

KEAMANAN SISTEM INFORMASI

Dr. Budi Raharjo, S.Kom., M.Kom., MM.

YAYASAN PRIMA AGUS TEKNIK

KEAMANAN SISTEM INFORMASI

Dr. Budi Raharjo, S.Kom., M.Kom., MM.

BIODATA PENULIS



Dr. Budi Raharjo, S.Kom, M.Kom, MM lahir di Semarang, tanggal 22 Februari 1985. Beliau adalah Alumni dari Universitas Bina Nusantara (BINUS University) Jakarta dan juga alumni Universitas Kristen Satya wacana (UKSW) Salatiga. Dr. Budi Raharjo telah menjadi Dosen pada Universitas STEKOM pada mata kuliah Kepemimpinan (Leadership), mata kuliah Pengantar Akuntansi, Manajemen Proses, Manajemen Akuntansi dan Manajemen Resiko Bisnis. Selain sebagai dosen Universitas STEKOM, Dr. Budi Raharjo, M.Kom, MM juga mempunyai bisnis sendiri dalam bidang perhotelan dan juga sebagai wirausaha dalam bidang pemasok unggas (ayam) beku, ke berbagai kota besar, khususnya Jakarta dan sekitarnya.

Pengalaman beliau berwirausaha menjadi bekal utama dalam penulisan buku ajar yang diterbitkan oleh Yayasan Prima Agus Teknik (YPAT) Semarang. Oleh sebab itu bukunya berisi langkah langkah praktis yang mudah diikuti oleh para mahasiswa, saat mahasiswa mengikuti proses perkuliahan pada Universitas Sains dan Teknologi Komputer (Universitas STEKOM). Jabatan struktural yang di embannya saat ini adalah Wakil Rektor 1 (Akademik) Universitas STEKOM Semarang.

YAYASAN PRIMA AGUS TEKNIK
Jl. Majapahit No. 605 Semarang
Telp. (024) 6723456. Fax. 024-6710144
Email : penerbit_ypat@stekom.ac.id

ISBN 978-623-6141-77-9 (PDF)

KEAMANAN
SISTEM INFORMASI

Dr. Budi Raharjo, S.Kom., M.Kom., MM.

YAYASAN PRIMA AGUS TEKNIK

KEAMANAN SISTEM INFORMASI

Penulis :
Dr. Budi Raharjo, S.Kom., M.Kom., MM.

ISBN : 9 786236 141779

Editor :
Dr. Mars Caroline Wibowo. S.T., M.Mm.Tech

Penyunting :
Dr. Joseph Teguh Santoso, M.Kom.

Desain Sampul dan Tata Letak :
Irdha Yunianto, S.Ds., M.Kom

Penebit :
Yayasan Prima Agus Teknik

Redaksi :
Jl. Majapahit no 605 Semarang
Telp. (024) 6723456
Fax. 024-6710144
Email : penerbit_ypat@stekom.ac.id

Distributor Tunggal :
Universitas STEKOM
Jl. Majapahit no 605 Semarang
Telp. (024) 6723456
Fax. 024-6710144
Email : info@stekom.ac.id

Hak cipta dilindungi undang-undang
Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara apapun tanpa ijin dari penulis

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa, bahwa buku yang berjudul “Keamanan Sistem Informasi Komputer" dapat diselesaikan dengan baik. Buku ini dapat membantu seorang pengguna yang ingin memahami lebih lanjut tentang cara kerja program keamanan komputer. Buku ini juga menyajikan latar belakang yang diperlukan tentang konsep sistem operasi atau jaringan, sebelum membahas tentang masalah keamanan terkait.

Keamanan komputer memberikan persyaratan terhadap komputer yang berbeda dari kebanyakan persyaratan sistem karena sering kali berbentuk pembatasan terhadap apa yang tidak boleh dilakukan komputer. Ini membuat keamanan komputer menjadi lebih menantang karena sudah cukup sulit untuk membuat program komputer melakukan segala apa yang sudah dirancang untuk dilakukan dengan benar

Buku ini ditujukan untuk orang-orang yang ingin belajar tentang keamanan komputer. Buku ini ditujukan untuk tiga kelompok orang: mahasiswa dan mahasiswa, profesional dan manajer komputasi, dan pengguna semua jenis sistem berbasis komputer. Semua ingin tahu hal yang sama: bagaimana mengendalikan risiko keamanan komputer. Tetapi Anda mungkin berbeda dalam seberapa banyak informasi yang Anda butuhkan tentang topik tertentu: Beberapa pembaca menginginkan survei yang luas, sementara yang lain ingin fokus pada topik tertentu, seperti jaringan atau pengembangan program.

Semoga buku ini dapat memberikan wawasan dan referensi bagi pembaca khususnya mahasiswa dalam memperdalam ilmu dasar-dasar keamanan sistem informasi. Penulis menyadari buku ini masih jauh dari sempurna, oleh karena itu penulis mengharapkan saran yang konstruktif dari para pembaca.

Semarang, Juni 2021

Dr. Budi Raharjo, S.Kom., M.Kom., MM

DAFTAR ISI

Kata Pengantar ..... **v**
Daftar Isi ..... **vi**

BAB 1 Pendahuluan **1**
1.1 Definisi Keamanan Komputer 3
1.1.1 Nilai Aset 4
1.1.2 Paradigma Kerentanan–Ancaman–Kontrol 5
1.2 Ancaman 6
1.2.1 Kerahasiaan (Confidentiality) 8
1.2.2 Integritas (Integrity) 9
1.2.3 Ketersediaan (Availability) 10
1.2.4 Jenis Ancaman 12
1.2.5 Ancaman Persisten Tingkat Lanjut 14
1.2.6 Jenis Penyerang 15
1.3 Kerugian 21
1.3.1 Risiko dan Berpikir Praktis 22
2.3.2 Metode – Peluang–Motif 25
1.4 Kerentanan 27
1.5 Kontrol 27
1.6 Kesimpulan 30

BAB 2 Otentikasi, Kontrol Akses, dan Kriptografi **33**
2.1 Otentikasi 35
2.1.1 Identifikasi Versus Otentikasi 36
2.1.2 Otentikasi Berdasarkan Frasa dan Fakta: Sesuatu yang Anda Ketahui 38
2.1.3 Penggunaan Kata Sandi 39
2.1.4 Dictionary Attack 40
2.1.5 Exhaustive Attack (Brute Force Attack) 45
2.2 Otentikasi Berdasarkan Biometrik 51
2.2.1 Contoh Authenticator Biometrik 52
2.3 Otentikasi Berdasarkan Token 62
2.3.1 Manajemen Federated Identity 64
2.3.2 Autentikasi Multifaktor 67
2.3.3 Otentikasi Aman 70
2.4 Kontrol Akses 71
2.4.1 Kebijakan Akses 72
2.4.2 Menerapkan Kontrol Akses 74
2.4.3 Kemampuan (Capability) 81
2.4.4 Procedure-Oriented Access Control 83
2.4.5 Role Based Access Control (RBAC) 84
2.5 Kriptografi 85
2.5.1 Masalah yang Ditangani dengan Enkripsi 86

2.5.2 Pembacaan Sandi 89
2.5.3 Sistem Enkripsi Simetris dan Asimetris 91
2.5.4 Streaming dan Blokir Cipher 92
2.5.5 DES: Data Encryption Standard 94
2.5.6 AES: Advanced Encryption System 97
2.5.7 Kriptografi\ Kunci Publik 100
2.5.8 Algoritma Rivest–Shamir–Adelman (RSA) 102
2.5.9 Kriptografi Kunci Publik untuk Bertukar Kunci Rahasia 103
2.4.10 Keaslian 108
2.5.11 Tanda Tangan (Digital Signature) 113
3.5.12 Kepercayaan 117
3.5.14 Sertifikat: Identitas Tepercaya dan Kunci Publik 121

Bab 3 Jaringan Komputer 130
3.1 Konsep Jaringan 131
3.1.1 Media Transmisi Jaringan 132
3.1.2 Lapisan Protokol 138
3.1.3 Addressing dan Routing 139
Bagian I : Perang terhadap Jaringan: Serangan Keamanan Jaringan 142
3.2 Ancaman Terhadap Komunikasi Jaringan 142
3.2.1 Intersepsi: Eavesdropping dan Penyadapan 143
3.2.2 Apa yang Membuat Jaringan Rentan terhadap Intersepsi? 146
3.2.3 Data Corruption : Modifikasi dan Fabrikasi 150
3.2.4 Interruption: Hilangnya Layanan 155
3.2.5 Pemindaian Port 157
3.2.6 Ringkasan Kerentanan Jaringan 162
3.3 Keamanan Jaringan Nirkabel 163
3.3.1 Latar Belakang WiFi 163
3.3.2 Kerentanan di Jaringan Nirkabel 169
3.3.3 Penanggulangan yang Gagal: WEP (Wired Equivalent Privacy) 176
3.3.4 Suite Protokol yang Lebih Kuat: WPA (WiFi Protected Access) 180
3.4 Denial of Service (DoS) 185
3.4.1 Bagaimana Layanan Ditolak 187
3.4.2 Flooding Jaringan dengan Resource Exhaustion 195
3.4.3 Denial of Service dengan Kegagalan Addressing 197
3.4.4 PengalihanTraffic 201
3.4.5 Serangan DNS 202
3.4.6 Memanfaatkan Kerentanan yang Diketahui 207
3.4.7 Pemutusan Secara Fisik 208
3.5 Distributed Denial of Service (DDoS) 210
3.5.1 Serangan Scripted Denial-of-Service 213
3.5.2 Mengatasi Serangan DDoS 220
Bagian II : Pertahanan Strategis: Penanggulangan Keamanan 220
3.6 Kriptografi dalam Keamanan Jaringan 221
3.6.1 Enkripsi Jaringan 221
3.6.2 Enkripsi Peramban (Browser Encryption) 225
3.6.3 IP Security Protocol Suite (IPsec) 231
3.6.4 Security Association 233

3.6.5 Virtual Private Network 236
3.6.6 Sistem Arsitektur 239
3.7 Firewall 241
3.7.1 Definisi Firewall 241
3.7.2 Desain Firewall 242
3.7.2 Jenis Firewal 244
3.7.3 Personal Firewall 253
3.7.4 Network Address Translation (NAT) 257
3.7.5 Data Loss Prevention 258
3.8 Intrusion Detection System 260
3.8.1 Pemahaman 260
3.8.2 Sistem Pencegahan Intrusi 264
3.7.2 Tujuan untuk Sistem Deteksi Intrusi (IDS) 268
3.9 Manajemen Jaringan 269
3.9.1 Manajemen untuk Memastikan Layanan 269
3.9.2 Security Information and Event Management (SIEM) 272
3.10 Kesimpulan 277

Bab 4 Database 281
4.1 Pengenalan Database..... 283
4.1.1 Konsep Database 283
4.1.2 Komponen Database 284
4.1.3 Keuntungan Menggunakan Database 287
4.2 Persyaratan Keamanan Basis Data 288
4.3 Keandalan dan Integritas 294
4.3.1 Fitur Perlindungan dari Sistem Operasi 295
4.3.2 Redundansi/Konsistensi Internal 297
4.4 Database Disclosure 299
4.4.1 Data Sensitif 299
4.5 Data Mining Dan Big Data 304
4.5.1 Data mining 304
4.5.2 Big Data 308
4.6 Kesimpulan 316

Bab 5 Kerahasiaan Privasi 319
5.1 Konsep Privasi 320
5.1.1 Aspek- Aspek Privacy Information (Security) 320
5.1.2 Masalah Privasi dalam Program Komputer 323
5.1.3 Kontrol dan Kepemilikan Data 326
5.2 Prinsip dan Kebijakan Privasi 329
5.2.1 Praktik Informasi yang Adil 329
5.2.2 Tindakan Individu untuk Melindungi Privasi 334
5.2.3 Pemerintah dan Privasi 337
5.2.4 Pencurian Identitas 339
5.3 Otentikasi dan Privasi 339
9.3.1 Apa Arti Otentikasi 340
5.4 Keamanan Privasi Data mining 245
5.4.1 Data Mining Pemerintah 346

5.5 Privasi di Web 348
5.5.1 Memahami Lingkungan Online 348
5.5.2 Pembayaran dalam Web 350
5.5.3 Pendaftaran Situs dan Portal 351
5.5.4 Tindakan Pencegahan untuk Web Surfing 353
5.5.5 Bug Web: Apakah ada Alat Pembasminya? 357
5.5.6 Belanja di Internet 360
5.6 Keamanan Email 362
5.6.1 Ke Mana Email Pergi, dan Siapa yang Dapat Mengaksesnya? 362
5.7 Dampak Privasi dari Teknologi yang Muncul 366
5.7.1 Radio Frequency Identification RFID) .....
5.7.2 Pemungutan Suara Elektronik 369
5.7.3 VoIP dan Skype 370
5.7.4 Privasi di Cloud 371
5.7.5 Kesimpulan tentang Emerging Technology 372
5.8 Kesimpulan 372

Bab 6 Kriptografi dalam Keamanan Komputer 375
6.1 Kriptologi 376
6.1.1 Kriptanalisis 376
6.1.2 Kriptografi Primitif 381
6.1.3 Analisis Statistik 388
6.2 Algoritma Enkripsi Simetris 391
6.2.1 Enkripsi Simetris DES 392
6.2.2 Pertanyaan Tentang Keamanan DES 400
6.2.3 Enkripsi AES 402
6.2.4 RC2, RC4, RC5, dan RC6 405
6.3 Enkripsi Asimetris dengan RSA 409
6.3.1 Algoritma RSA 410
6.4 Message Digest 414
6.4.1 Fungsi Hash 414
6.4.2 One-Way Hash Function 415
6.5 Tanda Tangan Digital 417
6.5.1 Kriptografi Kurva Eliptik 418
6.5.2 Algoritma Tanda Tangan Digital El Gamal 419
6.6 Kriptografi Kuantum 422
6.6.1 Fisika Kuantum 424
6.6.2 Penerimaan Foton 425
6.6.3 Kriptografi Dengan Foton 426
6.6.4 Penerapan 427
6.7 Kesimpulan 427

Daftar Pustaka 429

Bab 1

Pendahuluan

Bab ini akan membahas tentang:

- Ancaman, kerentanan, dan kontrol
- Kerahasiaan, Integritas, dan Ketersediaan (C-I-A)
- Penyerang dan jenis serangan; metode, kesempatan, dan motif
- Menilai aset

Tiap tahun, ancaman *cybersecurity* selalu menghantui seluruh penjuru dunia. Dari beberapa Studi Kasus yang terjadi, beberapa diantaranya masih menjadi ancaman yang menakutkan. Meski masing-masing memiliki berbagai tujuan berbeda, namun ancaman tersebut berdampak pula pada kepentingan lain karena sifatnya yang tidak bisa dibendung.

Serangan siber sudah banyak terjadi, baik dari jenis paling biasa hingga tercanggih sekalipun dengan dampak kerugian bagi pengguna. Dalam Studi Kasus terburuk yang pernah terjadi, misalnya pengguna dapat ancaman pembayaran tebusan di layar, ada juga senyap, yang bertindak tersembunyi untuk pencurian data.

Serangan siber tidak akan berakhir selagi ada perangkat komputasi yang terhubung ke Internet. Serangan siber terus meningkat terutama selama pandemi corona, menyusul banyak pengguna yang bekerja dan belajar dari rumah dengan memanfaatkan jaringan Internet.

Cisco Umbrella mencatat setidaknya terjadi peningkatan sebesar 40% serangan siber di tahun lalu. Karena itu pengguna harus mengetahui berbagai jenis serangan siber yang umumnya menyerang dan meningkatkan keamanan siber-nya.

Keamanan siber adalah praktik untuk melindungi sistem, jaringan, dan program dari ancaman atau serangan digital.

Serangan-serangan ini biasanya ditujukan untuk mengakses, mengubah, atau menghancurkan informasi sensitif; memeras uang dari korban; atau mengganggu proses bisnis.

Shadow Brokers Curi Senjata Cyber NSA

Shadow Brokers mengklaim telah berhasil mendapatkan senjata cyber berupa tool hacking milik National Security Agency (NSA). Senjata ini memiliki kemampuan untuk membobol seluruh sistem operasi Windows. Tindakan pencurian ini merupakan bentuk protes kelompok peretas tersebut terhadap NSA. Gambar 1-1 mengilustrasikan serangan ini.

NSA yang mendapatkan celah melalui tool tersebut bukannya melaporkan ke Microsoft ataupun perusahaan lainnya, namun malah menyimpannya untuk kepentingan keamanan mereka sendiri. Akhirnya setelah melakukan peretasan, Shadow Brokers menjual informasi tersebut kepada pihak yang membutuhkan dengan tawaran yang paling tinggi melalui sistem pelelangan.



Gambar 1-1 Ilustrasi Serangan terhadap NSA

Yang dikuatirkan, jika data tersebut jatuh ke tangan yang salah, hal ini tentu berpotensi membahayakan miliaran pengguna perangkat lunak di seluruh dunia. Untungnya Microsoft langsung merespon dan merilis perbaikan yang diperlukan. Hanya saja perbaikan tersebut hanya berlaku untuk sistem operasi Windows yang masih dalam dukungan Microsoft seperti Windows 7 keatas.

Pelaku hoaks ini tidak pernah tertangkap, juga tidak jelas bagaimana tepatnya hal itu dilakukan. Kemungkinan seseorang dapat mengakses sistem yang menyalurkan siaran darurat ke stasiun radio dan televisi lokal. Dengan kata lain, seorang hacker mungkin membobol sistem komputer.

Kita menghadapi komputer setiap hari dalam situasi yang tak terhitung jumlahnya, seringkali dalam Studi Kasus di mana hampir tidak menyadari komputer terlibat, seperti sistem peringatan darurat untuk media penyiaran. Komputer-komputer ini memindahkan uang, mengendalikan pesawat terbang, memantau kesehatan, mengunci pintu, memutar musik, memanaskan bangunan, mengatur hati, menyebarkan airbag, menghitung suara, mengarahkan komunikasi, mengatur lalu lintas, dan melakukan ratusan hal lain yang memengaruhi kehidupan, kesehatan, keuangan, dan kesejahteraan. Sebagian besar waktu komputer ini bekerja sebagaimana mestinya. Tetapi kadang-kadang mereka melakukan kesalahan yang mengerikan, baik karena kegagalan yang tidak berbahaya atau serangan yang berbahaya. Seperti diilustrasikan pada Gambar 1-2.



Gambar 1-2 Ilustrasi Serangan Siber

Buku ini membahas tentang keamanan komputer, datanya, dan perangkat serta objek yang terkait dengannya. Dalam buku ini Anda akan mempelajari beberapa cara komputer bisa gagal atau dibuat gagal—dan bagaimana melindungi dari kegagalan tersebut.

1.1 Definisi Keamanan Komputer

Definisi computer security atau keamanan komputer saat ini manjadi lebih luas atau bisa juga didefinisikan sebagai berikut: keamanan komputer dirancang untuk melindungi komputer dan segala sesuatu yang berkaitan dengan itu, bangunannya, workstation dan printer, kabel, dan disk dan media penyimpanan lainnya. Yang paling penting, keamanan komputer melindungi informasi yang disimpan dalam sistem anda. Keamanan komputer tidak hanya dirancang untuk melindungi terhadap penyusup dari luar yang masuk ke sistem, tetapi juga bahaya yang timbul dari dalam seperti berbagi password dengan teman, gagal atau tidak dilakukan untuk backup data, menumpahkan kopi pada keyboard dan sebagainya. Untuk menentukan apa yang harus dilindungi, pertama-tama kita harus mengidentifikasi apa yang dinamakan aset dan nilai aset yang ada dalam program kompoter.

Aset adalah data, perangkat, atau komponen lain dari sistem organisasi yang berharga – seringkali karena berisi data sensitif atau dapat digunakan untuk mengakses informasi tersebut. Misalnya, komputer desktop, laptop, atau telepon perusahaan karyawan akan dianggap sebagai aset, seperti halnya aplikasi pada perangkat tersebut. Demikian juga, infrastruktur penting, seperti server dan sistem pendukung, adalah aset.

Aset organisasi yang paling umum adalah aset informasi. Ini adalah hal-hal seperti database dan file fisik – yaitu data sensitif yang Anda simpan.

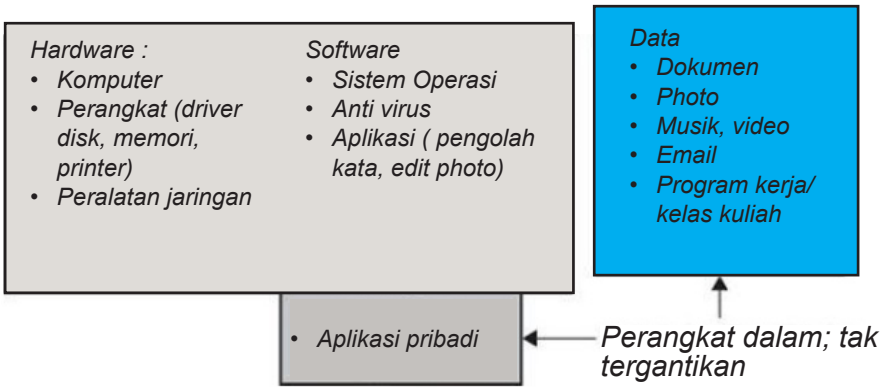
Konsep terkait adalah 'wadah aset informasi', di mana informasi itu disimpan. Dalam kasus database, aset ini akan menjadi aplikasi yang digunakan untuk membuat database. Untuk file fisik, akan menjadi lemari arsip tempat informasi berada.

Perangkat komputer (termasuk perangkat keras, komponen tambahan, dan aksesori) tentu saja merupakan aset. Karena sebagian besar perangkat keras komputer tidak berguna tanpa program, perangkat lunak juga merupakan aset. Perangkat lunak mencakup sistem operasi, utilitas, dan handler perangkat; aplikasi seperti pengolah kata, pemutar media atau handler email; dan bahkan

program yang mungkin Anda tulis sendiri. Banyak perangkat keras dan perangkat lunak tidak tersedia, artinya tersedia secara komersial (bukan dibuat khusus untuk tujuan Anda) dan Anda dapat dengan mudah mendapatkan penggantinya. Hal yang membuat komputer unik dan penting bagi pemakainya adalah apa yang ada dalam komputer tersebut : foto, lagu, makalah, pesan email, proyek, informasi kalender, ebook (dengan anotasi Anda), informasi kontak, kode yang dibuat, dan sejenisnya. Dengan demikian, item data di komputer juga merupakan aset. Tidak seperti kebanyakan perangkat keras dan perangkat lunak, data bisa sulit—jika bukan tidak mungkin—untuk dibuat ulang atau diganti. Semua aset ini ditunjukkan pada Gambar 1-3.



perangkat fisik: bisa diganti



Gambar 1-3 Nilai Aset Komputer

Ketiga hal ini—perangkat keras, perangkat lunak, dan data—berisi atau mengekspresikan hal-hal seperti desain untuk produk baru Anda berikutnya, foto-foto dari liburan terakhir Anda, bab-bab buku baru Anda, atau urutan genom yang dihasilkan dari penelitian terbaru Anda. Semua hal ini mewakili usaha intelektual atau kekayaan, dan mereka memiliki nilai yang berbeda dari satu orang atau organisasi ke orang lain. Nilai itulah yang membuat mereka menjadi aset yang layak dilindungi, dan mereka adalah elemen yang ingin kita lindungi. Aset lain—seperti akses ke data, kualitas layanan, proses, pengguna manusia, dan konektivitas jaringan—juga layak dilindungi; mereka dipengaruhi atau diaktifkan oleh perangkat keras, perangkat lunak, dan data. Jadi, dalam banyak Studi Kasus, melindungi perangkat keras, perangkat lunak, dan data juga mencakup aset lain ini.

1.1.1 Nilai Aset

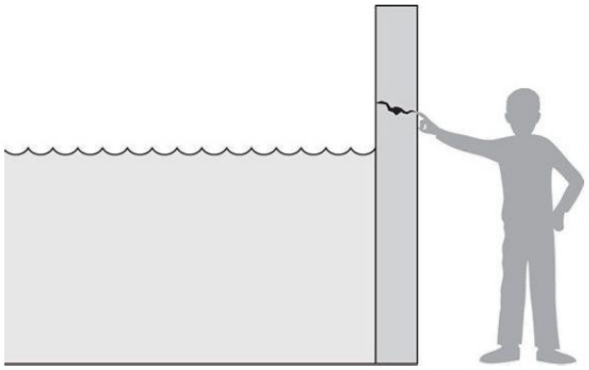
Setelah mengidentifikasi aset yang akan dilindungi, selanjutnya kami menentukan nilainya. Kita sering membuat keputusan berdasarkan nilai, bahkan ketika kita tidak menyadarinya. Misalnya, ketika Anda pergi berenang, Anda dapat meninggalkan sebotol air dan handuk di pantai, tetapi tidak dengan dompet atau ponsel Anda. Perbedaannya berkaitan dengan nilai aset.

Nilai suatu aset bergantung pada perspektif pemilik aset atau pengguna, dan mungkin tidak tergantung pada biaya moneter. Foto saudara perempuan Anda, yang hanya bernilai beberapa sen dalam bentuk kertas dan tinta, mungkin bernilai tinggi bagi Anda dan tidak bernilai bagi teman sekamar Anda. Nilai barang lain tergantung pada biaya penggantian; beberapa data komputer sulit atau tidak mungkin untuk diganti. Misalnya, foto Anda dan teman-teman Anda di sebuah pesta mungkin tidak dikenakan biaya apa pun, tetapi itu sangat berharga karena tidak ada salinan lain. Di sisi lain, DVD film favorit Anda mungkin menghabiskan sebagian besar gaji Anda, tetapi Anda dapat membeli yang lain jika DVD dicuri atau rusak. Demikian pula, waktu memiliki pengaruh pada nilai aset. Misalnya, nilai rencana lini produk baru perusahaan sangat tinggi, terutama bagi pesaing. Tetapi begitu produk baru dirilis, nilai paket turun drastis.

1.1.2 Paradigma Kerentanan–Ancaman–Kontrol

Tujuan dari keamanan komputer adalah melindungi aset berharga. Untuk mempelajari berbagai cara perlindungan, kami menggunakan kerangka kerja yang menjelaskan bagaimana aset dapat dirugikan dan bagaimana melawan atau mengurangi kerugian tersebut.

Kerentanan (*vulnerability*) adalah kelemahan dalam sistem, misalnya, dalam prosedur, desain, atau implementasi, yang dapat dieksploitasi untuk menyebabkan kerugian atau bahaya. Misalnya, sistem tertentu mungkin rentan terhadap manipulasi data yang tidak sah karena sistem tidak memverifikasi identitas pengguna sebelum mengizinkan akses data.



Gambar 1-4 Ancaman dan Kerentanan

Ancaman (*threat*) terhadap sistem komputasi adalah serangkaian keadaan yang berpotensi menyebabkan kerugian atau bahaya. Untuk melihat perbedaan antara ancaman dan kerentanan, perhatikan ilustrasi pada Gambar 1-4. Di sini, dinding menahan air. Air di sebelah kiri tembok merupakan ancaman bagi orang di sebelah kanan tembok: Air bisa naik, meluap ke orang itu, atau bisa tetap di bawah ketinggian tembok, menyebabkan tembok runtuh. Jadi ancaman celaka adalah potensi laki-laki untuk basah, terluka, atau tenggelam. Untuk saat ini, tembok itu masih utuh, jadi ancaman terhadap pria itu tidak disadari.

Namun, kita bisa melihat celah kecil di dinding—kerentanan yang mengancam keamanan pria itu. Jika air naik ke atau di luar tingkat retakan, itu akan mengeksploitasi kerentanan dan membahayakan manusia. Ada banyak ancaman terhadap sistem komputer, termasuk yang diprakarsai oleh manusia dan yang diprakarsai oleh komputer. Kita semua pernah mengalami akibat kesalahan manusia yang tidak disengaja, cacat desain perangkat keras, dan kegagalan perangkat lunak. Tapi bencana alam juga merupakan ancaman; mereka dapat meruntuhkan sistem saat ruang komputer kebanjiran atau pusat data ambruk akibat gempa, misalnya.

Seseorang yang mengeksploitasi kerentanan melakukan serangan pada sistem. Serangan juga dapat diluncurkan oleh sistem lain, seperti ketika satu sistem mengirim pesan yang sangat banyak

ke sistem lain, hampir mematikan kemampuan sistem kedua untuk berfungsi. Sayangnya, kami sering melihat jenis serangan ini, karena serangan penolakan layanan membanjiri server dengan lebih banyak pesan daripada yang dapat mereka tangani.

Bagaimana kita mengatasi masalah ini? Kami menggunakan kontrol atau penanggulangan sebagai perlindungan. Artinya, kontrol adalah tindakan, perangkat, prosedur, atau teknik yang menghilangkan atau mengurangi kerentanan. Pada Gambar 1-4, pria itu memasukkan jarinya ke dalam lubang, mengendalikan ancaman kebocoran air sampai dia menemukan solusi yang lebih permanen untuk masalah tersebut. Secara umum, kita dapat menggambarkan hubungan antara ancaman, kontrol, dan kerentanan dengan cara ini: Sebelum kita dapat melindungi aset, kita perlu mengetahui jenis bahaya yang harus kita lindungi, jadi sekarang kita mengeksplorasi ancaman terhadap aset berharga.

1.2 Ancaman

Ancaman adalah setiap insiden yang dapat berdampak negatif terhadap suatu aset – misalnya, jika hilang, rusak saat offline, atau diakses oleh pihak yang tidak berwenang. Ancaman dapat dikategorikan sebagai keadaan yang membahayakan kerahasiaan, integritas, atau ketersediaan aset, dan dapat disengaja atau tidak disengaja.

Ancaman yang disengaja mencakup hal-hal seperti peretasan kriminal atau orang dalam yang jahat mencuri informasi, sedangkan ancaman yang tidak disengaja umumnya melibatkan kesalahan karyawan, kerusakan teknis, atau peristiwa yang menyebabkan kerusakan fisik, seperti kebakaran atau bencana alam.
Kita dapat mempertimbangkan potensi kerusakan aset dengan dua cara: Pertama, kita dapat melihat hal buruk apa yang dapat terjadi pada aset, dan kedua, kita dapat melihat siapa atau apa yang dapat menyebabkan atau membiarkan hal buruk tersebut terjadi. Kedua perspektif ini memungkinkan kita untuk menentukan bagaimana melindungi aset.

Pikirkan sejenak tentang apa yang membuat komputer berharga bagi Anda. Pertama, Anda menggunakannya sebagai alat untuk mengirim dan menerima email, menelusuri web, menulis makalah, dan melakukan banyak tugas lainnya, dan Anda mengharapkannya tersedia untuk digunakan saat Anda menginginkannya. Tanpa komputer, tugas-tugas ini akan lebih sulit, jika bukan tidak mungkin. Kedua, Anda sangat bergantung pada integritas komputer Anda. Saat menulis kertas dan menyimpannya, Anda percaya bahwa kertas itu akan dimuat ulang persis seperti Anda menyimpannya. Demikian pula, Anda berharap foto yang diberikan teman pada Anda di flash drive akan tampak sama saat Anda memuatnya ke komputer seperti saat Anda melihatnya di komputer teman Anda. Terakhir, Anda mengharapkan aspek "pribadi" dari komputer pribadi tetap bersifat pribadi, artinya Anda menginginkannya untuk melindungi kerahasiaan Anda. Misalnya, Anda ingin pesan email Anda hanya antara Anda dan penerima yang terdaftar; Anda tidak ingin mereka disiarkan ke orang lain. Dan ketika menulis esai, Anda berharap tidak ada yang bisa menyalinnya tanpa izin Anda.

Ketiga aspek ini, kerahasiaan, integritas, dan ketersediaan, menjadikan komputer Anda berharga bagi Anda. Namun dilihat dari sudut pandang lain, ada tiga cara yang mungkin untuk membuatnya kurang berharga, yaitu menyebabkan kerugian bagi Anda. Jika seseorang mencuri komputer Anda,

mengacak data di disk Anda, atau melihat file data pribadi Anda, nilai komputer Anda telah berkurang atau penggunaan komputer Anda telah dirugikan. Karakteristik ini merupakan properti keamanan dasar dan objek ancaman keamanan.

Ketiga sifat tersebut dapat kita definisikan sebagai berikut.

- ketersediaan: kemampuan sistem untuk memastikan bahwa suatu aset dapat digunakan oleh pihak yang berwenang
- integritas: kemampuan sistem untuk memastikan bahwa aset hanya dimodifikasi oleh pihak yang berwenang
- kerahasiaan: kemampuan sistem untuk memastikan bahwa aset hanya dilihat oleh pihak yang berwenang

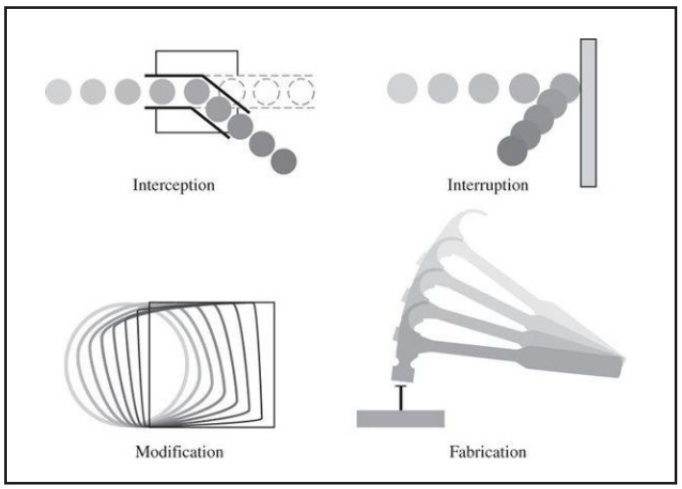
Ketiga properti ini, ciri-ciri keamanan yang solid, muncul dalam literatur sejak esai James P. Anderson tentang keamanan komputer (Anderson:73) dan sering muncul kembali dalam makalah dan diskusi keamanan komputer yang lebih baru. Secara bersama-sama (dan disusun ulang), properti disebut triad C-I-A atau triad keamanan. ISO 7498-2 (ISO:89) menambahkan dua properti lagi yang diinginkan, terutama dalam jaringan komunikasi:

- otentikasi: kemampuan sistem untuk mengkonfirmasi identitas pengirim
- nonrepudiation atau akuntabilitas: kemampuan sistem untuk mengkonfirmasi bahwa pengirim tidak dapat secara meyakinkan menyangkal telah mengirim sesuatu

Departemen Pertahanan AS menambahkan kemampuan audit: kemampuan sistem untuk melacak semua tindakan yang terkait dengan aset tertentu. Didalam information security sering juga dikenal CIA Triad atau segitiga ***C****onfidentiality* (kerahasiaan), ***I****ntegrity* (integritas), dan ***A****vailability* (ketersediaan). Kerahasiaan, integritas dan ketersediaan, yang dikenal sebagai segitiga CIA ini adalah model yang dirancang untuk memandu kebijakan untuk keamanan informasi dalam sebuah organisasi. Model ini juga kadang-kadang disebut sebagai triad AIC (ketersediaan, integritas dan kerahasiaan) untuk menghindari kebingungan dengan Central Intelligence Agency. Unsur-unsur dari tiga serangkai tersebut dianggap tiga komponen yang paling penting dari system keamanan.

Apa yang dapat terjadi untuk merusak kerahasiaan, integritas, atau ketersediaan aset komputer? Jika pencuri mencuri komputer Anda, Anda tidak lagi memiliki akses, sehingga Anda kehilangan ketersediaan; selanjutnya, jika pencuri melihat gambar atau dokumen yang Anda simpan, kerahasiaan Anda terganggu. Dan jika pencuri mengubah konten file musik Anda tetapi kemudian mengembalikannya dengan komputer Anda, integritas data Anda telah dirugikan. Anda dapat membayangkan banyak skenario berdasarkan tiga properti ini.

Triad C-I-A dapat dilihat dari perspektif yang berbeda: sifat kerugian yang ditimbulkan pada aset. Harm juga dapat dicirikan oleh empat tindakan: intersepsi, interupsi, modifikasi, dan fabrikasi. Keempat tindakan ini digambarkan pada Gambar 1-5. Dari sudut pandang ini, kerahasiaan dapat terganggu jika seseorang menyadap data, ketersediaan hilang jika seseorang atau sesuatu mengganggu aliran data atau akses ke komputer, dan integritas dapat gagal jika seseorang atau sesuatu memodifikasi data atau membuat data palsu. Memikirkan keempat jenis tindakan ini dapat membantu Anda menentukan ancaman apa yang mungkin ada terhadap komputer yang Anda coba lindungi.



Gambar 1-5 Empat Tindakan yang Menyebabkan Kerusakan Keamanan

Untuk menganalisis bahaya, kami selanjutnya menyempurnakan triad C-I-A, melihat lebih dekat setiap elemennya.

1.2.1 Kerahasiaan (*Confidentiality*)

Beberapa hal jelas membutuhkan perlindungan kerahasiaan. Misalnya, nilai siswa, transaksi keuangan, catatan medis, dan pengembalian pajak bersifat sensitif dan pribadi. Seorang siswa yang bangga mungkin berlari keluar kelas sambil berteriak, "Saya mendapat nilai A!" tetapi siswa harus menjadi orang yang memilih apakah akan mengungkapkan nilai itu kepada orang lain. Hal-hal lain, seperti rahasia diplomatik dan militer, rencana pemasaran dan pengembangan produk perusahaan, dan tes pendidik, juga harus dikontrol dengan hati-hati. Terkadang, bagaimanapun, tidak begitu jelas bahwa ada sesuatu yang sensitif. Misalnya, pesanan makanan militer mungkin tampak seperti informasi yang tidak berbahaya, tetapi peningkatan pesanan yang tiba-tiba bisa menjadi tanda keterlibatan yang baru mulai dalam konflik. Pembelian makanan, perubahan lokasi setiap jam, dan akses ke buku bukanlah hal yang biasanya Anda anggap rahasia, tetapi dapat mengungkapkan sesuatu yang ingin dirahasiakan seseorang.

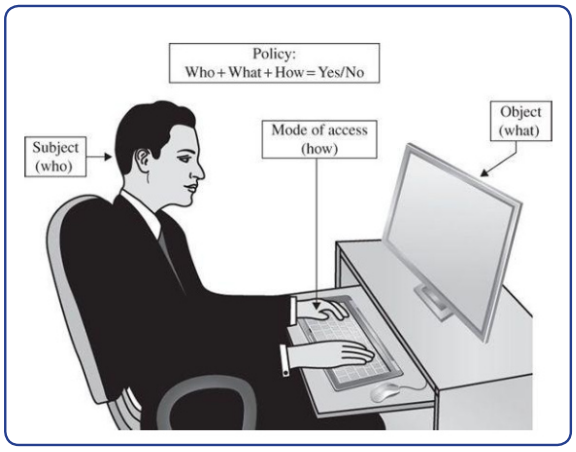
Definisi kerahasiaan sangat mudah: Hanya orang atau sistem yang berwenang yang dapat mengakses data yang dilindungi. Namun, seperti yang kita lihat di bab-bab selanjutnya, memastikan kerahasiaan bisa jadi sulit. Misalnya, siapa yang menentukan orang atau sistem mana yang berwenang untuk mengakses sistem saat ini? Dengan “mengakses” data, apakah yang kita maksud adalah bahwa pihak yang berwenang dapat mengakses satu bit? Seluruh koleksi? Potongan data di luar konteks? Dapatkah seseorang yang berwenang mengungkapkan data kepada pihak lain? Kadang-kadang bahkan ada pertanyaan tentang siapa yang memiliki data: Jika Anda mengunjungi halaman web, apakah Anda memiliki fakta bahwa Anda mengklik tautan, atau apakah pemilik halaman web, penyedia Internet, orang lain, atau Anda semua?

Sederhananya, *confidentiality* ini bisa berarti sama dengan privasi. Ini juga merupakan serangkaian langkah-langkah yang perlu dilakukan untuk mencegah terekspos nya informasi sensitif dari jangkauan tangan orang-orang yang tidak berwenang. Tak hanya itu, juga harus dipastikan bahwa orang yang tepat sudah benar-benar mendapatkannya data yang dibutuhkan.

Kerahasiaan berhubungan paling jelas dengan data, meskipun kita dapat memikirkan kerahasiaan perangkat keras (penemuan baru) atau seseorang (keberadaan penjahat yang dicari). Berikut adalah beberapa properti yang dapat berarti kegagalan kerahasiaan data:

- Orang yang tidak berwenang mengakses item data.
- Proses atau program yang tidak sah mengakses item data.
- Seseorang yang berwenang untuk mengakses data tertentu mengakses data lain yang tidak diotorisasi (yang merupakan versi khusus dari "orang yang tidak berwenang mengakses item data").
- Orang yang tidak berwenang mengakses perkiraan nilai data (misalnya, tidak mengetahui gaji pasti seseorang tetapi mengetahui bahwa gaji tersebut berada dalam kisaran tertentu atau melebihi jumlah tertentu).
- Orang yang tidak berwenang mengetahui keberadaan sepotong data (misalnya, mengetahui bahwa sebuah perusahaan sedang mengembangkan produk baru tertentu atau bahwa pembicaraan sedang berlangsung tentang penggabungan dua perusahaan).

Perhatikan pola umum dari pernyataan ini: Seseorang, proses, atau program (atau tidak) berwenang untuk mengakses item data dengan cara tertentu. Kami menyebut orang, proses, atau program sebagai subjek, item data sebagai objek, jenis akses (seperti membaca, menulis, atau mengeksekusi) mode akses, dan otorisasi sebagai kebijakan, seperti yang ditunjukkan pada Gambar 1-6. Keempat istilah ini muncul kembali di seluruh buku ini karena merupakan aspek fundamental dari keamanan komputer.



Gambar 1-6 Kontrol Akses

Satu kata yang menangkap sebagian besar aspek kerahasiaan adalah pandangan, meskipun Anda tidak boleh mengartikan istilah itu secara harfiah. Kegagalan kerahasiaan tidak selalu berarti bahwa seseorang melihat suatu objek dan, pada kenyataannya, hampir tidak mungkin untuk melihat bit dengan cara yang berarti (walaupun Anda mungkin melihat representasi mereka sebagai karakter atau gambar). Kata view memang berkonotasi aspek lain dari kerahasiaan dalam keamanan komputer, melalui asosiasi dengan menonton film atau lukisan di museum: lihat tapi jangan sentuh. Dalam keamanan komputer, kerahasiaan biasanya berarti memperoleh tetapi tidak memodifikasi. Modifikasi adalah subjek integritas, yang akan kita bahas di bagian selanjutnya.

1.2.2 Integritas (*Integrity*)

Contoh kegagalan integritas mudah ditemukan. Beberapa tahun yang lalu makro jahat dalam dokumen Word memasukkan kata "tidak" setelah beberapa contoh acak dari kata "adalah;" Anda bisa membayangkan kekacauan yang terjadi. Karena dokumen umumnya benar secara sintaksis, orang tidak segera mendeteksi perubahan tersebut. Dalam Studi Kasus lain, model chip komputer Pentium

menghasilkan hasil yang salah dalam keadaan tertentu dari aritmatika floating-point. Meskipun keadaan kegagalan jarang terjadi, Intel memutuskan untuk memproduksi dan mengganti chip tersebut. Banyak dari kita menerima email yang salah alamat karena seseorang salah mengetik ketika menyalin dari daftar tertulis. Situasi yang lebih buruk terjadi ketika ketidakakuratan itu disebarkan ke milis lain sehingga kita sepertinya tidak pernah bisa memperbaiki akar masalahnya. Di lain waktu kami menemukan bahwa spreadsheet tampaknya salah, hanya untuk menemukan bahwa seseorang mengetik "spasi 123" di sel, mengubahnya dari nilai numerik menjadi teks, sehingga program spreadsheet menyalahgunakan sel itu dalam perhitungan. Misalkan seseorang mengonversi data numerik menjadi angka romawi: Seseorang dapat berargumen bahwa IV sama dengan 4, tetapi IV tidak akan berguna di sebagian besar aplikasi, juga tidak akan berarti bagi seseorang yang mengharapkan 4 sebagai jawaban. Studi Kasus-Studi Kasus ini menunjukkan beberapa contoh luasnya kegagalan integritas.

Integriy ini berarti menjaga konsistensi, akurasi, dan kepercayaan terhadap data untuk setiap waktu hingga seterusnya. Data tidak boleh diubah pada saat transit. Kemudian juga langkah-langkah tertentu perlu dilakukan untuk memastikan bahwa data tidak bisa diubah-ubah oleh orang yang tidak punya kepentingan sejalan (misalnya, para peretas yang ingin melakukan manipulasi data dsb).

Langkah-langkah tersebut juga termasuk izin dalam mengakses file dan batasan kontrol bagi akses pengguna. Kontrol ini bisa dipakai untuk mencegah perubahan yang keliru atau penghapusan tidak disengaja dari pengguna resmi yang bisa juga menjadi masalah. Jadi, intinya backup/redundant harus tersedia untuk memulihkan data yang sudah kena masalah agar bisa kembali ke keadaan yang semula.
Misalnya, jika kita mengatakan bahwa kita telah mempertahankan integritas suatu barang, yang kita maksudkan adalah barang tersebut

- tepat
- akurat
- tidak dimodifikasi
- dimodifikasi hanya dengan cara yang dapat diterima
- dimodifikasi hanya oleh orang yang berwenang
- dimodifikasi hanya oleh proses resmi
- konsisten
- konsisten secara internal
- bermakna dan berguna

Integritas juga dapat berarti dua atau lebih dari sifat-sifat ini. Welke dan Mayfield mengenali tiga aspek tertentu dari integritas—tindakan yang diotorisasi, pemisahan dan perlindungan sumber daya, serta deteksi dan koreksi kesalahan. Integritas dapat ditegakkan dengan cara yang sama seperti kerahasiaan: dengan kontrol yang ketat terhadap siapa atau apa yang dapat mengakses sumber daya mana dengan cara apa.

1.2.3 Ketersediaan (*Availability*)

Mimpi terburuk pengguna komputer: Anda menyalakan sakelar dan komputer tidak melakukan apa-apa. Data dan program Anda mungkin masih ada, tetapi Anda tidak bisa mengaksesnya sama

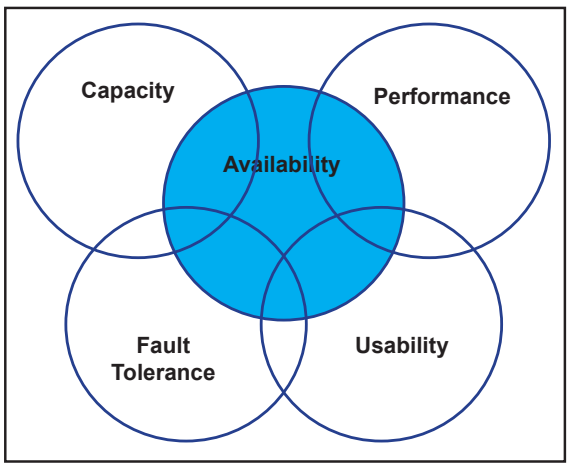
sekali. Untungnya, hanya sedikit dari kita yang mengalami kegagalan itu. Namun, banyak dari kita mengalami kelebihan beban: akses semakin lambat; komputer merespons tetapi tidak dengan cara yang kami anggap normal atau dapat diterima.

Ketersediaan berlaku untuk data dan layanan (yaitu, untuk informasi dan pemrosesan informasi), dan juga kompleks. Seperti halnya gagasan kerahasiaan, orang yang berbeda mengharapkan ketersediaan berarti hal yang berbeda. Misalnya, objek atau layanan dianggap tersedia jika hal berikut ini benar:

- Hal ini hadir dalam bentuk yang dapat digunakan.
- Memiliki kapasitas yang cukup untuk memenuhi kebutuhan layanan.
- Hal ini membuat kemajuan yang jelas, dan, jika dalam mode tunggu, memiliki waktu tunggu yang terbatas.
- Layanan selesai dalam jangka waktu yang dapat diterima.

Kita dapat membangun deskripsi ketersediaan secara keseluruhan dengan menggabungkan tujuan-tujuan ini. Berikut adalah beberapa kriteria untuk menentukan ketersediaan.

- Ada tanggapan yang tepat waktu untuk permintaan kami.
- Sumber daya dialokasikan secara adil sehingga beberapa pemohon tidak disukai daripada yang lain.
- Konkurensi dikendalikan; yaitu, akses simultan, manajemen kebuntuan, dan akses eksklusif didukung sesuai kebutuhan.
- Layanan atau sistem yang terlibat mengikuti filosofi toleransi kesalahan, di mana kesalahan perangkat keras atau perangkat lunak mengarah pada penghentian layanan yang lancar atau penyelesaian masalah alih-alih crash dan hilangnya informasi secara tiba-tiba. (Penghentian berarti akhir; apakah itu anggun atau tidak, pada akhirnya sistem tidak tersedia. Namun, dengan peringatan yang adil tentang penghentian sistem, pengguna mungkin dapat pindah ke sistem lain dan melanjutkan pekerjaan.)
- Layanan atau sistem dapat digunakan dengan mudah dan sesuai dengan tujuan penggunaannya. (Ini adalah karakteristik kegunaan, tetapi sistem yang tidak dapat digunakan juga dapat menyebabkan kegagalan ketersediaan.)



Gambar 1-7 Ketersediaan dan Aspek Terkait

Seperti yang Anda lihat, ekspektasi ketersediaan jauh jangkauannya. Pada Gambar 1-7 kami menggambarkan beberapa properti yang ketersediaannya tumpang tindih. Memang, komunitas keamanan baru mulai memahami apa yang tersirat dari ketersediaan dan bagaimana memastikannya.

Seseorang atau sistem dapat melakukan tiga hal dasar dengan item data: melihatnya, memodifikasinya, atau menggunakannya. Dengan demikian, melihat (kerahasiaan), memodifikasi (integritas), dan menggunakan (ketersediaan) adalah mode dasar akses yang ingin dipertahankan oleh keamanan komputer.

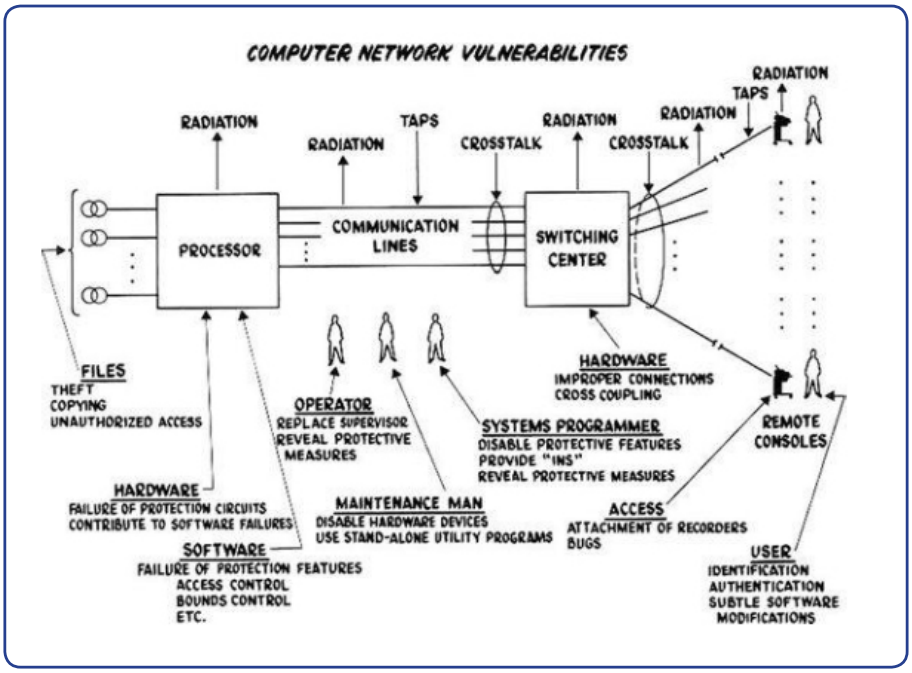
Paradigma keamanan komputer adalah kontrol akses: Untuk menerapkan kebijakan, keamanan komputer mengontrol semua akses oleh semua subjek ke semua objek yang dilindungi dalam semua mode akses. Kontrol akses yang kecil dan terpusat sangat penting untuk menjaga kerahasiaan dan integritas, tetapi tidak jelas apakah satu titik kontrol akses dapat menegakkan ketersediaan. Memang, para ahli ketergantungan akan mencatat bahwa titik kontrol tunggal dapat menjadi titik kegagalan tunggal, sehingga memudahkan penyerang untuk menghancurkan ketersediaan dengan menonaktifkan titik kontrol tunggal. Sebagian besar kesuksesan keamanan komputer di masa lalu berfokus pada kerahasiaan dan integritas; ada model kerahasiaan dan integritas, ketersediaan adalah tantangan besar keamanan berikutnya.

Kami baru saja menjelaskan triad C-I-A dan tiga sifat keamanan mendasar yang diwakilinya. Deskripsi kami tentang properti ini dalam konteks hal-hal yang membutuhkan perlindungan. Untuk memotivasi pemahaman Anda, kami memberikan beberapa contoh bahaya dan ancaman untuk menyebabkan kerugian. Langkah kita selanjutnya adalah memikirkan sifat ancaman itu sendiri.

1.2.4 Jenis Ancaman

Untuk beberapa gagasan tentang bahaya, lihat Gambar 1-8, diambil dari laporan Willis Ware (Ware:70).

Meskipun ditulis ketika komputer begitu besar, sangat mahal, dan sangat sulit dioperasikan sehingga hanya organisasi besar seperti universitas, perusahaan besar, atau departemen pemerintah yang memilikinya, diskusi Ware masih bersifat instruktif hingga saat ini. Ware terutama memperhatikan perlindungan data rahasia, yaitu menjaga kerahasiaan. Dalam gambar, ia menggambarkan manusia seperti programmer dan staf pemeliharaan mendapatkan akses ke data, serta radiasi dimana data dapat melarikan diri sebagai sinyal. Dari gambar tersebut Anda dapat melihat beberapa dari sekian banyak jenis ancaman terhadap sistem komputer.

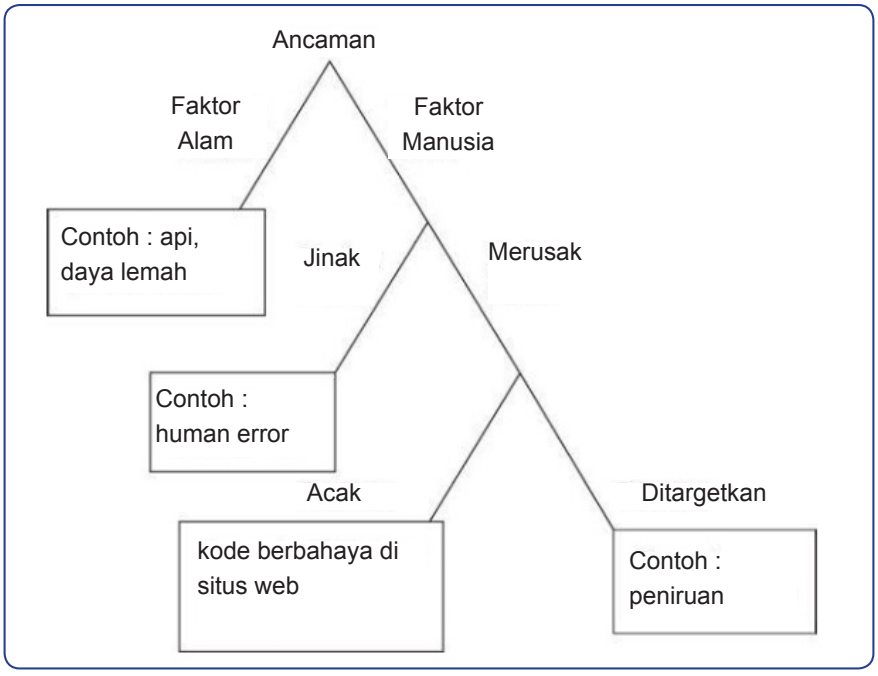


Gambar 1-8 Kerentanan Komputer

Salah satu cara untuk menganalisis bahaya adalah dengan mempertimbangkan penyebab atau sumbernya. Kami menyebut penyebab potensial bahaya sebagai ancaman. Kerugian dapat disebabkan oleh peristiwa bukan manusia atau manusia. Contoh ancaman non-manusia termasuk bencana alam seperti kebakaran atau banjir; kehilangan daya listrik; kegagalan komponen seperti kabel komunikasi, chip prosesor, atau drive disk; atau diserang oleh babi hutan.

Ancaman manusia dapat berupa jinak (nonmalicious) atau berbahaya. Jenis bahaya yang tidak berbahaya termasuk seseorang secara tidak sengaja menumpahkan minuman ringan di laptop, secara tidak sengaja menghapus teks, secara tidak sengaja mengirim pesan email ke orang yang salah, dan dengan ceroboh mengetik “12” bukannya “21” saat memasukkan nomor telepon atau mengklik “ya” dan bukan "tidak" untuk menimpa file. Kesalahan manusia yang tidak disengaja ini terjadi pada kebanyakan orang; kami hanya berharap bahwa keseriusan kerusakannya tidak terlalu besar, atau jika memang demikian, kami tidak akan mengulangi kesalahan itu.

Sebagian besar aktivitas keamanan komputer terkait dengan kerusakan berbahaya yang disebabkan manusia: Ancaman jahat sebenarnya ingin menyebabkan kerusakan, jadi kami sering menggunakan istilah serangan untuk peristiwa keamanan komputer yang berbahaya. Serangan berbahaya dapat dilakukan secara acak atau terarah. Dalam serangan acak, penyerang ingin membahayakan komputer atau pengguna mana pun; serangan seperti itu analog dengan menyapa pejalan kaki berikutnya yang berjalan di jalan. Contoh serangan acak adalah kode berbahaya yang diposting di situs web yang dapat dikunjungi oleh siapa saja.



Gambar 1-9 Jenis Ancaman

Dalam serangan terarah, penyerang bermaksud merusak komputer tertentu, mungkin di satu organisasi (pikirkan serangan terhadap organisasi politik) atau milik individu tertentu (pikirkan mencoba menguras rekening bank orang tertentu, misalnya, dengan peniruan identitas). Kelas serangan terarah lainnya adalah terhadap produk tertentu, seperti komputer mana pun yang menjalankan browser tertentu. (Kami tidak ingin membagi rambut tentang apakah serangan semacam

itu diarahkan—pada satu produk perangkat lunak itu—atau acak, terhadap pengguna produk itu; intinya bukanlah kesempurnaan semantik tetapi perlindungan terhadap serangan.) Rentang kemungkinan diarahkan serangan praktis tidak terbatas. Berbagai jenis ancaman ditunjukkan pada Gambar 1-9.

Kegagalan perangkat keras yang normal dapat tampak seperti serangan yang diarahkan dan berbahaya untuk menolak akses, dan peretas sering kali mencoba menyembunyikan aktivitas mereka agar terlihat seperti pengguna biasa yang berwenang. Sebagai ahli keamanan komputer kita perlu mengantisipasi hal buruk apa yang mungkin terjadi, daripada menunggu serangan terjadi atau memperdebatkan apakah serangan itu disengaja atau tidak disengaja.

Baik buku ini maupun daftar periksa atau metode apa pun tidak dapat menunjukkan kepada Anda semua jenis kerusakan yang dapat terjadi pada aset komputer. Ada terlalu banyak cara untuk mengganggu penggunaan Anda atas aset-aset ini. Namun, dua daftar retrospektif dari kerentanan yang diketahui menarik. *Common Vulnerabilities and Exposures* (CVE) (lihat http://cve.mitre.org/) adalah kamus kerentanan dan eksposur keamanan yang diketahui publik. Pengidentifikasi umum CVE memungkinkan pertukaran data antara produk keamanan dan menyediakan titik indeks dasar untuk mengevaluasi cakupan alat dan layanan keamanan. Untuk mengukur tingkat bahaya, *Common Vulnerability Scoring System* (CVSS) (lihat http://nvd.nist.gov/cvss.cfm) menyediakan sistem pengukuran standar yang memungkinkan penilaian dampak kerentanan yang akurat dan konsisten.

1.2.5 Ancaman Persisten Tingkat Lanjut

Pakar keamanan menjadi semakin khawatir tentang jenis ancaman yang disebut ancaman persisten tingkat lanjut. Penyerang tunggal mungkin membuat serangan acak yang menjerat beberapa, atau beberapa juta, individu, tetapi dampak yang dihasilkan terbatas pada apa yang dapat diatur dan dikelola oleh penyerang tunggal itu. Sekumpulan penyerang—misalnya, tentang kelompok cyber yang setara dengan geng jalanan atau regu kejahatan terorganisir—mungkin bekerja sama untuk mencuri nomor kartu kredit atau aset keuangan serupa untuk mendanai aktivitas ilegal lainnya. Penyerang seperti itu cenderung oportunistik, mengorek kantong korban dan beralih ke aktivitas lain.
Serangan ancaman persisten tingkat lanjut datang dari penyerang yang sabar dan terorganisir dengan dana yang baik. Seringkali berafiliasi dengan pemerintah atau kelompok kuasi-pemerintah, penyerang ini terlibat dalam kampanye jangka panjang. Mereka dengan hati-hati memilih target mereka, menyusun serangan yang secara khusus menarik target tersebut; pesan email yang disebut spear phishing dimaksudkan untuk merayu penerimanya. Biasanya serangan itu diam, menghindari dampak yang jelas yang akan mengingatkan korban, sehingga memungkinkan penyerang untuk mengeksploitasi hak akses korban dalam waktu yang lama.

Motif serangan semacam itu terkadang tidak jelas. Salah satu tujuan populer adalah spionase ekonomi. Serangkaian serangan, yang tampaknya diorganisir dan didukung oleh pemerintah China, digunakan pada 2012 dan 2013 untuk mendapatkan desain produk dari perusahaan kedirgantaraan di Amerika Serikat. Ada bukti rintisan kode serangan dimuat ke mesin korban jauh sebelum serangan; kemudian, penyerang memasang kode yang lebih kompleks dan mengekstrak data yang diinginkan.

Pada Mei 2014, Departemen Kehakiman mendakwa lima peretas China secara in absentia atas serangan ini.

Pada musim panas 2014 serangkaian serangan terhadap bank J.P. Morgan Chase dan hingga selusin lembaga keuangan serupa memungkinkan para penyerang mengakses 76 juta nama, nomor telepon, dan alamat email. Para penyerang—dan bahkan negara asal mereka—tetap tidak diketahui, begitu pula motifnya. Mungkin penyerang menginginkan data keuangan yang lebih sensitif, seperti nomor akun atau kata sandi, tetapi hanya bisa mendapatkan informasi kontak yang kurang berharga. Juga tidak diketahui apakah serangan ini terkait dengan serangan setahun sebelumnya yang mengganggu layanan ke bank itu dan beberapa lainnya.

Untuk membayangkan lanskap penuh dari kemungkinan serangan, Anda mungkin perlu mempertimbangkan jenis orang yang menyerang sistem komputer. Meskipun siapa pun berpotensi menjadi penyerang, kelas orang tertentu menonjol karena latar belakang atau tujuan mereka. Jadi, di bagian berikut kita melihat profil beberapa kelas penyerang.

1.2.6 Jenis Penyerang

Siapa penyerang? Seperti yang telah kita lihat, motivasi mereka berkisar dari kesempatan hingga target tertentu. Mengesampingkan serangan dari penyebab alami dan jinak, kita dapat mengeksplorasi siapa penyerang dan apa yang memotivasi mereka.

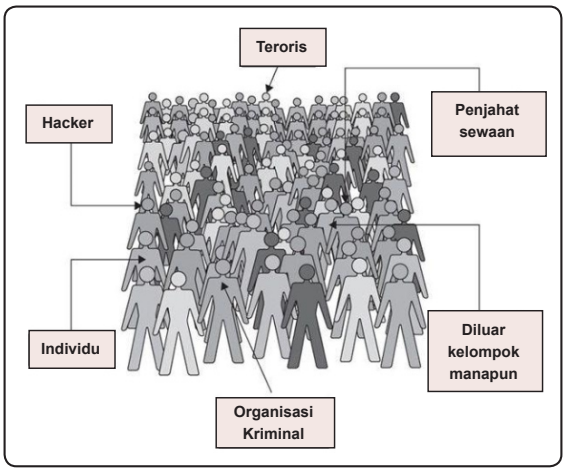
Sebagian besar penelitian tentang penyerang sebenarnya menganalisis penjahat komputer, yaitu orang yang benar-benar telah dihukum karena kejahatan, terutama karena kelompok itu mudah diidentifikasi dan dipelajari. Orang yang lolos atau yang melakukan penyerangan tanpa ketahuan mungkin memiliki ciri-ciri yang berbeda dengan penjahat yang tertangkap. Lebih buruk lagi, dengan hanya mempelajari penjahat yang telah kita tangkap, kita mungkin tidak belajar bagaimana menangkap penyerang yang tahu cara menyalahgunakan sistem tanpa ditangkap.

Seperti apa penjahat dunia maya itu? Di televisi dan film, para penjahat mengenakan pakaian lusuh, tampak kejam dan menyeramkan, dan tinggal di geng di suatu tempat di luar kota. Sebaliknya, sheriff berpakaian bagus, berdiri dengan bangga dan tinggi, dikenal dan dihormati oleh semua orang di kota, dan menimbulkan ketakutan di hati sebagian besar penjahat.

Yang pasti, beberapa penjahat komputer adalah tipe yang kejam dan jahat. Tetapi lebih banyak lagi yang mengenakan setelan bisnis, memiliki gelar sarjana, dan tampak menjadi pilar komunitas mereka. Ada yang masih SMA atau mahasiswa. Lainnya adalah eksekutif bisnis paruh baya. Beberapa secara mental gila, sangat bermusuhan, atau sangat berkomitmen untuk suatu tujuan, dan mereka menyerang komputer sebagai simbol. Lainnya adalah orang biasa yang tergoda oleh keuntungan pribadi, balas dendam, tantangan, kemajuan, atau keamanan pekerjaan—seperti pelaku kejahatan apa pun, menggunakan komputer atau tidak.

Para peneliti telah mencoba menemukan ciri-ciri psikologis yang membedakan penyerang, seperti yang dijelaskan di Studi Kasus 1-1. Studi-studi ini jauh dari konklusif, bagaimanapun, dan ciri-ciri yang mereka identifikasi mungkin menunjukkan korelasi tetapi belum tentu kausalitas. Untuk menghargai poin ini, misalkan sebuah penelitian menemukan bahwa jumlah orang yang dihukum karena kejahatan komputer adalah kidal. Apakah hasil itu menyiratkan bahwa semua orang kidal

adalah penjahat komputer atau hanya orang kidal? Tentu tidak. Tidak ada satu profil pun yang menangkap karakteristik penyerang komputer "tipikal", dan karakteristik beberapa penyerang terkenal juga cocok dengan banyak orang yang bukan penyerang. Seperti yang ditunjukkan pada Gambar 1-10, penyerang terlihat seperti orang lain di tengah kerumunan.



Gambar 1-10 Penyerang

Studi Kasus 1-1 : Profil Psikologis Penyerang

Temple Grandin, seorang profesor ilmu hewan di Colorado State University dan seorang penderita gangguan mental yang disebut *Asperger Syndrome* (AS), berpikir bahwa Kevin Mitnick dan beberapa peretas lainnya yang digambarkan secara luas menunjukkan gejala klasik sindrom Asperger. Meskipun dengan cepat menunjukkan bahwa tidak ada penelitian yang menemukan hubungan antara AS dan peretasan, Grandin mencatat ciri-ciri perilaku serupa di antara Mitnick, dirinya sendiri, dan penderita AS lainnya. Sebuah artikel di USA Today (29 Maret 2001) mencantumkan ciri-ciri AS berikut:

- keterampilan sosial yang buruk, sering dikaitkan dengan menjadi penyendiri selama masa kanak-kanak; "kutu buku komputer" klasik
- gelisah, gelisah, tidak mampu melakukan kontak mata, kurang merespon isyarat dalam interaksi sosial, seperti ekspresi wajah atau bahasa tubuh
- kemampuan luar biasa untuk mengingat deretan angka yang panjang
- kemampuan untuk fokus pada masalah teknis secara intens dan untuk waktu yang lama, meskipun mudah teralihkan pada masalah lain dan tidak mampu mengelola beberapa tugas sekaligus
- kejujuran yang mendalam dan rasa hormat terhadap hukum

Donn Parker telah mempelajari peretasan dan kejahatan komputer untuk banyak orang selama bertahun-tahun. Dia menyatakan “peretas dicirikan oleh sikap yang tidak dewasa dan terlalu idealis … Mereka senang menampilkan diri mereka sendiri ke media sebagai orang yang baik dan idealis, juara dari yang diunggulkan.”

Pertimbangkan kutipan berikut dari wawancara dengan “Mixter,” programmer Jerman yang mengakui bahwa dia adalah penulis perangkat lunak penyerang yang tersebar luas yang disebut Tribal Flood Network (TFN) dan sekuelnya TFN2K:

T : Mengapa Anda menulis perangkat lunak?
J : Saya pertama kali mendengar tentang Trin00 [perangkat lunak penyerang lainnya] pada Juli '99 dan saya menganggapnya menarik dari perspektif teknis, tetapi juga berpotensi kuat secara negatif. Saya mengetahui beberapa fakta tentang cara kerja Trin00, dan karena saya tidak berhasil mendapatkan sumber atau binari Trin00 pada waktu itu, saya menulis jaringan klien-server saya sendiri yang mampu melakukan penolakan layanan.
T : Apakah Anda terlibat … dalam salah satu serangan profil tinggi baru-baru ini?
J : Tidak. Fakta bahwa saya menulis alat ini sama sekali tidak berarti bahwa saya memaafkan penggunaan aktifnya. Saya harus mengakui bahwa saya cukup terkejut mendengar tentang serangan terbaru. Tampaknya para penyerang adalah orang-orang yang tidak tahu apa-apa yang menyalahgunakan sumber daya dan alat yang kuat untuk kegiatan yang umumnya berbahaya dan tidak masuk akal hanya "karena mereka bisa."

Perhatikan bahwa dari beberapa informasi tentang serangan penolakan layanan, dia menulis jaringan klien-servernya sendiri dan kemudian serangan yang canggih. Tapi dia "cukup terkejut" mendengar mereka digunakan untuk menyakiti.

Diperlukan lebih banyak penelitian sebelum kita dapat menentukan profil seorang peretas. Dan bahkan lebih banyak pekerjaan akan diperlukan untuk memperluas profil itu ke profil penyerang (berbahaya). Tidak semua peretas menjadi penyerang; beberapa peretas menjadi administrator sistem, pengembang, atau pakar keamanan yang sangat berdedikasi dan teliti. Tetapi beberapa psikolog melihat di AS dasar-dasar profil peretas.

Perorangan

Awalnya, penyerang komputer adalah individu, bertindak dengan motif kesenangan, tantangan, atau balas dendam. Penyerang awal bertindak sendiri. Dua yang paling terkenal di antara mereka adalah Robert Morris Jr., mahasiswa pascasarjana Cornell University yang meruntuhkan Internet pada tahun 1988, dan Kevin Mitnick, pria yang membobol dan mencuri data dari puluhan komputer, termasuk San Pusat Superkomputer Diego.

Di Indonesia sendiri, hacker sudah ada sejak abad 20 saat tanah air menjadi ladang subur perkembangan internet. Beberapa kelompok hacker di Indonesia saat itu cukup banyak, di antaranya hackerlink, anti-hackerlink, kecoa elektronik, dan echo.

Hacker di Indonesia mencapai masa keemasan pada kisaran tahun 2000, yaitu AntiHackerlink. Pihaknya mampu membobol puluhan situs internet kala itu baik dari dalam dan luar negeri. Uniknya, pendiri dari Antihackerlink ini adalah seorang anak yang belum genap berumur 17 tahun bernama Wenas Agustiawan yang biasa dikenal dengan nama hC (hantu Crew)

Kelompok Lintas Negara yang Terorganisir

Serangan yang lebih baru telah melibatkan sekelompok orang. Serangan terhadap pemerintah negara diyakini merupakan ledakan yang tidak terkoordinasi dari federasi penyerang dari seluruh dunia. Kevin Poulsen mengutip Tim Rosenberg, seorang profesor riset di Universitas George Washington, memperingatkan "kelompok peretas multinasional yang didukung oleh kejahatan terorganisir" dan menunjukkan kecanggihan mafia era larangan. Dia juga melaporkan bahwa Christopher Painter, wakil direktur kejahatan komputer Departemen Kehakiman AS, berpendapat bahwa penjahat cyber dan artis penipuan serius semakin bekerja dalam konser atau satu dan sama. Menurut Painter, kelompok penjahat yang terhubung secara longgar di seluruh dunia bekerja sama untuk membobol sistem dan mencuri serta menjual informasi, seperti nomor kartu kredit. Misalnya, pada Oktober 2004, otoritas AS dan Kanada menangkap 28 orang dari 6 negara yang terlibat dalam jaringan kejahatan dunia maya yang terorganisir untuk membeli dan menjual informasi dan identitas kartu kredit.

Sedangkan motif awal penyerang komputer seperti Morris dan Mitnick bersifat pribadi, seperti prestise atau prestasi, serangan baru-baru ini sangat dipengaruhi oleh keuntungan finansial. Perusahaan keamanan McAfee melaporkan “Penjahat telah menyadari keuntungan finansial besar yang bisa diperoleh dari Internet dengan sedikit risiko. Mereka membawa keterampilan, pengetahuan, dan koneksi yang dibutuhkan untuk skala besar, perusahaan kriminal bernilai tinggi yang, bila dikombinasikan dengan keterampilan komputer, memperluas cakupan dan risiko kejahatan dunia maya.”

Kejahatan Terorganisir

Sasaran penyerang termasuk penipuan, pemerasan, pencucian uang, dan perdagangan narkoba, area di mana kejahatan terorganisir memiliki kehadiran yang mapan. Bukti berkembang bahwa kelompok kejahatan terorganisir terlibat dalam kejahatan komputer. Faktanya, penjahat tradisional merekrut peretas untuk bergabung dengan dunia kejahatan dunia maya yang menguntungkan. Misalnya, Albert Gonzales dijatuhi hukuman 20 tahun penjara pada Maret 2010 karena bekerja dengan jaringan kejahatan untuk mencuri 40 juta nomor kartu kredit dari pengecer TJMaxx dan lainnya, dengan biaya lebih dari $200 juta (Reuters, 26 Maret 2010).

Kejahatan terorganisir dapat menggunakan kejahatan komputer (seperti mencuri nomor kartu kredit atau rincian rekening bank) untuk membiayai aspek kejahatan lainnya. Serangan baru-baru ini menunjukkan bahwa penjahat profesional telah menemukan betapa menguntungkannya kejahatan komputer. Mike Danseglio, seorang manajer proyek keamanan di Microsoft, mengatakan, “Pada tahun 2006, para penyerang ingin membayar sewa. Mereka tidak ingin menulis worm yang menghancurkan perangkat keras Anda. Mereka ingin mengasimilasi komputer Anda dan menggunakannya untuk menghasilkan uang.” Mikko Hyppönen, Chief Research Officer perusahaan keamanan Finlandia f-Secure, setuju bahwa serangan hari ini sering datang dari Rusia, Asia, dan Brasil; motifnya sekarang untung, bukan ketenaran. Ken Dunham, Direktur Tim Respon Cepat untuk VeriSign mengatakan bahwa dia “yakin bahwa kelompok-kelompok mafia yang terorganisir dengan baik telah menguasai jaringan kejahatan global bernilai miliaran dolar yang didukung oleh peretas yang terampil.”

McAfee juga menjelaskan Studi Kasus hacker-for-hire: seorang pengusaha yang menyewa seorang hacker New Jersey berusia 16 tahun untuk menyerang situs web pesaingnya. Peretas menyerang situs tersebut selama periode lima bulan dan merusak tidak hanya perusahaan target tetapi juga

penyedia layanan Internet (ISP) mereka dan perusahaan lain yang tidak terkait yang menggunakan ISP yang sama. Menurut perkiraan FBI, serangan itu merugikan semua perusahaan lebih dari $2 juta; FBI menangkap hacker dan pengusaha pada Maret 2005.

Brian Snow (Snow:05) mengamati bahwa peretas menginginkan skor atau semacam bukti untuk memberi mereka hak membual. Kejahatan terorganisir menginginkan sumber daya; penjahat tersebut ingin tetap berada di bawah radar untuk dapat mengambil keuntungan dari sistem dari waktu ke waktu. Tujuan berbeda ini mengarah pada pendekatan yang berbeda untuk kejahatan komputer: Peretas pemula dapat menggunakan serangan kasar, sedangkan penyerang profesional menginginkan metode yang rapi, kuat, dan tidak terdeteksi yang dapat memberikan hadiah untuk waktu yang lama.

Teroris

Hubungan antara keamanan komputer dan terorisme cukup jelas. Kami melihat teroris menggunakan komputer dalam empat cara:

- Komputer sebagai target serangan: Serangan *Denial-of-Service* dan perusakan situs web adalah kegiatan populer untuk organisasi politik mana pun karena mereka menarik perhatian pada penyebabnya dan membawa perhatian negatif yang tidak diinginkan ke objek serangan.
- Komputer sebagai metode serangan: Meluncurkan serangan ofensif membutuhkan penggunaan komputer. Stuxnet, contoh kode komputer berbahaya yang disebut worm, diketahui menyerang sistem kontrol otomatis, khususnya model sistem kontrol yang diproduksi oleh Siemens. Para ahli mengatakan kode tersebut dirancang untuk menonaktifkan mesin yang digunakan dalam pengendalian reaktor nuklir di Iran. Orang-orang di balik serangan itu tidak diketahui, tetapi infeksi diyakini telah menyebar melalui USB flash drive yang dibawa oleh para insinyur yang memelihara pengontrol komputer.
- Komputer sebagai pemicu serangan: Situs web, log web, dan daftar email adalah cara yang efektif, cepat, dan murah untuk memungkinkan banyak orang berkoordinasi. Menurut Dewan Hubungan Luar Negeri, teroris yang bertanggung jawab atas serangan November 2008 yang menewaskan lebih dari 200 orang di Mumbai menggunakan sistem GPS untuk memandu kapal mereka, Blackberry untuk komunikasi mereka, dan Google Earth untuk merencanakan rute mereka.
- Komputer sebagai penambah serangan: Internet telah terbukti menjadi sarana yang sangat berharga bagi teroris untuk menyebarkan propaganda dan merekrut agen. Pada bulan Oktober 2009 FBI menangkap Colleen LaRose, juga dikenal sebagai JihadJane, setelah dia menghabiskan waktu berbulan-bulan menggunakan email, YouTube, MySpace, dan papan pesan elektronik untuk merekrut radikal di Eropa dan Asia Selatan untuk “melakukan jihad kekerasan,” menurut dakwaan federal.

Kita tidak dapat mengukur secara akurat sejauh mana teroris menggunakan komputer, karena teroris merahasiakan sifat kegiatan mereka dan karena definisi dan alat pengukuran kami agak lemah. Namun, insiden seperti yang dijelaskan di Studi Kasus 1-2 memberikan bukti bahwa keempat aktivitas ini meningkat.

Studi Kasus 1-2 : The Terrorists, Inc., Departemen TI

Pada tahun 2001, seorang reporter untuk Wall Street Journal membeli komputer bekas di Afghanistan. Sangat mengejutkan, ia menemukan bahwa hard drive berisi apa yang tampak seperti file dari seorang agen senior Al Qaeda. Reporter, Alan Cullison, melaporkan bahwa dia menyerahkan komputer itu ke FBI. Dalam ceritanya yang diterbitkan pada tahun 2004 di The Atlantic, dia dengan hati-hati menghindari mengungkapkan apa pun yang dia pikir mungkin sensitif. Disk berisi lebih dari 1.000 dokumen, banyak di antaranya dienkripsi dengan enkripsi yang relatif lemah. Cullison menemukan rancangan rencana misi dan kertas putih yang mengemukakan argumen ideologis dan filosofis untuk serangan 11 September 2001. Juga ditemukan salinan berita tentang kegiatan teroris. Beberapa dokumen yang ditemukan menunjukkan bahwa al Qaeda pada awalnya tidak tertarik pada senjata kimia, biologi, atau nuklir, tetapi menjadi tertarik setelah membaca artikel berita publik yang menuduh al Qaeda memiliki kemampuan tersebut.

Mungkin yang paling tak terduga adalah pesan email dari jenis yang akan ditemukan di kantor biasa: rekomendasi untuk promosi, pembenaran untuk pengeluaran kas kecil, dan argumen tentang anggaran.

Komputer tersebut tampaknya telah digunakan oleh al Qaeda dari tahun 1999 hingga 2001. Cullison mencatat bahwa Afghanistan pada akhir tahun 2001 adalah tempat kekacauan, dan kemungkinan pemilik laptop melarikan diri dengan cepat, meninggalkan komputernya, di mana ia jatuh ke tangan seorang pedagang barang bekas yang tidak tahu isinya.

Tetapi isi komputer ini menggambarkan aspek penting dari keamanan dan kerahasiaan komputer: Kita tidak pernah bisa memprediksi waktu di mana bencana keamanan akan terjadi, dan dengan demikian kita harus selalu siap untuk bertindak segera jika itu terjadi secara tiba-tiba.

Jika seseorang di televisi bersin, Anda tidak perlu khawatir tentang kemungkinan terkena flu. Tetapi jika seseorang yang berdiri di sebelah Anda bersin, bisa jadi Anda menjadi khawatir. Pada bagian selanjutnya kita akan membahas bahaya yang dapat timbul dari adanya ancaman keamanan komputer pada sistem komputer Anda sendiri.

1.3 Kerugian

Konsekuensi negatif dari ancaman yang diaktualisasikan adalah kerugian (*Harm*); kita melindungi diri kita dari ancaman untuk mengurangi atau menghilangkan bahaya. Kami telah menjelaskan banyak contoh kerusakan komputer: komputer yang dicuri, file yang diubah atau hilang, surat pribadi yang terungkap, atau akses yang ditolak ke data. Peristiwa ini menyebabkan kerugian yang ingin kita hindari.

Dalam diskusi kami sebelumnya tentang aset, kami mencatat bahwa nilai bergantung pada persepsi dan kebutuhan pemilik atau orang luar. Beberapa aspek nilai tidak dapat diukur, seperti nilai makalah yang harus Anda serahkan kepada profesor Anda besok; jika Anda kehilangan kertas (yaitu, jika ketersediaannya hilang), tidak ada jumlah uang yang akan menggantinya. Barang-barang yang Anda nilai sedikit atau tidak sama sekali mungkin lebih berharga bagi orang lain; misalnya, foto grup yang diambil di pesta semalam dapat mengungkapkan bahwa teman Anda tidak berada di tempat yang dia katakan kepada istrinya. Meskipun mungkin sulit untuk menetapkan angka tertentu sebagai nilai suatu aset, Anda biasanya dapat menetapkan nilai pada skala umum, seperti sedang atau sangat kecil atau sangat tinggi, tergantung pada tingkat kerugian yang diakibatkan oleh kerugian atau kerusakan tersebut. objek akan menyebabkan. Atau Anda dapat menetapkan nilai relatif terhadap aset lain, berdasarkan kerugian yang sebanding: Versi file ini lebih berharga bagi Anda daripada versi itu.

Dalam laporan ancaman Internet global 2010, perusahaan keamanan Symantec mensurvei jenis barang dan jasa yang ditawarkan untuk dijual di halaman web bawah tanah. Item yang paling sering ditawarkan pada tahun 2009 dan 2008 adalah nomor kartu kredit, dengan harga mulai dari $0,85 hingga $30,00 masing-masing. (Bandingkan harga-harga tersebut dengan upaya individu untuk menangani efek sebauh pencurian kartu kredit atau jumlah potensial yang hilang oleh bank penerbit.) Kedua yang paling sering adalah kredensial rekening bank, pada $15 sampai $850; ini ditawarkan untuk dijual di 19% situs web di kedua tahun. Akun email berikutnya di $1 sampai $20, dan daftar alamat email pergi untuk $1.70 sampai $15.00 per seribu. Pada posisi 10 pada tahun 2009 adalah kredensial administrasi situs web, dengan biaya hanya $2 hingga $30. Situs web pasar gelap ini menunjukkan bahwa harga pasar aset komputer dapat sangat berbeda dari nilainya bagi pemilik yang sah.

Nilai dari banyak aset dapat berubah dari waktu ke waktu, sehingga tingkat kerusakan (dan karena itu tingkat keparahan ancaman) juga dapat berubah. Dengan waktu, uang, dan kemampuan yang tidak terbatas, kita mungkin mencoba melindungi dari segala macam bahaya. Tetapi karena sumber daya kita terbatas, kita harus memprioritaskan perlindungan kita, menjaga hanya dari ancaman serius dan yang dapat kita kendalikan. Memilih ancaman yang kami coba mitigasi melibatkan proses yang disebut manajemen risiko, dan itu termasuk menimbang keseriusan ancaman terhadap kemampuan kami untuk melindungi.

1.3.1 Risiko dan Berpikir Praktis

Jumlah dan jenis ancaman praktis tidak terbatas karena merancang serangan membutuhkan imajinasi aktif, tekad, ketekunan, dan waktu (serta akses dan sumber daya). Sifat dan jumlah ancaman di dunia komputer mencerminkan kehidupan secara umum: Penyebab kerusakan tidak terbatas dan sebagian besar tidak dapat diprediksi. Bencana alam seperti gunung berapi dan gempa bumi terjadi dengan sedikit atau tanpa peringatan, seperti halnya kecelakaan mobil, serangan jantung, influenza, dan tindakan kekerasan secara acak. Untuk melindungi dari kecelakaan atau flu, Anda mungkin memutuskan untuk tinggal di dalam rumah, tidak pernah berkeliaran di luar. Tetapi dengan melakukan itu, Anda menukar satu set risiko dengan risiko lainnya; saat Anda berada di dalam, Anda rentan terhadap keruntuhan bangunan. Ada terlalu banyak kemungkinan penyebab kerusakan bagi kita untuk melindungi diri kita sendiri—atau komputer kita—sepenuhnya terhadap semua itu.

Dalam kehidupan nyata, kami membuat keputusan setiap hari tentang cara terbaik untuk memberikan keamanan kami. Misalnya, meskipun kita mungkin memilih untuk tinggal di daerah yang tidak rawan gempa, kita tidak bisa sepenuhnya menghilangkan risiko gempa. Beberapa pilihan dilakukan secara sadar, seperti memutuskan untuk tidak berjalan di gang gelap di lingkungan yang tidak aman; di lain waktu alam bawah sadar kita membimbing kita, dari pengalaman atau keahlian, untuk mengambil tindakan pencegahan. Kami mengevaluasi kemungkinan dan tingkat keparahan bahaya, dan kemudian mempertimbangkan cara (disebut penanggulangan atau kontrol) untuk mengatasi ancaman dan menentukan efektivitas kontrol.

Keamanan komputer serupa. Karena kami tidak dapat melindungi dari segalanya, kami memprioritaskan: Hanya begitu banyak waktu, energi, atau uang yang tersedia untuk perlindungan, jadi kami mengatasi beberapa risiko dan membiarkan yang lain tergelincir. Atau kami mempertimbangkan tindakan alternatif, seperti mentransfer risiko dengan membeli asuransi atau bahkan tidak melakukan apa-apa jika efek samping dari tindakan pencegahan bisa lebih buruk daripada kemungkinan kerugiannya. Risiko yang tetap tidak tercakup oleh pengendalian disebut risiko residual.

Model dasar manajemen risiko melibatkan pengguna menghitung nilai semua aset, menentukan jumlah bahaya dari semua kemungkinan ancaman, menghitung biaya perlindungan, memilih pengamanan (yaitu, kontrol atau penanggulangan) berdasarkan tingkat risiko dan sumber daya yang terbatas, dan menerapkan pengamanan untuk mengoptimalkan bahaya yang dihindari. Pendekatan manajemen risiko ini adalah pendekatan yang logis dan masuk akal untuk perlindungan, tetapi memiliki kelemahan yang signifikan. Pada kenyataannya, sulit untuk menilai nilai setiap aset; seperti yang telah kita lihat, nilai dapat berubah tergantung pada konteks, waktu, dan sejumlah karakteristik lainnya. Yang lebih sulit lagi adalah menentukan dampak dari semua kemungkinan ancaman. Jangkauan kemungkinan ancaman secara efektif tidak terbatas, dan sulit (jika bukan tidak mungkin dalam beberapa situasi) untuk mengetahui dampak jangka pendek dan jangka panjang dari suatu tindakan. Misalnya, Studi Kasus 1-3 menjelaskan studi tentang dampak pelanggaran keamanan dari waktu ke waktu pada keuangan perusahaan, menunjukkan bahwa ancaman harus dievaluasi dari waktu ke waktu, tidak hanya pada satu contoh.

Sudah lama diasumsikan bahwa pelanggaran keamanan akan berdampak buruk bagi bisnis: bahwa pelanggan, yang takut kehilangan data mereka, akan menyimpang dari bisnis yang tidak aman dan menuju bisnis yang lebih aman. Tetapi studi empiris menunjukkan bahwa gambarannya lebih rumit. Studi awal tentang efek pelanggaran keamanan, seperti yang dilakukan Campbell, meneliti efek pelanggaran pada harga saham. Mereka menemukan bahwa dampak pelanggaran dapat bergantung pada sifat pelanggaran itu sendiri; efeknya lebih tinggi ketika pelanggaran melibatkan akses tidak sah ke data rahasia. Cavusoglu dkk. menemukan bahwa pelanggaran mempengaruhi nilai tidak hanya perusahaan yang mengalami pelanggaran tetapi juga perusahaan keamanan: Rata-rata, perusahaan yang dilanggar kehilangan 2,1 persen dari nilai pasar dalam dua hari setelah pengungkapan pelanggaran, tetapi nilai pasar pengembang keamanan justru meningkat 1,36 persen.

Myung Ko dan Carlos Dorantes melihat dampak keuangan jangka panjang dari pelanggaran yang diumumkan secara publik. Berdasarkan Campbell et al. studi, mereka memeriksa data selama empat kuartal setelah pengumuman akses tidak sah ke data rahasia. Ko dan Dorantes mencatat banyak jenis kemungkinan biaya terkait pelanggaran:

“Contoh biaya jangka pendek antara lain biaya perbaikan, biaya penggantian sistem, kehilangan bisnis karena terganggunya operasional bisnis, dan hilangnya produktivitas karyawan. Ini juga dianggap sebagai biaya nyata. Di sisi lain, biaya jangka panjang termasuk kehilangan pelanggan yang sudah ada karena kehilangan kepercayaan, gagal menarik pelanggan potensial di masa depan karena reputasi negatif dari pelanggaran, kehilangan mitra bisnis karena kehilangan kepercayaan, dan potensi kewajiban hukum dari pelanggaran. Sebagian besar biaya ini adalah biaya tidak berwujud yang sulit untuk dihitung tetapi sangat penting dalam menilai keseluruhan biaya pelanggaran keamanan bagi organisasi.”

Ko dan Dorantes membandingkan dua kelompok perusahaan: satu set (kelompok perlakuan) dengan pelanggaran data, dan yang lain (kelompok kontrol) tanpa pelanggaran tetapi cocok untuk ukuran dan industri. Temuan mereka sangat mengejutkan. Bertentangan dengan apa yang Anda duga, perusahaan yang dilanggar tidak mengalami penurunan kinerja untuk kuartal setelah pelanggaran, tetapi laba atas aset mereka menurun pada kuartal ketiga. Perbandingan perlakuan dengan perusahaan kontrol mengungkapkan bahwa perusahaan kontrol umumnya mengungguli perusahaan yang dilanggar. Namun, perusahaan yang dilanggar mengungguli perusahaan kontrol pada kuartal keempat.

Hasil ini sesuai dengan hasil peneliti lain yang menyimpulkan bahwa ada dampak ekonomi jangka panjang yang minimal dari pelanggaran keamanan. Ada banyak alasan mengapa demikian. Misalnya, pelanggan mungkin berpikir bahwa semua perusahaan yang bersaing memiliki kerentanan dan ancaman yang sama, sehingga

berpindah ke vendor lain tidak mengurangi risiko. Penjelasan lain yang mungkin adalah persepsi bahwa perusahaan yang dilanggar memiliki keamanan yang lebih baik karena pelanggaran tersebut memaksa perusahaan untuk memperkuat kontrol dan dengan demikian mengurangi kemungkinan pelanggaran serupa. Namun penjelasan lain mungkin hanya rentang perhatian pelanggan yang pendek; seiring berjalannya waktu, pelanggan melupakan pelanggaran dan kembali ke bisnis seperti biasa.

Semua penelitian ini memiliki keterbatasan, termasuk ukuran sampel yang kecil dan kurangnya data yang memadai. Tetapi mereka dengan jelas menunjukkan kesulitan dalam mengukur dan memverifikasi dampak risiko keamanan, dan menunjukkan perbedaan antara efek jangka pendek dan jangka panjang.

Meskipun kita tidak boleh menerapkan perlindungan secara sembarangan, kita tentu akan melindungi dari ancaman yang kita anggap paling mungkin atau paling merusak. Untuk alasan ini, penting untuk memahami bagaimana kita memandang ancaman dan mengevaluasi kemungkinan terjadinya dan dampaknya. Studi Kasus 1-4 merangkum beberapa penelitian yang relevan dalam persepsi risiko dan pengambilan keputusan. Penelitian tersebut menunjukkan bahwa, untuk Studi Kasus yang relatif jarang seperti masalah keamanan berdampak tinggi, kita harus mempertimbangkan cara orang lebih fokus pada dampak daripada kemungkinan kejadian yang sebenarnya.

Studi Kasus 1-4 : Persepsi Risiko Kejadian Ekstrim

Ketika suatu jenis peristiwa yang merugikan sering terjadi, kita dapat menghitung kemungkinan dan dampaknya dengan memeriksa frekuensi dan sifat dari rangkaian peristiwa kolektif. Misalnya, kita dapat menghitung kemungkinan hujan minggu ini dan menebak jumlah inci curah hujan yang akan kita terima; hujan cukup sering terjadi. Tetapi masalah keamanan sering kali merupakan peristiwa yang ekstrem: jarang terjadi dan dalam berbagai keadaan, sehingga sulit untuk melihatnya sebagai sebuah kelompok dan menarik kesimpulan umum. Karya Paul Slovic tentang risiko membahas kesulitan khusus dengan peristiwa ekstrem. Dia menunjukkan bahwa mengevaluasi risiko dalam Studi Kasus-Studi Kasus seperti itu dapat menjadi upaya politik dan juga ilmiah. Dia mencatat bahwa kita cenderung membiarkan nilai, proses, kekuatan, dan kepercayaan memengaruhi analisis risiko kita.

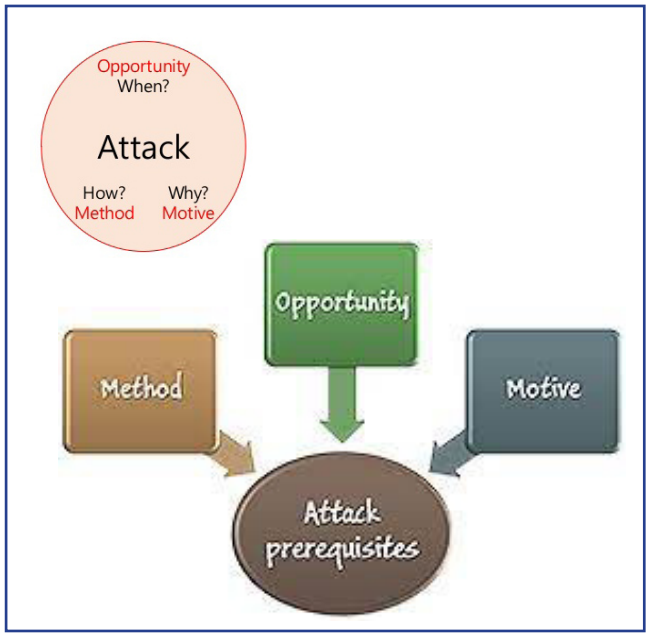
Dimulai dengan Fischoff et al., para peneliti mengkarakterisasi risiko ekstrem di sepanjang dua sumbu berbasis persepsi: ketakutan akan risiko dan sejauh mana risiko itu tidak diketahui. Perasaan tentang risiko ini, yang disebut pengaruh oleh psikolog, memungkinkan peneliti untuk mendiskusikan risiko relatif dengan menempatkannya pada bidang yang ditentukan oleh dua persepsi sebagai sumbu. Sebuah studi oleh Loewenstein et al. menjelaskan bagaimana persepsi risiko dipengaruhi oleh asosiasi (dengan peristiwa yang sudah dialami) dan oleh pengaruh setidaknya jika tidak lebih dari alasan. Bahkan, jika dua pengaruh bersaing, perasaan biasanya mengalahkan akal.

Karakteristik analisis risiko ini diperkuat oleh teori prospek: studi tentang bagaimana orang membuat keputusan dengan menggunakan akal dan perasaan. Kahneman dan Tversky menunjukkan bahwa orang cenderung melebih-lebihkan kemungkinan kejadian langka dan tidak berpengalaman karena perasaan takut dan ketidaktahuan mereka biasanya mendominasi penalaran analitis tentang kemungkinan kejadian yang rendah. Sebaliknya, jika orang mengalami hasil dan kemungkinan yang sama, perasaan takut mereka berkurang dan mereka benar-benar dapat meremehkan kejadian langka. Dengan kata lain, jika dampak dari kejadian langka tinggi (ketakutan tinggi), maka orang-orang fokus pada dampaknya, terlepas dari kemungkinannya. Tetapi jika dampak dari kejadian langka kecil, maka mereka memperhatikan kemungkinannya.

Mari kita lihat lebih hati-hati sifat ancaman keamanan. Kita telah melihat bahwa satu aspek—potensi bahayanya—adalah jumlah kerusakan yang dapat ditimbulkannya; aspek ini merupakan komponen dampak dari risiko. Kami juga mempertimbangkan besarnya kemungkinan ancaman. Ancaman yang mungkin terjadi bukan hanya ancaman yang mungkin ingin dilakukan seseorang, melainkan ancaman yang benar-benar dapat terjadi. Beberapa orang mungkin melamun tentang menjadi kaya dengan merampok bank; sebagian besar, bagaimanapun, akan menolak gagasan itu karena kesulitannya (jika bukan amoralitas atau risikonya). Salah satu aspek kemungkinan adalah kelayakan: Apakah mungkin untuk menyelesaikan serangan? Jika jawabannya tidak, maka kemungkinannya adalah nol, dan oleh karena itu juga risikonya. Jadi tempat yang baik untuk memulai dalam menilai risiko adalah dengan melihat apakah tindakan yang diusulkan layak dilakukan. Tiga faktor menentukan kelayakan, seperti yang kami jelaskan selanjutnya.

2.3.2 Metode – Peluang–Motif

Seorang hacker harus memiliki tiga hal untuk memastikan keberhasilan serangan jahatnya: metode, peluang, dan motif, yang digambarkan pada Gambar 1-11. Secara kasar, metode adalah caranya; peluang adalah kapan; dan motif, mengapa menyerang. Dengan menolak penyerang dengan salah satu dari ketiganya dan serangan tidak akan berhasil. Mari kita bahas properti ini satu per satu.



Gambar 1-11 Metode–Peluang–Motif

Metode (*Method*)

Yang kami maksud dengan metode adalah keterampilan, pengetahuan, alat, dan hal-hal lain yang digunakan untuk melakukan serangan. Pikirkan tokoh komik yang ingin melakukan sesuatu, misalnya, untuk mencuri perhiasan berharga, tetapi karakternya sangat tidak kompeten sehingga setiap gerakannya

pasti akan gagal. Orang-orang ini tidak memiliki kemampuan atau metode untuk berhasil, sebagian karena tidak ada kelas dalam pencurian permata atau buku tentang perampokan untuk boneka. Namun, siapa pun dapat menemukan banyak kursus dan buku tentang komputasi. Pengetahuan tentang model tertentu dari sistem komputer tersedia secara luas di toko buku dan di Internet. Sistem pasar massal (seperti sistem operasi Microsoft atau Apple atau Unix) sudah tersedia untuk dibeli, seperti juga produk perangkat lunak umum, seperti pengolah kata atau sistem manajemen basis data, sehingga penyerang potensial bahkan dapat memperoleh perangkat keras dan perangkat lunak untuk bereksperimen dan menyempurnakan serangan.

Beberapa produsen merilis spesifikasi terperinci tentang bagaimana sistem dirancang atau bagaimana sistem itu beroperasi, sebagai panduan bagi pengguna dan integrator yang ingin menerapkan produk pelengkap lainnya. Berbagai alat serangan—skrip, program model, dan alat untuk menguji kelemahan—tersedia dari situs peretas di Internet, sampai-sampai banyak serangan hanya membutuhkan kemampuan penyerang untuk mengunduh dan menjalankan program. Istilah script kiddie menggambarkan seseorang yang mengunduh paket kode serangan lengkap dan hanya perlu memasukkan beberapa detail untuk mengidentifikasi target dan membiarkan skrip melakukan serangan. Seringkali, hanya waktu dan kecenderungan yang membatasi penyerang.

Kesempatan (*Opportunity*)

Opportunity adalah waktu dan akses untuk melakukan serangan. Anda mendengar bahwa apartemen yang luar biasa baru saja dibangun dan mulai dipasarkan, jadi Anda bergegas ke agen persewaan, hanya untuk menemukan orang lain menyewanya lima menit sebelumnya. Anda melewatkan kesempatan Anda.

Banyak sistem komputer memberikan banyak kesempatan untuk menyerang. Sistem yang tersedia untuk publik, menurut definisi, dapat diakses; seringkali pemiliknya sangat berhati-hati untuk membuatnya tersedia sepenuhnya sehingga jika salah satu komponen perangkat keras gagal, pemilik memiliki suku cadang yang langsung siap untuk digunakan. Orang lain tidak menyadari kebutuhan untuk melindungi komputer mereka, sehingga laptop tanpa pengawasan dan koneksi jaringan yang tidak aman memberikan banyak peluang untuk diserang. Beberapa sistem memiliki titik masuk pribadi atau tidak berdokumen untuk administrasi atau pemeliharaan, tetapi penyerang juga dapat menemukan dan menggunakan titik masuk tersebut untuk menyerang sistem.

Motif

Terakhir, seorang penyerang harus memiliki motif atau alasan ingin menyerang. Anda mungkin memiliki banyak kesempatan dan kemampuan untuk melempar batu melalui jendela tetangga Anda, tetapi Anda tidak melakukannya. Kenapa tidak? Karena Anda tidak punya alasan untuk ingin menyakiti tetangga Anda: Anda tidak memiliki motif.

Kami telah menjelaskan beberapa motif kejahatan komputer: uang, ketenaran, harga diri, politik, teror. Seringkali sulit untuk menentukan motif serangan. Beberapa tempat adalah "target yang menarik", yang berarti mereka sangat menarik bagi penyerang. Target populer termasuk penegakan hukum dan komputer departemen pertahanan, mungkin karena mereka dianggap terlindungi dengan baik terhadap serangan (sehingga mereka menghadirkan tantangan dan serangan yang berhasil menunjukkan kehebatan penyerang). Sistem lain diserang karena mudah diserang. Dan beberapa sistem diserang secara acak hanya karena mereka ada di sana.

Dengan menunjukkan kelayakan, faktor metode, peluang, dan motif menentukan apakah suatu serangan dapat berhasil. Faktor-faktor ini memberikan keuntungan bagi penyerang karena merupakan kualitas atau kekuatan yang harus dimiliki penyerang. Faktor lain, kali ini memberikan keuntungan bagi pembela, menentukan apakah serangan akan berhasil: Penyerang membutuhkan kerentanan, tempat yang tidak dipertahankan untuk menyerang. Jika bek menghilangkan kerentanan, penyerang tidak bisa menyerang.

1.4 Kerentanan

Seperti yang kita catat sebelumnya dalam bab ini, kerentanan adalah kelemahan dalam keamanan sistem komputer, misalnya, dalam prosedur, desain, atau implementasi, yang mungkin dieksploitasi untuk menyebabkan kerugian atau bahaya. Pikirkan sebuah bank, dengan penjaga bersenjata di pintu depan, kaca antipeluru yang melindungi teller, dan brankas logam berat yang membutuhkan banyak kunci untuk masuk. Untuk merampok bank, Anda harus memikirkan cara mengeksploitasi kelemahan yang tidak tercakup oleh pertahanan ini. Misalnya, Anda mungkin menyuap teller atau menyamar sebagai pekerja pemeliharaan.

Sistem komputer juga memiliki kerentanan. Dalam buku ini kami mempertimbangkan banyak hal, seperti otentikasi yang lemah, kurangnya kontrol akses, kesalahan dalam program, sumber daya yang terbatas atau tidak mencukupi, dan perlindungan fisik yang tidak memadai. Dipasangkan dengan serangan yang kredibel, masing-masing kerentanan ini dapat membahayakan kerahasiaan, integritas, atau ketersediaan. Setiap vektor serangan berusaha untuk mengeksploitasi kerentanan tertentu.

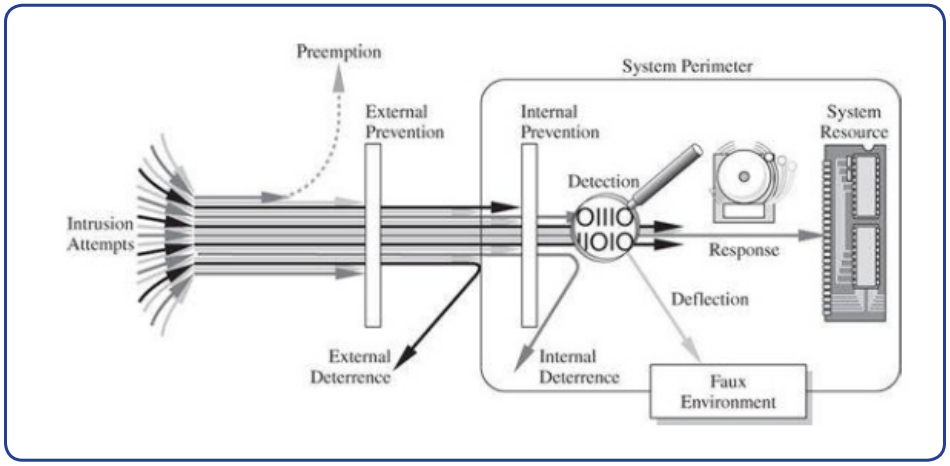
Analis keamanan berbicara tentang permukaan serangan sistem, yang merupakan rangkaian lengkap kerentanan sistem—aktual dan potensial. Dengan demikian, permukaan serangan mencakup bahaya fisik, serangan jahat oleh orang luar, pencurian data sembunyi-sembunyi oleh orang dalam, kesalahan, dan peniruan identitas. Meskipun serangan semacam itu berkisar dari yang mudah hingga yang sangat tidak mungkin, analis harus mempertimbangkan semua kemungkinan. Langkah kami selanjutnya adalah menemukan cara untuk memblokir ancaman dengan menetralkan kerentanan.

1.5 Kontrol

Kontrol atau penanggulangan adalah sarana untuk melawan ancaman. Harm terjadi ketika ancaman diwujudkan terhadap kerentanan. Untuk melindungi dari bahaya, maka, kita dapat menetralisir ancaman, menutup kerentanan, atau keduanya. Kemungkinan terjadinya kerugian disebut risiko. Kita dapat menangani kerusakan dengan beberapa cara:

- mencegahnya, dengan memblokir serangan atau menutup kerentanan
- mencegahnya, dengan membuat serangan lebih keras tetapi bukan tidak mungkin
- menangkisnya, dengan membuat target lain lebih menarik (atau yang ini kurang menarik)
- menguranginya, dengan membuat dampaknya tidak terlalu parah
- mendeteksinya, baik saat itu terjadi atau beberapa saat setelah fakta
- pulih dari efeknya

Tentu saja, lebih dari satu kontrol ini dapat digunakan secara bersamaan. Jadi, misalnya, kami mungkin mencoba mencegah penyusupan—tetapi jika kami curiga kami tidak dapat mencegah semuanya, kami mungkin juga memasang perangkat pendeteksi untuk memperingatkan serangan yang akan segera terjadi. Dan kita harus memiliki prosedur tanggap insiden untuk membantu pemulihan jika penyusupan berhasil.



Gambar 1-12 Pengaruh Kontrol

Untuk mempertimbangkan kontrol atau tindakan pencegahan yang berupaya mencegah eksploitasi kerentanan sistem komputasi, kita mulai dengan memikirkan cara tradisional untuk meningkatkan keamanan fisik. Pada Abad Pertengahan, kastil dan benteng dibangun untuk melindungi orang-orang dan properti berharga di dalamnya.

Benteng mungkin memiliki satu atau lebih karakteristik keamanan, termasuk

- gerbang atau pintu yang kuat untuk mengusir penjajah
- dinding berat untuk menahan benda yang dilempar atau diproyeksikan ke sana
- parit di sekitarnya untuk mengontrol akses
- celah panah untuk membiarkan pemanah menembak musuh yang mendekat
- crenellations untuk memungkinkan penghuni bersandar keluar dari atap dan menuangkan cairan panas atau keji ke penyerang
- jembatan gantung untuk membatasi akses ke orang yang berwenang
- portcullis untuk membatasi akses di luar jembatan gantung
- penjaga gerbang untuk memverifikasi bahwa hanya orang dan barang yang berwenang yang bisa masuk

Demikian pula, hari ini kami menggunakan pendekatan multi-cabang untuk melindungi rumah dan kantor kami. Kami dapat menggabungkan kunci kuat di pintu dengan alarm pencuri, jendela yang diperkuat, dan bahkan tetangga yang usil untuk mengawasi barang-barang berharga kami. Dalam setiap Studi Kasus, kami memilih satu atau lebih cara untuk mencegah penyusup atau penyerang, dan kami mendasarkan pilihan kami tidak hanya pada nilai dari apa yang kami lindungi tetapi juga pada upaya yang kami pikir akan dilakukan penyerang atau penyusup untuk masuk ke dalam.

Keamanan komputer memiliki karakteristik yang sama. Kami memiliki banyak kontrol yang dimiliki. Beberapa lebih mudah digunakan atau diimplementasikan daripada yang lain. Beberapa lebih murah daripada yang lain untuk digunakan atau diimplementasikan. Dan beberapa lebih sulit daripada yang lain untuk ditimpa oleh penyusup. Gambar 1-12 mengilustrasikan bagaimana kami menggunakan kombinasi kontrol untuk mengamankan sumber daya kami yang berharga. Kami menggunakan satu atau lebih kontrol, sesuai dengan apa yang kami lindungi, bagaimana biaya perlindungan dibandingkan dengan risiko kerugian, dan seberapa keras kami berpikir penyusup akan bekerja untuk mendapatkan apa yang mereka inginkan.

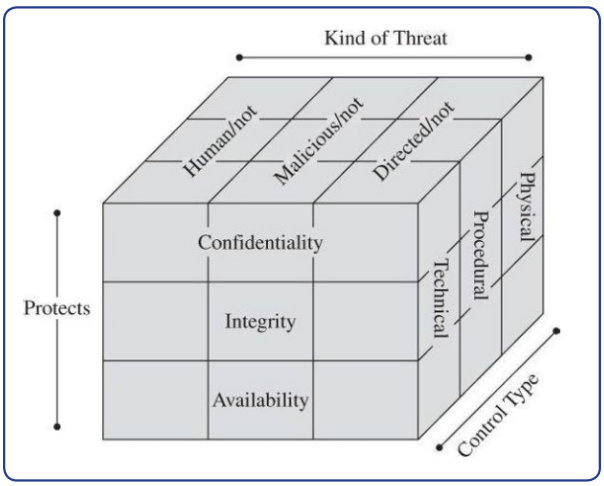
Di bagian ini, akan disajikan ikhtisar kontrol yang tersedia. Di sisa buku ini, kita akan membahas cara menggunakan kontrol terhadap jenis ancaman tertentu.

Kita dapat mengelompokkan kontrol ke dalam tiga kelas yang sebagian besar independen. Daftar berikut menunjukkan kelas dan beberapa contoh dari setiap jenis kontrol.

- Kontrol fisik menghentikan atau memblokir serangan dengan menggunakan sesuatu yang nyata juga, seperti tembok dan pagar
  - kunci
  - (manusia) penjaga
  - alat penyiram dan alat pemadam kebakaran lainnya
- Kontrol prosedural atau administratif menggunakan perintah atau persetujuan yang
  - membutuhkan atau menasihati orang bagaimana harus bertindak; sebagai contoh,
  - hukum, peraturan
  - kebijakan, prosedur, pedoman
  - hak cipta, paten
  - kontrak, perjanjian
- Kontrol teknis melawan ancaman dengan teknologi (perangkat keras atau perangkat lunak), termasuk
  - kata sandi
  - kontrol akses program atau sistem operasi
  - protokol jaringan
  - firewall, sistem deteksi intrusi
  - enkripsi
  - pengatur arus lalu lintas jaringan

(Perhatikan bahwa istilah "kontrol logis" juga digunakan, tetapi beberapa orang menggunakannya untuk berarti kontrol administratif, sedangkan yang lain menggunakannya untuk berarti kontrol teknis. Untuk menghindari kebingungan, kami tidak menggunakan istilah itu.)

Seperti yang ditunjukkan pada Gambar 1-13, Anda dapat memikirkan properti yang akan dilindungi dan jenis ancaman ketika Anda memilih jenis tindakan pencegahan yang tepat. Tak satu pun dari kelas-kelas ini selalu lebih baik dari atau lebih baik dari yang lain; mereka bekerja dengan cara yang berbeda dengan hasil yang berbeda. Dan bisa efektif untuk menggunakan kontrol yang tumpang tindih atau pertahanan secara mendalam: lebih dari satu kontrol atau lebih dari satu kelas kontrol untuk mencapai perlindungan.



Gambar 1-13 Jenis Penanggulangan

1.6 Kesimpulan

Keamanan komputer berusaha untuk memastikan kerahasiaan, integritas, dan ketersediaan sistem komputasi dan komponennya. Tiga bagian utama dari sistem komputasi tunduk pada serangan: perangkat keras, perangkat lunak, dan data. Ketiganya, dan komunikasi di antara mereka, rentan terhadap kerentanan keamanan komputer. Pada gilirannya, orang-orang dan sistem yang tertarik untuk mengkompromikan sistem dapat merancang serangan yang mengeksploitasi kerentanan.

Dalam bab ini kami telah menjelaskan konsep keamanan komputer berikut:

- Situasi keamanan muncul dalam banyak aktivitas sehari-hari, meskipun terkadang sulit untuk membedakan antara serangan keamanan dan gangguan manusia atau teknologi biasa. Sayangnya, penyerang yang cerdik menyadari kebingungan ini, sehingga mereka mungkin membuat serangan mereka tampak seperti kegagalan acak yang sederhana.
- Ancaman adalah kejadian yang dapat menimbulkan kerugian. Kerentanan adalah kelemahan di mana kerusakan dapat terjadi. Kedua masalah ini digabungkan: Tanpa yang lain tidak menyebabkan kerugian, tetapi ancaman yang menggunakan kerentanan berarti kerusakan. Untuk mengendalikan situasi seperti itu, kita dapat memblokir atau mengurangi ancaman, atau menutup kerentanan (atau keduanya).
- Jarang sekali kita dapat mencapai keamanan yang sempurna: tidak ada ancaman yang dapat dijalankan dan tidak ada kerentanan yang dapat dijalankan. Terkadang kita gagal mengenali ancaman, atau di lain waktu kita mungkin tidak mampu atau tidak mau menutup kerentanan. Keamanan yang tidak lengkap bukanlah situasi yang buruk; sebaliknya, ini menunjukkan tindakan penyeimbang: Mengontrol ancaman dan kerentanan tertentu, menerapkan tindakan pencegahan yang masuk akal, dan menerima risiko kerugian dari Studi Kasus yang tidak dapat diatasi.
- Seorang penyerang membutuhkan tiga hal: metode—keterampilan dan pengetahuan untuk melakukan serangan yang berhasil; kesempatan—waktu dan akses untuk menyerang; dan motif—Alasan untuk ingin menyerang. Sayangnya, tidak satu pun dari ketiganya kekurangan pasokan, yang berarti serangan tidak dapat dihindari.

Dalam bab ini kami telah memperkenalkan gagasan tentang ancaman dan bahaya, kerentanan, serangan dan penyerang, dan penanggulangan. Penyerang memanfaatkan ancaman yang mengeksploitasi kerentanan terhadap aset berharga untuk menyebabkan kerugian, dan kami berharap dapat merancang tindakan balasan untuk menghilangkan sarana, peluang, dan motif. Konsep-konsep inilah yang menjadi dasar kita perlu mempelajari, memahami, dan menguasai keamanan komputer.

Penanggulangan dan kontrol dapat diterapkan pada data, program, sistem, perangkat fisik, hubungan komunikasi, lingkungan, dan personel. Terkadang beberapa kontrol diperlukan untuk menutupi satu kerentanan, tetapi terkadang satu kontrol mengatasi banyak masalah sekaligus.

Bahan Evaluasi

1. Jelaskan perbedaan antara kerentanan, ancaman, dan kontrol.
2. Pencurian biasanya mengakibatkan beberapa jenis kerusakan. Misalnya, jika seseorang mencuri mobil Anda, Anda mungkin menderita kerugian finansial, ketidaknyamanan (karena kehilangan moda transportasi), dan gangguan emosional (karena invasi properti dan ruang pribadi Anda). Sebutkan tiga jenis kerugian yang mungkin dialami perusahaan dari pencurian peralatan komputer.
3. Sebutkan setidaknya tiga jenis kerugian yang dapat dialami perusahaan dari spionase elektronik atau melihat materi rahasia perusahaan secara tidak sah.
4. Sebutkan setidaknya tiga jenis kerusakan yang dapat diderita perusahaan ketika integritas program atau data perusahaan dikompromikan.
5. Sebutkan setidaknya tiga jenis kerugian yang dapat dihadapi perusahaan dari hilangnya layanan, yaitu kegagalan ketersediaan. Buat daftar produk atau kemampuan yang aksesnya hilang, dan jelaskan bagaimana kerugian ini merugikan perusahaan.
6. Jelaskan situasi di mana Anda mengalami kerugian sebagai akibat dari kegagalan keamanan komputer. Apakah kegagalan itu berbahaya atau tidak? Apakah serangan itu menargetkan Anda secara khusus atau umum dan Anda adalah korban yang malang?
7. Jelaskan dua contoh kerentanan dalam mobil yang kontrolnya telah ditetapkan oleh pabrikan mobil. Jelaskan mengapa menurut Anda pengendalian ini efektif, agak efektif, atau tidak efektif.
8. Salah satu kontrol terhadap penghapusan perangkat lunak yang tidak disengaja adalah dengan menyimpan semua versi lama dari suatu program. Tentu saja, kontrol ini sangat mahal dalam hal biaya penyimpanan. Sarankan kontrol yang lebih murah terhadap penghapusan perangkat lunak yang tidak disengaja. Apakah kontrol Anda efektif terhadap semua kemungkinan penyebab penghapusan perangkat lunak? Jika tidak, ancaman apa yang tidak tercakup?
9. Di komputer pribadi Anda, siapa yang dapat menginstal program? Siapa yang dapat mengubah data sistem operasi? Siapa yang dapat menggantikan bagian dari sistem operasi? Bisakah salah satu dari tindakan ini dilakukan dari jarak jauh?
10. Misalkan sebuah program untuk mencetak gaji secara diam-diam membocorkan daftar nama karyawan yang berpenghasilan lebih dari jumlah tertentu setiap bulannya. Kontrol apa yang dapat diterapkan untuk membatasi kerentanan kebocoran ini?

11. Menjaga kerahasiaan, integritas, dan ketersediaan data merupakan pernyataan kembali kekhawatiran atas interupsi, intersepsi, modifikasi, dan fabrikasi. Bagaimana tiga konsep pertama berhubungan dengan empat konsep terakhir? Artinya, apakah salah satu dari empat itu setara dengan satu atau lebih dari ketiganya? Apakah salah satu dari tiga dicakup oleh satu atau lebih dari empat?
12. Apakah menurut Anda mencoba masuk ke (yaitu, mendapatkan akses ke atau menggunakan) sistem komputasi tanpa otorisasi harus ilegal? Mengapa atau mengapa tidak??
13. Jelaskan contoh (selain yang disebutkan dalam bab ini) data yang kerahasiaannya memiliki ketepatan waktu yang singkat, katakanlah, sehari atau kurang. Jelaskan contoh data yang kerahasiaannya memiliki ketepatan waktu lebih dari satu tahun.
14. Apakah saat ini Anda menggunakan tindakan pengendalian keamanan komputer? Jika demikian, apa? Terhadap serangan apa yang Anda coba lindungi?
15. Jelaskan sebuah contoh di mana penolakan mutlak layanan kepada pengguna (yaitu, pengguna tidak mendapat tanggapan dari komputer) merupakan masalah serius bagi pengguna tersebut. Jelaskan contoh lain di mana 10 persen penolakan layanan kepada pengguna (yaitu, komputasi pengguna berlangsung, tetapi pada tingkat 10 persen lebih lambat dari biasanya) merupakan masalah serius bagi pengguna tersebut. Bisakah akses oleh orang yang tidak berwenang ke sistem komputasi menghasilkan penolakan layanan 10 persen kepada pengguna yang sah? Bagaimana?
16. Ketika Anda mengatakan bahwa perangkat lunak berkualitas tinggi, apa maksud Anda? Bagaimana keamanan sesuai dengan definisi kualitas Anda? Misalnya, dapatkah suatu aplikasi tidak aman dan tetap “baik”?
17. Pengembang sering memikirkan kualitas perangkat lunak dalam hal kesalahan dan kegagalan. Kesalahan adalah masalah (misalnya, perulangan yang tidak pernah berhenti atau salah menempatkan koma dalam pernyataan) yang dapat dilihat oleh pengembang dengan melihat kodenya. Kegagalan adalah masalah, seperti sistem crash atau permintaan fungsi yang salah, yang terlihat oleh pengguna. Dengan demikian, kesalahan bisa ada dalam program tetapi tidak pernah menjadi kegagalan, karena kondisi di mana kesalahan menjadi kegagalan tidak pernah tercapai. Bagaimana kerentanan perangkat lunak cocok dengan skema kesalahan dan kegagalan ini? Apakah setiap kesalahan adalah kerentanan? Apakah setiap kerentanan adalah kesalahan?
18. Pertimbangkan sebuah program untuk menampilkan waktu dan suhu kota Anda saat ini di situs web Anda. Siapa yang mungkin ingin menyerang program Anda? Jenis kerusakan apa yang mungkin ingin mereka timbulkan? Jenis kerentanan apa yang mungkin mereka eksploitasi untuk menyebabkan kerusakan?
19. Pertimbangkan sebuah program yang memungkinkan konsumen memesan produk dari web. Siapa yang mungkin ingin menyerang program? Jenis kerusakan apa yang mungkin ingin mereka timbulkan? Jenis kerentanan apa yang mungkin mereka eksploitasi untuk menyebabkan kerusakan?
20. Pertimbangkan program untuk menerima dan mentabulasi suara dalam pemilihan. Siapa yang mungkin ingin menyerang program? Jenis kerusakan apa yang mungkin ingin mereka timbulkan? Jenis kerentanan apa yang mungkin mereka eksploitasi untuk menyebabkan kerusakan?

Bab 2

Otentikasi, Kontrol Akses, dan Kriptografi

Bab ini akan membahas tentang:

- Otentikasi, kemampuan, dan keterbatasan
- Tiga dasar otentikasi: pengetahuan, karakteristik, kepemilikan
- Kekuatan mekanisme otentikasi
- Implementasi kontrol akses
- Menggunakan enkripsi
- Enkripsi simetris dan asimetris
- Message Digest
- Tanda tangan digital dan sertifikat

Dalam keamanan sistem komputer yang perlu kita lakukan adalah untuk mempersulit orang lain mengganggu sistem yang kita pakai, baik kita menggunakan komputer yang sifatnya sendiri, jaringan local maupun jaringan global. Harus dipastikan system bisa berjalan dengan baik dan kondusif, selain itu program aplikasinya masih bisa dipakai tanpa ada masalah.

Pada saat computer diperkenalkan pertama kali, ukuran komputer sangat besar, langka, dan sangat mahal. Oleh karena itu organisasi atau perusahaan yang cukup beruntung memiliki komputer akan mencoba dengan cara terbaik untuk melindungi computer tersebut. Keamanan komputer hanya salah satu aspek dari keamanan secara keseluruhan dari asset organisasi. Keamanan difokuskan pada fisik pembobolan, pencurian peralatan komputer, dan pencurian atau perusakan kemasan disk, gulungan pita, dan media lainnya. Hanya sedikit orang yang tahu bagaimana menggunakan komputer, dan dengan demikian pengguna harus dengan hati-hati dipilih. Pada saat itu computer tidak terhubung dengan jaringan internet sehingga memang masalah keamanan hanya berfokus pada fisik dan lingkungannya saja.

Dalam beberapa hal, keamanan tidak berubah sejak makhluk hidup mulai mengumpulkan hal-hal yang layak dilindungi. Pemilik sistem menetapkan kebijakan keamanan, secara formal atau informal, eksplisit atau implisit — sesederhana semboyan "*no one is allowed to take my food*" (tak seorang boleh mengambil makanan saya) — dan mulai mengambil tindakan untuk menegakkan kebijakan

itu. Karakter ancaman berubah ketika sang penjaga keamanan mulai bertranformasi dari cara tradisional yang kuno dengan mengukiti perkembangan kemajuan IT atau Internet, seperti halnya sifat perlindungan yang tersedia, tetapi esensi strategis mereka sebagian besar tetap konstan: Seorang penyerang menginginkan sesuatu yang dimiliki oleh sistem keamanan, jadi penyerang mengejarnya. Penjaga keamanan memiliki sejumlah opsi — bertarung, membangun penghalang atau sistem alarm, berlari dan bersembunyi, mengurangi daya tarik target terhadap penyerang — dan semua opsi ini memiliki analog dalam keamanan komputer modern. Spesifiknya berubah, tetapi garis besarnya tetap sama.

Dalam bab ini, kami meletakkan dasar untuk keamanan komputer dengan mempelajari goresan luas tersebut. Kami melihat sejumlah strategi keamanan di mana-mana, mengidentifikasi ancaman terhadap mana masing-masing strategi tersebut efektif, dan memberikan contoh tindakan pencegahan yang representatif. Sepanjang sisa buku ini, saat kami mempelajari langkah-langkah keamanan teknis khusus yang digunakan dalam sistem operasi, pemrograman, situs web dan browser, dan jaringan, kami meninjau kembali strategi yang sama ini berulang kali. Bertahun-tahun dari sekarang, ketika kita semua menggunakan teknologi yang bahkan belum pernah dibayangkan, bab ini harus relevan seperti sekarang ini.

Seorang profesional keamanan menganalisis situasi dengan menemukan ancaman dan kerentanan terhadap kerahasiaan, integritas, dan/atau ketersediaan sistem komputasi. Untuk engendalikan ancaman dan kerentanan ini melibatkan kebijakan yang menentukan siapa (subjek mana) dapat mengakses apa (objek mana) bagaimana (dengan cara apa). Agar efektif, penegakan kebijakan harus menentukan siapa secara akurat. Artinya, jika kebijakan mengatakan bahwa A dapat mengakses sesuatu, keamanan gagal jika orang lain meniru A. Jadi, untuk menegakkan kebijakan keamanan dengan benar, kami memerlukan cara untuk menentukan tanpa keraguan bahwa identitas subjek itu akurat. Properti identifikasi yang akurat disebut ***otentikasi***. Alat penting pertama untuk profesional keamanan adalah otentikasi dan teknik serta teknologinya.

Ketika pertama kali memperkenalkan kebijakan keamanan, tidak secara eksplisit menyatakan sebaliknya: Subjek diizinkan untuk mengakses objek dalam mode tertentu tetapi, kecuali diizinkan, semua subjek lain tidak diizinkan untuk mengakses objek. Kebijakan tanpa batasan seperti itu praktis menjadi sia-sia belaka. Apa gunanya mengatakan satu subjek dapat mengakses objek jika subjek lain dapat melakukannya tanpa diizinkan oleh kebijakan. Akibatnya, kita memerlukan cara untuk membatasi akses hanya pada subjek-subjek dalam daftarnya, seperti mengizinkan pengunjung teater untuk bermain (dengan tiket) atau memasukkan undangan ke pesta (pada daftar tamu). Seseorang atau sesuatu mengontrol akses, misalnya, seorang penerima tamu mengumpulkan tiket atau tuan rumah mengelola daftar tamu. Mengizinkan secara tepat akses yang diotorisasi disebut kontrol akses. Mekanisme untuk menerapkan kontrol akses adalah alat keamanan komputer mendasar lainnya.

Misalkan Anda mencoba membatasi akses ke pertandingan sepak bola yang diadakan di taman terbuka di kota berpenduduk padat. Tanpa pagar, gerbang, atau parit, Anda tidak bisa membatasi siapa yang bisa melihat permainan. Tapi misalkan Anda memiliki kekuatan super dan bisa menyelubungi para pemain dengan seragam tembus pandang. Anda akan mengeluarkan kacamata khusus hanya untuk orang yang diizinkan untuk melihat pertandingan; orang lain mungkin melihat tetapi tidak melihat apa-apa. Meskipun skenario ini murni fantasi, teknologi tembus pandang seperti itu memang ada, yang disebut enkripsi. Sederhananya, enkripsi adalah alat yang dengannya kita dapat mengubah

data sehingga hanya penerima yang dituju (yang memiliki kunci, setara dengan kacamata anti-penyelubungan) yang dapat menyimpulkan bit yang disembunyikan. Alat keamanan mendasar ketiga dan terakhir dalam bab ini adalah enkripsi.

Dalam bab ini kami menjelaskan alat-alat ini dan kemudian memberikan beberapa contoh untuk membantu memahami cara kerja alat-alat tersebut. Tetapi sebagian besar aplikasi dari alat-alat ini datang di bab-bab selanjutnya, di mana kami menguraikan penggunaannya dalam konteks situasi keamanan yang lebih lengkap. Di sebagian besar bab ini, kami membedah tiga alat keamanan sebagai dasar utama : otentikasi, kontrol akses, dan enkripsi.

2.1 Otentikasi

Kepercayaan dan pengakuan telah menjadi pokok hubungan manusia sejak awal waktu, memaksa manusia purba untuk menciptakan cara mengidentifikasi satu sama lain melalui penggunaan tanda tangan, fitur wajah, nama, dan baru-baru ini melalui penggunaan dokumen seperti identifikasi resmi dan paspor. Konsep otentikasi telah menjadi sangat rumit dengan munculnya dan penggunaan internet dalam kehidupan sehari-hari. Administrator duduk di belakang layar, tidak dapat memverifikasi identitas pengguna yang tidak terlihat dengan melihat, nama, atau tanda tangan – sekarang mereka menggunakan teknologi untuk melindungi jaringan mereka dari mereka yang ingin melewati metode otentikasi dengan maksud jahat. Mari kita lihat beberapa informasi tentang autentikasi digital untuk mempelajari apa yang berhasil dan apa yang tidak, dan bagaimana autentikasi perlu berkembang untuk mendukung masa depan online kita.

Komputer telah menggantikan banyak interaksi tatap muka dengan sistem elektronik. Tanpa ada pihak lain yang waspada untuk menyadari bahwa ada sesuatu yang salah, orang membutuhkan mekanisme lain untuk memisahkan pihak yang berwenang dari pihak yang tidak berwenang. Untuk alasan ini, dasar keamanan komputer adalah akses terkontrol: seseorang berwenang untuk mengambil tindakan atas sesuatu. Kita memeriksa kontrol akses nanti dalam bab ini. Tetapi agar kontrol akses berfungsi, kita perlu yakin siapa "seseorang" itu. Di bagian ini kami memperkenalkan otentikasi, proses memastikan atau mengkonfirmasi identitas.

Otentikasi adalah aspek cybersecurity yang paling dikenal untuk sebagian besar pengguna, ditemui dan digunakan beberapa kali per hari di berbagai perangkat dan layanan. Ini pada dasarnya tentang memastikan bahwa orang yang tepat memiliki akses, dengan memeriksa bahwa mereka adalah siapa yang mereka klaim. Akibatnya, otentikasi mewakili aspek kunci dari keamanan, karena biasanya mewakili garis depan perlindungan yang berdiri antara sistem dan calon penipu.

Proses otentikasi dalam konteks sistem komputer berarti jaminan dan konfirmasi identitas pengguna. Sebelum pengguna mencoba mengakses informasi yang tersimpan di jaringan, dia harus membuktikan identitas dan izin mereka untuk mengakses data. Saat masuk ke jaringan , pengguna harus memberikan informasi masuk unik termasuk nama pengguna dan kata sandi, praktik yang dirancang untuk melindungi jaringan dari penyusupan oleh peretas. Otentikasi telah berkembang lebih jauh dalam beberapa tahun terakhir untuk memerlukan lebih banyak informasi pribadi pengguna, misalnya, biometrik, untuk memastikan keamanan akun dan jaringan dari mereka yang memiliki keterampilan teknis untuk memanfaatkannya.

Sebuah sistem komputer tidak memiliki isyarat yang kita lakukan dengan komunikasi tatap muka yang memungkinkan kita mengenali teman-teman kita. Sebaliknya komputer bergantung pada data untuk mengenali orang lain. Menentukan siapa seseorang sebenarnya terdiri dari dua langkah terpisah:

- Identifikasi adalah tindakan untuk menegaskan siapa seseorang.
- Otentikasi adalah tindakan membuktikan identitas yang ditegaskan: bahwa orang tersebut adalah siapa yang dia katakan.

Kami telah mengungkapkan langkah-langkah ini dari sudut pandang seseorang yang ingin dikenali, menggunakan istilah "orang" untuk kesederhanaan. Bahkan, pengakuan tersebut terjadi antara orang-orang, proses komputer (pelaksana program), koneksi jaringan, perangkat, dan entitas aktif serupa. Dalam keamanan, semua entitas ini disebut subjek.

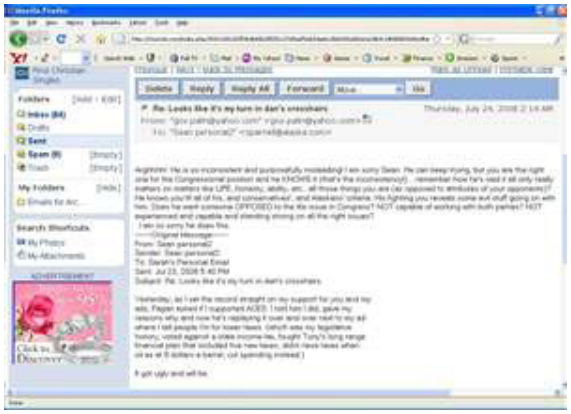
Dua konsep identifikasi dan otentikasi mudah dan sering membingungkan. Identitas, seperti nama, sering kali dikenal publik, dan tidak dilindungi. Di sisi lain, otentikasi harus dilindungi. Jika identitas seseorang bersifat publik, siapa pun dapat mengklaim sebagai orang itu. Apa yang membedakan orang yang berpura-pura dari orang yang sebenarnya adalah bukti dengan otentikasi.

2.1.1 Identifikasi Versus Otentikasi

Identitas sering dikenal, dapat diprediksi, atau dapat ditebak. Jika Anda mengirim email ke seseorang, Anda secara implisit mengirimkan ID akun email Anda sehingga orang lain dapat membalas Anda. Dalam diskusi online, Anda dapat memposting komentar di bawah nama layar sebagai cara untuk menautkan berbagai postingan Anda. Nomor rekening bank Anda tercetak pada cek yang Anda tulis; nomor rekening kartu debit Anda ditampilkan di kartu Anda, dan seterusnya. Dalam setiap Studi Kasus ini Anda mengungkapkan sebagian dari identitas Anda. Perhatikan bahwa identitas Anda lebih dari sekadar nama Anda: Nomor rekening bank, nomor kartu debit, alamat email, dan hal-hal lain adalah cara orang dan proses mengidentifikasi Anda.
Beberapa ID akun tidak sulit ditebak. Beberapa tempat menetapkan ID pengguna sebagai nama belakang pengguna diikuti dengan inisial depan. Yang lain menggunakan tiga inisial atau skema lain yang dapat diprediksi dengan mudah oleh orang luar. Seringkali untuk transaksi online ID akun Anda adalah alamat email Anda, untuk memudahkan Anda mengingatnya. Akun lain mengidentifikasi Anda melalui telepon, jaminan sosial, atau nomor identitas lainnya. Dengan terlalu banyak akun untuk diingat, Anda mungkin menyambut tempat-tempat yang mengidentifikasi Anda dengan sesuatu yang Anda ketahui dengan baik karena Anda sering menggunakannya. Tetapi menggunakannya sering juga berarti orang lain dapat mengetahui atau menebaknya juga. Karena alasan ini, banyak orang dapat dengan mudah, meskipun salah, mengaku sebagai Anda dengan menunjukkan salah satu pengenal Anda yang diketahui.

Otentikasi, di sisi lain, harus dapat diandalkan. Jika identifikasi menegaskan identitas Anda, otentikasi menegaskan bahwa Anda adalah siapa yang Anda maksudkan. Meskipun pengidentifikasi dapat diketahui secara luas atau mudah ditentukan, otentikasi harus bersifat pribadi. Namun, jika proses otentikasi tidak cukup kuat, itu tidak akan aman. Pertimbangkan, misalnya, bagaimana email seorang kandidat politik disusupi oleh proses autentikasi yang tidak cukup pribadi seperti yang dijelaskan di Studi Kasus 2-1.

Studi Kasus 2-1 : Email Calon Wakil Presiden Sarah Palin Terungkap



Salah satu screenshot akun Palin

Selama kampanye presiden AS 2008, akun email pribadi calon wakil presiden Sarah Palin diretas. Isi pesan email dan daftar kontak Palin diposting di papan buletin publik. Seorang mahasiswa Universitas Tennessee berusia 20 tahun, David Kernell, kemudian dihukum karena akses tidak sah untuk mendapatkan informasi dari komputernya dan dijatuhi hukuman satu tahun satu hari.

Bagaimana mungkin seorang mahasiswa mengakses komputer pejabat publik terkenal yang pada saat itu adalah gubernur Alaska dan calon wakil presiden AS di bawah perlindungan Dinas Rahasia AS? Mudah: Dia hanya berpura-pura menjadi dirinya. Tapi tentunya tidak ada seorang pun (selain, mungkin, komedian Tina Fey) yang berhasil menyamar sebagai dirinya. Berikut adalah betapa mudahnya serangan itu.

Akun email Gubernur Palin adalah gov.palin@yahoo.com. ID akun terkenal karena laporan berita tentang insiden sebelumnya yang melibatkan Palin menggunakan akun pribadinya untuk komunikasi resmi negara; bahkan tanpa publisitas nama akun tidak akan sulit ditebak.

Tapi kata sandinya? Tidak, siswa tidak menebak kata sandinya. Yang harus dia lakukan hanyalah berpura-pura menjadi Palin dan mengklaim bahwa dia lupa kata sandinya. Yahoo menanyakan Kernell pertanyaan keamanan yang diajukan Palin ke Yahoo saat membuka akun: tanggal lahir (ditemukan dari Wikipedia), kode pos (pengetahuan publik, terutama karena dia mendapat perhatian publik karena tidak menggunakan rumah gubernur resmi), dan di mana dia bertemu dengannya. suami (bagian dari biografi tidak resminya yang beredar selama kampanye: dia dan suaminya bertemu di sekolah menengah). Dengan tiga jawaban itu, Kernell dapat mengubah kata sandinya (menjadi “popcorn,” sesuatu yang menarik bagi sebagian besar mahasiswa). Sejak saat itu, Kernell bukan hanya Palin yang efektif, Palin yang asli tidak dapat mengakses akun emailnya sendiri karena dia tidak mengetahui kata sandi barunya.

Mekanisme otentikasi menggunakan salah satu dari tiga kualitas untuk mengonfirmasi identitas pengguna:

- Sesuatu yang diketahui pengguna. Kata sandi, nomor PIN, frasa sandi, jabat tangan rahasia, dan nama gadis ibu adalah contoh yang mungkin diketahui pengguna.
- Sesuatu pengguna. Pengautentikasi ini, yang disebut biometrik, didasarkan pada karakteristik fisik pengguna, seperti sidik jari, pola suara seseorang, atau wajah (gambar). Metode otentikasi

ini sudah lama (kami mengenali teman secara langsung dari wajah mereka atau melalui telepon dengan suara mereka) tetapi baru mulai digunakan dalam otentikasi komputer.
- Sesuatu yang dimiliki pengguna. Lencana identitas, kunci fisik, SIM, atau seragam adalah contoh umum dari hal-hal yang dimiliki orang yang membuatnya dapat dikenali.

Dua atau lebih bentuk dapat digabungkan; misalnya, kartu bank dan PIN menggabungkan sesuatu yang dimiliki pengguna (kartu) dengan sesuatu yang diketahui pengguna (PIN).

Meskipun kata sandi adalah bentuk otentikasi komputer pertama dan tetap populer, bentuk lain ini menjadi lebih mudah digunakan, lebih murah, dan lebih umum. Pada bagian berikut, kami memeriksa masing-masing bentuk otentikasi ini.

2.1.2 Otentikasi Berdasarkan Frasa dan Fakta: Sesuatu yang Anda Ketahui

Perlindungan kata sandi tampaknya menawarkan sistem yang relatif aman untuk mengonfirmasi informasi terkait identitas, tetapi praktik manusia terkadang menurunkan kualitasnya. Mari kita jelajahi kerentanan dalam otentikasi, dengan fokus pada parameter otentikasi yang paling umum, kata sandi. Di bagian ini kami mempertimbangkan sifat kata sandi, kriteria untuk memilihnya, dan cara menggunakannya untuk otentikasi. Saat Anda membaca diskusi berikut tentang kerentanan kata sandi, pikirkan seberapa baik serangan identitas ini akan bekerja melawan pertanyaan keamanan dan skema otentikasi lain yang mungkin Anda kenal. Dan ingat berapa banyak informasi tentang kita yang diketahui — terkadang karena kita mengungkapkannya sendiri — seperti yang dijelaskan di Studi Kasus 2-2.

Studi Kasus 2-2 : Halaman Facebook Menjawab Pertanyaan Keamanan

George Bronk, 23 tahun penduduk Sacramento, California, mengaku bersalah pada 13 Januari 2011 atas tuduhan termasuk intrusi komputer, peniruan identitas palsu, dan kepemilikan pornografi anak. Kejahatannya melibatkan peniruan identitas wanita dengan data yang diperoleh dari akun Facebook mereka.

Menurut berita Associated Press, Bronk memindai halaman Facebook untuk halaman yang menunjukkan alamat email wanita. Dia kemudian membaca profil Facebook mereka dengan hati-hati untuk petunjuk yang dapat membantunya menjawab pertanyaan keamanan, seperti warna favorit atau nama tengah ayah. Dengan petunjuk profil ini, Bronk kemudian beralih ke penyedia akun email. Menggunakan teknik yang sama seperti Kernell, Bronk berpura-pura lupa kata sandinya dan terkadang berhasil menjawab pertanyaan keamanan yang diperlukan untuk memulihkan kata sandi yang terlupakan. Dia terkadang menggunakan teknik yang sama untuk mendapatkan akses ke akun Facebook.

Setelah dia memiliki kata sandi wanita, dia meneliti folder surat terkirim mereka untuk foto-foto memalukan; dia terkadang mengirimkannya ke kontak korban atau mempostingnya di halaman Facebook-nya. Dia melakukan kegiatannya dari Desember 2009 hingga Oktober 2010. Ketika polisi menyita komputernya dan menganalisis

isinya, mereka menemukan 3.200 kontak Internet dan 172 file email yang berisi foto-foto eksplisit; polisi mengirim surat ke semua kontak untuk menanyakan apakah mereka telah menjadi korban, dan 46 menjawab bahwa mereka pernah. Para korban tinggal di Inggris, Washington, D.C., dan 17 negara bagian dari California hingga New Hampshire.

Kantor jaksa agung California menyarankan mereka yang menggunakan email dan situs jejaring sosial untuk memilih pertanyaan keamanan dan jawaban yang tidak diposting di situs publik, atau menambahkan angka atau karakter lain ke jawaban keamanan umum. Kiat keamanan tambahan ada di situs web jaksa agung.

2.1.3 Penggunaan Kata Sandi

Penggunaan kata sandi cukup mudah, karena Anda mungkin sudah tahu dari pengalaman. Seorang pengguna memasukkan beberapa bagian identifikasi, seperti nama atau ID pengguna yang ditetapkan; Identifikasi ini dapat tersedia untuk umum atau mudah ditebak karena tidak memberikan perlindungan yang nyata. Sistem perlindungan kemudian meminta kata sandi dari pengguna. Jika kata sandi cocok dengan yang ada di file untuk pengguna, pengguna diautentikasi dan diizinkan mengakses. Jika pencocokan kata sandi gagal, sistem meminta kata sandi lagi, jika pengguna salah ketik.
Meskipun kata sandi digunakan secara luas, mereka mengalami beberapa kesulitan penggunaan:

- Penggunaan. Memberikan kata sandi untuk setiap akses ke suatu objek bisa merepotkan dan memakan waktu.
- Pengungkapan. Jika pengguna mengungkapkan kata sandi kepada individu yang tidak berwenang, objek tersebut segera dapat diakses. Jika pengguna kemudian mengubah kata sandi untuk melindungi kembali objek, pengguna harus memberi tahu pengguna lain yang sah tentang kata sandi baru karena kata sandi lama mereka akan gagal.
- Pencabutan/Pembatalan. Untuk mencabut hak akses satu pengguna ke suatu objek, seseorang harus mengubah kata sandi, sehingga menyebabkan masalah yang sama seperti pengungkapan.
- Hilang. Bergantung pada bagaimana kata sandi diterapkan, mungkin tidak mungkin untuk mengambil kata sandi yang hilang atau terlupakan. Operator atau administrator sistem tentu saja dapat mengintervensi dan memberikan kata sandi baru, tetapi seringkali mereka tidak dapat menentukan kata sandi apa yang telah dipilih pengguna sebelumnya. Jika pengguna kehilangan (atau lupa) kata sandi, administrator harus menetapkan yang baru.

Menyerang dan Melindungi Kata Sandi

Seberapa amankah kata sandi itu sendiri? Kata sandi agak terbatas sebagai perangkat perlindungan karena jumlah bit informasi yang dikandungnya relatif kecil. Lebih buruk lagi, orang memilih kata sandi yang bahkan tidak memanfaatkan jumlah bit yang tersedia: Memilih string yang terkenal, seperti qwerty, kata sandi, atau 123456 mengurangi ketidakpastian atau kesulitan penyerang pada dasarnya menjadi nol.

Knight dan Hartley [KNI98] daftar, secara berurutan, 12 langkah yang mungkin dicoba penyerang untuk menentukan kata sandi. Langkah-langkah ini meningkatkan tingkat kesulitan (jumlah tebakan), dan karenanya menunjukkan jumlah pekerjaan yang harus dilakukan penyerang untuk mendapatkan kata sandi. Berikut adalah langkah-langkah menebak kata sandi mereka:

- tanpa kata sandi
- sama dengan ID pengguna
- adalah, atau berasal dari, nama pengguna
- pada daftar kata umum (misalnya, kata sandi, rahasia, pribadi) ditambah nama dan pola umum (misalnya, qwerty, aaaaaa)
- terdapat dalam kamus singkat perguruan tinggi
- terdapat dalam daftar kata bahasa Inggris lengkap
- terkandung dalam kamus umum non-bahasa Inggris
- terdapat dalam kamus singkat perguruan tinggi dengan huruf kapital (PassWorD) atau substitusi (angka 0 untuk huruf O, dan seterusnya)
- terdapat dalam kamus bahasa Inggris lengkap dengan huruf besar atau substitusi
- terdapat dalam kamus umum non-Inggris dengan kapitalisasi atau substitusi
- diperoleh dengan kekerasan, mencoba semua kemungkinan kombinasi karakter alfabet
- diperoleh dengan kekerasan, mencoba semua kemungkinan kombinasi dari set karakter lengkap

Meskipun langkah terakhir akan selalu berhasil, langkah-langkah yang mendahuluinya sangat memakan waktu sehingga akan menghalangi semua kecuali penyerang yang paling berdedikasi yang waktu bukan merupakan faktor pembatas.

Kami sekarang memperluas beberapa pendekatan ini.

2.1.4 Dictionary Attack

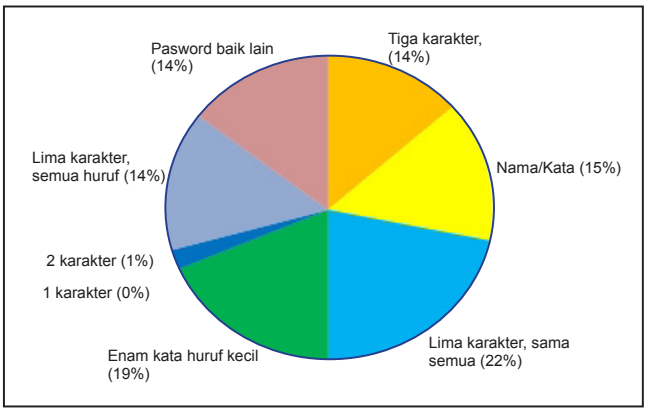
Dictionary attack dianggap efektif karena memanfaatkan psikologi manusia, yaitu kebiasaan bahwa pengguna akun akan menggunakan kata-kata yang lumrah dan mudah diingat sebagai password suatu akun tertentu, misalnya tanggal lahir, makanan kesukaan, nama orang tua, dan lain-lain. Kata-kata ini merupakan kata yang terdapat dalam kamus (karena merupakan kata yang digunakan sehari-hari) dan mudah diingat. Psikologi manusia dalam membuat password juga telah dikaji dan dibuktikan dengan penelitian. Saat ini dictionary attack telah dikembangkan variasinya dan semakin tinggi efektivitasnya, sehingga teknik ini masih sering digunakan untuk membobol password pengguna sebuah akun. Sekarang situs Internet menawarkan apa yang disebut perangkat lunak pemulihan kata sandi sebagai freeware atau shareware dengan harga di bawah $20. (Ini adalah program pembobol kata sandi.)

Orang berpikir mereka bisa pintar dengan memilih kata sandi yang sederhana dan mengganti karakter tertentu, seperti 0 (nol) untuk huruf O, 1 (satu) untuk huruf I atau L, 3 (tiga) untuk huruf E atau @ (at) untuk huruf A. Tetapi pengguna bukan satu-satunya orang yang dapat memikirkan penggantian ini.

Menyimpulkan 'Kemungkinan' Kata Sandi Pengguna

Salah satu kunci keamanan online adalah memiliki kata sandi atau password yang kuat. Namun sebagian besar pengguna sulit membuat kata sandi berbeda yang tepat, dan mengingatnya dengan benar. Kata sandi merupakan hal yang penting. Alasannya, jika data pengguna disusupi, kata sandi yang lemah bisa menimbulkan konsekuensi serius seperti pencurian identitas.

Jika seseorang memilih kata sandi, dia mungkin tidak memilih kata secara acak. Kemungkinan besar kata sandi seseorang adalah sesuatu yang berarti baginya. Orang biasanya memilih kata sandi pribadi, seperti nama pasangan, anak, anggota keluarga lain, atau hewan peliharaan. Untuk setiap orang tertentu, jumlah kemungkinan seperti itu hanya selusin atau dua. Mencoba banyak kata sandi ini dengan komputer membutuhkan waktu kurang dari satu detik! Bahkan seseorang yang bekerja dengan tangan dapat mencoba sepuluh kandidat yang mungkin dalam satu atau dua menit.



Gambar 2-1 Distribusi Jenis Password (Morris dan Thompson)

Morris dan Thompson mengkonfirmasi ketakutan kami dalam laporan mereka tentang hasil mengumpulkan kata sandi dari banyak pengguna, ditunjukkan pada Gambar 2-1 (berdasarkan data dari penelitian itu) menunjukkan karakteristik dari 3.289 kata sandi yang dikumpulkan. Hasil dari penelitian itu menyedihkan, dan situasi saat ini kemungkinan besar akan sama. Dari kata sandi itu, 86 persen dapat ditemukan dalam pengujian 24 jam sehari selama sekitar satu minggu, menggunakan perkiraan yang sangat murah hati 1 milidetik per pemeriksaan kata sandi.

Menurut LogMeIn, perusahaan di balik pengelola kata sandi LastPass, terdapat 5.183 pelanggaran data pada 2019 yang mengungkapkan informasi pribadi seperti alamat rumah, dan kredensial masuk yang dapat dengan mudah digunakan untuk mencuri identitas pengguna atau melakukan penipuan. *(https://www.suara.com/tekno/2020/09/05/064500/ini-5-cara-membuat-dan-mengingat-kata-sandi-yang-kuat-dan-aman)*

Kaspersky Lab melakukan penelitian terhadap penggunaan kata sandi oleh netizen. Dari hasil penelitian yang dirilis beberapa waktu lalu oleh Kaspersky, para netizen mengalami dilema terhadap kata sandi.

Beberapa netizen mengaku, kata sandi yang rumit memang aman namun sulit untuk mengingat. Berbeda dengan kata sandi berkombinasi mudah tetapi rentan diretas.

Berdasarkan penelitian tersebut, banyak responden memahami kebutuhan kata sandi yang kuat untuk mengamankan akun. Tiga akun online yang membutuhkan kata sandi kuat versi reponden, yakni akun perbankan (63 persen), aplikasi pembayaran termasuk e-wallet (42 persen), dan aplikasi belanja online (41 persen). Namun para responden mengaku bahwa kata sandi kuat cukup sulit untuk dihapal.

Kesulitan tersebut membuat responden memiliki kecenderungan lupa dengan kata sandi. Akibatnya, responden tidak bisa mengakses akun tersebut. Sebanyak 38 persen pengguna tidak dapat dengan cepat mengingat kata sandi untuk akun online pribadi. Bahkan kondisi tersebut membuat pengguna frustasi hingga stres sehingga mengganggu aktivitas normal. Kemudian sebesar 51 persen dari responden tidak menyimpan kata sandi dengan aman, sementara 23 persen lainnya menulis kata sandi di dalam notepad untuk memudahkan ingatan.

Dari hasil penelitian, untuk menghindari frustasi tersebut akhirnya para netizan menggunakan kata sandi lemah agar mudah diingat. Sebanyak 10 persen hanya menggunakan satu kata sandi saja untuk semua akun online. Hal tersebut membuat 17 persen responden sudah mendapat ancaman hingga mengalami peretasan akun selama satu tahun terakhir.

Sebanyak 41 persen email responden menjadi target peretasan, kemudian 37 persen akun sosial media, 18 persen rekening perbankan, dan 18 persen akun belanja online. Head of Consumer Business Kaspersky Lab Andrei Mochola mengatakan, jika pengguna mempunyai kata sandi kuat yang dapat diingat setiap saat maka akun online akan jauh lebih aman. "Hal ini sangat penting bagi pengguna," kata Mochola.

Mochola mengatakan, apabila hal tersebut dirasa sulit maka pengguna memerlukan sebuah platform yang bisa menjadi solusi pengingat. Kaspersky memiliki platform bernama 'Password Manager' sehingga netizen bisa memiliki kata sandi yang kuat tanpa harus menuliskannya pada notepad. Bahkan pengguna tidak perlu mengingat rangkaian huruf rumit dari kata sandi tersebut. Pengguna hanya cukup mengingat satu kata sandi utama untuk mengakses semua akun online. *(https://republika.co.id/berita/p2mqdb359/netizen-dilema-terhadap-pengguna-kata-sandi)*

Kesimpulan yang kami tarik dari insiden ini adalah bahwa orang lebih sering memilih kata sandi yang lemah dan mudah ditebak daripada yang diperkirakan. Jelas, orang menemukan sesuatu dalam proses kata sandi yang sulit atau tidak menyenangkan: Entah orang tidak dapat memilih kata sandi yang baik, mungkin karena tekanan situasi, atau mereka takut akan lupa kata sandi yang solid. Dalam kedua Studi Kasus tersebut, kata sandi tidak selalu merupakan autentikator yang baik.

Menebak Kemungkinan Kata Sandi

Pikirkan sebuah kata. Apakah kata yang Anda pikirkan panjang? Apakah itu tidak biasa? Apakah sulit untuk mengeja atau mengucapkan? Jawaban untuk ketiga pertanyaan ini mungkin tidak.

'Penyusup' yang mencari kata sandi menyadari karakteristik yang sangat manusiawi ini dan menggunakannya untuk keuntungan mereka. Oleh karena itu, penyusup mencoba teknik yang kemungkinan besar dengan cepat dapat berhasil mengetahui kata sandi seseorang. Jika orang lebih suka kata sandi yang pendek daripada kata sandi yang panjang, penyusup akan mencoba semua kata sandi tetapi mencoba mengurutkannya berdasarkan panjangnya. Hanya ada 261 + 262 + 263 = 18.278 (tidak peka huruf besar/kecil) kata sandi dengan panjang 3 atau kurang. Menguji banyak kata sandi akan sulit tetapi mungkin bagi manusia, tetapi pengujian kata sandi berulang adalah aplikasi komputer yang mudah. Pada tingkat asumsi satu kata sandi per milidetik, semua kata sandi ini dapat diperiksa dalam 18,278 detik, hampir tidak ada tantangan dengan komputer. Bahkan memperluas percobaan ke 4 atau 5 karakter meningkatkan hitungan hanya menjadi 475 detik (sekitar 8 menit) atau 12.356 detik (sekitar 3,5 jam), masing-masing.

Analisis ini mengasumsikan bahwa orang memilih kata sandi seperti '*vxlag'* dan '*msms'* sesering mereka memilih *enter* dan *beer*. Namun, orang cenderung memilih nama atau kata yang dapat mereka ingat. Banyak sistem komputasi memiliki pemeriksa ejaan yang dapat digunakan untuk memeriksa kesalahan ejaan dan kesalahan tipografi dalam dokumen. Pemeriksa ejaan ini terkadang membawa kamus online dari kata-kata bahasa Inggris yang paling umum. Satu berisi kamus 80.000 kata. Mencoba semua kata ini sebagai kata sandi hanya membutuhkan waktu 80 detik dengan perkiraan yang tidak realistis, yaitu satu tebakan per milidetik.

Sebuah survey dilakukan oleh Keeper Security untuk menemukan password apa yang paling sering digunakan oleh pengguna komputer dan internet di seluruh dunia. Dan ditemukan kalau password urutan angka masih jadi pilihan favorit.

Keeper Security mencatatkan setidaknya ada 1,7 juta akun dalam surveinya yang menggunakan password '123456'. Sementara data lainnya mengatakan sekitar 17% dari 10 juta akun yang kena hack menggunakan password tersebut. (*https://inet.detik.com/cyberlife)*

Berikut daftar 25 password paling populer sepanjang 2016,

1. 123456
2. 123456789
3. qwerty
4. 12345678
5. 111111
6. 1234567890
7. 1234567
8. password
9. 123123
10. 987654321
11. qwertyuiop
12. mynoob
13. 123321
14. 666666
15. 18atcskd2w
16. 7777777
17. 1q2w3e4r
18. 654321
19. 555555
20. 3rjs1la7qe
21. google
22. 1q2w3e4r5t
23. 123qwe
24. zxcvbnm
25. 1q2w3e

'Mengalahkan' Penyembunyian Password

Lebih mudah daripada menebak kata sandi hanya dengan membaca satu dari tabel, seperti tabel contoh yang ditunjukkan pada Tabel 2.1. Sistem operasi mengotentikasi pengguna dengan meminta nama dan kata sandi, yang kemudian harus divalidasi, kemungkinan besar dengan membandingkan dengan nilai yang disimpan dalam tabel. Tetapi tabel itu kemudian menjadi harta karun bagi para pelaku kejahatan: Mendapatkan tabel tersebut memberikan akses ke semua akun karena tidak hanya berisi satu tetapi semua ID pengguna dan kata sandi yang sesuai.

Tabel 2.1 Contoh Tabel Kata Sandi

Identitas	Kata Sandi
Rizky	qwerty
Wahyu	aaaaa
Karla	guessme
Jaelani	zxcvbnm
Titin	aaaaa
Rasyid	261220
Gabriel	3rjs1la7qe

Sistem operasi menghalangi pendekatan itu dengan menyimpan kata sandi tidak dalam bentuk publiknya tetapi dalam bentuk tersembunyi (menggunakan enkripsi, yang akan kami jelaskan nanti dalam bab ini), seperti yang ditunjukkan pada Tabel 2.2. Ketika pengguna membuat kata sandi, sistem operasi menerima dan segera menyembunyikannya, menyimpan versi yang tidak dapat dibaca. Kemudian ketika pengguna mencoba untuk mengautentikasi dengan memasukkan ID pengguna dan kata sandi, sistem operasi menerima apa pun yang diketik, menerapkan fungsi penyembunyian yang sama, dan membandingkan versi yang disembunyikan dengan apa yang disimpan. Jika kedua formulir cocok, otentikasi lolos.

Kami menggunakan istilah "menyembunyikan" di paragraf sebelumnya karena terkadang sistem operasi melakukan beberapa bentuk pengacakan yang sebenarnya bukan enkripsi, atau terkadang itu adalah bentuk enkripsi terbatas. Satu-satunya poin penting adalah bahwa prosesnya satu arah: Mengubah kata sandi ke bentuk penyembunyiannya sederhana, tetapi sebaliknya (dimulai dengan versi tersembunyi dan mendapatkan kata sandi yang sesuai) secara efektif tidak mungkin. (Untuk alasan ini, di beberapa situs web jika Anda lupa kata sandi Anda, sistem dapat mengatur ulang kata sandi Anda ke nilai acak yang baru, tetapi sistem tidak dapat memberi tahu Anda apa kata sandi Anda yang terlupakan.)

Tabel 2.2 Contoh Tabel Kata Sandi dengan Nilai Kata Sandi Tersembunyi

Identity	Password
Rizky	0x471aa2d2
Wahyu	0x13b9c32f
Jaelani	0x01c142be
Titin	0x13b9c32f
Titin	0x5202aae2
Rasyid	0x488b8c27

Untuk otentikasi aktif, yaitu memasukkan identitas dan autentikator untuk dapat mengakses sistem, sebagian besar sistem mengunci pengguna yang gagal dalam sejumlah kecil upaya login berturut-turut. Jumlah kegagalan ini mencegah penyerang mencoba lebih dari beberapa tebakan. (Namun, perhatikan bahwa fitur penguncian ini memberi penyerang cara untuk mencegah akses oleh pengguna yang sah: cukup masukkan kata sandi yang salah sehingga sistem memblokir akun.) Namun, jika penyerang memperoleh tabel kata sandi terenkripsi dan mempelajari algoritma penyembunyian, program komputer dapat dengan mudah menguji ratusan ribu tebakan dalam hitungan menit.

Seperti yang dikonfirmasi oleh banyak penelitian dalam bab ini, orang sering menggunakan salah satu dari beberapa kata sandi yang dapat diprediksi. Pencegat dapat membuat apa yang disebut tabel pelangi, daftar bentuk tersembunyi dari kata sandi umum, seperti yang ditunjukkan pada Tabel 2.3. Mencari entri yang cocok dalam tabel kata sandi yang dicegat, penyusup dapat mengetahui bahwa kata sandi Rizky adalah 123456 dan kata sandi Manda adalah qwerty. Penyerang mengurutkan tabel untuk melakukan pencarian dengan cepat.

Tabel 2.3 Contoh Tabel Pelangi untuk Kata Sandi Umum

Original Password	Encrypted Password
asdfg	0x023c94fc
p@55w0rd	0x04ff38d9
aaaaaa	0x13b9c32f
password	0x2129f30d
qwerty	0x471aa2d2
12345678	0x4f2c4dd8
123456	0x5903c34d
aaaaa	0x8384a8c8
	etc.

Kata sandi yang diacak memiliki kerentanan lain. Perhatikan pada Tabel 2-2 bahwa Wahyu dan Titin keduanya memilih kata sandi yang sama. Kedua salinan akan memiliki nilai tersembunyi yang sama, sehingga seseorang yang mencegat tabel dapat mengetahui bahwa pengguna Wahyu dan Titin memiliki kata sandi yang sama. Mengetahui hal itu, pencegat juga dapat menebak bahwa Wahyu dan Titin sama-sama memilih kata sandi yang sama, dan mulai mencoba yang biasa; ketika satu bekerja, yang lain juga akan bekerja.

Untuk mengatasi kedua ancaman ini, beberapa sistem menggunakan bagian tambahan yang disebut Salt. Salt adalah bidang data tambahan yang berbeda untuk setiap pengguna, mungkin tanggal akun dibuat atau bagian dari nama pengguna. Nilai Salt digabungkan ke kata sandi sebelum kombinasi diubah dengan penyembunyian.

Dengan cara ini, Wahyu+aaaaaa memiliki nilai penyembunyian yang berbeda dari Titin+aaaaaa, seperti yang ditunjukkan pada Tabel 2.4. Selain itu, penyerang tidak dapat membuat tabel pelangi karena kata sandi umum sekarang semuanya memiliki komponen unik juga.

Tabel 2-4 Contoh Tabel Kata Sandi dengan Nilai Kata Sandi Tersembunyi yang Dipersonalisasi

Identity	ID+password (not stored in table)	Stored Authentication Value
Rizky	Riz+qwerty	0x1d46e346
Wahyu	Wah+aaaaaa	0x2d5d3e44
Karla	Kar+oct31witch	0xc23c04d8
Titin	Tit+aaaaaa	0xe30f4d27
Jaelani	Jae+guessme	0x8127f48d
Rasyid	Ras+261220	0x5209d942

2.1.5 Exhaustive Attack (*Brute Force Attack*)

Dalam Exhaustive attack atau brute force attack, penyerang mencoba semua kemungkinan kata sandi, biasanya dengan cara otomatis. Tentu saja, jumlah kemungkinan kata sandi tergantung pada penerapan sistem komputasi tertentu. Misalnya, jika kata sandi adalah kata yang terdiri dari 26 karakter A–Z dan dapat terdiri dari 1 hingga 8 karakter, ada 261 kata sandi dengan 1 karakter, 262 kata sandi dengan 2 karakter, dan 268 kata sandi dengan 8 karakter. Oleh karena itu, sistem secara keseluruhan memiliki 261 + 262 + … + 268 = 269 – 1 5 * 1012 atau lima juta kemungkinan

kata sandi. Jumlah itu tampaknya cukup keras. Jika kita menggunakan komputer untuk membuat dan mencoba setiap kata sandi dengan kecepatan pemeriksaan satu kata sandi per milidetik, dibutuhkan waktu 150 tahun untuk menguji semua kata sandi delapan huruf. Tetapi jika kami dapat mempercepat pencarian menjadi satu kata sandi per mikrodetik, faktor kerja turun menjadi sekitar dua bulan. Jumlah waktu ini masuk akal bagi penyerang untuk berinvestasi jika hadiahnya besar. Misalnya, seorang penyusup dapat mencoba secara brutal untuk memecahkan kata sandi pada file nomor kartu kredit atau informasi rekening bank.

Tetapi waktu istirahat dapat dibuat lebih mudah diatur dalam beberapa cara. Mencari satu kata sandi tertentu tidak selalu mengharuskan semua kata sandi dicoba; penyusup hanya perlu mencoba sampai kata sandi yang benar diidentifikasi. Jika kumpulan semua kemungkinan kata sandi didistribusikan secara merata, penyusup kemungkinan hanya perlu mencoba setengah dari ruang kata sandi: jumlah pencarian yang diharapkan untuk menemukan kata sandi tertentu. Namun, penyusup juga dapat memanfaatkan distribusi kata sandi yang tidak merata. Karena kata sandi harus diingat, orang cenderung memilih kata sandi yang sederhana; oleh karena itu, penyusup harus mencoba kombinasi karakter yang pendek sebelum mencoba yang lebih panjang. Fitur ini mengurangi waktu rata-rata untuk menemukan kecocokan karena mengurangi bagian dari ruang kata sandi yang dicari sebelum menemukan kecocokan. Dan seperti yang kami jelaskan sebelumnya, penyerang dapat membuat tabel pelangi dari kata sandi umum, yang mengurangi upaya serangan menjadi pencarian tabel sederhana.

Semua teknik untuk mengalahkan kata sandi ini, dikombinasikan dengan masalah kegunaan, menunjukkan bahwa kita perlu mencari metode otentikasi lain. Di bagian berikutnya, kami mengeksplorasi cara menerapkan otentikasi yang kuat sebagai kontrol terhadap serangan peniruan identitas. Untuk contoh lain dari masalah otentikasi, lihat Studi Kasus 2-3.

Studi Kasus 2-3 : Akankah Earl of Buckingham yang Asli mau tampil di muka umum?

Seorang pria yang mengaku sebagai Earl of Buckingham diidentifikasi sebagai Charlie Stopford, seorang pria yang telah menghilang dari keluarganya di Florida pada tahun 1983 dan mengambil identitas Christopher Buckingham, bayi berusia 8 bulan yang meninggal pada tahun 1963. Dipertanyakan di Inggris pada tahun 2005 setelah pemeriksaan detail paspor mengungkapkan hubungan dengan bayi Buckingham yang telah meninggal, Stopford ditangkap ketika dia tidak mengetahui detail keluarga terkait lainnya. (Pekerjaannya pada saat penangkapannya? Konsultan keamanan komputer.)

Pihak berwenang Inggris tahu dia bukan Christopher Buckingham, tapi apa identitas aslinya? Jawabannya ditemukan hanya karena keluarganya di Amerika Serikat mengira itu mengenalinya dari foto dan berita: Stopford adalah seorang suami dan ayah yang telah menghilang lebih dari 20 tahun sebelumnya. Karena dia pernah berada di Angkatan Laut AS (dalam intelijen militer, tidak kurang) dan sidik jari dewasanya tercatat, pihak berwenang dapat membuat identifikasi positif.

Adapun gelar yang dia ambil untuk dirinya sendiri, tidak ada Earl of Buckingham sejak 1687.

Dalam masyarakat modern kita terbiasa dengan jejak kertas penuh yang mendokumentasikan peristiwa sejak lahir hingga mati, tetapi tidak semua orang cocok dengan model itu. Pertimbangkan Studi Kasus orang-orang tertentu yang karena berbagai alasan perlu mengubah identitas mereka.

Ketika pemerintah mengubah identitas seseorang (misalnya, ketika seorang saksi bersembunyi), identitas baru itu mencakup catatan sekolah, alamat, catatan pekerjaan, dan sebagainya.

Kata Sandi yang Baik

Dipilih dengan hati-hati, kata sandi bisa menjadi autentikator yang kuat. Istilah "kata sandi" menyiratkan satu kata, tetapi Anda sebenarnya dapat menggunakan kata atau frasa yang tidak ada. Jadi 2Brn2Bti? bisa berupa kata sandi (berasal dari “menjadi atau tidak, itulah pertanyaannya”) seperti halnya “PayTaxesApril15th.” Perhatikan bahwa pilihan ini memiliki beberapa karakteristik penting: String panjang, dipilih dari kumpulan karakter yang besar, dan tidak muncul dalam kamus. Properti ini membuat kata sandi sulit (tetapi, tentu saja, bukan tidak mungkin) untuk ditentukan. Jika kami menggunakan kata sandi, kami dapat meningkatkan keamanannya dengan beberapa praktik sederhana:

- ***Gunakan karakter selain hanya a–***z. Jika kata sandi dipilih dari huruf a– z, hanya ada 26 kemungkinan untuk setiap karakter. Menambahkan angka akan memperluas jumlah kemungkinan menjadi 36. Menggunakan huruf besar dan kecil ditambah angka akan menambah jumlah karakter yang mungkin menjadi 62. Meskipun perubahan ini tampak kecil, efeknya besar ketika seseorang menguji ruang penuh dari semua kemungkinan kombinasi karakter. Dibutuhkan sekitar 100 jam untuk menguji semua kata 6-huruf yang dipilih dari huruf-huruf satu huruf saja, tetapi dibutuhkan sekitar 2 tahun untuk menguji semua kata sandi 6-simbol dari huruf besar dan kecil serta angka. Meskipun 100 jam masuk akal, 2 tahun cukup menekan untuk membuat serangan ini jauh kurang menarik.
- ***Pilih kata sandi yang panjang***. Eksplosi secara kombinatorial kesulitan menebak kata sandi dimulai sekitar panjang 4 atau 5. Memilih kata sandi yang lebih panjang membuat kecil kemungkinan kata sandi akan terungkap. Ingatlah bahwa penyusup brute force dapat berhenti segera setelah kata sandi ditemukan. Beberapa penyusup akan mencoba Studi Kasus yang mudah—kata-kata yang diketahui dan kata sandi pendek—dan beralih ke target lain jika serangan itu gagal.
- ***Hindari nama atau kata yang sebenarnya***. Secara teoritis, ada 26, atau sekitar 300 juta "kata" 6 huruf (berarti kombinasi huruf apa pun), tetapi hanya ada sekitar 150.000 kata dalam kamus perguruan tinggi yang bagus, mengabaikan panjangnya. Dengan memilih salah satu dari 99,95 persen bukan kata, Anda memaksa penyerang untuk menggunakan pencarian brute-force yang lebih lama daripada pencarian kamus yang disingkat.
- ***Gunakan deret kata yang Anda ingat.*** Pilihan kata sandi adalah ikatan ganda. Untuk mengingat kata sandi dengan mudah, Anda menginginkan kata sandi yang memiliki arti khusus bagi Anda.

Namun, Anda tidak ingin orang lain dapat menebak arti khusus ini. Satu kata sandi yang mudah diingat adalah UcnB2s. Kekacauan yang tampak tidak biasa itu adalah transformasi sederhana dari "Anda tidak akan pernah bisa terlalu aman." Huruf pertama kata dari sebuah lagu, beberapa huruf dari kata yang berbeda dari frasa pribadi, atau sesuatu yang melibatkan skor bola basket yang mudah diingat adalah contoh kata sandi yang masuk akal. Tapi jangan terlalu mencolok. Alat peretas kata sandi juga menguji penggantian seperti 0 (nol) untuk o atau O (huruf “oh”) dan 1 (satu) untuk l (huruf “ell”) atau $ untuk S (huruf “ess”). Jadi l10v3U sudah ada di file pencarian.

- ***Gunakan varian untuk beberapa sandi***. Dengan akun, situs web, dan langganan, seseorang dapat dengan mudah mengumpulkan 50 atau 100 kata sandi, yang jelas terlalu banyak untuk diingat. Kecuali jika Anda menggunakan trik. Mulailah dengan frasa seperti pada saran sebelumnya: Ih1b2s (Saya punya satu saudara laki-laki, dua saudara perempuan). Kemudian tambahkan beberapa pola yang melibatkan beberapa vokal dan konsonan pertama dari entitas untuk kata sandi: Ih1b2sIvs untuk vIsa, Ih1b2sAfc untuk fAcebook, dan seterusnya.
- ***Ganti sandi secara berkala***. Bahkan jika Anda tidak memiliki alasan untuk mencurigai bahwa seseorang telah mengkompromikan kata sandi, Anda harus mengubahnya dari waktu ke waktu. Penyusup dapat merusak sistem kata sandi dengan mendapatkan daftar lama atau bekerja secara mendalam pada daftar terenkripsi.
- ***Jangan ditulis***. Catatan: Nasihat jangka panjang ini hanya relevan jika keamanan fisik merupakan risiko serius. Orang yang memiliki akun di banyak mesin dan server, dan dengan banyak aplikasi atau situs, mungkin mengalami kesulitan mengingat semua kode akses. Menyetel semua kode sama atau menggunakan kata sandi yang tidak aman tetapi mudah diingat mungkin lebih berisiko daripada menulis kata sandi pada daftar yang cukup terlindungi. (Jelas, Anda tidak boleh menempelkan PIN Anda ke kartu bank Anda atau memposting kata sandi Anda di layar komputer Anda.)
- ***Jangan beri tahu orang lain***. Serangan termudah adalah rekayasa sosial, di mana penyerang menghubungi administrator sistem atau pengguna untuk mendapatkan kata sandi dengan cara tertentu. Misalnya, penyerang dapat menelepon pengguna, mengklaim sebagai "administrasi sistem", dan meminta pengguna untuk memverifikasi kata sandi pengguna. Dalam situasi apa pun Anda tidak boleh memberikan kata sandi pribadi Anda; administrator yang sah dapat menghindari kata sandi Anda jika perlu, dan orang lain hanya mencoba menipu Anda.

Prinsip-prinsip ini mengarah pada pemilihan kata sandi yang solid, tetapi mereka menyebabkan masalah yang berbeda: Orang memilih kata sandi sederhana karena mereka harus membuat dan mengingat begitu banyak kata sandi. Rekening bank, akses email, layanan perpustakaan, banyak situs web, dan aplikasi lain semuanya tampaknya memerlukan kata sandi. Kami tidak dapat menyalahkan pengguna karena tergoda untuk menggunakan satu kata sandi sederhana untuk mereka semua ketika alternatifnya mencoba mengingat lusinan atau bahkan ratusan kata sandi yang kuat, seperti yang dibahas di Studi Kasus 2-4.

Studi Kasus 2.4 : Kegunaan dalam Kecil versus Kegunaan dalam Besar

Bagi pengembang aplikasi yang mencari kontrol yang wajar, kata sandi tampaknya merupakan mekanisme langsung untuk melindungi aset. Tetapi ketika banyak aplikasi memerlukan kata sandi, pekerjaan sederhana pengguna untuk mengingat satu atau dua kata sandi berubah menjadi mimpi buruk untuk melacak sejumlah besar kata sandi. Memang, kunjungan ke http://www.passwordbook.com menunjukkan bahwa pengguna sering mengalami kesulitan mengelola kumpulan kata sandi. Situs ini memperkenalkan Anda pada kata sandi dan pengatur login yang murah dan mudah dibeli. Dalam kata-kata vendor, ini adalah “Pengelola kata sandi lengkap untuk master Web atau administrator jaringan yang sibuk … Aman dan mudah, buku tidak macet! Sekarang Anda dapat mengelola semua kata sandi Anda dalam satu buku hardbound.”

Meskipun mengelola satu kata sandi atau token untuk suatu aplikasi mungkin tampak mudah (kami menyebutnya "kegunaan dalam skala kecil"), mengelola banyak kata sandi atau token sekaligus menjadi tugas yang menakutkan ("kegunaan dalam skala besar"). Masalah mengingat berbagai macam item telah didokumentasikan dalam literatur psikologi sejak 1950-an, ketika Miller [MIL56] menunjukkan bahwa orang mengingat sesuatu dengan memecahnya menjadi potongan-potongan yang mudah diingat, apakah itu angka, huruf, kata, atau lainnya. entitas yang dapat diidentifikasi. Miller awalnya mendokumentasikan bagaimana orang dewasa muda memiliki rentang memori tujuh (plus atau minus dua) potongan. Penelitian selanjutnya mengungkapkan bahwa rentang memori tergantung pada sifat potongan: potongan yang lebih panjang menyebabkan rentang memori yang lebih pendek: tujuh untuk digit, enam untuk huruf, dan lima untuk kata. Faktor lain juga mempengaruhi rentang memori seseorang. Cowan menyarankan bahwa kita mengasumsikan rentang memori kerja empat potongan untuk dewasa muda, dengan rentang yang lebih pendek untuk anak-anak dan warga lanjut usia. Untuk alasan ini, kegunaan harus menginformasikan tidak hanya pilihan konstruksi kata sandi yang sesuai (kecil) tetapi juga arsitektur keamanan itu sendiri (besar).

Hal Lain yang Diketahui

Kata sandi, atau lebih tepatnya sesuatu yang hanya diketahui oleh pengguna, adalah salah satu bentuk otentikasi yang kuat. Kata sandi mudah dibuat dan dikelola, tidak mahal untuk digunakan, dan mudah dimengerti. Namun, pengguna terlalu sering memilih kata sandi yang mudah diingat, tetapi tidak mudah ditebak orang lain. Selain itu, pengguna dapat lupa kata sandi atau memberitahukannya kepada orang lain. Kata sandi berasal dari faktor otentikasi sesuatu yang diketahui pengguna, dan sayangnya otak orang tidak sempurna.

Akibatnya, beberapa pendekatan lain untuk "sesuatu yang diketahui pengguna" telah diusulkan. Misalnya, Studi Kasus 2-5 menjelaskan pendekatan otentikasi yang menggunakan pengetahuan pengguna alih-alih kata sandi. Namun, beberapa teknik otentikasi pengetahuan pengguna telah diuji dengan baik dan sedikit yang ditingkatkan dengan cara yang bermanfaat; pendekatan ini masih diteliti.

Studi Kasus 2-5 : Menggunakan Pola Pribadi untuk Otentikasi

Lamande melaporkan bahwa sistem otentikasi GrIDSure (http://www.gridsure.com) telah diintegrasikan ke dalam platform Unified Access Gateway (UAG) Microsoft. Sistem ini memungkinkan pengguna untuk mengotentikasi dirinya dengan kode akses satu kali berdasarkan pola kotak yang dipilih dari kisi. Ketika pengguna menginginkan akses, dia disajikan dengan kotak yang berisi nomor yang ditetapkan secara acak; dia kemudian memasukkan sebagai kode sandinya angka-angka yang sesuai dengan pola yang dipilihnya. Karena nomor kisi yang ditampilkan berubah setiap kali kisi disajikan, pola memungkinkan kode sandi yang dimasukkan menjadi kode satu kali. GrIDSure adalah upaya untuk menskalakan pendekatan "pengetahuan pengguna" dari kegunaan dalam skala kecil ke kegunaan dalam skala besar. Banyak peneliti telah memeriksa aspek keamanan dan kegunaan GrIDSure, dengan hasil yang beragam. Masih harus dilihat bagaimana penggunaan GrIDSure dibandingkan dengan penggunaan kumpulan kata sandi tradisional.

Demikian pula, produk ImageShield dari Confident Technologies (www.confidenttechnologies.com) meminta pengguna untuk mendaftar dengan memilih tiga kategori dari daftar; kategorinya mungkin kucing, mobil, dan bunga, misalnya. Kemudian pada waktu otentikasi, pengguna diperlihatkan kisi-kisi gambar, beberapa dari kategori pengguna dan yang lainnya tidak. Setiap gambar memiliki 1 karakter huruf atau angka. String akses satu kali pengguna adalah karakter yang dilampirkan ke gambar dari kategori yang dipilih sebelumnya oleh pengguna. Jadi, jika gambar tersebut termasuk kucing dengan label A, bunga dengan label 7, dan tujuh gambar lainnya, nilai akses pengguna adalah A7. Gambar, karakter, dan posisi berubah untuk setiap akses, sehingga autentikatornya juga berbeda.

Skema otentikasi seperti ini didasarkan pada teka-teki sederhana yang dapat dipecahkan pengguna dengan mudah tetapi penipu tidak akan berhasil menebaknya. Namun, dengan skema otentikasi baru, kita harus menyadari fenomena kegunaan dalam skala kecil dan besar: Dapatkah pengguna mengingat kotak pada kisi dan kategori gambar dan tempat liburan favorit dan rumus 2a+c dan banyak lainnya hal-hal yang tidak jelas?

Pertanyaan Keamanan

Sebagai ganti kata sandi, beberapa perusahaan menggunakan pertanyaan yang (mungkin) hanya orang yang tepat yang tahu jawabannya. Siapakah nama guru pertama Anda? Apa makanan favorit Anda? Pertanyaan-pertanyaan tersebut biasanya muncul saat Anda melakukan login akun pertama kali. Sebenarnya, apa maksud dari pertanyaan tersebut? Pertanyaan tersebut merupakan fitur pertanyaan keamanan atau *security questions*. Fitur ini berfungsi untuk mengamankan dan memastikan bahwa tidak ada orang lain yang dapat mengutak-atik akun Anda, seperti pada saat perubahan password atau pembaruan data. Akan tetapi, kebanyakan orang masih menyepelekan langkah ini dan mengisinya hanya sebagai formalitas. Para pengguna cenderung lupa akan pertanyaan dan jawaban yang telah dimasukkan. Padahal fitur ini dapat menjadi penolong Anda jika sewaktu-waktu ingin melakukan perubahan data atau memperbarui password pada akun Anda

Masalah dengan pertanyaan seperti itu adalah bahwa jawaban untuk beberapa pertanyaan dapat ditentukan dengan sedikit kesulitan, seperti yang terjadi pada akun email Sarah Palin. Dengan jumlah catatan publik yang tersedia secara online, nama gadis ibu dan nama jalan sering tersedia, dan teman-teman sekolah dapat menebak sejumlah kecil kemungkinan guru favorit. Joseph Bonneau dan Sören Preibusch melakukan survei terperinci tentang metode otentikasi situs web dan menemukan sedikit keseragaman, banyak kelemahan, dan tidak ada korelasi yang jelas antara nilai data situs dan persyaratan otentikasinya. Kata sandi menjadi semakin menindas karena banyak situs web sekarang meminta pengguna untuk masuk. Tetapi ketika dihadapkan dengan sistem yang sulit ditangani, pengguna sering kali mengambil jalan yang mudah: memilih kata sandi yang mudah dan menggunakannya kembali di banyak situs. Untuk mengatasi kelemahan itu, beberapa sistem menggunakan bentuk otentikasi yang tidak dapat dicuri, diduplikasi, dilupakan, dipinjamkan, atau hilang: properti pengguna, seperti yang akan kita bahas di bagian selanjutnya. Teknologi untuk meneruskan karakteristik pribadi ke server jarak jauh membutuhkan lebih dari sekadar keyboard dan perangkat penunjuk, tetapi pendekatan semacam itu menjadi lebih layak, terutama karena pelanggaran tabel kata sandi meningkat.

2.2 Otentikasi Berdasarkan Biometrik

Penggunaan password atau kata sandi untuk melindungi sistem online disebutkan tidak lama lagi bakal segera digantikan oleh otentikasi berbasis teknologi biometrik.

Pasalnya, teknologi biometrik seperti pemindai sidik jari atau retina mata, dinilai lebih aman, cepat, dan nyaman, dibandingkan dengan teknologi tradisional seperti password atau PIN (personal identification number).

Kepala bidang teknologi cybersecurity Microsoft, Diana Kelley, mengatakan, industri digital di seluruh dunia saat ini menghadapi kondisi bahwa semakin banyak kesulitan yang dihadapi kebanyakan pengguna dalam mengingat dan merancang kata sandi yang kompleks. Bahkan, menurut laporan dari perusahaan perangkat lunak pengelola kata sandi, Lastpass, rata-rata karyawan di sebuah perusahaan harus mengingat atau merancang sebanyak 191 kata sandi.

Biometrik adalah sifat biologis, berdasarkan beberapa karakteristik fisik tubuh manusia. Daftar teknologi otentikasi biometrik masih terus bertambah. Sekarang perangkat dapat mengenali biometrik berikut:

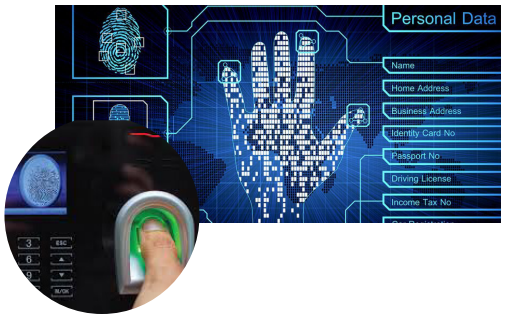
- sidik jari
- geometri tangan (bentuk dan ukuran jari)
- retina dan iris (bagian dari mata)
- suara
- tulisan tangan, tanda tangan, gerakan tangan
- karakteristik mengetik
- pembuluh darah di jari atau tangan
- wajah
- fitur wajah, seperti bentuk hidung atau jarak mata

Otentikasi dengan biometrik memiliki keunggulan dibandingkan kata sandi karena biometrik tidak dapat hilang, dicuri, dilupakan, atau dibagikan dan selalu tersedia, selalu ada, sehingga dapat dikatakan. Karakteristik ini sulit, jika bukan tidak mungkin, untuk ditempa.

2.2.1 Contoh Authenticator Biometrik

Banyak karakteristik fisik yang memungkinkan sebagai autentikator. Di bagian ini kita berikan contoh dua di antaranya, satu untuk ukuran dan bentuk tangan, dan satu lagi untuk pola pembuluh darah di tangan.

Gambar 2-2 menunjukkan pembaca geometri tangan. Pengguna meletakkan tangan pada sensor, yang mendeteksi panjang dan lebar jari, kelengkungan, dan karakteristik lainnya. Perangkat otentikasi dari Fujitsu membaca pola pembuluh darah di tangan. Perangkat ini tidak memerlukan kontak fisik antara tangan dan pembaca, yang merupakan keuntungan untuk kebersihan. Pabrikan mengklaim tingkat penerimaan palsu 0,00008 persen dan tingkat penolakan palsu 0,01 persen, dengan waktu respons kurang dari satu detik. Gambar 2-3 menunjukkan perangkat ini tertanam di mouse komputer, sehingga pengguna secara otomatis diautentikasi.



Gambar 2.2 Pembaca Geometri Tangan (https://www.roboratorymedan.com/)



Gambar 2.3 Pembaca Vena Tangan https://www.fortress-land.com/

Masalah dengan Penggunaan Biometrik

Beberap permasalahan yang muncul dalam Biometrik antara lain:

- Biometrik relatif baru, dan beberapa orang menganggap penggunaannya mengganggu. Misalnya, orang-orang di beberapa budaya dihina dengan harus tunduk pada sidik jari, karena mereka berpikir bahwa hanya penjahat yang sidik jarinya. Geometri tangan dan pengenalan wajah (yang dapat dilakukan dari kamera di seberang ruangan) hampir tidak invasif, tetapi orang-orang benar-benar khawatir tentang mengintip sinar laser atau memasukkan jari ke dalam slot. (Lihat [SCH06a] untuk beberapa contoh orang yang menolak biometrik.)
- Perangkat pengenalan biometrik mahal, meskipun perangkat menjadi lebih populer, biaya per perangkatnya akan turun. Tetap saja, melengkapi setiap workstation pengguna dengan pembaca bisa mahal untuk perusahaan besar dengan banyak karyawan.
- Pembaca biometrik dan perbandingan bisa menjadi satu titik kegagalan. Pertimbangkan aplikasi ritel di mana pengenalan biometrik terkait dengan skema pembayaran: Seperti yang dikatakan seorang pengguna, “Jika kartu kredit saya gagal untuk mendaftar, saya selalu dapat mengeluarkan kartu kedua, tetapi jika sidik jari saya tidak dikenali, saya harus hanya satu jari itu.” (Pengenalan

sidik jari khusus untuk satu jari; pola satu jari tidak sama dengan yang lain.) Pekerja manual sebenarnya dapat menghapus sidik jari mereka dari waktu ke waktu, dan luka atau iritasi dapat mengacaukan pembaca sidik jari. Lupa kata sandi adalah kesalahan pengguna; gagal otentikasi biometrik tidak.

- Semua pembaca biometrik menggunakan pengambilan sampel dan menetapkan ambang batas untuk menerima kecocokan yang dekat. Perangkat harus mengambil sampel biometrik, sering mengukur ratusan titik kunci, dan membandingkan rangkaian pengukuran tersebut dengan templat. Fiturnya sedikit berbeda dari satu bacaan ke bacaan berikutnya, misalnya, jika wajah Anda dimiringkan, jika Anda menekan satu sisi jari lebih dari yang lain, atau jika suara Anda terkena infeksi sinus. Variasi mengurangi akurasi.
- Meskipun akurasi peralatan meningkat, pembacaan yang salah masih terjadi. Kami memberi label *false positive* atau *false accept* suatu bacaan yang diterima ketika seharusnya ditolak (yaitu, autentikator tidak cocok) dan *false negative* atau *false reject* yang menolak ketika seharusnya menerima. Seringkali, mengurangi tingkat positif palsu meningkatkan negatif palsu, dan sebaliknya. Studi Kasusu 2-6 menjelaskan mengapa kita tidak pernah bisa menghilangkan semua positif dan negatif palsu. Konsekuensi negatif palsu biasanya lebih kecil daripada positif palsu, sehingga sistem yang dapat diterima mungkin memiliki tingkat positif palsu 0,001 persen tetapi tingkat negatif palsu 1 persen. Namun, jika populasinya besar dan asetnya sangat berharga, bahkan persentase kecil ini dapat menyebabkan hasil yang sangat buruk.

Studi Kasus 2-6 : Pengertian Sebenarnya False Positives dan False Negatives

Sistem screening harus dapat menilai sejauh mana skema pencocokan mereka bekerja dengan baik. Artinya, mereka harus dapat menentukan apakah mereka secara efektif mengidentifikasi orang-orang yang dicari tanpa merugikan orang-orang yang tidak dicari. Ketika sistem penyaringan membandingkan sesuatu yang dimilikinya (seperti sidik jari yang tersimpan) dengan sesuatu yang diukurnya (seperti karakteristik jari), kami menyebutnya sistem atau tes dikotomis: Ada yang cocok atau tidak.

Kita dapat menggambarkan dikotomi dengan menggunakan Standar Referensi, seperti yang digambarkan pada Tabel 2.6, di bawah ini. Standar Referensi adalah seperangkat aturan yang menentukan kapan tes positif berarti hasil positif. Kami ingin menghindari dua jenis kesalahan: False Positive/positif palsu (bila ada kecocokan tetapi sebenarnya tidak cocok) dan false Negative/negatif palsu (ketika tidak ada kecocokan tetapi sebenarnya ada).

Tabel 2.6 Standar Referensi untuk Menggambarkan Tes Dikotomis

	Apakah ada Orang mengakui	Apakah tidak ada Orang mengakui
Hasil tes Positif (Ada kecocokan)	True Positive (TP)	False Positive (FP)
Hasil tes Negatif (Tidak ada kecocokan)	False Negative (FN)	True Negatuve (TN)

Kita dapat mengukur keberhasilan layar dengan menggunakan empat ukuran standar: sensitivitas, prevalensi, akurasi, dan spesifisitas. Untuk melihat cara kerjanya, kami menetapkan variabel ke entri pada Tabel 2.6, seperti yang ditunjukkan pada Tabel 2.7.

Tabel 2.7 Standar Referensi dengan Variabel

	Apakah ada Orang mengakui	**Apakah tidak ada Orang mengakui**
Hasil tes Positif	True Positive = *a*	False Positive = *b*
Hasil tes Negatif	False Negative = *c*	True Negatuve = *d*

Sensitivitas mengukur sejauh mana layar memilih orang-orang yang namanya benar-benar cocok dengan orang yang dicari. Ini adalah proporsi hasil positif di antara semua kemungkinan kecocokan yang benar dan dihitung sebagai *a / (a + c)*. Spesifisitas mengukur proporsi hasil negatif di antara semua orang yang tidak dicari; itu dihitung sebagai *d / (b + d)*.

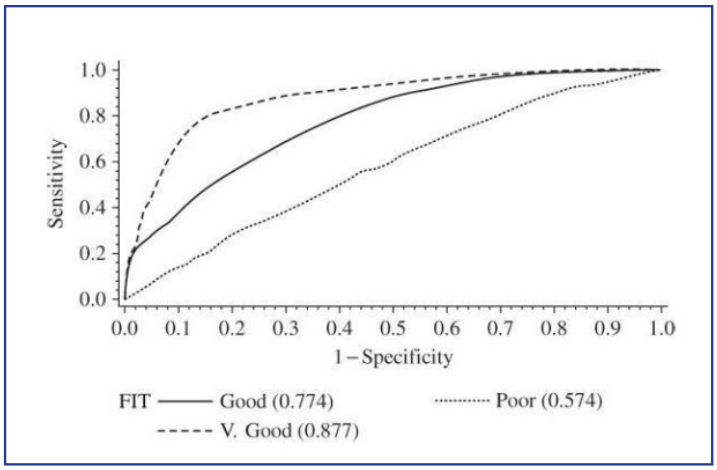
Sensitivitas dan spesifisitas menggambarkan seberapa baik tes membedakan antara Studi Kasus dengan dan tanpa kondisi tertentu.
Akurasi atau kemanjuran mengukur sejauh mana tes atau layar dengan benar menandai kondisi atau situasi; itu diukur sebagai *(a + d) / (a + b + c + d)*. Prevalensi memberitahu kita seberapa umum kondisi atau situasi tertentu. Ini diukur sebagai *(a + c) / (a + b + c + d)*.

Sensitivitas dan spesifisitas secara statistik terkait: Ketika salah satu meningkat, yang lain menurun. Dengan demikian, Anda tidak bisa begitu saja mengatakan bahwa Anda akan mengurangi atau menghapus positif palsu; tindakan seperti itu pasti akan meningkatkan negatif palsu. Sebagai gantinya, Anda harus menemukan keseimbangan antara jumlah kesalahan yang dapat diterima positif dan negatif palsu. Untuk membantu kami, kami menghitung nilai prediksi positif dari sebuah tes: angka yang menyatakan berapa kali kecocokan positif benar-benar mewakili identifikasi orang yang dicari. Nilai prediksi positif adalah *a / (a + b*). Demikian pula, kita dapat menghitung nilai prediksi negatif dari tes sebagai *d / (c + d)*. Kita dapat menggunakan nilai-nilai prediktif untuk memberi kita gambaran tentang kapan suatu hasil cenderung positif atau negatif. Misalnya, hasil positif dari suatu kondisi yang memiliki prevalensi tinggi kemungkinan besar akan positif. Namun, hasil positif untuk kondisi yang tidak biasa cenderung menjadi positif palsu (FP).

Sensitivitas dan spesifisitas berubah untuk tes tertentu, tergantung pada tingkat tes yang menentukan kecocokan. Misalnya, pengujian dapat menyebutnya sebagai kecocokan hanya jika itu adalah kecocokan persis: hanya 'Smith' yang akan cocok dengan 'Smith.' Kriteria kecocokan seperti itu akan memiliki hasil positif yang lebih sedikit (yaitu, lebih sedikit situasi yang dianggap cocok) daripada yang menggunakan Soundex untuk menyatakan bahwa dua nama adalah sama: 'Smith' sama dengan 'Smythe', 'Smeth', 'Smitt', dan nama lain yang terdengar mirip. Akibatnya, kedua

tes bervariasi dalam sensitivitasnya. Kriteria Soundex kurang ketat dan cenderung menghasilkan lebih banyak kecocokan positif; oleh karena itu, ini adalah tes yang lebih sensitif tetapi kurang spesifik. Secara umum, pertimbangkan kisaran sensitivitas yang dapat dihasilkan saat kita mengubah kriteria pengujian. Kita dapat meningkatkan sensitivitas dengan membuat kriteria untuk tes positif menjadi kurang ketat. Demikian pula, kita dapat meningkatkan spesifisitas dengan membuat kriteria untuk tes positif lebih ketat.

Kurva KOP (kurva karakteristik operasi penerima) – *Receiver Operating Characteristic*/ (ROC) − adalah grafik yang menunjukkan performa model klasifikasi di semua batas klasifikasi. Kurva ini menandai dua parameter: Rasio Positif Benar. Rasio Positif Palsu. Secara tradisional, grafik ROC menunjukkan tingkat positif palsu (1 – spesifisitas) pada sumbu x dan tingkat positif sebenarnya (sensitivitas atau 1 – tingkat negatif palsu) pada sumbu y. Keakuratan tes sesuai dengan area di bawah kurva. Area 1 mewakili tes yang sempurna, sedangkan area 0,5 adalah tes yang tidak berharga. Idealnya, kami ingin tes berada sejauh mungkin dan setinggi mungkin pada grafik, mewakili tes dengan tingkat positif benar yang tinggi dan tingkat positif palsu yang rendah. Artinya, semakin besar area di bawah kurva, semakin tes mengidentifikasi positif benar dan meminimalkan positif palsu. Gambar 2.4 menunjukkan contoh kurva ROC dan hubungannya dengan sensitivitas dan spesifisitas.



Gambar 2.4 Kurva ROC

Untuk sistem pencocokan atau penyaringan, seperti halnya pengujian apa pun, administrator sistem harus menentukan tingkat sensitivitas dan spesifisitas apa yang dapat diterima. Tingkat tergantung pada tujuan tes, pengaturan, prevalensi kriteria target, metode alternatif untuk mencapai tujuan yang sama, dan biaya dan manfaat pengujian.

- Kecepatan di mana pengenalan harus dilakukan membatasi akurasi. Kami mungkin idealnya ingin mengambil beberapa bacaan dan menggabungkan hasil atau mengevaluasi kecocokan terdekat. Tetapi otentikasi dilakukan untuk memungkinkan pengguna melakukan sesuatu: Otentikasi bukanlah tujuan akhir tetapi gerbang yang menjaga pengguna dari tujuan. Dapat dimengerti bahwa pengguna ingin melewati gerbang dan menjadi frustrasi dan kesal jika otentikasi terlalu lama.

- Meskipun kami suka menganggap biometrik sebagai bagian unik dari seorang individu, pemalsuan adalah mungkin. Beberapa contoh pemalsuan dijelaskan di Studi Kasus 2-7.

Biometrik bergantung pada karakteristik fisik yang dapat bervariasi dari satu hari ke hari berikutnya atau seiring bertambahnya usia. Pertimbangkan tangan Anda, misalnya: Pada beberapa hari, suhu, tingkat aktivitas Anda, atau faktor lain dapat menyebabkan tangan Anda membengkak, sehingga mengubah karakteristik fisik tangan Anda. Tetapi otentikasi tidak boleh gagal hanya karena hari itu panas. Pengenalan biometrik juga tergantung pada bagaimana sampel diambil. Untuk geometri tangan, misalnya, Anda meletakkan tangan Anda di atas templat, tetapi pengukurannya akan sedikit berbeda tergantung pada bagaimana Anda memposisikan tangan Anda.

Studi Kasus 2-7 : Pemalsuan Biometrik

Pemalsuan yang paling terkenal adalah sidik jari buatan yang diproduksi oleh para peneliti di Jepang menggunakan gelatin yang murah dan mudah didapat. Para peneliti menggunakan cetakan yang dibuat dengan menekan jari-jari hidup atau dengan memproses gambar sidik jari dari cetakan pada permukaan kaca. Hasil "jari bergetah" sering diterima oleh 11 perangkat sidik jari tertentu dengan sensor optik atau kapasitif.
Menurut cerita lain dari berita BBC (13 Mar 2013) seorang dokter di Brasil ditangkap dengan enam belas jari: sepuluh asli dan enam terbuat dari silikon yang dia gunakan untuk masuk ke sistem kartu waktu rumah sakit atas nama sesama dokter.

Dalam sebuah penelitian yang diterbitkan pada tahun 2014, para peneliti melihat apakah lensa kontak dapat digunakan untuk mengelabui perangkat otentikasi yang melihat pola iris (cincin mata berwarna). Tujuan dari penelitian ini adalah untuk menentukan apakah sistem pengenalan iris secara andal mendeteksi hal positif yang sebenarnya; yaitu, apakah subjek akan diautentikasi secara andal oleh sistem. Para peneliti menunjukkan bahwa lensa kontak berwarna dapat mengelabui sistem untuk menolak kecocokan ketika benar-benar ada. Subjek mungkin menggunakan lensa kontak agar tidak terlihat sebagai buronan, misalnya. Meskipun sulit dan tidak umum, pemalsuan akan menjadi masalah setiap kali hadiah untuk hasil yang salah cukup tinggi.

Otentikasi dengan biometrik menggunakan pola atau template, seperti garis dasar, yang mewakili pengukuran karakteristik. Saat Anda menggunakan biometrik untuk otentikasi, serangkaian pengukuran saat ini diambil dan dibandingkan dengan template. Namun, sampel saat ini tidak harus sama persis dengan template. Otentikasi berhasil jika kecocokan "cukup dekat", yang berarti dalam toleransi yang telah ditentukan, misalnya, jika 90 persen nilai cocok atau jika setiap parameter berada dalam 5 persen dari nilai yang diharapkan. Mengukur, membandingkan, dan menilai kedekatan untuk kecocokan membutuhkan waktu, tentu lebih lama daripada perbandingan “kecocokan persis atau tidak” untuk kata sandi. (Pertimbangkan hasil yang dijelaskan di Studi Kasus 2-8.) Oleh karena itu, kecepatan dan keakuratan biometrik merupakan faktor dalam menentukan kesesuaiannya untuk lingkungan penggunaan tertentu.

Ingat bahwa identifikasi menyatakan identitas, sedangkan otentikasi mengkonfirmasi identitas, seperti yang digambarkan pada Gambar 2-5. Biometrik dapat diandalkan untuk otentikasi tetapi kurang dapat diandalkan untuk identifikasi. Alasannya adalah matematika. Semua pembaca biometrik beroperasi dalam dua fase. Pertama, pengguna mendaftar dengan pembaca, selama waktu itu karakteristik pengguna (misalnya, geometri tangan) ditangkap dan direduksi menjadi satu set titik data. Selama pendaftaran, pengguna mungkin diminta untuk menunjukkan tangan beberapa kali sehingga perangkat lunak pendaftaran dapat menyesuaikan variasi, seperti bagaimana tangan diposisikan. Registrasi menghasilkan pola, yang disebut template, dari titik data khusus untuk pengguna tertentu. Pada fase kedua pengguna kemudian mencari otentikasi dari sistem, selama waktu itu sistem mengukur ulang tangan dan membandingkan pengukuran baru dengan template yang disimpan. Jika pengukuran baru cukup dekat dengan template, sistem menerima otentikasi; jika tidak, sistem akan menolaknya. Studi Kasus 2-9 menunjukkan masalah dalam identifikasi dan otentikasi yang membingungkan.

Studi Kasus 2-8 : Pengambilan Sidik Jari—Tidak Begitu Cepat!

Merekam atau menangkap sidik jari harus menjadi proses yang mudah. Beberapa negara menggunakan sidik jari untuk melacak pengunjung asing yang masuk ke negara tersebut, sehingga mereka ingin mengetahui dampaknya terhadap pemrosesan pengunjung di perbatasan. Di televisi dan di film sepertinya mendapatkan gambar sidik jari yang bagus hanya membutuhkan waktu satu atau dua detik.

Para peneliti di Institut Standar dan Teknologi Nasional AS (NIST) melakukan eksperimen terkontrol yang melibatkan lebih dari 300 subjek yang umumnya mewakili populasi AS. Mereka menemukan bahwa berbeda dengan televisi, mendapatkan sampel berkualitas dari sepuluh jari membutuhkan waktu antara 45 detik dan satu menit.

Studi Kasus 2-9 : DNA untuk Identifikasi atau Otentikasi

Pada bulan Desember 1972, seorang perawat di San Francisco diserang secara seksual dan dibunuh secara brutal di apartemennya. Sang induk semang, yang menghadapi seorang pria saat ia bergegas keluar dari apartemen, memberikan deskripsi fisik kepada polisi. Di TKP, polisi mengumpulkan bukti, termasuk sampel DNA dari pembunuh yang diduga. Namun, setelah berbulan-bulan penyelidikan, polisi tidak dapat fokus pada tersangka dan Studi Kasus itu akhirnya diturunkan ke tumpukan Studi Kasus yang belum terpecahkan.

Tiga puluh tahun kemudian, Departemen Kepolisian San Francisco mendapat hibah untuk menggunakan DNA untuk memecahkan Studi Kasus terbuka dan, setelah membuka kembali Studi Kasus tahun 1972, mereka menemukan satu slide dengan sampel DNA yang rusak. Untuk tujuan investigasi, para ilmuwan mengisolasi 13 ciri, yang disebut penanda, dalam sampel DNA. Peluang dua orang berbeda yang cocok

pada semua 13 penanda adalah 1 dalam 1 kuadriliun (1*1015). Namun, seperti yang dijelaskan dalam cerita Los Angeles Times oleh Jason Felch dan Maura Dolan [FEL08], sampel lama dalam Studi Kasus ini telah memburuk dan hanya 5½ dari 13 penanda yang dapat diandalkan. Dengan hanya banyak penanda, kemungkinan dua orang akan cocok turun menjadi 1 dalam 1,1 juta, dan ingat bahwa untuk tujuan di sini, pencocokan DNA dua orang berarti setidaknya satu sampel bukan milik penjahat.

Selanjutnya, polisi ingin membandingkan sampel dengan database negara bagian California dari sampel DNA penjahat yang dihukum. Tetapi untuk menjalankan perbandingan seperti itu, administrator memerlukan setidaknya 7 spidol dan polisi hanya memiliki 5½. Untuk mencari database, polisi menggunakan nilai dari dua penanda lain yang terlalu redup untuk dianggap konklusif. Dengan tujuh penanda, polisi mensurvei database 338.000 dan menghasilkan satu kecocokan, seorang pria kemudian diadili dan dihukum karena kejahatan ini, seorang pria yang pengacara pembelanya sangat yakin tidak bersalah. Dia tidak memiliki hubungan dengan korban, sidik jarinya tidak cocok dengan yang dikumpulkan di TKP, dan keyakinannya sebelumnya untuk kejahatan seks memiliki pola yang berbeda.

Masalahnya adalah bahwa polisi menggunakan kecocokan DNA sebagai pengenal, bukan autentikator. Jika polisi memiliki bukti lain terhadap tersangka tertentu dan DNA tersangka yang ditemukan di TKP, kemungkinan identifikasi yang benar meningkat. Namun, jika polisi hanya mencari orang yang DNA-nya cocok dengan sampel, kemungkinan kecocokan palsu meningkat secara dramatis. Ingatlah bahwa dengan rasio kecocokan palsu 1 dalam 1,1 juta, jika Anda mengumpulkan 1,1 juta orang, Anda akan berharap bahwa satu akan salah mencocokkan sampel Anda, atau dengan 0,5 juta orang Anda akan berpikir kemungkinan kecocokan menjadi sekitar 1 dalam 2. Kemungkinan kecocokan palsu turun menjadi 1 dari 1,1 juta orang hanya jika Anda memeriksa hanya satu orang.

Pikirkan analogi ini: Jika Anda membeli satu tiket lotre dalam 1,1 juta tiket lotre, peluang Anda untuk menang adalah 1 banding 1,1 juta. Jika Anda membeli dua tiket, peluang Anda meningkat menjadi 2 dalam 1,1 juta, dan jika Anda membeli 338.000 tiket, peluang Anda menjadi 338.000 dalam 1,1 juta, atau kira-kira 1 dalam 3. Untuk alasan ini, ketika mencari identifikasi, bukan otentikasi, baik DNA FBI dewan penasehat dan panel Dewan Riset Nasional merekomendasikan mengalikan probabilitas umum (1 dalam 1,1 juta) dengan jumlah sampel dalam database untuk mendapatkan kemungkinan kecocokan acak—tidak bersalah—.

Meskipun kita tidak mengetahui apakah terpidana dalam Studi Kasus ini bersalah atau tidak, namun alasan tersebut mengingatkan kita untuk berhati-hati dalam membedakan antara identifikasi dan otentikasi.

Akurasi Biometrik

Kami menganggap biometrik—atau teknologi otentikasi apa pun—sebagai biner: Seseorang lulus atau gagal, dan jika kita mengatur parameter dengan benar, sebagian besar orang yang tepat akan lulus dan sebagian besar orang yang salah akan gagal. Artinya, mekanismenya tidak membeda-bedakan. Faktanya, prosesnya bias, disebabkan oleh keseimbangan antara sensitivitas dan selektivitas: Beberapa orang lebih mungkin untuk lulus dan yang lain lebih mungkin untuk gagal. Studi Kasus 2-10 menjelaskan bagaimana ini bisa terjadi.

Sampai saat ini polisi dan sistem peradilan berasumsi bahwa sidik jari itu unik.
Namun, sebenarnya tidak ada dasar matematis atau ilmiah untuk asumsi ini. Faktanya, identifikasi sidik jari telah terbukti salah, dan sistem perbandingan sidik jari manusia dan komputerisasi juga menunjukkan kegagalan. Bagian dari masalah perbandingan berkaitan dengan fakta bahwa tidak seluruh sidik jari dibandingkan, hanya karakteristik pada tonjolan yang signifikan pada cetakan. Dengan demikian, manusia atau mesin hanya memeriksa fitur yang menonjol, yang disebut template dari cetakan itu.

Kecuali setiap template unik, yaitu, tidak ada dua orang yang memiliki nilai yang sama, sistem tidak dapat mengidentifikasi subjek secara unik. Namun, selama penipu tidak mungkin memiliki template biometrik yang sama dengan pengguna sebenarnya, sistem dapat mengautentikasi. Dalam autentikasi, kami tidak memeriksa semua templat untuk melihat siapa yang mungkin cocok dengan serangkaian fitur terukur; kita cukup menentukan apakah fitur seseorang cocok dengan template yang disimpannya. Otentikasi biometrik layak hari ini; identifikasi biometrik sebagian besar masih menjadi topik penelitian.

Mengukur akurasi otentikasi biometrik sulit karena otentikasi tidak unik. Dalam pengaturan eksperimental, untuk satu subjek atau kumpulan subjek mana pun, kita dapat menghitung tingkat negatif palsu dan positif palsu karena kita mengetahui subjek dan identitas mereka yang sebenarnya. Tetapi kami tidak dapat memperkirakan hasil tersebut kepada dunia dan bertanya berapa banyak orang lain yang dapat diautentikasi sebagai seseorang. Kami terbatas karena populasi dan latar penelitian kami mungkin tidak mencerminkan dunia nyata. Vendor produk membuat banyak klaim tentang keakuratan biometrik atau fitur biometrik tertentu, tetapi hanya sedikit peneliti independen yang benar-benar mencoba membuktikan klaim tersebut. Dalam satu percobaan yang dijelaskan di Studi Kasus 2-10, pemeriksa sidik jari ahli, orang-orang yang bersaksi tentang bukti sidik jari di persidangan, beberapa kali gagal.

Studi Kasus 2-10 : Apakah Ada Orang Biasa-biasa saja?

Apakah ada orang yang sistem biometriknya tidak berfungsi? Yaitu, apakah ada orang, misalnya, yang ciri-cirinya begitu tidak dapat dibedakan sehingga mereka akan selalu dianggap sebagai orang lain?

Doddington dkk. memeriksa sistem dan pengguna untuk menemukan contoh spesifik dari orang-orang yang cenderung sering ditolak secara salah, mereka yang profilnya cenderung cocok dengan subjek lain, dan mereka yang cenderung cocok dengan banyak profil yang tidak biasa.

Untuk kelas-kelas ini Yager dan Dunstone menambahkan orang-orang yang cenderung cocok dan menyebabkan tingkat positif palsu yang tinggi dan orang-orang yang tidak mungkin cocok dengan diri mereka sendiri atau orang lain. Para penulis kemudian mempelajari algoritma analisis biometrik yang berbeda dalam kaitannya dengan Studi Kasus-Studi Kasus sulit ini.

Yager dan Dunstone mengutip kepercayaan populer bahwa 2 persen dari populasi memiliki sidik jari yang secara inheren sulit untuk dicocokkan. Setelah menganalisis basis data sidik jari yang besar (kumpulan sidik jari US-VISIT dari pengunjung asing ke Amerika Serikat), mereka menyimpulkan bahwa sedikit, jika ada, orang pada dasarnya sulit untuk dicocokkan, dan tentu saja bukan 2 persen.

Mereka memeriksa teknologi biometrik tertentu dan menemukan bahwa beberapa kesalahan terkait dengan teknologi, bukan pada manusia. Misalnya, mereka melihat database sistem pengenalan iris mata orang yang gagal dicocokkan, tetapi mereka menemukan bahwa banyak dari orang-orang itu memakai kacamata ketika mereka mendaftar dalam sistem; mereka berspekulasi bahwa kacamata mempersulit sistem untuk mengekstrak fitur pola iris individu. Dalam Studi Kasus lain, mereka melihat sistem pengenalan wajah. Mereka menemukan bahwa orang-orang yang gagal dicocokkan sistem berasal dari satu kelompok etnis tertentu dan berspekulasi bahwa algoritma analisis telah disesuaikan dengan perbedaan wajah kelompok etnis lain. Dengan demikian, mereka menyimpulkan bahwa kesalahan pencocokan lebih mungkin disebabkan oleh masalah pendaftaran dan kelemahan algoritme daripada properti bawaan apa pun dari fitur orang.

Namun, untuk sistem biometrik yang mereka pelajari, mereka menemukan bahwa untuk karakteristik dan algoritma analisis tertentu, beberapa karakteristik pengguna berkinerja lebih baik daripada karakteristik pengguna lain. Penelitian ini memperkuat kebutuhan untuk menerapkan sistem tersebut dengan hati-hati sehingga keterbatasan yang melekat pada algoritma, perhitungan, atau penggunaan tidak mempengaruhi hasil secara tidak proporsional.

Otentikasi sangat penting untuk sistem komputasi karena identifikasi pengguna yang akurat adalah kunci untuk hak akses individu. Sebagian besar sistem operasi dan administrator sistem komputasi telah menerapkan langkah-langkah keamanan yang wajar namun ketat untuk mengunci pengguna yang tidak sah sebelum mereka dapat mengakses sumber daya sistem. Namun, seperti yang dilaporkan di Studi Kasus 2.11, terkadang mekanisme yang tidak sesuai dipaksa untuk digunakan sebagai perangkat otentikasi.

Kehilangan atau melupakan otentikasi biometrik hampir tidak mungkin karena biometrik bergantung pada karakteristik manusia. Tetapi karakteristiknya dapat berubah seiring waktu (pikirkan warna atau berat rambut); oleh karena itu, otentikasi biometrik mungkin kurang tepat daripada otentikasi berbasis pengetahuan. Anda tahu kata sandi atau tidak. Tapi sidik jari bisa menjadi 99 persen cocok atau 95 persen atau 82 persen, bagian dari variasi tergantung pada faktor-faktor seperti bagaimana Anda memposisikan jari saat sidik jari dibaca, apakah jari terluka, dan jika tangan dingin atau kulit

kering atau kotor. Stres juga dapat memengaruhi faktor biometrik, seperti pengenalan suara, yang berpotensi mengganggu keamanan. Bayangkan situasi kritis di mana Anda perlu segera mengakses komputer, tetapi kegugupan memengaruhi suara Anda. Jika sistem gagal otentikasi Anda dan menawarkan kesempatan untuk mencoba lagi, tekanan tambahan dapat membuat suara Anda lebih buruk, yang mengancam ketersediaan.

Biometrik dapat dilakukan dengan cukup cepat dan mudah, dan terkadang kita dapat menyesuaikan sensitivitas dan spesifisitas untuk menyeimbangkan hasil positif palsu dan negatif palsu. Tetapi karena biometrik memerlukan perangkat untuk membaca, penggunaannya untuk otentikasi jarak jauh terbatas. Faktor otentikasi ketiga, sesuatu yang Anda miliki, menawarkan kekuatan dan kelemahan yang berbeda dari dua faktor lainnya.

Studi Kasus 2-11 : Menggunakan Cookie untuk Otentikasi

Di web, cookie sering digunakan untuk otentikasi. Cookie adalah sepasang item data yang dikirim ke browser web oleh situs web yang dikunjungi. Item data terdiri dari kunci dan nilai, yang dirancang untuk mewakili status sesi saat ini antara pengguna yang berkunjung dan situs web yang dikunjungi. Setelah cookie ditempatkan pada sistem pengguna (biasanya dalam direktori dengan cookie lain), browser terus menggunakannya untuk interaksi selanjutnya antara pengguna dan situs web tersebut. Setiap cookie seharusnya memiliki tanggal kedaluwarsa, tetapi tanggal itu bisa jauh di masa depan, dan dapat diubah nanti atau bahkan diabaikan.

Misalnya, situs web The Wall Street Journal, wsj.com, membuat cookie saat pengguna pertama kali masuk. Dalam transaksi berikutnya, cookie bertindak sebagai pengenal; pengguna tidak lagi memerlukan kata sandi untuk mengakses situs tersebut. (Situs lain menggunakan pendekatan yang sama atau serupa.) Pengguna harus dilindungi dari paparan dan pemalsuan. Artinya, pengguna mungkin tidak ingin seluruh dunia mengetahui situs apa yang telah mereka kunjungi. Mereka juga tidak ingin seseorang memeriksa informasi atau membeli barang dagangan secara online dengan peniruan identitas dan penipuan. Dan selanjutnya, pada komputer bersama, satu pengguna dapat bertindak sebagai orang lain jika situs penerima menggunakan cookie untuk melakukan otentikasi otomatis.

Sit and Fu menunjukkan bahwa cookie tidak dirancang untuk perlindungan. Tidak ada cara untuk menetapkan atau mengonfirmasi integritas cookie, dan tidak semua situs mengenkripsi informasi dalam cookie mereka. Sit and Fu juga menunjukkan bahwa sistem operasi server harus sangat waspada untuk melindungi dari penyadapan: “Kebanyakan pertukaran [lalu lintas web] tidak menggunakan [enkripsi] untuk melindungi dari penyadapan; siapa pun di jaringan antara dua komputer dapat mendengar lalu lintas. Kecuali server mengambil tindakan pencegahan yang kuat, penyadap dapat mencuri dan menggunakan kembali cookie, menyamar sebagai pengguna tanpa batas.”

2.3 Otentikasi Berdasarkan Token

Sesuatu yang dimiliki berarti Anda memiliki objek fisik yang dimiliki. Satu autentikator fisik yang mungkin Anda kenal adalah kuncinya. Saat memasukkan kunci ke dalam gembok, tonjolan pada kunci berinteraksi dengan pin di kunci untuk memungkinkan mekanisme berputar. Dalam arti kunci mengotentikasi Anda untuk entri resmi karena Anda memiliki kunci yang sesuai. Tentu saja, Anda dapat kehilangan kunci atau menggandakannya dan memberikan duplikatnya kepada orang lain, sehingga autentikasinya tidak sempurna. Tapi itu tepat: Hanya kunci Anda yang berfungsi, dan kunci Anda hanya berfungsi dengan kunci Anda. (Untuk contoh ini, kami sengaja mengabaikan kunci master.)

Contoh token yang familiar lainnya adalah lencana dan kartu identitas. Anda mungkin memiliki "kartu afinitas": kartu dengan kode yang memberi Anda diskon di toko. Banyak siswa dan karyawan memiliki lencana identitas yang memungkinkan mereka mengakses gedung. Anda harus memiliki kartu identitas atau paspor untuk naik pesawat atau memasuki negara asing. Dalam Studi Kasus ini, Anda memiliki objek yang dikenali orang lain untuk memungkinkan Anda mengakses atau mendapatkan hak istimewa.

Jenis lain dari token otentikasi memiliki data untuk berkomunikasi secara tidak terlihat. Contoh token semacam ini termasuk kartu kredit dengan strip magnetik, kartu kredit dengan chip komputer tertanam, atau kartu akses dengan teknologi nirkabel pasif atau aktif. Anda memperkenalkan token ke pembaca yang sesuai, dan pembaca merasakan nilai dari kartu. Jika identitas dan nilai Anda dari token Anda cocok, korespondensi ini menambah keyakinan bahwa Anda adalah siapa yang Anda katakan.

Token Aktif dan Pasif

Seperti namanya, token pasif tidak melakukan apa-apa, dan token aktif mengambil beberapa tindakan. Foto atau kunci adalah contoh token pasif karena isi token tidak pernah berubah. (Dan, tentu saja, keabadian foto bisa menjadi masalah, karena orang mengubah gaya atau warna rambut dan wajah mereka berubah seiring waktu.)

Token aktif dapat memiliki beberapa variabilitas atau interaksi dengan lingkungannya. Misalnya, beberapa sistem transportasi umum menggunakan kartu dengan strip magnetik. Saat Anda memasukkan kartu ke pembaca, mesin membaca saldo saat ini, mengurangi harga perjalanan dan menulis ulang saldo baru untuk penggunaan berikutnya. Dalam hal ini, token hanyalah repositori untuk menyimpan nilai saat ini. Bentuk lain dari token aktif memulai komunikasi dua arah dengan pembacanya, seringkali dengan sinyal nirkabel atau radio. Token ini mengarah pada perbedaan berikutnya antara token, interaksi statis dan dinamis.

Token Statis dan Dinamis

Nilai token statis tetap. Kunci, kartu identitas, paspor, kartu kredit dan kartu strip magnetik lainnya, dan kartu pemancar radio (disebut perangkat RFID) adalah contoh token statis. Token statis paling berguna untuk otentikasi di tempat: Ketika seorang penjaga melihat lencana gambar Anda, fakta bahwa Anda memiliki lencana tersebut dan bahwa wajah Anda terlihat (setidaknya samar-samar) seperti gambar menyebabkan penjaga melewati otentikasi dan mengizinkan Anda mengakses.

Kami juga tertarik dengan otentikasi jarak jauh, yaitu, kemampuan Anda untuk membuktikan identitas Anda kepada seseorang atau komputer di tempat lain. Dengan contoh lencana gambar, mungkin tidak mudah untuk mengirimkan gambar lencana dan penampilan wajah Anda untuk dibandingkan dengan komputer jarak jauh. Lebih buruk lagi, jarak meningkatkan kemungkinan pemalsuan: Seorang penjaga lokal dapat mengetahui apakah Anda mengenakan topeng, tetapi seorang penjaga mungkin tidak mendeteksinya dari gambar jarak jauh. Otentikasi jarak jauh rentan terhadap masalah token yang dipalsukan.

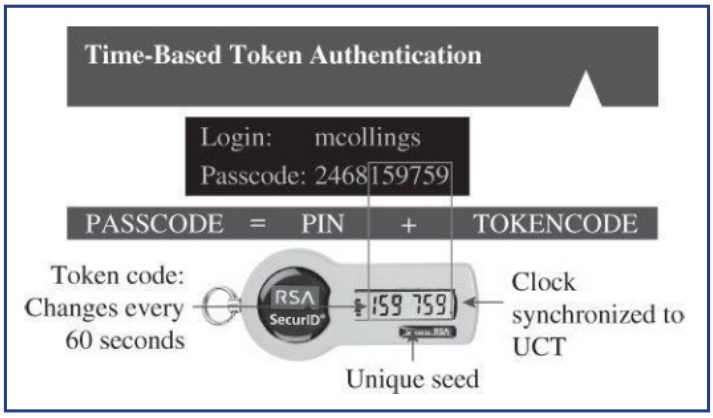
Token rentan terhadap serangan yang disebut skimming. Skimming adalah penggunaan perangkat untuk menyalin data otentikasi secara diam-diam dan menyampaikannya ke penyerang. Anjungan tunai mandiri (ATM) dan pembaca kartu kredit di tempat penjualan sangat rentan terhadap skimming. Di ATM, pencuri memasang perangkat kecil di atas slot tempat Anda memasukkan kartu bank. Karena semua kartu bank sesuai dengan format standar (sehingga Anda dapat menggunakan kartu Anda di ATM atau pedagang mana pun), pencuri dapat menulis perangkat lunak sederhana untuk menyalin dan menyimpan informasi yang direkam pada strip magnetik pada kartu bank Anda. Beberapa skimmer juga memiliki kamera kecil untuk merekam penekanan tombol saat Anda memasukkan PIN pada keypad. Baik secara instan (menggunakan komunikasi nirkabel) atau lebih baru (mengumpulkan perangkat fisik), pencuri mendapatkan nomor akun Anda dan PIN-nya. Pencuri hanya membuat kartu dummy dengan nomor akun Anda yang tercatat dan, menggunakan PIN untuk otentikasi, mengunjungi ATM dan menarik uang tunai dari akun Anda atau membeli barang dengan kartu kredit kloning.

Catatan : Perhatikan bahwa diskusi ini mengacu pada kartu strip magnetik yang populer di Amerika Serikat. Sebagian besar negara lain menggunakan kartu chip komputer tertanam yang secara substansial kurang rentan terhadap skimming.

Bentuk penyalinan lain terjadi dengan kata sandi. Jika Anda harus memasukkan atau mengucapkan kata sandi Anda, orang lain dapat melihat dari balik bahu Anda atau mendengar Anda, dan sekarang autentikator itu mudah disalin atau dipalsukan. Untuk mengatasi penyalinan token fisik atau kata sandi, kita dapat menggunakan token dinamis. Token dinamis adalah token yang nilainya berubah. Meskipun ada beberapa bentuk yang berbeda, token autentikasi dinamis pada dasarnya adalah perangkat yang menghasilkan nilai yang tidak dapat diprediksi yang mungkin kita sebut sebagai nomor pass. Beberapa perangkat mengubah nomor pada interval tertentu, misalnya, sekali dalam satu menit; yang lain mengubah angka saat Anda menekan tombol, dan yang lain menghitung angka baru sebagai respons terhadap input, terkadang disebut tantangan. Dalam semua Studi Kasus, tidak masalah jika orang lain melihat atau mendengar Anda memberikan nomor pass, karena satu nilai itu akan berlaku hanya untuk satu akses (milik Anda), dan mengetahui bahwa satu nilai tidak akan memungkinkan orang luar untuk menebak atau menghasilkan nomor lulus berikutnya.

Generator token dinamis berguna untuk otentikasi jarak jauh, terutama dari seseorang ke komputer. Contoh token dinamis adalah token SecurID dari RSA Laboratories, yang ditunjukkan pada Gambar 2-5. Untuk menggunakan token SecurID, Anda memasukkan nomor saat ini yang ditampilkan pada token saat diminta oleh aplikasi autentikasi. Setiap token menghasilkan serangkaian angka yang berbeda dan hampir tidak dapat diprediksi yang berubah setiap menit, sehingga sistem otentikasi mengetahui angka yang diharapkan dari token Anda setiap saat. Dengan cara ini, dua orang dapat memiliki token SecurID, tetapi setiap token hanya mengautentikasi pemiliknya yang ditetapkan.

Memasukkan nomor dari token lain tidak melewati otentikasi Anda. Dan karena token menghasilkan nomor baru setiap menit, memasukkan nomor dari otentikasi sebelumnya juga gagal.

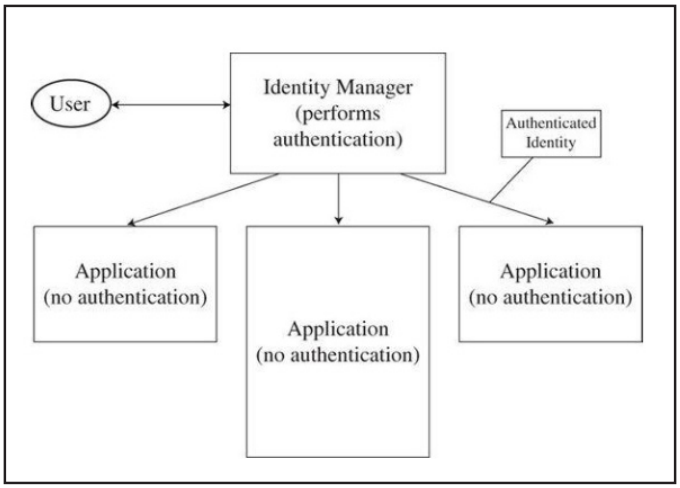


Gambar 2.5 SecurID Token (Foto milik RSA, divisi keamanan EMS)

Kami sekarang telah memeriksa tiga dasar otentikasi: sesuatu yang Anda ketahui, sedang, atau miliki. Digunakan dalam pengaturan yang sesuai, masing-masing dapat menawarkan keamanan yang wajar. Pada bagian selanjutnya kita melihat beberapa cara untuk meningkatkan keamanan dasar dari ketiga bentuk ini.

2.3.1 Manajemen Federated Identity

Federated identity adalah kondisi dimana telah ada saling percaya dari masing masing organisasi/ perusahaan yang terlibat, standarisasi yang sama, infrastruktur yang tersedia dan adanya kolaborasi identity yang sama untuk saling berbagi identity, proses bisnis. Federated menggambarkan proses dan duku-ngan teknologi dimana satu identity dapat digunakan untuk mengakses banyak sumber daya informasi karena banyak variable identity telah di mapping menjadi global identity.



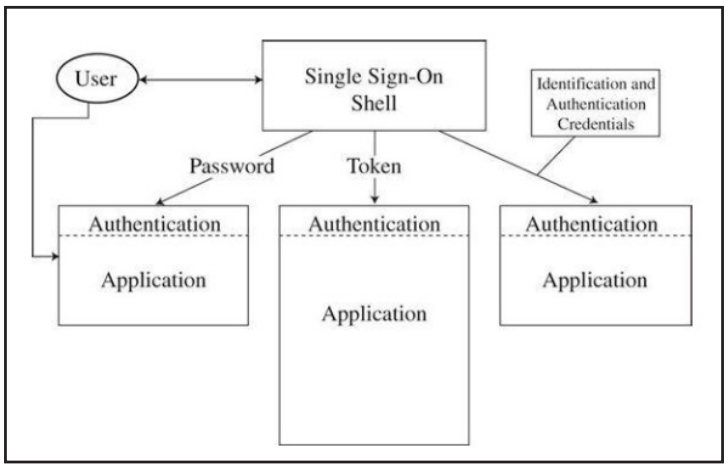
GAMBAR 2-6 Manajer Identitas Federasi

Jika berbagai bentuk otentikasi ini tampak membingungkan dan berlebihan, itu bisa saja terjadi. Pertimbangkan bahwa beberapa sistem akan memerlukan kata sandi, yang lain pemindaian sidik jari, yang lain token aktif, dan lainnya beberapa kombinasi teknik. Seperti yang sudah Anda ketahui, mengingat identitas dan kata sandi yang berbeda untuk banyak sistem itu menantang. Orang-orang

yang harus menggunakan beberapa sistem yang berbeda secara bersamaan (email, pelacakan pelanggan, inventaris, dan penjualan, misalnya) segera menjadi lelah keluar dari satu, ke yang lain, menyegarkan login yang telah habis, dan membuat dan memperbarui profil pengguna. Pengguna berhak meminta komputer untuk menangani pembukuan.

Skema manajemen identitas federasi adalah penyatuan sistem identifikasi dan otentikasi yang terpisah. Alih-alih mempertahankan profil pengguna yang terpisah, skema federasi mempertahankan satu profil dengan satu metode otentikasi. Sistem yang terpisah berbagi akses ke database identitas yang diautentikasi. Dengan demikian, otentikasi dilakukan di satu tempat, dan proses dan sistem yang terpisah menentukan bahwa pengguna yang sudah diautentikasi akan diaktifkan. Proses seperti itu ditunjukkan pada Gambar 2-6.

Terkait erat adalah single sign-on digambarkan pada Gambar 2-7. Pikirkan prosedur payung yang Anda login sekali per sesi (misalnya, sekali sehari).Single Sign-On merupakan fasilitas One Register Multiple Access. Hanya dengan memiliki sebuah Account E-Mail yang telah aktif di UGM, maka pengguna bisa mengakses fasilitas lainnya hanya dengan satu Account tersebut. Misalnya mengakses Internet Messaging Address Book, dan WebMail. (http://im.ugm.ac.id/)



Gambar 2-7 Single Sign-On

Teknologi Single-sign-on (sering disingkat menjadi SSO) adalah teknologi yang mengizinkan pengguna jaringan agar dapat mengakses sumber daya dalam jaringan hanya dengan menggunakan satu akun pengguna saja. Teknologi ini sangat diminati, khususnya dalam jaringan yang sangat besar dan bersifat heterogen (di saat sistem operasi serta aplikasi yang digunakan oleh komputer adalah berasal dari banyak vendor, dan pengguna dimintai untuk mengisi informasi dirinya ke dalam setiap platform yang berbeda tersebut yang hendak diakses oleh pengguna). Dengan menggunakan SSO, seorang pengguna hanya cukup melakukan proses autentikasi sekali saja untuk mendapatkan izin akses terhadap semua layanan yang terdapat di dalam jaringan.

Selain mendatangkan manfaat, SSO juga dapat mendatangkan bencana. Dari cara pandang seperti ini, beberapa pengamat memperkirakan bahwa penggunaan SSO dapat menghemat biaya untuk memelihara password yang rumit yang dapat mencapai ratusan dolar setiap pengguna tiap tahun. Tetapi, implementasi SSO dalam sebuah jaringan yang heterogen adalah rumit, sehingga banyak administrator jaringan kurang begitu giat dalam mengimplementasikannya.

Contoh dari sistem SSO adalah protokol Kerberos, yang telah dimasukkan ke dalam sistem operasi Windows 2000 ke atas. Protokol yang sama dapat juga digunakan di dalam keluarga sistem operasi UNIX. Novell juga telah menawarkan fungsi SSO miliknya sendiri, yang disebut sebagai Novell Single Sign On (NSSO) yang dapat digunakan dalam lingkungan Windows/NetWare. Beberapa perusahaan, seperti Entrust Technologies dan RSA Security menawarkan fungsi SSO yang berbasis kriptografi kunci publik. (http://id.wikipedia.org/wiki/Single-sign-on)

Single Sign On (SSO) adalah sebuah sistem yang memfasilitasi penanganan user account untuk beberapa server dengan hanya menggunakan satu username dan password saja. Sistem ini memiliki beberapa keuntungan, antara lain :

- User tidak perlu mengingat banyak username dan password.
- Kemudahan pemrosesan data. Jika setiap server memiliki data user masing-masing, maka pemrosesan data user (penambahan, pengurangan, perubahan) harus dilakukan pada setiap server yang ada. Sedangkan dengan menggunakan SSO, cukup hanya melakukan 1 kali pemrosesan.

Dua hal diatas jelas menyatakan bahwa penggunaan SSO meningkatkan kepraktisan, Namun di sisi lain, kepraktisan tersebut menimbulkan suatu kelemahan dalam hal keamanan. Jika password system administrator diketahui oleh orang yang tidak berhak, maka orang tersebut dapat melakukan perubahan terhadap semua data yang ada di dalamnya.

Adabeberapa macam framework yang menyediakan keamanan web dan single sign on (SSO), salah satunya adalah JA-SIG Central Authentication Service (CAS) (Aaslund et al, 2007).

Single sign on (SSO) dalam sebuah lingkungan jaringan biasanya menyimpan credentials dalam sebuah server terpusat atau dalam sebuah direktori. Sistem berbasiskan direktori harus menyediakan kemampuan yang tinggi dengan mereplikasi penyimpanan credential. Untuk menyediakan fleksibilitas yang lebih baik, sebuah single sign on harus menyediakan baik server terpusat dan metode dari penyimpanan credential. Lightweight Directory Access Protocol (LDAP) didesain untuk meng-update dan mencari direktori yang berjalan lewat jaringan TCP/IP. Banyak pengembang seperti Microsoft dengan Active Directory, Novell dengan Novell® eDirectory™ dan Netscape dengan Netscape Directory Server semua telah mengambil LDAP sebagai sebuah standar untuk direktori servis. Karena penggunaan yang luas dari LDAP direktori, sebuah produk single sign on harus menyediakan dukungan built-in untuk LDAP sehingga produk ini dapat bekerja secara efektif dengan infrastruktur modern seperti sekarang.

Perbedaan antara kedua pendekatan ini adalah bahwa federated identity management melibatkan modul manajemen identitas tunggal yang menggantikan identifikasi dan otentikasi di semua sistem lain. Jadi semua sistem ini memanggil modul manajemen identitas. Sementara dengan sistem single sing-on, sistem masih meminta identifikasi dan otentikasi individu, tetapi melakukan interaksi tersebut atas nama pengguna.

2.3.2 Autentikasi Multifaktor

Autentikasi merupakan cara untuk mengetahui apakah pengguna yang mengakses ke suatu sistem merupakan pengguna yang berhak dan sesuai atau tidak. Untuk memperkuat keamanan pada suatu sistem, autentikasi terhadap banyak faktor yang berhubungan dengan pengguna digunakan sehingga autentikasi tersebut tidak hanya password saja atau PIN saja.

Pendekatan otentikasi faktor tunggal yang dibahas dalam bab ini menawarkan keuntungan dan kerugian. Misalnya, token hanya berfungsi selama Anda tidak memberikannya (atau kehilangan atau dicuri), dan penggunaan kata sandi gagal jika seseorang dapat melihat Anda memasukkan kata sandi Anda dengan mengintip dari balik bahu Anda. Kita dapat mengkompensasi keterbatasan satu bentuk otentikasi dengan menggabungkannya dengan bentuk lain.

Kartu identitas, seperti SIM, sering kali berisi gambar dan tanda tangan. Kartu itu sendiri adalah token, tetapi siapa pun yang melihat kartu itu dapat membandingkan wajah Anda dengan gambar dan mengonfirmasi bahwa kartu itu milik Anda. Atau orang tersebut dapat meminta Anda untuk menulis nama Anda dan dapat membandingkan tanda tangan. Dengan cara itu, otentikasi berbasis token dan biometrik (karena penampilan Anda dan cara Anda menandatangani nama Anda adalah sifat bawaan Anda). Perhatikan bahwa kartu kredit Anda memiliki ruang untuk tanda tangan Anda di bagian belakang, tetapi di Amerika Serikat beberapa pedagang membandingkan tanda tangan itu dengan slip penjualan yang Anda tanda tangani. Memiliki faktor otentikasi yang tersedia tidak berarti kami menggunakannya.

Selama prosesnya tidak terlalu berat, otentikasi dapat menggunakan dua, tiga, empat, atau lebih faktor. Misalnya, untuk mengakses sesuatu, Anda harus mengetikkan kode rahasia, menggeser lencana Anda, dan memegang tangan Anda di atas piring.

Menggabungkan informasi otentikasi disebut otentikasi multifaktor. Dua bentuk otentikasi (yang, tidak mengherankan, dikenal sebagai otentikasi dua faktor) dianggap lebih baik dari satu, dengan asumsi tentu saja bahwa kedua bentuk itu kuat. Tetapi seiring bertambahnya jumlah formulir, ketidaknyamanan pengguna juga meningkat. Setiap faktor otentikasi memerlukan sistem dan administratornya serta pengguna untuk mengelola lebih banyak informasi keamanan.

Autentikasi biasa digunakan pada sistem yang penggunanya mengakses sistem tersebut dari jauh, atau ketika seorang pengguna mengakses sebagai hak penuh, dan mengakses database yang berisi informasi rahasia.
Dengan Autentikasi banyak faktor memverifikasi identitas pengguna dengan menggunakan dua dari tiga mekanisme berikut:

- Sesuatu yang diketahui pengguna, seperti password atau passphrase atau PIN
- Sesuatu yang dimiliki pengguna, seperti token atau sertifikat software
- Sesuatu yang berada di pengguna, seperti sidik jari atau retina

Ketika autentikasi banyak faktor diterapkan dengan benar, seorang penyerang akan lebih sulit untuk mengakses sistem dan mendapatkan informasi-informasi rahasia. Hal ini disebabkan karena

seorang pengguna harus memktikan bahwa pengguna memiliki akses secara fisik kepada faktor kedua yang dimiliki oleh pengguna (contoh: paspor, token, kartu autentikasi) atau yang berada di diri pengguna (contoh: sidik jari, retina, kontur wajah) atau menggunakan kode yang berubah-ubah setiap kali pengguna ingin mengakses sebuah sistem.

Beberapa kombinasi faktor-faktor autentikasi yang biasa digunakan adalah:

- Token yang dapat membangkitkan angka acak yang digunakan sebagai PIN atau password
- Smartcard yang digunakan bersama dengan smartcard-reader dan PIN atau password
- Kode acak yang berubah-ubah setiap pengguna mengakses sistem yang dikirimkan lewat SMS dan digunakan bersama dengan PIN atau password
- Biometrik seperti sidik jari atau tekstur wajah yang digunakan bersamaan dengan PIN atau password
- Sertifikat software yang disimpan di dalam komputer pengguna atau perangkat media penyimpanan yang dapat diakses dengan menggunakan PIN atau password

Setiap kombinasi autentikasi banyak faktor memiliki kelebihan dan kekurangannya masing-masing. Penggunaannya harus disesuaikan dengan kebutuhan sistem yang akan digunakan. Autentikasi banyak faktor yang penerapannya tidak baik dapat menyebabkan masalah keamanan dan dapat menimbulkan kerentanan terhadap sistem. Autentikasi banyak faktor menjadi sangat efektif apabila salah satu faktornya terpisah secara fisik dari komputer tempat pengguna mengakses sistem, seperti menggunakan token.

Ada beberapa hal yang harus diperhatikan untuk memaksimalkan efektifitas dari autentikasi banya faktor:

- Server tempat autentikasi tidak terhubung ke jaringan lain selain untuk autentikasi. Hal ini dapat dilakukan dengan:
- Mengaplikasikan semua saran keamanan dari produk yang dikeluarkan oleh vendor
- Mengimplementasikan atau membagi-bagi jaringan sesuai dengan ranah kerjanya untuk membatasi mesin dan pengguna yang di dalam jaringan yang dapat mengakses server tempat autentikasi
- Pengguna menggunakan password atau PIN yang sulit dan kompleks sebagai langkah keamanan apabila token atau informasi lainnya mengenai autentikasi dikopi, hilang, atau dicuri
- Jika tidak menggunakan autentikasi banyak faktor, PIN atau password pengguna yang digunakan untuk akses ke sistem dari jauh sebaiknya berbeda dengan password pengguna yang biasa untuk akses ke jaringan.

Pengguna tidak menyimpan perangkat token dengan perangkat tempat pengguna melakukan akses ke sistem dari jauh.

Two Factor Authentication

Mekanisme one time password atau OTP di tahun 1980. Namun tantangan yang di dapat pada tahun 1980-an ini ada dua hal:

- bagaimana metode yang tepat untuk membuat one time password yang baru setiap waktu yang tidak dapat di tebak namun dapat di validasi oleh system yang terpusat?
- bagaimana metode pengiriman one time password yang aman kepada pengguna?

Akhirnya di tahun 1984, Security Dynamics Technologies, Inc. mempaten kan metodologi OTP dengan mempergunakan device hardware yang kemudian di sebut dengan istilah hardware token. Angka OTP yang di generate akan terus berubah berdasarkan waktu atau istilahnya timed-based OTP. Kemunculan teknologi ini menjadi awal dari istilah Multi Factor Authentication yang sekarang ini kita kenal. Meskipun pada era yang sama, di berikan istilah lain, yaitu: Two Factor Authentication. Periode di mana One Time Password menjadi verifikasi tambahan setelah user memasukkan password yang benar.

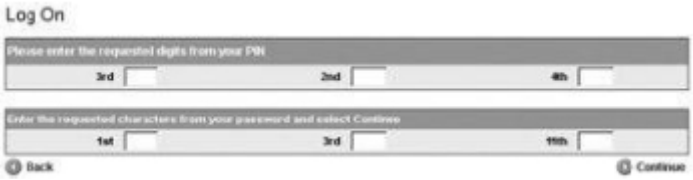
Hingga tahun 2020 ini, ada lima faktor yang di jadikan sebagai bagian dari Multi Factor Authentication. Ke lima faktor tersebut saya sebutkan dalam urutannya adalah:

- ***Something you are***: artinya autentikasi dilakukan dengan membandingkan biometric pengguna data verifikasi yang tersimpan.
- ***Somewhere you are***: artinya proses akan memperhatikan lokasi geografis pengguna saat pengguna melakukan autentikasi. Pengguna yang berasal pada daerah yang memiliki perbedaan lokasi geografis saat proses autentikasi di lakukan akan berpotensi di duga sebagai peretas.
- ***Something you Know***: artinya sesuatu yang diketahui oleh pengguna. Biasanya adalah password, namun pada Studi Kasus lainnya ada yang di sebut dengan istilah PIN.
- ***Something you do***: artinya proses verifikasi dilakukan berdasarkan gesture atau touch yang di lakukan oleh pengguna. Misalnya, pengguna di minta menggambar lingkaran atau bentuk lainnya yang kemudian akan di rekam dan di jadikan bagian dari autentikasi. Kecepatan pengguna dan arah garis yang di gambarkan saat verifikasi akan di jadikan pertimbangan dalam autentikasi.
- ***Something you Have***: artinya sesuatu yang dimiliki oleh pengguna. Ada yang berupa SMS yang diterima. Selain itu ada yang berupa software token yang terinstall pada mobile phone, contohnya Authy, Google Authenticator, dan masih banyak lagi. Sebagian lainnya mempergunakan hardware token yang menampilkan angka yang terus berubah.

Kami berasumsi bahwa lebih banyak faktor menyiratkan kepercayaan diri yang lebih tinggi, meskipun beberapa penelitian mendukung asumsi itu. Dan dua jenis otentikasi menyiratkan dua buah perangkat lunak dan mungkin dua jenis pembaca, serta waktu untuk melakukan dua otentikasi. Memang, bahkan jika otentikasi multifaktor lebih unggul daripada faktor tunggal, kita tidak tahu nilai n mana yang membuat otentikasi n-faktor optimal. Dari sudut pandang kegunaan, nilai n yang besar dapat menyebabkan frustrasi pengguna dan mengurangi keamanan, seperti yang ditunjukkan di Studi Kasus 2-13.

Studi Kasus 2-13 : Ketika Lebih Banyak Faktor Berarti Lebih Sedikit Keamanan

Blog Dave Concannon di www.apeofsteel.com/tag/ulsterbank menggambarkan rasa frustrasinya dalam menggunakan sistem perbankan online Ulsterbank. Proses logon melibatkan beberapa langkah. Pertama, pengguna memberikan nomor identifikasi pelanggan (faktor otentikasi pertama). Selanjutnya, ID pengguna terpisah diperlukan (faktor 2). Ketiga, PIN digunakan untuk memberikan satu set digit (faktor 3), seperti yang ditunjukkan pada gambar di bawah ini: Sistem meminta tiga digit berbeda yang dipilih secara acak (pada gambar, digit ketiga, kedua, dan keempat harus dimasukkan). Terakhir, sistem memerlukan frasa sandi minimal sepuluh karakter, beberapa di antaranya harus berupa angka (faktor 4).



Dalam blognya, Concannon menceritakan tentang kesulitan tidak hanya untuk masuk tetapi juga mengubah kata sandinya. Dengan empat faktor yang perlu diingat, pengguna Ulsterbank kemungkinan besar akan, dengan frustrasi, menuliskan faktor-faktor tersebut dan membawanya di dompet mereka, sehingga mengurangi keamanan sistem perbankan.

2.3.3 Otentikasi Aman

Kata sandi, biometrik, dan token semuanya dapat berpartisipasi dalam otentikasi yang aman. Tentu saja, hanya menggunakan salah satu atau semuanya tidak menjamin bahwa pendekatan otentikasi akan aman. Untuk mencapai keamanan yang sebenarnya, kita perlu berpikir dengan hati-hati tentang masalah yang kita coba pecahkan dan alat yang kita miliki; kita juga perlu memikirkan untuk memblokir kemungkinan serangan dan penyerang.

Misalkan kita ingin mengontrol akses ke sistem komputasi. Selain nama dan kata sandi, kami dapat menggunakan informasi lain yang tersedia untuk mengautentikasi pengguna. Misalkan Adams bekerja di departemen akuntansi selama shift antara 8:00 dan 5:00, Senin sampai Jumat. Setiap upaya akses yang sah oleh Adams harus dilakukan selama waktu itu, melalui stasiun kerja di kantor departemen akuntansi. Dengan membatasi Adams untuk masuk dalam kondisi tersebut, sistem melindungi dari dua masalah:

- Seseorang dari luar mungkin mencoba meniru Adams. Upaya ini akan digagalkan oleh waktu akses atau port tempat akses dicoba.
- Adams mungkin mencoba mengakses sistem dari rumah atau di akhir pekan, berencana menggunakan sumber daya yang tidak diizinkan atau melakukan sesuatu yang terlalu berisiko dengan orang lain di sekitarnya.

Membatasi pengguna ke stasiun kerja tertentu atau waktu akses tertentu dapat menyebabkan komplikasi (seperti ketika pengguna secara sah perlu bekerja lembur, seseorang harus mengakses sistem saat berada di luar kota untuk urusan bisnis, atau stasiun kerja tertentu gagal). Namun, beberapa perusahaan menggunakan teknik otentikasi ini karena keamanan tambahan yang mereka berikan melebihi ketidaknyamanan. Sebagai analis keamanan, kita perlu melatih pikiran kita untuk mengenali kualitas yang membedakan aktivitas normal dan diperbolehkan.

Seperti yang Anda lihat, praktisi keamanan memiliki berbagai mekanisme otentikasi yang siap digunakan. Tidak ada yang sempurna; semua memiliki kekuatan dan kelemahan, dan bahkan kombinasi mekanisme tidak sempurna. Seringkali antarmuka pengguna tampak sederhana dan sangat mudah (apa yang bisa lebih mudah daripada meletakkan jari di atas piring kaca?), tetapi seperti yang telah kami jelaskan, di balik kesederhanaan itu terdapat ketidakpastian, ambiguitas, dan kerentanan. Namun demikian, di bagian ini Anda telah melihat jenis dan contoh sehingga ketika Anda mengembangkan sistem dan aplikasi yang memerlukan otentikasi, Anda akan dapat menggunakan latar belakang ini untuk memilih pendekatan yang memenuhi kebutuhan keamanan Anda.

2.4 Kontrol Akses

Di bagian ini kita membahas bagaimana melindungi objek umum, seperti file, tabel, akses ke perangkat keras atau koneksi jaringan, dan sumber daya lainnya. Secara umum, kami menginginkan struktur yang fleksibel, sehingga pengguna tertentu dapat menggunakan sumber daya dalam satu cara (misalnya, hanya-baca), yang lain dengan cara yang berbeda (misalnya, mengizinkan modifikasi), dan yang lainnya tidak sama sekali. Kami menginginkan teknik yang kuat, mudah digunakan, dan efisien.

Kita mulai dengan paradigma kontrol akses dasar, diartikulasikan oleh Scott Graham dan Peter Denning : Subjek diizinkan untuk mengakses objek dalam mode tertentu, dan hanya akses resmi yang diizinkan.

- Subyek adalah pengguna manusia, sering diwakili oleh program pengganti yang berjalan atas nama pengguna.
- Objek adalah hal-hal di mana suatu tindakan dapat dilakukan: File, tabel, program, objek memori, perangkat keras, string, bidang data, koneksi jaringan, dan prosesor adalah contoh objek. Begitu juga pengguna, atau lebih tepatnya program atau proses yang mewakili pengguna, karena sistem operasi (program yang mewakili administrator sistem) dapat bertindak atas pengguna, misalnya, memungkinkan pengguna untuk menjalankan program, menghentikan pengguna, atau memberikan hak istimewa untuk pengguna.
- Mode akses adalah tindakan subjek yang dapat dikontrol pada objek, termasuk, namun tidak terbatas pada, membaca, menulis, memodifikasi, menghapus, mengeksekusi, membuat, menghancurkan, menyalin, mengekspor, mengimpor, dan sebagainya.

Pemisahan yang efektif akan menjaga subjek yang tidak berwenang dari akses yang tidak sah ke objek, tetapi celah pemisahan harus dilewati untuk subjek dan mode yang berwenang. Di bagian ini kami mempertimbangkan cara untuk mengizinkan semua dan hanya akses yang diizinkan.

2.4.1 Kebijakan Akses

Kontrol akses adalah proses mekanis, mudah diimplementasikan oleh tabel dan proses komputer: Subjek tertentu dapat atau tidak dapat mengakses objek tertentu dengan cara tertentu. Mendasari keputusan langsung adalah keputusan yang kompleks dan bernuansa yang aksesnya harus diizinkan; keputusan ini didasarkan pada kebijakan keamanan formal atau informal.

Keputusan kontrol akses (atau tidak boleh) dibuat secara tiba-tiba. Pat mendapatkan akses ke file ini karena dia mengerjakan proyek yang membutuhkan data; Sol adalah administrator dan harus dapat menambah dan menghapus pengguna untuk sistem. Memiliki basis menyederhanakan pembuatan keputusan serupa untuk pengguna dan objek lain. Kebijakan juga menyederhanakan penetapan aturan kontrol akses, karena hanya mencerminkan kebijakan yang ada.

Jadi, sebelum mencoba menerapkan kontrol akses, organisasi perlu meluangkan waktu untuk mengembangkan kebijakan keamanan tingkat yang lebih tinggi, yang kemudian akan mendorong semua aturan kontrol akses.

Implementasi Kebijakan yang Efektif

Melindungi objek melibatkan beberapa tujuan yang saling melengkapi.

- ***Periksa setiap akses***. Kami mungkin ingin mencabut hak istimewa pengguna untuk mengakses suatu objek. Jika sebelumnya kami telah mengizinkan pengguna untuk mengakses objek, kami tidak bermaksud bahwa pengguna harus mempertahankan akses tak terbatas ke objek. Bahkan, dalam beberapa situasi, kami mungkin ingin mencegah akses lebih lanjut segera setelah kami mencabut otorisasi, misalnya, jika kami mendeteksi pengguna yang menyamar. Untuk alasan ini, kita harus bertujuan untuk memeriksa setiap akses oleh pengguna ke suatu objek.
- ***Terapkan hak istimewa paling rendah***. Prinsip hak istimewa terkecil menyatakan bahwa subjek harus memiliki akses ke jumlah objek terkecil yang diperlukan untuk melakukan beberapa tugas. Bahkan jika informasi tambahan tidak berguna atau tidak berbahaya jika subjek memiliki akses, subjek tidak boleh memiliki akses tambahan itu. Misalnya, sebuah program tidak boleh memiliki akses ke alamat memori absolut yang diterjemahkan oleh referensi nomor halaman, meskipun program tidak dapat menggunakan alamat tersebut dengan cara yang efektif. Tidak mengizinkan akses ke objek yang tidak perlu menjaga dari kelemahan keamanan jika bagian dari mekanisme perlindungan gagal.
- Verifikasi penggunaan yang dapat diterima. Kemampuan untuk mengakses adalah keputusan ya atau tidak. Tetapi yang sama pentingnya adalah memeriksa apakah aktivitas yang akan dilakukan pada suatu objek sudah sesuai. Misalnya, struktur data seperti tumpukan memiliki operasi tertentu yang dapat diterima, termasuk push, pop, clear, dan sebagainya. Kita mungkin ingin tidak hanya mengontrol siapa atau apa yang memiliki akses ke tumpukan tetapi juga untuk memastikan bahwa semua akses yang dilakukan adalah akses tumpukan yang sah.

Pelacakan/Tracking

- Verifikasi penggunaan yang dapat diterima. Kemampuan untuk mengakses adalah keputusan ya atau tidak. Tetapi yang sama pentingnya adalah memeriksa apakah aktivitas yang akan dilakukan pada suatu objek sudah sesuai. Misalnya, struktur data seperti tumpukan memiliki

operasi tertentu yang dapat diterima, termasuk push, pop, clear, dan sebagainya. Kita mungkin ingin tidak hanya mengontrol siapa atau apa yang memiliki akses ke tumpukan tetapi juga untuk memastikan bahwa semua akses yang dilakukan adalah akses tumpukan yang sah.

Menerapkan kebijakan yang tepat bukanlah akhir dari administrasi akses. Terkadang administrator perlu meninjau kembali kebijakan akses untuk menentukan apakah itu berfungsi sebagaimana mestinya. Apakah seseorang telah ada untuk waktu yang lama dan telah memperoleh sejumlah besar hak yang tidak lagi dibutuhkan? Apakah begitu banyak pengguna memiliki akses ke satu objek yang tidak lagi perlu dikendalikan? Atau haruskah itu dipecah menjadi beberapa objek sehingga individu dapat diizinkan mengakses hanya bagian yang mereka butuhkan? Administrator perlu mempertimbangkan pertanyaan semacam ini pada kesempatan untuk menentukan apakah kebijakan dan implementasi melakukan apa yang seharusnya. Kami mengeksplorasi sisi manajemen dalam mendefinisikan kebijakan keamanan di Bab 10, tetapi kami meninjau beberapa masalah di sini karena memiliki pengaruh teknis pada kontrol akses.

Granularitas

Yang kami maksud dengan granularity adalah kehalusan atau kekhususan kontrol akses. Ini adalah spektrum: Di satu ujung Anda dapat mengontrol akses ke setiap bit atau byte, setiap kata dalam dokumen, setiap nomor pada spreadsheet, setiap foto dalam koleksi. Tingkat kekhususan itu umumnya berlebihan dan rumit untuk diterapkan. Semakin halus granularitasnya, semakin besar jumlah keputusan kontrol akses yang harus dibuat, sehingga ada penalti kinerja. Di ekstrem lain Anda cukup mengatakan Adam memiliki akses lengkap ke komputer C1. Pendekatan itu mungkin berhasil jika komputer hanya untuk digunakan Adam, tetapi jika komputer C1 digunakan bersama, maka sistem tidak memiliki dasar untuk mengontrol atau mengatur pembagian itu. Dengan demikian, titik tengah yang masuk akal harus diterapkan.

Biasanya file, program, atau ruang data adalah unit terkecil yang aksesnya dikendalikan. Namun, perhatikan bahwa aplikasi dapat mengimplementasikan kontrol aksesnya sendiri. Jadi, misalnya, seperti yang kami jelaskan di Bab 7, sistem manajemen basis data dapat memiliki akses ke basis data yang lengkap, tetapi sistem tersebut kemudian mengukir basis data menjadi unit yang lebih kecil dan membagi akses: Pengguna ini dapat melihat nama tetapi tidak gaji, pengguna tersebut dapat hanya melihat data karyawan di kantor barat.

Perangkat keras, blok memori, ruang pada disk tempat kode program disimpan, aplikasi khusus, semua ini kemungkinan merupakan objek yang aksesnya dikendalikan.

Akses Log

Setelah membuat keputusan akses, sistem bertindak untuk mengizinkan akses itu dan meninggalkan pengguna dan objek untuk menyelesaikan transaksi. Sistem juga mencatat akses mana yang telah diizinkan, menciptakan apa yang disebut log audit. Log ini dibuat dan dipelihara oleh sistem, dan disimpan untuk analisis selanjutnya. Beberapa alasan untuk akses logging adalah sebagai berikut:

- Catatan akses dapat membantu merencanakan peralatan baru atau yang ditingkatkan, dengan menunjukkan item mana yang telah banyak digunakan.

- Jika sistem gagal, catatan ini dapat menunjukkan akses apa yang sedang berlangsung dan mungkin membantu mengidentifikasi penyebab kegagalan.
- Jika pengguna menyalahgunakan objek, log akses menunjukkan dengan tepat objek mana yang diakses pengguna.
- Dalam hal kompromi eksternal, log audit dapat membantu mengidentifikasi bagaimana penyerang memperoleh akses dan item data mana yang diakses (dan karena itu terungkap atau dikompromikan). Data ini untuk analisis forensik setelah fakta sangat membantu dalam menangani insiden besar.

Sebagai bagian dari aktivitas kontrol akses, sistem membangun dan melindungi log audit ini. Jelas, granularity penting: Sebuah log yang mencatat setiap byte memori yang diakses terlalu panjang untuk menjadi banyak nilai praktis, tetapi log yang mengatakan "8:01 pengguna menyalakan sistem; 17:21 pengguna mematikan sistem" mungkin berisi terlalu sedikit detail untuk membantu.

Pada bagian berikutnya kami mempertimbangkan mekanisme perlindungan yang sesuai untuk objek umum dari jenis yang tidak ditentukan, seperti jenis objek yang tercantum di atas. Untuk membuat penjelasan lebih mudah dipahami, terkadang kita menggunakan contoh objek tertentu, seperti file. Namun, perhatikan bahwa mekanisme umum dapat digunakan untuk melindungi semua jenis objek yang terdaftar.

Hak Istimewa Terbatas

Sistem yang melarang semua akses ke apa pun oleh siapa pun tentu saja mencapai kerahasiaan dan integritas, tetapi sama sekali gagal dalam ketersediaan dan kegunaan. Jadi, kami mencari titik tengah yang menyeimbangkan kebutuhan akan beberapa akses dengan risiko akses yang berbahaya dan tidak pantas. Tentu saja, kami tidak mengharapkan pengguna atau proses menyebabkan kerugian. Namun menyadari bahwa tidak semua pengguna beretika atau bahkan kompeten dan bahwa tidak semua proses berfungsi sebagaimana dimaksud, kami ingin membatasi paparan dari pengguna yang berperilaku tidak semestinya atau proses yang tidak berfungsi. Hak istimewa terbatas adalah cara untuk membatasi eksposur itu.

Hak istimewa terbatas adalah konsep manajemen, bukan kontrol teknis. Proses menganalisis pengguna dan menentukan hak istimewa yang mereka butuhkan adalah langkah pertama yang diperlukan untuk memberi otorisasi dalam batasan tersebut. Setelah menetapkan batasan, kami beralih ke teknologi kontrol akses untuk menerapkan batasan tersebut. Dalam Bab 3 kami kembali mengangkat konsep hak istimewa terbatas ketika kami menjelaskan desain dan implementasi program yang menjamin keamanan. Prinsip desain keamanan yang pertama kali ditulis oleh Jerome Saltzer dan Michael Schroeder [SAL75] menjelaskan keuntungan dari membatasi hak istimewa yang digunakan pengguna dan program mereka untuk dijalankan.

2.4.2 Menerapkan Kontrol Akses

Kontrol akses sering dilakukan oleh sistem operasi. Hanya sistem operasi yang dapat mengakses objek primitif, seperti file, untuk mengontrolnya, dan sistem operasi membuat dan menghentikan program yang mewakili pengguna (subjek). Namun, desain perangkat keras saat ini tidak selalu mendukung sistem operasi dalam mengimplementasikan kontrol akses yang terdiferensiasi dengan

baik atau fine-grained. Sistem operasi biasanya tidak melihat di dalam file atau objek data, misalnya, sehingga tidak dapat melakukan kontrol akses tingkat baris atau elemen dalam database. Juga, sistem operasi tidak dapat dengan mudah membedakan antara jenis lalu lintas jaringan. Dalam Studi Kasus ini, sistem operasi tunduk pada manajer database atau perangkat jaringan dalam menerapkan beberapa aspek kontrol akses. Dengan jenis hak istimewa yang terbatas untuk dialokasikan, sistem operasi tidak dapat dengan mudah mengontrol manajer database dan memungkinkan manajer database mengontrol pengguna. Dengan demikian, desain perangkat keras saat ini membatasi beberapa desain sistem operasi.

Monitor Referensi

James Anderson dan komite studinya (Anderson:72) memberi nama dan struktur pada versi digital dari sebuah konsep yang telah ada selama ribuan tahun. Untuk melindungi benteng abad pertengahan mereka, para penguasa memiliki satu gerbang yang sangat dilindungi sebagai satu-satunya cara masuk. Para jenderal mengepung markas pasukan dengan benteng dan penjaga jaga. Para bankir menyimpan uang tunai dan barang-barang berharga lainnya di brankas dengan pintu yang tidak dapat ditembus yang hanya dimiliki oleh segelintir orang tepercaya. Penjahat dongeng mengunci gadis-gadis di menara. Semua contoh ini menunjukkan kontrol akses yang kuat karena desain yang gagal-aman.

Dalam formulasi Anderson untuk komputer, kontrol akses bergantung pada kombinasi perangkat keras dan perangkat lunak yang:

- selalu dipanggil; memvalidasi setiap upaya akses
- kebal dari gangguan
- pasti benar

Anderson menyebut konstruksi ini sebagai monitor referensi. Harus jelas mengapa ketiga sifat ini penting.

Monitor referensi adalah gagasan, bukan alat yang dapat Anda beli untuk dicolokkan ke port. Itu bisa disematkan dalam aplikasi (untuk mengontrol objek aplikasi), bagian dari sistem operasi (untuk objek yang dikelola sistem) atau bagian dari alat. Namun, Anda akan melihat tiga sifat yang sama ini muncul berulang kali dalam buku ini. Untuk memiliki monitor referensi yang efektif, kita perlu mempertimbangkan cara yang efektif dan efisien untuk menerjemahkan kebijakan, dasar validasi, ke dalam tindakan. Bagaimana kami merepresentasikan kebijakan dalam data biner memiliki implikasi pada seberapa efisien dan bahkan seberapa efektif mediasinya.

Pada bagian berikutnya kami menyajikan beberapa model tentang bagaimana hak akses dapat dipertahankan dan diimplementasikan oleh monitor referensi.

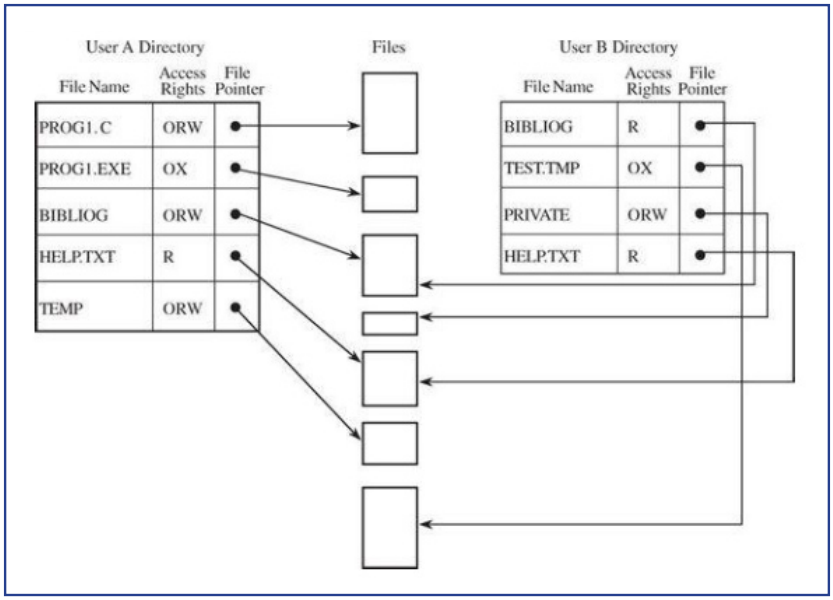
Direktori Kontrol Akses

Salah satu cara sederhana untuk melindungi suatu objek adalah dengan menggunakan mekanisme yang bekerja seperti direktori file. Bayangkan kita mencoba melindungi file (kumpulan objek) dari pengguna sistem komputasi (kumpulan subjek). Setiap file memiliki pemilik unik yang memiliki hak akses "kontrol" (termasuk hak untuk menyatakan siapa yang memiliki akses apa) dan untuk mencabut

akses setiap orang kapan saja. Setiap pengguna memiliki direktori file, yang mencantumkan semua file yang dapat diakses oleh pengguna tersebut.

Jelas, tidak ada pengguna yang diizinkan untuk menulis di direktori file, karena itu akan menjadi cara untuk memalsukan akses ke file. Oleh karena itu, sistem operasi harus memelihara semua direktori file, di bawah perintah dari pemilik file. Hak yang jelas untuk file adalah membaca, menulis, dan mengeksekusi yang umum pada banyak sistem bersama. Selanjutnya, hak lain, pemilik, dimiliki oleh pemilik, yang mengizinkan pengguna tersebut untuk memberikan dan mencabut hak akses. Gambar 2-8 menunjukkan contoh direktori file.

Pendekatan ini mudah diterapkan karena menggunakan satu daftar per pengguna, yang menyebutkan semua objek yang diizinkan untuk diakses oleh pengguna. Namun, beberapa kesulitan dapat muncul. Pertama, daftar menjadi terlalu besar jika banyak objek bersama, seperti pustaka subprogram atau tabel umum pengguna, dapat diakses oleh semua pengguna. Direktori setiap pengguna harus memiliki satu entri untuk setiap objek bersama tersebut, bahkan jika pengguna tidak berniat mengakses objek tersebut. Penghapusan harus tercermin di semua direktori.

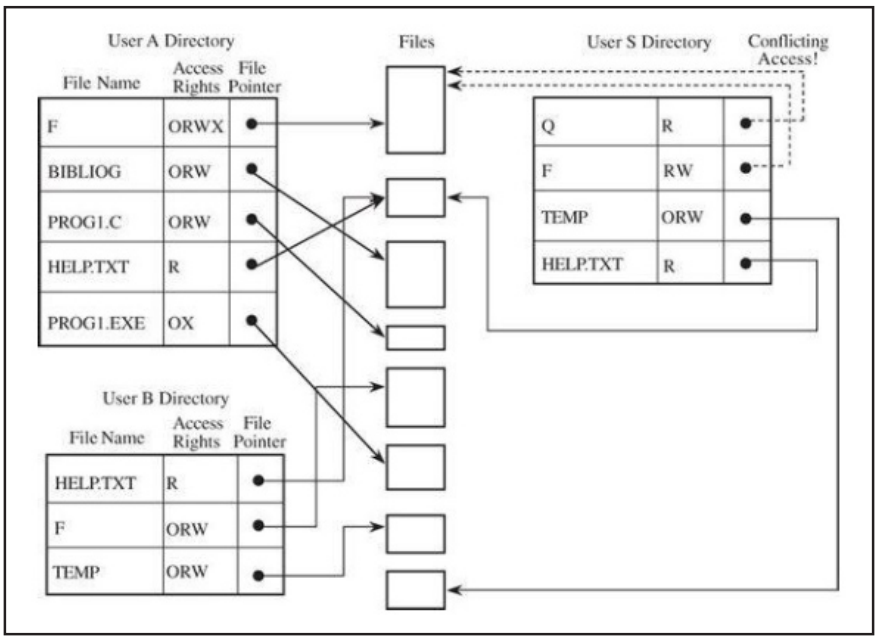


Gambar 2.8 Hak Akses Direktori

Kesulitan kedua adalah pencabutan akses. Jika pemilik A telah memberikan hak kepada pengguna B untuk membaca file F, entri untuk F dibuat dalam direktori untuk B. Pemberian akses ini menyiratkan tingkat kepercayaan antara A dan B. Jika A kemudian mempertanyakan kepercayaan itu, A dapat ingin mencabut hak akses B. Sistem operasi dapat dengan mudah menanggapi permintaan tunggal untuk menghapus hak B untuk mengakses F, karena tindakan itu melibatkan penghapusan satu entri dari direktori tertentu. Tetapi jika A ingin menghapus hak setiap orang untuk mengakses F, sistem operasi harus mencari setiap direktori individu untuk entri F, suatu aktivitas yang dapat memakan waktu pada sistem yang besar. Misalnya, sistem besar atau jaringan sistem yang lebih kecil dapat dengan mudah memiliki 5.000 hingga 10.000 akun aktif. Selain itu, B mungkin telah memberikan hak akses untuk F ke pengguna lain C, situasi yang dikenal sebagai propagasi hak akses, sehingga A mungkin tidak mengetahui bahwa akses C ada dan harus dicabut. Masalah ini sangat serius dalam jaringan.

Kesulitan ketiga melibatkan nama samaran. Pemilik A dan B mungkin memiliki dua file berbeda bernama F, dan mereka berdua mungkin ingin mengizinkan akses oleh S. Jelas, direktori untuk S tidak dapat berisi dua entri dengan nama yang sama untuk file yang berbeda. Oleh karena itu, S harus mampu mengidentifikasi secara unik F untuk A (atau B). Salah satu pendekatan adalah memasukkan penunjukan pemilik asli seolah-olah itu adalah bagian dari nama file, dengan notasi seperti A:F (atau B:F).

Misalkan, bagaimanapun, S ingin menggunakan nama selain F untuk membuat isi file lebih jelas. Sistem dapat mengizinkan S untuk memberi nama F dengan nama apa pun yang unik untuk direktori S. Kemudian, F dari A dapat dipanggil Q ke S. Seperti yang ditunjukkan pada Gambar 2-9, S mungkin lupa bahwa Q adalah F dari A, dan jadi S meminta akses lagi dari A untuk F. Tapi sekarang A mungkin lebih percaya pada S, jadi A mentransfer F dengan hak yang lebih besar dari sebelumnya. Tindakan ini membuka kemungkinan bahwa satu subjek, S, mungkin memiliki dua set hak akses yang berbeda ke F, satu di bawah nama Q dan satu di bawah nama F. Dengan cara ini, mengizinkan nama samaran dapat menyebabkan beberapa izin yang belum tentu konsisten. Dengan demikian, pendekatan direktori mungkin terlalu sederhana untuk sebagian besar situasi perlindungan objek.

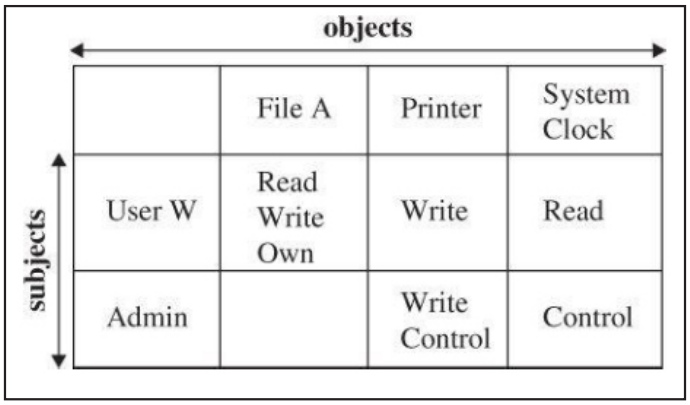


Gambar 2.9 Hak Akses Ambigu

Matriks Kontrol Akses

Kita dapat menganggap direktori sebagai daftar objek yang dapat diakses oleh satu subjek, dan daftar akses sebagai tabel yang mengidentifikasi subjek yang dapat mengakses satu objek. Data dalam dua representasi ini setara, perbedaannya adalah kemudahan penggunaan dalam situasi tertentu.

Sebagai alternatif, kita dapat menggunakan matriks kontrol akses, yang ditunjukkan pada Gambar 2-10, sebuah tabel di mana setiap baris mewakili subjek, setiap kolom mewakili objek, dan setiap entri adalah kumpulan hak akses untuk subjek itu ke objek itu.



Gambar 2.10 Matriks Kontrol Akses

Contoh representasi matriks kontrol akses yang lebih rinci ditunjukkan pada Tabel 2-8. Hak akses yang ditunjukkan pada tabel tersebut adalah O, own; R, baca; W, tulis; dan X, jalankan. Secara umum, matriks kontrol akses jarang (artinya sebagian besar sel kosong): Sebagian besar subjek tidak memiliki hak akses ke sebagian besar objek. Matriks akses dapat direpresentasikan sebagai daftar tiga kali lipat, masing-masing memiliki bentuk subjek, objek, hak, seperti yang ditunjukkan pada Tabel 2-9.

Tabel 2.8 Matriks Kontrol Akses

	Bibliog	Temp	F	Help .txt	C_ Comp	Linker	Clock	Printer
USER A	ORW	ORW	ORW	R	X	X	R	W
USER B	R	—	—	R	X	X	R	W
USER S	RW	—	R	R	X	X	R	W
USER T	—	—	R	X	X	X	R	W
SYS MGR	—	—	—	RW	OX	OX	ORW	O
USER SVCS	—	—	—	O	X	X	R	W

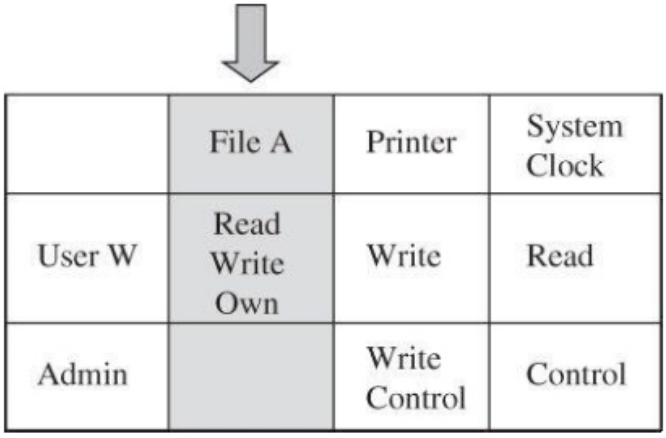
Tabel 2.9 Daftar Tiga Kali Kontrol Akses

Subject	Object	Right
USER A	Bibliog	ORW
USER B	Bibliog	R
USER S	Bibliog	RW
USER A	Temp	ORW
USER A	F	ORW
USER S	F	R
etc.		

Representasi ini mungkin lebih efisien daripada matriks kontrol akses karena tidak ada triple untuk setiap sel kosong dari matriks (seperti USER T, Bibliog, –). Meskipun triple dapat diurutkan berdasarkan subjek atau objek sesuai kebutuhan, pencarian sejumlah besar dari triple ini cukup tidak efisien sehingga implementasi ini jarang digunakan.

Daftar Kontrol Akses

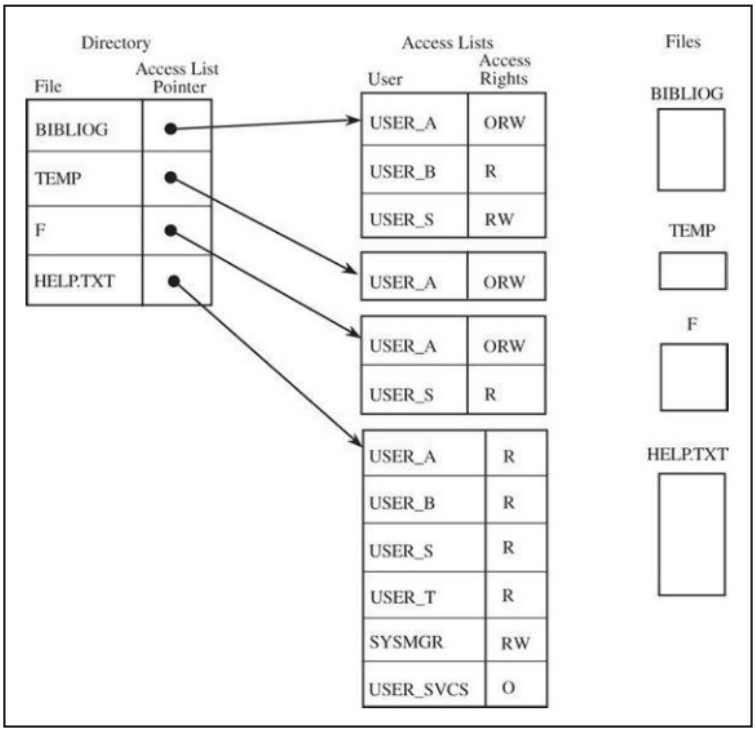
Representasi alternatif adalah daftar kontrol akses; seperti yang ditunjukkan pada Gambar 2-11, representasi ini sesuai dengan kolom dari matriks kontrol akses. Ada satu daftar tersebut untuk setiap objek, dan daftar tersebut menunjukkan semua subjek yang harus memiliki akses ke objek dan apa akses mereka. Pendekatan ini berbeda dari daftar direktori karena ada satu daftar kontrol akses per objek; direktori dibuat untuk setiap mata pelajaran. Meskipun perbedaan ini tampak kecil, ada beberapa keuntungan signifikan dari pendekatan ini.



	File A	Printer	System Clock
User W	Read Write Own	Write	Read
Admin		Write Control	Control

Gambar 2-.11 Daftar Kontrol Akses

Representasi daftar kontrol akses dapat menyertakan hak default. Pertimbangkan subjek A dan S, keduanya memiliki akses ke objek F. Sistem operasi akan mempertahankan hanya satu daftar akses untuk F, menunjukkan hak akses untuk A dan S, seperti yang ditunjukkan pada Gambar 2-12. Daftar kontrol akses dapat menyertakan entri default umum untuk setiap pengguna. Dengan cara ini, pengguna tertentu dapat memiliki hak eksplisit, dan semua pengguna lain dapat memiliki serangkaian hak default. Dengan organisasi ini, semua pengguna sistem yang mungkin dapat berbagi file atau program publik tanpa memerlukan entri untuk objek di direktori individu setiap pengguna.



Gambar 2.12 Daftar Kontrol Akses dengan Dua Subjek

Sistem operasi Multics menggunakan bentuk daftar kontrol akses di mana setiap pengguna termasuk dalam tiga kelas perlindungan: pengguna, grup, dan kompartemen. Penunjukan pengguna mengidentifikasi subjek tertentu, dan penunjukan grup menyatukan subjek yang memiliki minat yang sama, seperti menjadi rekan kerja dalam sebuah proyek. Kompartemen membatasi objek yang tidak dipercaya; program yang dieksekusi di satu kompartemen tidak dapat mengakses objek di kompartemen lain tanpa izin khusus. Kompartemen juga merupakan cara untuk mengumpulkan objek yang terkait, seperti semua file untuk satu proyek.

Untuk melihat bagaimana jenis perlindungan ini dapat bekerja, anggaplah setiap pengguna yang memulai akses ke sistem mengidentifikasi grup dan kompartemen yang dapat digunakan untuk bekerja. Jika Adams masuk sebagai pengguna Adams di grup Decl dan kompartemen Art2, hanya objek yang memiliki Adams-Decl-Art2 dalam daftar kontrol akses yang dapat diakses dalam sesi.

Dengan sendirinya, mekanisme semacam ini akan terlalu membatasi untuk dapat digunakan. Adams tidak dapat membuat file umum untuk digunakan dalam sesi apa pun. Lebih buruk lagi, objek bersama tidak hanya harus mencantumkan Adams sebagai subjek yang sah tetapi juga harus mencantumkan Adams di bawah semua kelompok yang dapat diterima dan semua kompartemen yang dapat diterima untuk setiap kelompok.

Solusinya adalah penggunaan kartu liar, yang berarti placeholder yang menunjuk "pengguna mana pun" (atau "grup apa pun" atau "kompartemen apa pun"). Daftar kontrol akses mungkin menentukan akses oleh Adams-Decl-Art1, memberikan hak khusus kepada Adams jika bekerja dalam grup Decl pada kompartemen Art1. Daftar mungkin juga menentukan Adams-*-Art1, artinya Adams dapat mengakses objek dari grup mana pun di kompartemen Art1. Demikian juga, notasi *-Decl-* akan berarti "pengguna mana pun dalam grup Decl di kompartemen mana pun." Penempatan yang berbeda dari notasi wildcard * memiliki interpretasi yang jelas.

Unix menggunakan pendekatan serupa dengan izin pengguna-grup-dunia. Setiap pengguna termasuk dalam grup pengguna terkait—siswa di kelas yang sama, pekerja di proyek bersama, atau anggota departemen yang sama. Izin akses untuk setiap objek adalah triple (u,g,w) di mana u adalah untuk hak akses pengguna, g untuk anggota grup lainnya, dan w untuk semua pengguna lain di dunia.

Daftar kontrol akses dapat dipertahankan dalam urutan yang diurutkan, dengan * diurutkan sebagai datang setelah semua nama tertentu. Misalnya, Adams-Decl-* akan muncul setelah semua penunjukan kompartemen khusus untuk Adams. Pencarian izin akses berlanjut hanya sampai pertandingan pertama. Dalam protokol, semua penunjukan eksplisit diperiksa sebelum wildcard di posisi apa pun, sehingga hak akses tertentu akan didahulukan daripada hak wildcard. Entri terakhir pada daftar akses dapat berupa *-*-*, yang menentukan hak yang diizinkan untuk pengguna mana pun yang tidak secara eksplisit ada dalam daftar akses. Dengan perangkat wildcard ini, objek publik bersama dapat memiliki daftar akses yang sangat pendek, secara eksplisit menyebutkan beberapa subjek yang seharusnya memiliki hak akses yang berbeda dari default.

Daftar Hak Istimewa

Daftar hak istimewa, kadang-kadang disebut direktori, adalah deretan matriks akses, yang menunjukkan semua hak istimewa atau hak akses untuk subjek tertentu (ditunjukkan pada Gambar 2-13). Salah satu keuntungan dari daftar hak istimewa adalah kemudahan pencabutan: Jika pengguna

dihapus dari sistem, daftar hak istimewa menunjukkan semua objek yang dapat diakses pengguna sehingga hak tersebut dapat dihapus dari objek.

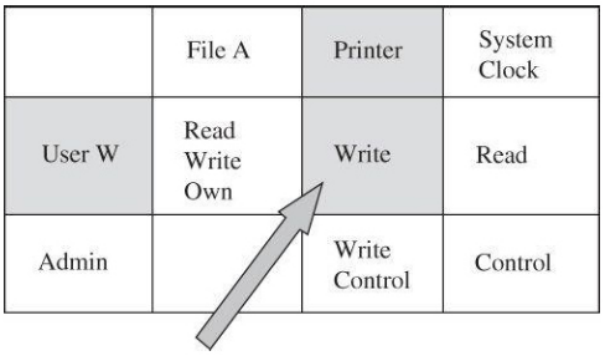


	File A	Printer	System Clock
User W	Read Write Own	Write	Read
Admin		Write Control	Control

Gambar 2.13 Daftar Kontrol Hak Istimewa

2.4.3 Kemampuan (*Capability*)

Sejauh ini, kami telah memeriksa skema perlindungan di mana sistem operasi harus melacak semua objek dan hak perlindungan. Tetapi pendekatan lain memberikan beberapa beban pada pengguna. Misalnya, pengguna mungkin diharuskan memiliki tiket atau pass yang memungkinkan akses, seperti tiket atau kartu identitas yang tidak dapat digandakan.



	File A	Printer	System Clock
User W	Read Write Own	Write	Read
Admin		Write Control	Control

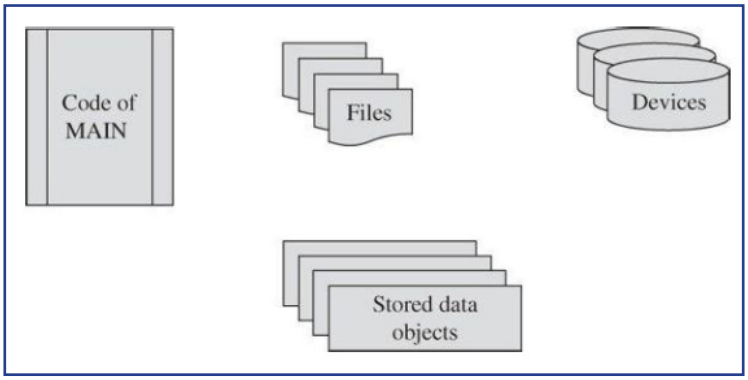
Gambar 2.14 Kemampuan

Secara lebih formal, kami mengatakan bahwa kapabilitas adalah tanda yang tidak dapat dipalsukan yang memberikan pemiliknya hak tertentu atas suatu objek. Sistem Multics, dan Hydra menggunakan kemampuan untuk kontrol akses. Seperti yang ditunjukkan pada Gambar 2-14, kapabilitas hanyalah satu kontrol akses tiga kali lipat dari subjek, objek, dan hak. Secara teori, subjek dapat membuat objek baru dan dapat menentukan operasi yang diizinkan pada objek tersebut. Misalnya, pengguna dapat membuat objek seperti file, segmen data, atau subproses dan juga dapat menentukan jenis operasi yang dapat diterima, seperti membaca, menulis, dan mengeksekusi. Tetapi pengguna juga dapat membuat objek yang benar-benar baru, seperti struktur data baru, dan dapat menentukan jenis akses yang sebelumnya tidak diketahui oleh sistem.

Pikirkan kemampuan sebagai tiket yang memberikan izin kepada subjek untuk memiliki jenis akses tertentu ke suatu objek. Untuk kemampuan menawarkan perlindungan yang solid, tiket harus tidak dapat dipalsukan. Salah satu cara untuk membuatnya tidak dapat dimaafkan adalah dengan tidak memberikan tiket secara langsung kepada pengguna. Sebaliknya, sistem operasi menyimpan semua tiket atas nama pengguna. Sistem operasi mengembalikan kepada pengguna sebuah penunjuk ke struktur data sistem operasi, yang juga menautkan ke pengguna. Kemampuan hanya dapat dibuat dengan permintaan khusus dari pengguna ke sistem operasi. Setiap kemampuan juga mengidentifikasi akses yang diizinkan. Atau, kemampuan dapat dienkripsi di bawah kunci yang hanya tersedia untuk mekanisme kontrol akses. Jika kemampuan terenkripsi berisi identitas pemilik yang sah, pengguna A tidak dapat menyalin kemampuan dan memberikannya kepada pengguna B.

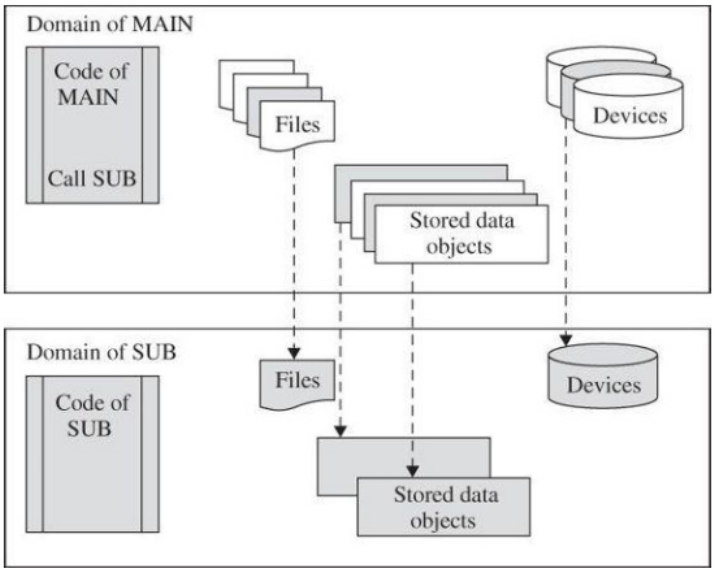
Salah satu kemungkinan hak akses ke suatu objek adalah transfer atau propagasi. Subjek yang memiliki hak ini dapat memberikan salinan kemampuan kepada subjek lain. Pada gilirannya, masing-masing kemampuan ini juga memiliki daftar jenis akses yang diizinkan, salah satunya mungkin juga transfer. Dalam hal ini, proses A dapat mengirimkan salinan kemampuan ke B, yang kemudian dapat meneruskan salinan ke C. B dapat mencegah distribusi lebih lanjut dari kemampuan (dan karena itu mencegah penyebaran lebih lanjut dari hak akses) dengan menghilangkan hak transfer dari hak yang diberikan dalam kemampuan ke C. B mungkin masih memberikan hak akses tertentu ke C, tetapi bukan hak untuk menyebarkan hak akses ke subjek lain.

Saat proses dijalankan, ia beroperasi di domain atau ruang nama lokal. Domain adalah kumpulan objek yang dapat diakses oleh proses. Domain untuk pengguna pada waktu tertentu mungkin mencakup beberapa program, file, segmen data, dan perangkat I/O seperti printer dan terminal. Contoh domain ditunjukkan pada Gambar 2-15.



Gambar 2.15 Contoh Domain

Saat eksekusi berlanjut, proses dapat memanggil subprosedur, melewati beberapa objek yang memiliki akses sebagai argumen ke subprosedur. Domain subprosedur tidak harus sama dengan prosedur pemanggilannya; pada kenyataannya, prosedur pemanggilan hanya dapat melewatkan beberapa objeknya ke subprosedur, dan subprosedur mungkin memiliki hak akses ke objek lain yang tidak dapat diakses oleh prosedur pemanggilan, seperti yang ditunjukkan pada Gambar 2-16.



Gambar 2.16 Melewati Objek ke Domain

Penelepon juga dapat melewati hanya sebagian dari hak aksesnya untuk objek yang diteruskannya ke subprosedur. Sebagai contoh, suatu prosedur dapat memberikan hak untuk membaca kepada subprosedur tetapi tidak untuk mengubah nilai data tertentu.

Karena setiap kapabilitas mengidentifikasi satu objek dalam domain, kumpulan kapabilitas mendefinisikan domain. Ketika suatu proses memanggil subprosedur dan meneruskan objek tertentu ke subprosedur, sistem operasi membentuk tumpukan semua kemampuan prosedur saat ini. Sistem operasi kemudian menciptakan kemampuan baru untuk subprosedur.

Secara operasional, kemampuan adalah cara mudah untuk melacak hak akses subjek ke objek selama eksekusi. Kemampuan didukung oleh tabel yang lebih komprehensif, seperti matriks kontrol akses atau daftar kontrol akses. Setiap kali suatu proses berusaha menggunakan objek baru, sistem operasi memeriksa daftar induk objek dan subjek untuk menentukan apakah objek tersebut dapat diakses. Jika demikian, sistem operasi menciptakan kemampuan untuk objek itu.

Kemampuan harus disimpan dalam memori yang tidak dapat diakses oleh pengguna biasa. Salah satu cara untuk mencapai ini adalah dengan menyimpan kapabilitas dalam segmen yang tidak ditunjuk oleh tabel segmen pengguna atau untuk memasukkannya ke dalam memori yang dilindungi seperti dari sepasang register basis/batas. Pendekatan lain adalah dengan menggunakan mesin arsitektur yang ditandai untuk mengidentifikasi kemampuan sebagai struktur yang membutuhkan perlindungan. Selama eksekusi, hanya kemampuan objek yang telah diakses oleh proses saat ini yang tetap tersedia. Pembatasan ini meningkatkan kecepatan akses ke objek dapat diperiksa.

Kemampuan dapat dicabut. Ketika subjek penerbit mencabut suatu kapabilitas, tidak boleh ada akses lebih lanjut di bawah kapabilitas yang dicabut tersebut. Tabel kapabilitas dapat berisi petunjuk ke kapabilitas aktif yang muncul di bawahnya sehingga sistem operasi dapat melacak hak akses apa yang harus dihapus jika kapabilitas dicabut. Masalah serupa adalah menghapus kemampuan untuk pengguna yang tidak lagi aktif.

Ketiga struktur dasar ini, direktori, matriks kontrol akses dan subsetnya, serta kapabilitas, merupakan dasar dari sistem kontrol akses yang diterapkan saat ini. Terlepas dari implementasi mekanis dari matriks kontrol akses atau substrukturnya, dua model akses berhubungan lebih khusus dengan tujuan kontrol akses: menghubungkan akses ke peran subjek atau konteks akses. Kami menyajikan model-model itu selanjutnya.

2.4.4 Procedure-Oriented Access Control

Salah satu tujuan dari kontrol akses adalah membatasi tidak hanya subjek yang memiliki akses ke suatu objek, tetapi juga apa yang dapat mereka lakukan terhadap objek tersebut. Akses baca versus tulis dapat dikontrol dengan lebih mudah oleh sebagian besar aplikasi dan sistem operasi, tetapi kontrol yang lebih kompleks tidak mudah dicapai.

Paradigma Prosedural atau dikenal juga dengan paradigma imperatif menggunakan metode pemrograman dengan mengeluarkan perintah yang akan dieksekusi oleh komputer. Baris demi baris dieksekusi secara berurutan mulai dari baris atas hingga bawah, dimana semua data dan kode digabung menjadi satu bagian dalam satu program. Menurut – Wikipedia Procedural Programming

(PP) adalah paradigma pemrograman, yang berasal dari pemrograman terstruktur, berdasarkan konsep pemanggilan prosedur. Procedural programming, juga dikenal sebagai rutinitas, subrutin, atau fungsi, cukup berisi serangkaian langkah komputasi yang harus dilakukan. Bahasa pemrograman prosedural utama pertama muncul sekitar tahun 1957–1964, termasuk Fortran, ALGOL, COBOL, PL / I dan BASIC. Pascal dan C diterbitkan sekitar tahun 1970-1972.

Procedural programming merupakan pemrograman secara inline dengan menggunakan konsep top-down approach dengan kata lain, PP dapat diartikan dengan memberi intruksi berupa tahapan kegiatan untuk menyelesaikan suatu aktivitas tergantung procedures dan routines terkait.

Dengan Procedure-OrientedAccess Control, menyiratkan adanya prosedur yang mengontrol akses ke objek (misalnya, dengan melakukan otentikasi penggunanya sendiri untuk memperkuat perlindungan dasar yang disediakan oleh sistem operasi dasar). Intinya, prosedur membentuk kapsul di sekitar objek, hanya mengizinkan akses tertentu yang ditentukan.

Prosedur dapat memastikan bahwa akses ke objek dilakukan melalui antarmuka tepercaya. Misalnya, baik pengguna maupun rutinitas sistem operasi umum mungkin tidak diizinkan mengakses langsung ke tabel pengguna yang valid. Sebaliknya, satu-satunya akses yang diizinkan mungkin melalui tiga prosedur: satu untuk menambahkan pengguna, satu untuk menghapus pengguna, dan satu untuk memeriksa apakah nama tertentu sesuai dengan pengguna yang valid. Prosedur ini, terutama menambah dan menghapus, dapat menggunakan pemeriksaan mereka sendiri untuk memastikan bahwa panggilan ke mereka sah.

Proteksi berorientasi prosedur menerapkan prinsip penyembunyian informasi karena cara mengimplementasikan objek hanya diketahui oleh prosedur kontrol objek. Tentu saja, tingkat perlindungan ini membawa konsekuensi ketidakefisienan. Dengan perlindungan berorientasi prosedur, tidak ada pemeriksaan akses yang sederhana dan cepat, bahkan jika objek tersebut sering digunakan.

2.4.5 Role Based Access Control (RBAC)

Kami belum membedakan jenis pengguna, tetapi kami ingin beberapa pengguna (seperti administrator) memiliki hak istimewa yang signifikan, dan kami ingin orang lain (seperti pengguna biasa atau tamu) memiliki hak istimewa yang lebih rendah. Di perusahaan dan institusi pendidikan, ini bisa menjadi rumit ketika pengguna biasa menjadi administrator atau pembuat roti pindah ke grup pembuat kandil. Kontrol akses berbasis peran memungkinkan kami mengaitkan hak istimewa dengan grup, seperti semua administrator dapat melakukan ini atau pembuat candlestick dilarang melakukan itu. Mengelola keamanan lebih mudah jika kita dapat mengontrol akses berdasarkan tuntutan pekerjaan, bukan oleh orang. Kontrol akses mengikuti orang yang mengubah tanggung jawab, dan administrator sistem tidak harus memilih pengaturan kontrol akses yang sesuai untuk seseorang. Untuk detail selengkapnya tentang nuansa kontrol akses berbasis peran.

RBAC memberikan hak akses untuk roles bukan menerapkan hak akses pelaku atau subjek. Pengguna tidak bisa mengambil hak akses pengguna lain merupakan dasar perbedaan antara RBAC dan DAC. Dalam organisasi, roles dibuat untuk fungsi kerja yang berbeda sesuai dengan peran keanggotaan yang didasarkan pada kompetensi, tugas, dan kewenangan. Tugas keamanan

RBAC diberikan pada kontrol akses sebagai prioritas tertinggi untuk mengontrol akses ke sumber daya. Administrator sangat berperan dalam memberikan hak kepada pengguna, sehingga subjek akan mendapatkan akses ke objek melalui role yang telah diberikan oleh administrator. Struktur RBAC meliputi roles, permission, user, dan session. RBAC memerlukan penunggasan hak akses untuk role pada setiap subjek, dengan cara menghubungkan subjek dengan role. Hak akses yang ditugaskan ke roles akan digunakan oleh pengguna untuk membuat sebuah session, dan session akan mendapatkan izin (permission) dari role yang didapatkan oleh pengguna untuk mengaktifkan perannya (roles).

Dalam RBAC, role didefinisikan sebagai suatu gagasan yang merupakan dasar dari kebijakan kontrol akses (Sandhu & Bhamidipati, 1999). Role didefinisikan dalam ilmu perilaku sebagai pola yang ditentukan sesuai perilaku seseorang dalam situasi tertentu berdasarkan posisi orang .

Kesimpulannya, studi kami tentang mekanisme kontrol akses telah dengan sengaja berkembang dari yang sederhana ke yang kompleks. Secara historis, karena mekanisme telah memberikan fleksibilitas yang lebih besar, mereka melakukannya dengan harga overhead yang meningkat. Misalnya, mengimplementasikan kemampuan yang harus diperiksa pada setiap akses jauh lebih sulit daripada menerapkan struktur direktori sederhana yang diperiksa hanya pada akses pertama subjek ke objek. Kompleksitas ini jelas bagi pengguna dan pelaksana. Pengguna mengetahui fitur perlindungan tambahan, tetapi pengguna yang naif mungkin merasa frustrasi atau terintimidasi karena harus memilih opsi perlindungan dengan sedikit pemahaman tentang kegunaannya. Kompleksitas implementasi menjadi jelas dalam respons yang lambat terhadap pengguna. Keseimbangan antara kesederhanaan dan fungsionalitas adalah perjuangan berkelanjutan dalam keamanan.

2.5 Kriptografi

Selanjutnya kami memperkenalkan yang ketiga dari alat keamanan dasar kami, kriptografi. Dalam bab ini kami hanya menyajikan dasar-dasar topik, cukup sehingga Anda dapat melihat bagaimana hal itu dapat digunakan dan apa yang dapat dicapai. Kami meninggalkan internal untuk Bab 12 di akhir buku ini. Kami melakukan itu karena sebagian besar praktisi keamanan komputer akan kesulitan untuk menjelaskan atau menerapkan kriptografi yang baik dari awal, yang membuat poin kami bahwa Anda tidak perlu memahami internal kriptografi hanya untuk menggunakannya dengan sukses. Saat Anda membaca bab ini, Anda mungkin bertanya mengapa sesuatu dilakukan dengan cara tertentu atau bagaimana sesuatu benar-benar bekerja. Bab ini berfokus pada alat dan penggunaannya, meninggalkan cara kerja internal untuk masa depan.

Enkripsi atau kriptografi—nama yang berarti penulisan rahasia—mungkin merupakan pertahanan terkuat dalam gudang perlindungan keamanan komputer. Data yang disamarkan dengan baik tidak dapat dengan mudah dibaca, dimodifikasi, atau dibuat-buat. Sederhananya, enkripsi seperti mesin: Anda meletakkan data di satu ujung, roda gigi berputar dan lampu berkedip, dan Anda menerima data yang dimodifikasi di ujung lainnya. Faktanya, beberapa perangkat enkripsi yang digunakan selama Perang Dunia II beroperasi dengan roda gigi dan rotor yang sebenarnya, dan perangkat ini efektif untuk menghalangi (walaupun tidak selalu mencegah) pihak yang berlawanan untuk membaca pesan yang dilindungi. Sekarang mesin telah digantikan oleh algoritma komputer, tetapi prinsipnya sama: Transformasi membuat data sulit untuk ditafsirkan oleh orang luar.

Kita mulai dengan memeriksa apa yang dilakukan enkripsi dan bagaimana cara kerjanya. Kami memperkenalkan prinsip-prinsip dasar algoritma enkripsi, memperkenalkan dua jenis enkripsi dengan kegunaan yang berbeda. Karena enkripsi yang lemah atau cacat hanya menciptakan ilusi perlindungan, kami juga melihat bagaimana enkripsi bisa gagal. Kami menjelaskan secara singkat teknik yang digunakan untuk menembus tutup pelindung untuk membatalkan keamanan. Berdasarkan jenis enkripsi dasar ini, kami menunjukkan cara menggabungkannya untuk mengatasi beberapa masalah umum komputasi dan komunikasi dengan aman.

2.5.1 Masalah yang Ditangani dengan Enkripsi

Terkadang kami menggambarkan enkripsi dalam konteks pengiriman pesan rahasia. Pembingkaian ini hanya untuk memudahkan deskripsi: Konsep yang sama berlaku untuk melindungi file data atau informasi sensitif dalam memori. Jadi ketika kita berbicara tentang mengirim pesan, Anda juga harus memikirkan untuk melindungi objek digital apa pun untuk akses hanya oleh orang yang berwenang.

Pertimbangkan langkah-langkah yang terlibat dalam pengiriman pesan dari pengirim, S, ke penerima, R. Jika S mempercayakan pesan ke T, yang kemudian mengirimkannya ke R, T kemudian menjadi media transmisi. Jika orang luar, O, ingin mengakses pesan (untuk membaca, mengubah, atau bahkan menghancurkannya), kami menyebut O sebagai pencegat atau penyusup. Setiap saat setelah S mengirimkan pesan melalui T, ia rentan terhadap eksploitasi, dan O mungkin mencoba mengaksesnya dengan salah satu cara berikut:

- memblokirnya, dengan mencegahnya mencapai R, sehingga mempengaruhi ketersediaan pesan
- mencegatnya, dengan membaca atau mendengarkan pesan, sehingga mempengaruhi kerahasiaan pesan
- memodifikasinya, dengan mengambil pesan dan mengubahnya dengan cara tertentu, yang mempengaruhi integritas pesan
- membuat pesan yang tampak otentik, mengaturnya agar disampaikan seolah-olah itu berasal dari S, sehingga juga mempengaruhi integritas pesan

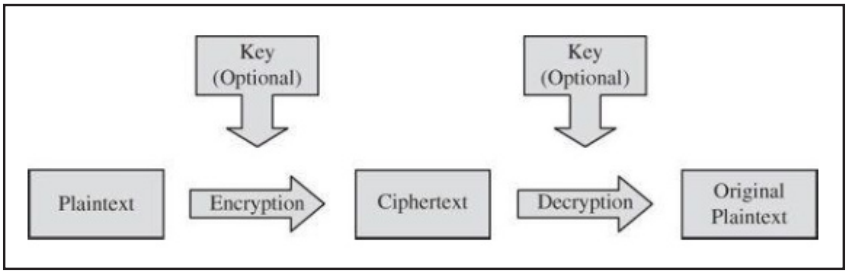
Seperti yang Anda lihat, kerentanan pesan mencerminkan empat kemungkinan kegagalan keamanan yang kami identifikasi di Bab 1. Untungnya, enkripsi adalah teknik yang dapat mengatasi semua masalah ini. Enkripsi adalah cara untuk menjaga keamanan data di lingkungan yang tidak aman. Dalam buku ini, kita mempelajari enkripsi sebagai teknik keamanan, dan kita melihat bagaimana enkripsi digunakan dalam melindungi program, database, jaringan, dan komunikasi elektronik.

Terminologi

Enkripsi adalah proses penyandian pesan sehingga maknanya tidak jelas; dekripsi adalah proses sebaliknya, mengubah pesan terenkripsi kembali ke bentuk aslinya yang normal. Sebagai alternatif, istilah encode dan decode atau encipher dan decipher digunakan sebagai ganti mengenkripsi dan mendekripsi. Artinya, kita katakan kita menyandikan, mengenkripsi, atau menyandikan pesan asli untuk menyembunyikan maknanya. Kemudian, kami mendekode, mendekripsi, atau menguraikannya untuk mengungkapkan pesan asli. Sebuah sistem untuk enkripsi dan dekripsi disebut kriptosistem.

Terdapat sedikit perbedaan makna dari ketiga pasangan kata encode dan decode atau encipher dan decipher, meskipun tidak signifikan dalam konteks pembahasan ini. Tegasnya, encoding adalah proses menerjemahkan seluruh kata atau frasa ke kata atau frasa lain, sedangkan enciphering adalah menerjemahkan huruf atau simbol secara individual; enkripsi adalah istilah grup yang mencakup pengkodean dan penyandian.

Bentuk asli dari sebuah pesan dikenal sebagai plaintext, dan bentuk terenkripsi disebut ciphertext. Hubungan ini ditunjukkan pada Gambar 2-18. Pikirkan enkripsi sebagai bentuk cat buram yang mengaburkan atau menghilangkan teks biasa, mencegahnya terlihat atau ditafsirkan secara akurat. Kemudian, dekripsi adalah proses pengelupasan cat untuk mengungkapkan kembali plaintext asli.



Gambar 2.18 Plaintext dan Ciphertext

Kami menggunakan notasi formal untuk menggambarkan transformasi antara plaintext dan ciphertext. Sebagai contoh, kita menulis C = E(P) dan P = D(C), di mana C mewakili ciphertext, E adalah aturan enkripsi, P adalah plaintext, dan D adalah aturan dekripsi. Apa yang kita cari adalah kriptosistem yang P = D(E(P)). Dengan kata lain, kami ingin dapat mengubah pesan plaintext menjadi ciphertext untuk melindunginya dari penyusup, tetapi kami juga ingin mendapatkan kembali pesan asli sehingga penerima dapat membacanya dengan benar.

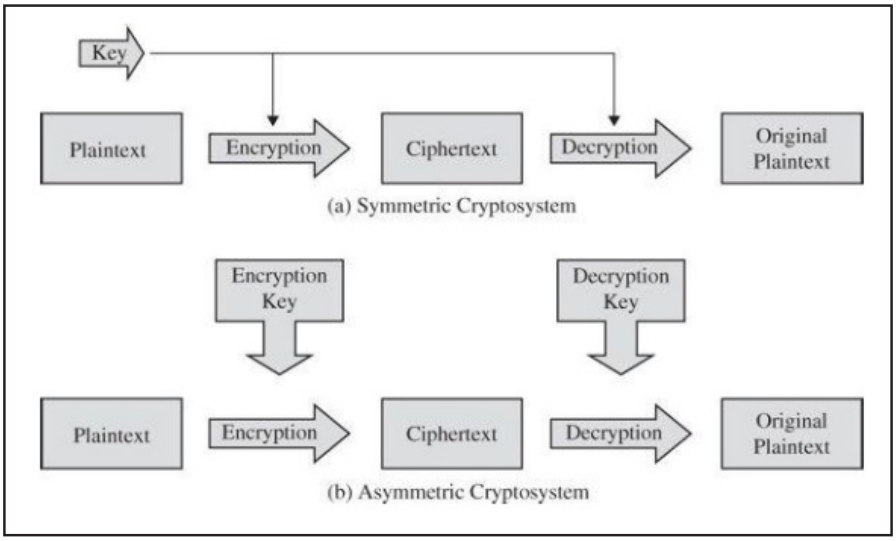
Kunci Enkripsi

Sebuah cryptosystem melibatkan seperangkat aturan tentang bagaimana mengenkripsi plaintext dan mendekripsi ciphertext. Aturan enkripsi dan dekripsi, yang disebut algoritma, sering menggunakan perangkat yang disebut kunci, dilambangkan dengan K, sehingga ciphertext yang dihasilkan tergantung pada pesan plaintext asli, algoritma, dan nilai kunci. Kami menulis ketergantungan ini sebagai C = E(K, P). Pada dasarnya, E adalah satu set algoritma enkripsi, dan kunci K memilih satu algoritma tertentu dari set.

Proses ini mirip dengan menggunakan kunci yang diproduksi secara massal di rumah. Sebagai pemilik rumah, Anda akan membayar mahal untuk membuat kontrak dengan seseorang untuk menemukan dan membuat kunci hanya untuk rumah Anda. Selain itu, Anda tidak akan tahu apakah kunci penemu tertentu benar-benar kokoh atau bagaimana dibandingkan dengan kunci penemu lain. Solusi yang lebih baik adalah memiliki beberapa perusahaan yang terkenal dan dihormati yang memproduksi kunci standar yang berbeda menurut kunci (fisik). Kemudian, Anda dan tetangga Anda mungkin memiliki merek dan gaya kunci yang sama, tetapi kunci Anda hanya akan membuka gembok Anda. Dengan cara yang sama, berguna untuk memiliki beberapa algoritma enkripsi yang diperiksa dengan baik untuk digunakan semua orang, tetapi kunci yang berbeda akan mencegah seseorang membobol data yang Anda coba lindungi.

Terkadang kunci enkripsi dan dekripsi sama, jadi P = D(K,E(K,P)), artinya kunci yang sama, K, digunakan baik untuk mengenkripsi pesan dan kemudian mendekripsinya. Bentuk ini disebut enkripsi simetris atau kunci tunggal atau kunci rahasia karena D dan E adalah proses bayangan cermin. Sebagai contoh sepele, algoritma enkripsi mungkin menggeser setiap huruf plaintext ke depan n posisi dalam alfabet. Untuk n = 1, A diubah menjadi b, B menjadi c, … P menjadi q, … dan Z menjadi a, jadi kita katakan nilai kuncinya adalah n, memindahkan n posisi maju untuk enkripsi dan mundur untuk dekripsi. (Anda mungkin memperhatikan bahwa kami telah menulis plaintext dalam huruf besar dan ciphertext yang sesuai dalam huruf kecil; kriptografer terkadang menggunakan konvensi itu untuk membantu mereka membedakan keduanya.)

Di lain waktu, kunci enkripsi dan dekripsi datang berpasangan. Kemudian, kunci dekripsi, KD, membalikkan enkripsi kunci KE, sehingga P = D(KD, E(KE,P)). Algoritma enkripsi bentuk ini disebut asimetris atau kunci publik karena mengubah C kembali ke P melibatkan serangkaian langkah dan kunci yang berbeda dari langkah dan kunci E. Perbedaan antara enkripsi simetris dan asimetris ditunjukkan pada Gambar 2-18.



Gambar 2.18 Enkripsi Simetris dan Asimetris

Kunci memberi kita fleksibilitas dalam menggunakan skema enkripsi. Kita dapat membuat enkripsi yang berbeda dari satu pesan teks biasa hanya dengan mengubah kuncinya. Selain itu, menggunakan kunci memberikan keamanan tambahan. Jika algoritma enkripsi harus jatuh ke tangan pencegat, pesan masa depan masih dapat dirahasiakan karena pencegat tidak akan mengetahui nilai kunci. Studi Kasus 2-14 menjelaskan bagaimana Inggris menangani kunci dan kode tertulis dalam Perang Dunia II. Skema enkripsi yang tidak memerlukan penggunaan kunci disebut cipher tanpa kunci.

Studi Kasus 2-14 : Silken Codes (Kode Sutra)

Leo Marks menggambarkan kehidupannya sebagai pembuat kode di Inggris selama Perang Dunia II. Artinya, Inggris menyewa Marks dan lainnya untuk merancang kode yang dapat digunakan oleh mata-mata dan tentara di lapangan. Pada hari-hari awal, skema enkripsi bergantung pada puisi yang ditulis untuk setiap mata-mata, dan bergantung pada kemampuan mata-mata untuk menghafal dan mengingat puisi dengan benar.

Tanda mengurangi risiko kesalahan dengan memperkenalkan skema pengkodean yang dicetak pada potongan sutra. Sutra yang tersembunyi di balik pakaian tidak bisa dirasakan ketika mata-mata itu ditepuk dan digeledah. Dan, tidak seperti kertas, sutra terbakar dengan cepat dan sempurna, sehingga mata-mata dapat menghancurkan bukti yang memberatkan, juga memastikan bahwa musuh tidak bisa mendapatkan bahkan potongan kode yang berharga. Ketika ditekan oleh atasan mengapa Inggris harus menggunakan sutra langka (yang juga dibutuhkan untuk kebutuhan masa perang seperti parasut) untuk kode, Marks mengatakan bahwa itu adalah pilihan. "antara sutra dan sianida."

Sejarah enkripsi sangat menarik; itu didokumentasikan dengan baik dalam buku David Kahn [KAH96]. Claude Shannon dianggap sebagai bapak kriptografi modern karena ia meletakkan dasar matematis formal untuk keamanan informasi dan menguraikan beberapa prinsip untuk kriptografi yang aman di awal komputasi digital [SHA49]. Enkripsi telah digunakan selama berabad-abad untuk melindungi komunikasi diplomatik dan militer, terkadang tidak berhasil sepenuhnya. Kata kriptografi mengacu pada praktik menggunakan enkripsi untuk menyembunyikan teks. Seorang cryptanalyst mempelajari enkripsi dan pesan terenkripsi, berharap menemukan makna tersembunyi. Seorang cryptanalyst mungkin juga bekerja secara defensif, menyelidiki kode dan cipher untuk melihat apakah mereka cukup solid untuk melindungi data secara memadai.

Baik kriptografer maupun kriptanalis berusaha menerjemahkan materi yang dikodekan kembali ke bentuk aslinya. Biasanya, kriptografer bekerja atas nama pengirim atau penerima yang sah, sedangkan kriptoanalis bekerja atas nama pencegat yang tidak sah. Akhirnya, kriptologi adalah penelitian dan studi tentang enkripsi dan dekripsi; itu mencakup baik kriptografi dan kriptanalisis.

2.5.2 Pembacaan Sandi

Tugas seorang cryptanalyst adalah memecahkan enkripsi. Artinya, cryptanalyst mencoba untuk menyimpulkan arti asli dari pesan ciphertext. Lebih baik lagi, cryptanalyst berharap untuk menentukan algoritma dekripsi mana, dan idealnya kunci mana, yang cocok dengan algoritma enkripsi untuk dapat memecahkan pesan lain yang dikodekan dengan cara yang sama.

Misalnya, anggaplah dua negara sedang berperang dan negara pertama telah mencegat pesan terenkripsi dari negara kedua. Cryptanalysts dari negara pertama ingin menguraikan pesan tertentu untuk mengantisipasi pergerakan dan sumber daya yang kedua. Tetapi yang lebih baik adalah menemukan metode dekripsi yang sebenarnya; maka negara pertama dapat menembus enkripsi semua pesan yang dikirim oleh negara kedua.

Seorang analis bekerja dengan berbagai informasi: pesan terenkripsi, algoritma enkripsi yang diketahui, teks biasa yang dicegat, item data yang diketahui atau diduga berada dalam pesan teks sandi, alat dan teknik matematika atau statistik, dan properti bahasa, serta banyak kecerdikan dan keberuntungan. Setiap bukti dapat memberikan petunjuk, dan analis menyatukan petunjuk untuk mencoba membentuk gambaran yang lebih besar tentang makna pesan dalam konteks bagaimana enkripsi dilakukan. Ingatlah bahwa tidak ada aturan. Seorang pencegat dapat menggunakan segala cara yang tersedia untuk mencari tahu arti dari pesan tersebut.

Work Factor

Sebuah algoritma enkripsi disebut *breakable* atau 'dapat dipecahkan' ketika, dengan waktu dan data yang cukup, seorang analis dapat menentukan algoritmanya. Namun, sebuah algoritme yang secara teoritis dapat dipecahkan mungkin sebenarnya tidak praktis untuk dicoba dipecahkan. Untuk mengetahui alasannya, pertimbangkan pesan 25 karakter yang diungkapkan hanya dengan huruf besar. Skema cipher yang diberikan mungkin memiliki 2625 (sekitar 1035) kemungkinan dekripsi, jadi tugasnya adalah memilih yang benar dari 2625. Jika komputer Anda dapat melakukan urutan 1010 operasi per detik, menemukan dekripsi ini akan membutuhkan urutan 1025 detik, atau kira-kira 1017 tahun. Dalam hal ini, meskipun kita tahu bahwa secara teoritis kita dapat menghasilkan solusi, menentukan algoritma penguraian dengan memeriksa semua kemungkinan dapat diabaikan karena tidak layak dengan teknologi saat ini.

Kesulitan memecahkan enkripsi disebut *work factor* (faktor kerja). Sekali lagi, analogi dengan kunci fisik mungkin terbukti bermanfaat. Seperti yang Anda ketahui, kunci fisik memiliki takik atau ketidakteraturan lainnya, dan takik menyebabkan pin bergerak di dalam kunci, memungkinkan kunci terbuka. Beberapa kunci sederhana, seperti yang dijual dengan koper, hanya memiliki satu takik, sehingga kunci ini sering kali dapat dibuka hanya dengan seutas kawat bengkok; lebih buruk lagi, beberapa produsen hanya memproduksi sedikit (dan terkadang hanya satu!) desain pin internal yang berbeda; Anda mungkin dapat membuka kunci semacam itu hanya dengan beberapa tombol. Jelas kunci ini hanya kosmetik.

Kunci rumah biasa memiliki lima atau enam takik, dan setiap takik dapat memiliki sepuluh kedalaman. Untuk membuka kunci seperti itu, diperlukan kombinasi yang tepat dari kedalaman takik, yang mungkin ada hingga satu juta kemungkinan, jadi membawa cincin sebanyak itu tidak mungkin dilakukan. Meskipun secara teori seseorang dapat membuka salah satu dari kunci ini dengan mencoba semua kunci yang mungkin, dalam praktiknya jumlah kemungkinan tersebut menjadi penghalang. Kami mengatakan bahwa faktor kerja untuk membuka salah satu kunci ini tanpa kunci yang sesuai cukup besar untuk mencegah sebagian besar serangan. Begitu juga dengan kriptografi: Sebuah enkripsi cukup jika pekerjaan mendekripsi tanpa mengetahui kunci enkripsi lebih besar dari nilai data yang dienkripsi.

Dua masalah penting lainnya harus ditangani ketika mempertimbangkan pemecahan algoritma enkripsi. Pertama, cryptanalyst tidak dapat diharapkan untuk mencoba hanya dengan cara yang keras dan panjang. Dalam contoh yang baru saja disajikan, dekripsi yang jelas mungkin memerlukan 2625 operasi mesin, tetapi pendekatan yang lebih cerdik mungkin hanya membutuhkan 1015 operasi. Pada kecepatan 1010 operasi per detik, 1015 operasi memakan waktu sedikit lebih dari satu hari. Pendekatan yang cerdik tentu layak dilakukan. Bahkan, surat kabar terkadang mencetak teka-teki kriptogram yang diselesaikan manusia dengan pena dan kertas saja, jadi jelas ada jalan pintas ke mesin komputer kita yang memperkirakan waktu bertahun-tahun atau bahkan satu hari usaha. Permainan surat kabar memberikan petunjuk tentang panjang kata dan karakter berulang, sehingga manusia memecahkan masalah yang lebih mudah. Seperti yang kami katakan, bagaimanapun, cryptanalyst juga menggunakan setiap informasi yang mereka miliki.

Beberapa algoritme yang kami pelajari dalam buku ini didasarkan pada masalah "sulit" yang diketahui yang membutuhkan waktu lama untuk diselesaikan. Tetapi cryptanalyst tidak harus memecahkan masalah mendasar untuk memecahkan enkripsi satu pesan. Penggunaan kontrol yang ceroboh

dapat mengungkapkan kemungkinan kata atau frasa, dan seorang analis dapat menggunakan tebakan terdidik yang dikombinasikan dengan analisis yang cermat untuk menghasilkan semua atau sebagian besar pesan penting. Atau cryptanalyst mungkin mempekerjakan mata-mata untuk mendapatkan plaintext sepenuhnya di luar sistem; analis kemudian dapat menggunakan sepasang plaintext dan ciphertext yang sesuai untuk menyimpulkan algoritme atau kunci yang digunakan untuk diterapkan ke pesan berikutnya.

Kedua, perkiraan breakability didasarkan pada teknologi saat ini. Kemajuan teknologi besar komputasi telah terjadi sejak tahun 1950. Hal-hal yang tidak mungkin dilakukan di 1940 menjadi mungkin pada 1950-an, dan setiap dekade berikutnya telah membawa perbaikan yang lebih besar. Sebuah dugaan yang dikenal sebagai "Hukum Moore" menegaskan bahwa kecepatan prosesor berlipat ganda setiap 1,5 tahun, dan dugaan ini telah berlaku selama lebih dari tiga dekade. Kami tidak berani mengatakan suatu algoritma aman hanya karena tidak dapat dipecahkan dengan teknologi saat ini, atau lebih buruk lagi, bahwa itu belum rusak.

Dalam buku ini kami menulis bahwa ada sesuatu yang tidak mungkin; misalnya, tidak mungkin mendapatkan plaintext dari ciphertext tanpa kunci dan algoritma yang sesuai. Harap dipahami bahwa dalam kriptografi ada beberapa hal yang benar-benar tidak mungkin: tidak mungkin atau sangat sulit, mungkin, tetapi tidak mungkin, tidak.

2.5.3 Sistem Enkripsi Simetris dan Asimetris

Ingatlah bahwa dua jenis enkripsi dasar adalah simetris (juga disebut "kunci rahasia") dan asimetris (juga disebut "kunci publik"). Algoritma simetris menggunakan satu kunci, yang berfungsi untuk enkripsi dan dekripsi. Biasanya, algoritma dekripsi terkait erat dengan enkripsi, pada dasarnya menjalankan enkripsi secara terbalik.

Sistem simetris menyediakan saluran dua arah kepada penggunanya: A dan B berbagi kunci rahasia, dan keduanya dapat mengenkripsi informasi untuk dikirim ke yang lain serta mendekripsi informasi dari yang lain. Selama kunci tetap rahasia, sistem juga memberikan keaslian, bukti bahwa pesan yang diterima tidak dibuat-buat oleh orang lain selain pengirim yang dinyatakan. Ini adalah buku keamanan, kami menunjukkan bahwa buktinya sebenarnya adalah bahwa pesan tersebut dikirim oleh seseorang yang memiliki atau dapat mensimulasikan efek dari kunci pengirim. Dengan banyak ancaman keamanan, ada risiko kecil, tetapi bukan nol, bahwa pesan tersebut sebenarnya bukan dari pengirim tetapi merupakan pemalsuan yang kompleks. Keaslian terjamin karena hanya pengirim yang sah yang dapat menghasilkan pesan yang akan didekripsi dengan baik dengan yang dibagikan kunci.

Simetri adalah keuntungan utama dari jenis enkripsi ini, tetapi juga menimbulkan masalah: Bagaimana dua pengguna A dan B mendapatkan kunci rahasia bersama mereka? Dan hanya A dan B yang dapat menggunakan kunci itu untuk komunikasi terenkripsi mereka. Jika A ingin berbagi komunikasi terenkripsi dengan pengguna lain C, A dan C memerlukan kunci rahasia bersama yang berbeda. Mengelola kunci adalah kesulitan utama dalam menggunakan enkripsi simetris. Secara umum, n pengguna yang ingin berkomunikasi secara berpasangan membutuhkan n * (n – 1)/2 kunci. Dengan kata lain, jumlah kunci yang dibutuhkan meningkat pada tingkat yang sebanding dengan kuadrat

dari jumlah pengguna! Jadi properti dari sistem enkripsi simetris adalah bahwa mereka memerlukan sarana distribusi kunci.

Sistem kunci asimetris atau publik, di sisi lain, biasanya memiliki pasangan kunci yang sama persis. Kunci diproduksi bersama atau satu diturunkan secara matematis dari yang lain. Dengan demikian, suatu proses menghitung kedua kunci sebagai satu set.

Tetapi untuk kedua jenis enkripsi, kunci harus disimpan dengan baik. Setelah kunci simetris atau privat diketahui oleh orang luar, semua pesan yang ditulis sebelumnya atau di masa depan dapat didekripsi (dan karenanya dibaca atau dimodifikasi) oleh orang luar. Jadi, untuk semua algoritma enkripsi, manajemen kunci adalah masalah utama. Ini melibatkan penyimpanan, pengamanan, dan pengaktifan kunci.

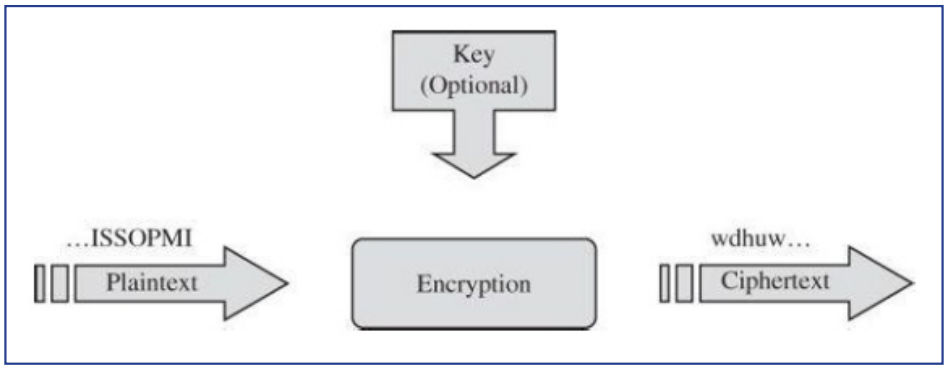
Sistem asimetris unggul dalam manajemen kunci. Dengan pendekatan kunci publik, Anda dapat mengirim kunci publik dalam pesan email atau mempostingnya di direktori publik. Hanya kunci pribadi yang sesuai, yang mungkin tidak diungkapkan, yang dapat mendekripsi apa yang telah dienkripsi dengan kunci publik.

2.5.4 Streaming dan Blokir Cipher

A block cipher adalah algoritma serbaguna yang mengimplementasikan permutasi kunci yang bergantung pada nilai-nilai yang merupakan urutan dari sejumlah bit yang tetap (disebut "blok"). Ini dapat digunakan untuk berbagai peran dalam berbagai jenis protokol kriptografi. Salah satu peran tersebut adalah enkripsi massal aliran data yang panjang; untuk mencapai hal seperti itu, cipher blok harus digunakan dengan mode operasi yang sesuai (alias "mode chaining"), yang tradisional adalah CBC, dan mode baru yang trendi menjadi RKT.

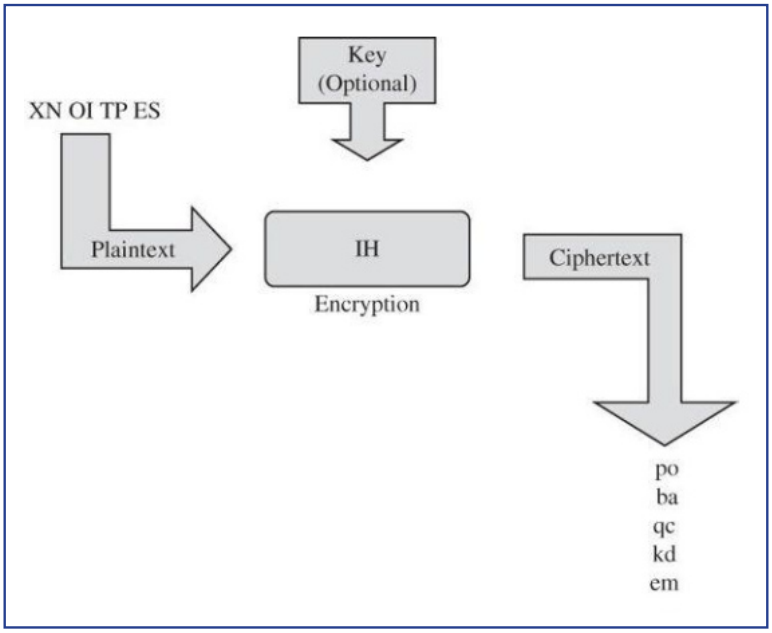
A stream cipher adalah algoritma khusus untuk tujuan enkripsi massal dari aliran data yang panjang. Idenya adalah bahwa, dengan kehilangan fleksibilitas cipher blok, akan mungkin untuk membuat algoritma yang lebih efisien (mis. Sesuatu yang mengenkripsi data lebih cepat).

Salah satu karakterisasi terakhir dari algoritma enkripsi berkaitan dengan sifat data yang akan disembunyikan. Misalkan Anda sedang streaming video, mungkin film, dari satelit. Aliran mungkin datang dalam semburan, tergantung pada hal-hal seperti beban pada satelit dan kecepatan di mana pengirim dan penerima dapat beroperasi. Untuk aplikasi tersebut Anda dapat menggunakan apa yang disebut enkripsi aliran, di mana setiap bit, atau mungkin setiap byte, dari aliran data dienkripsi secara terpisah. Sebuah model stream enciphering ditunjukkan pada Gambar 2-19. Perhatikan bahwa simbol input diubah satu per satu. Keuntungan dari stream cipher adalah dapat langsung diterapkan ke item data apa pun yang siap dikirim. Tetapi kebanyakan algoritma enkripsi melibatkan transformasi yang kompleks; untuk melakukan transformasi ini pada satu atau beberapa bit pada suatu waktu mahal.



Gambar 2-19 Penyandian Aliran

Untuk mengatasi masalah ini dan mempersulit cryptanalyst untuk memecahkan kode, kita dapat menggunakan cipher blok. Sebuah cipher blok mengenkripsi sekelompok simbol plaintext sebagai satu blok. Sebuah algoritma block cipher melakukan pekerjaannya pada sejumlah data plaintext sekaligus. Seperti mesin yang memotong 24 cookie sekaligus, algoritme ini memanfaatkan skala ekonomi dengan mengoperasikan sejumlah besar data sekaligus. Blok untuk algoritma tersebut biasanya 64, 128, 256 bit atau lebih. Ukuran blok tidak perlu memiliki hubungan khusus dengan ukuran karakter. Block cipher bekerja pada blok plaintext dan menghasilkan blok ciphertext, seperti yang ditunjukkan pada Gambar 2-20. Pada gambar, kotak tengah mewakili mesin enkripsi: Pasangan plaintext sebelumnya diubah menjadi po, yang saat ini dikonversi adalah IH, dan mesin akan segera mengonversi ES.



Gambar 2-20 Blok Cipher

Cipher aliran mengenkripsi satu bit atau satu byte pada satu waktu; cipher blok mengenkripsi sejumlah bit tetap sebagai potongan tunggal.

Meskipun keduanya adalah sandi simetris, stream cipher didasarkan pada menghasilkan keystream cryptograpic "tak terbatas", dan menggunakannya untuk mengenkripsi satu bit atau byte pada suatu waktu (mirip dengan pad satu kali), sedangkan cipher blok bekerja pada potongan data yang lebih

besar. (yaitu blok) pada suatu waktu, sering menggabungkan blok untuk keamanan tambahan (misalnya AES dalam mode CBC).

- Stream cipher biasanya lebih cepat dari pada block, tetapi itu memiliki harganya sendiri.
- Cipher blok biasanya memerlukan lebih banyak memori, karena mereka bekerja pada potongan data yang lebih besar dan sering memiliki "terbawa" dari blok sebelumnya, sedangkan sejak stream cipher bekerja pada hanya beberapa bit pada satu waktu mereka memiliki persyaratan memori yang relatif rendah (dan karena itu lebih murah untuk terapkan dalam skenario terbatas seperti perangkat tertanam, firmware, dan perangkat keras khusus).
- Stream cipher lebih sulit diimplementasikan dengan benar, dan rentan terhadap kelemahan berdasarkan penggunaan - karena prinsipnya mirip dengan pad sekali pakai, keystream memiliki persyaratan yang sangat ketat. Di sisi lain, itu biasanya bagian yang sulit, dan dapat diturunkan ke mis. kotak eksternal.
- Karena cipher blok mengenkripsi seluruh blok pada satu waktu (dan lebih jauh lagi memiliki mode "umpan balik" yang paling direkomendasikan), mereka lebih rentan terhadap kebisingan dalam transmisi, yaitu jika Anda mengacaukan satu bagian dari data, sisanya mungkin tidak dapat dipulihkan. Sedangkan dengan stream cipher byte byte dienkripsi secara individu tanpa koneksi ke potongan data lainnya (dalam kebanyakan cipher/mode), dan sering memiliki dukungan untuk gangguan pada saluran.

Juga, stream cipher tidak memberikan perlindungan integritas atau otentikasi, sedangkan beberapa cipher blok (tergantung pada mode) dapat memberikan perlindungan integritas, selain kerahasiaan.
Karena semua hal di atas, stream cipher biasanya terbaik untuk Studi Kasus-Studi Kasus di mana jumlah data tidak diketahui, atau kontinu - seperti aliran jaringan. Blok cipher, di sisi lain, atau lebih berguna ketika jumlah data sudah diketahui sebelumnya - seperti file, bidang data, atau protokol permintaan/respons, seperti HTTP di mana panjang total pesan diketahui sudah ada di awal.

Dengan deskripsi karakteristik algoritma enkripsi yang berbeda ini, kita sekarang dapat beralih ke beberapa algoritma enkripsi yang banyak digunakan. Kami menyajikan bagaimana masing-masing bekerja, sedikit konteks historis dan motivasi untuk masing-masing, dan beberapa kekuatan dan kelemahan. Kami mengidentifikasi algoritme ini berdasarkan nama karena nama-nama ini muncul dalam literatur populer. Kami juga memperkenalkan algoritme simetris lainnya di Bab 12. Tentu saja Anda harus mengetahui bahwa ini hanyalah contoh algoritme populer; dari waktu ke waktu algoritma ini dapat digantikan oleh orang lain. Untuk sebagian besar kriptografi telah menjadi plug-and-play, yang berarti bahwa dalam sebuah aplikasi pengembang dapat mengganti satu algoritma untuk yang lain dari jenis yang sama dan karakteristik yang sama. Dengan demikian kemajuan di bidang kriptografi tidak mengharuskan semua aplikasi yang menggunakan kriptografi harus ditulis ulang.

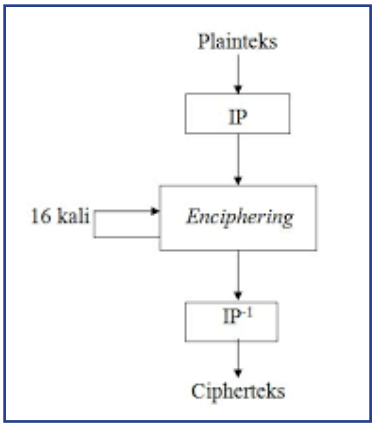
2.5.5 DES: Data Encryption Standard

DES merupakan salah satu algoritma kriptografi cipher block dengan ukuran blok 64 bit dan ukuran kuncinya 56 bit. Algoritma DES dibuat di IBM, dan merupakan modifikasi daripada algoritma terdahulu yang bernama Lucifer. Lucifer merupakan algoritma cipher block yang beroperasi pada blok masukan 64 bit dan kuncinya berukuran 128 bit.

Pengurangan jumlah bit kunci pada DES dilakukan dengan alasan agar mekanisme algoritma ini bisa diimplementasikan dalam satu chip. DES pertama kali dipublikasikan di Federal Register pada 17 Maret 1975. Setelah melalui banyak diskusi, akhirnya algortima DES diadopsi sebagai algoritma standar yang digunakan oleh NBS (National Bureau of Standards) pada 15 Januari 1977. Sejak saat itu, DES banyak digunakan pada dunia penyebaran informasi untuk melindungi data agar tidak bisa dibaca oleh orang lain.

Skema Algoritma DES

Algoritma DES dikembangkan di IBM dibawah kepemimpinan W.L. Tuchman pada tahun 1972. Algoritma ini didasarkan pada algoritma Lucifer yang dibuat oleh Horst Feistel. Algoritma ini telah disetujui oleh National Bureau of Standard (NBS) setelah penilaian kekuatannya oleh National Security Agency (NSA) Amerika Serikat. DES termasuk ke dalam kriptografi kunci-simetri dan tergolong jenis cipher blok. DES beroperasi pada ukuran blok 64 bit. Lihat Gambar 2.21.



Gambar 2.21 Skema Global Algoritma DES

Panjang kunci ekternal = 64 bit (sesuai ukuran blok), tetapi hanya 56 bit yang dipakai (8 bit paritas tidak digunakan). Setiap blok (plainteks atau cipherteks) dienkripsi dalam 16 putaran. Setiap putaran menggunakan kunci internal berbeda. Kunci internal (56-bit) dibangkitkan dari kunci eksternal. Setiap blok mengalami permutasi awal (IP), 16 putaran enciphering, dan inversi permutasi awal (IP-1).

Algoritma dimulai dengan mengenkripsi plaintext sebagai blok 64 bit. Kuncinya panjangnya 64 bit, tetapi sebenarnya bisa berupa angka 56-bit. (Ekstra 8 bit sering digunakan sebagai digit cek tetapi tidak mempengaruhi enkripsi dalam implementasi normal. Jadi kami mengatakan bahwa DES menggunakan kunci, kekuatannya adalah 56 bit.) Pengguna dapat memilih kunci baru kapan saja di sana adalah ketidakpastian tentang keamanan kunci lama.

DES mengenkripsi blok 64-bit dengan menggunakan kunci 56-bit

DES hanya menggunakan operasi aritmatika dan logika standar pada data biner hingga panjang 64 bit, sehingga cocok untuk diimplementasikan dalam perangkat lunak pada sebagian besar komputer saat ini. Enkripsi dengan DES melibatkan 16 iterasi, masing-masing menggunakan penggantian blok bit (disebut langkah substitusi), mengacak bit (disebut langkah permutasi), dan mencampur bit dari kunci (disebut transformasi kunci). Meskipun rumit, prosesnya berbasis tabel dan berulang, sehingga cocok untuk diimplementasikan pada chip tujuan tunggal. Faktanya, beberapa chip seperti itu tersedia di pasaran untuk digunakan sebagai komponen dasar pada perangkat yang menggunakan enkripsi DES dalam suatu aplikasi.

Double dan Triple DES

Seperti yang Anda ketahui, daya komputasi telah meningkat pesat selama beberapa dekade terakhir, dan ia berjanji untuk terus melakukannya. Untuk alasan ini, panjang kunci DES 56-bit tidak cukup panjang untuk kenyamanan sebagian orang. Sejak tahun 1970-an, para peneliti dan praktisi telah tertarik dengan versi DES yang lebih panjang. Tapi kami punya masalah: Desain algoritma DES diperbaiki ke kunci 56-bit.

Double DES

Untuk mengatasi ketidaknyamanan, beberapa peneliti menyarankan menggunakan enkripsi ganda untuk kerahasiaan yang lebih besar. Enkripsi ganda bekerja dengan cara berikut. Ambil dua kunci, k1 dan k2, dan lakukan dua enkripsi, satu di atas yang lain: E(k2, E(k1,m)). Secara teori, pendekatan ini harus melipatgandakan kesulitan memecahkan enkripsi, seperti halnya dua kunci lebih sulit untuk dipilih daripada satu.

Sayangnya, anggapan itu salah. Ralph Merkle dan Martin Hellman menunjukkan bahwa dua enkripsi hampir tidak lebih baik dari satu: dua enkripsi dengan kunci 56-bit yang berbeda setara dalam faktor kerja dengan satu enkripsi dengan kunci 57-bit. Dengan demikian, enkripsi ganda hanya menambahkan sedikit pekerjaan ekstra untuk penyerang yang mencoba menyimpulkan kunci di mana sepotong ciphertext dienkripsi. Seperti yang akan segera kami jelaskan, beberapa kunci DES 56-bit telah diturunkan hanya dalam beberapa hari; dua kali hari masih berhari-hari, ketika harapannya adalah menambahkan bulan jika tidak bertahun-tahun bekerja untuk enkripsi kedua. Sayangnya, DES ganda pada dasarnya tidak menambahkan keamanan lagi.

Triple DES

Namun, trik sederhana memang meningkatkan keamanan DES. Menggunakan tiga kunci menambah kekuatan yang signifikan.

Yang disebut prosedur triple DES adalah C = E(k3, E(k2, E(k1,m))). Artinya, Anda mengenkripsi dengan satu kunci, lalu dengan yang kedua, dan terakhir dengan yang ketiga. Proses ini memberikan kekuatan yang kira-kira setara dengan kunci 112-bit (karena serangan DES ganda mengalahkan kekuatan salah satu dari tiga kunci, tetapi tidak berpengaruh pada kunci ketiga).

Variasi kecil dari triple DES, yang oleh sebagian orang juga membingungkan disebut triple DES, adalah C = E(k1, D(k2, E(k1,m))). Artinya, Anda mengenkripsi dengan satu kunci, mendekripsi dengan yang kedua, dan mengenkripsi dengan yang pertama lagi. Versi ini hanya membutuhkan dua kunci. (Langkah dekripsi kedua juga membuat proses ini bekerja untuk enkripsi tunggal dengan satu kunci: Dekripsi membatalkan enkripsi pertama, sehingga hasil akhirnya adalah satu enkripsi. Bentuk enkripsi-dekripsi-enkripsi berguna karena satu algoritma dapat menghasilkan hasil untuk kedua konvensional DES satu kunci dan metode dua kunci yang lebih aman.) Versi dua kunci, tiga langkah ini tunduk pada serangan rumit lainnya, sehingga kekuatannya dinilai hanya sekitar 80 bit. Namun, 80 bit berada di luar kemampuan cracking yang wajar untuk perangkat keras saat ini.

Singkatnya, DES biasa memiliki ruang kunci 56 bit, DES ganda hampir tidak lebih baik, tetapi DES tiga kunci dua memberikan panjang efektif 80 bit, dan DES tiga kunci memberikan kekuatan 112 bit. Ingat mengapa kita begitu terpaku pada ukuran kunci: Jika tidak ada cara lain yang berhasil, penyerang selalu dapat mencoba semua kunci yang mungkin. Kunci yang lebih panjang berarti secara signifikan lebih banyak pekerjaan agar serangan ini membuahkan hasil, dengan faktor kerja berlipat ganda untuk setiap bit tambahan dalam panjang kunci. Sekarang, kira-kira setengah abad setelah DES dibuat, kunci 56-bit tidak memadai untuk kerahasiaan yang serius, tetapi ukuran kunci efektif 80- dan 112-bit memberikan keamanan yang wajar.

Sejak pertama kali diumumkan, DES menjadi kontroversi. Banyak peneliti mempertanyakan keamanan yang diberikannya. Karena hubungannya dengan pemerintah AS, khususnya Badan Keamanan Nasional AS (NSA) yang membuat perubahan tertentu yang tidak dapat dijelaskan antara apa yang diusulkan IBM dan apa yang sebenarnya diterbitkan oleh NBS, beberapa orang menduga bahwa algoritme entah bagaimana melemah, untuk memungkinkan pemerintah mengintip data terenkripsi. Sebagian besar kontroversi ini telah muncul dalam literatur terbuka, tetapi fitur DES tertentu belum diungkapkan oleh para desainer atau disimpulkan oleh analis luar.

Whitfield Diffie dan Martin Hellman berpendapat pada tahun 1977 bahwa kunci 56-bit terlalu pendek. Pada tahun 1977, dilarang untuk menguji semua 256 (sekitar 1015) kunci pada komputer saat itu. Tetapi mereka berpendapat bahwa seiring waktu, komputer akan menjadi lebih kuat dan algoritma DES akan tetap tidak berubah; akhirnya, kecepatan komputer akan melebihi kekuatan DES. Persis itu terjadi sekitar 20 tahun kemudian. Pada tahun 1997, para peneliti yang menggunakan jaringan lebih dari 3.500 mesin secara paralel dapat menyimpulkan kunci DES dalam pekerjaan empat bulan. Dan pada tahun 1998 dengan biaya sekitar $200.000, peneliti AS membangun mesin “DES cracker” khusus yang dapat menemukan kunci DES dalam waktu sekitar empat hari, hasilnya kemudian ditingkatkan menjadi beberapa jam.

Apakah ini berarti DES tidak aman? Tidak, tidak persis. Belum ada yang menunjukkan kelemahan serius dalam algoritma DES itu sendiri. Serangan tahun 1997 membutuhkan banyak kerja sama, dan mesin tahun 1998 agak mahal. Tetapi bahkan jika DES konvensional dapat diserang, triple DES masih jauh di luar kekuatan serangan ini. Ingat dampak pertumbuhan eksponensial: Mari kita katakan, untuk kesederhanaan, DES kunci tunggal dapat dipecahkan dalam satu jam. Versi kunci ganda yang sederhana kemudian dapat dipecahkan dalam dua jam. Tapi 2, yang lebih dari 16.700.000, berarti akan memakan waktu 16 juta jam, hampir 2.000 tahun, untuk mengalahkan enkripsi DES tiga kunci dua, dan jauh lebih lama untuk versi tiga kunci.

Namun demikian, struktur dasar DES dengan kunci 56-bit dengan panjang tetap dan jumlah iterasi yang tetap menunjukkan perlunya algoritma baru yang lebih kuat, dan lebih fleksibel. Pada tahun 1995, NIST mulai mencari algoritma enkripsi baru yang kuat. Tanggapan terhadap pencarian itu telah menjadi Standar Enkripsi Lanjutan, atau AES.

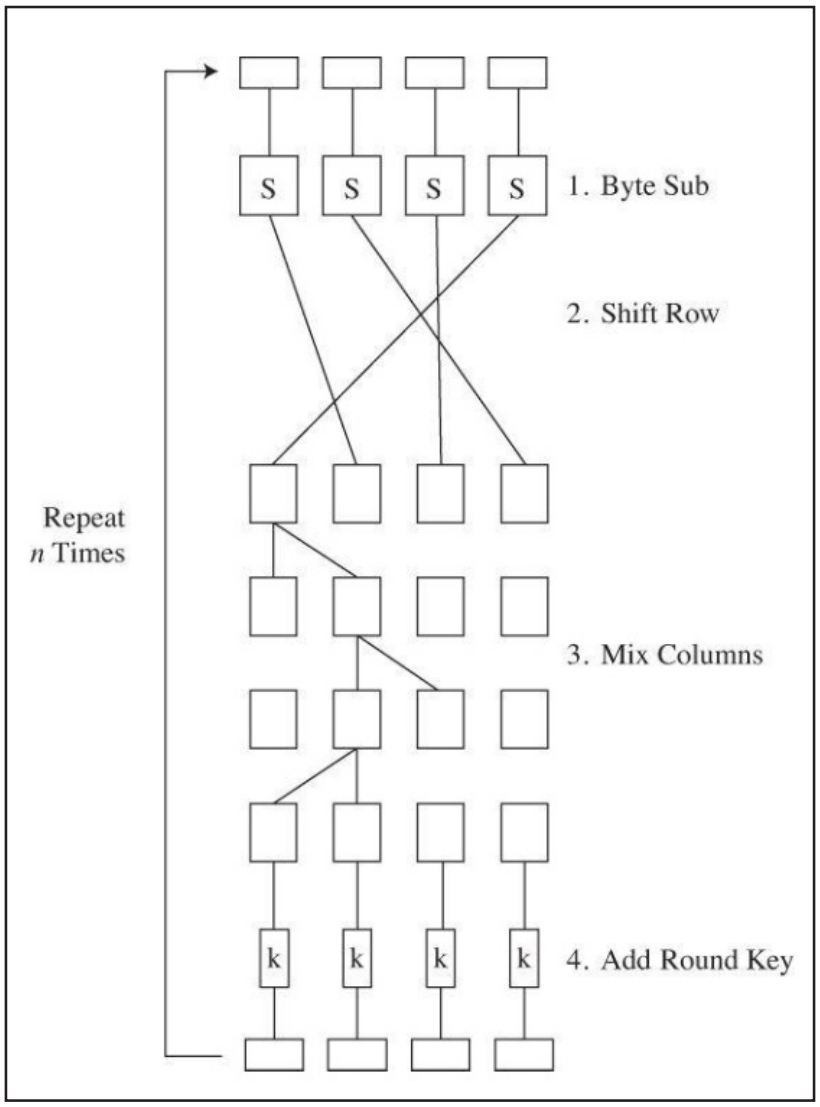
2.5.6 AES: Advanced Encryption System

Setelah kompetisi dan peninjauan publik, NIST memilih algoritme bernama Rijndael sebagai sistem enkripsi canggih baru; Rijndael sekarang lebih dikenal sebagai AES. AES diadopsi untuk digunakan oleh pemerintah AS pada bulan Desember 2001 dan menjadi Federal Information Processing Standard 197. AES kemungkinan akan menjadi algoritme simetris kelas komersial pilihan selama bertahun-tahun, jika bukan beberapa dekade. Mari kita lihat lebih dekat.

Sekilas tentang Rijndael

Rijndael adalah algoritma cepat yang dapat dengan mudah diimplementasikan pada prosesor sederhana. Meskipun memiliki dasar matematis yang kuat, ia terutama menggunakan operasi substitusi, transposisi, pergeseran, OR eksklusif, dan penjumlahan. Seperti DES, AES menggunakan siklus berulang.

Ada 10, 12, atau 14 siklus untuk kunci masing-masing 128, 192, dan 256 bit. Di Rijndael, siklusnya disebut “putaran”. Setiap putaran terdiri dari empat langkah yang menggantikan dan mengacak bit. Bit dari kunci sering digabungkan dengan bit hasil antara, jadi bit kunci juga tersebar dengan baik di seluruh hasil. Selanjutnya, keempat langkah ini sangat cepat. Algoritma AES digambarkan pada Gambar 2-22.



Gambar 2-22 Algoritma Enkripsi AES

Kekuatan Algoritma

Karakteristik dan kekuatan nyata dari DES dan AES dibandingkan pada Tabel 2-12. Ingat, tentu saja, bahwa angka-angka kekuatan ini hanya berlaku jika penerapan dan penggunaannya kuat; algoritma yang kuat kehilangan kekuatan jika digunakan dengan kelemahan yang memungkinkan orang luar menentukan properti kunci dari data terenkripsi.

Tabel 2-12 Perbandingan DES dan AES

Dasar untuk Perbandingan	DES (Standar Enkripsi Data)	AES (Standar Enkripsi Lanjutan)
Dasar	Dalam DES, blok data dibagi menjadi dua.	Di AES seluruh blok data diproses sebagai matriks tunggal.
Prinsip	DES bekerja pada struktur Feistel Cipher.	AES bekerja pada Prinsip Substitusi dan Permutasi.
Plaintext	Plaintext adalah dari 64 bit	Plaintext bisa dari 128.192, atau 256 bit
Ukuran kunci	DES dibandingkan dengan AES memiliki ukuran kunci yang lebih kecil.	AES memiliki ukuran kunci yang lebih besar dibandingkan dengan DES.
Babak	16 putaran	10 putaran untuk 128-bit algo 12 putaran untuk 192-bit algo 14 putaran untuk 256-bit algo
Rounds Names	Permutasi Ekspansi, Xor, S-box, P-box, Xor dan Swap.	Subbytes, Shiftrows, Campur kolom, Addroundkeys.
Keamanan	DES memiliki kunci yang lebih kecil yang kurang aman.	AES memiliki kunci rahasia yang besar sehingga, lebih aman.
Kecepatan	DES relatif lebih lambat.	AES lebih cepat.

Selain itu, jumlah siklus dapat diperpanjang secara alami. Dengan DES algoritme didefinisikan untuk 16 siklus secara tepat; untuk memperluas jumlah itu akan membutuhkan redefinisi algoritma yang substansial. Struktur internal AES tidak memiliki batasan apriori pada jumlah siklus. Jika cryptanalyst pernah menyimpulkan bahwa 10 atau 12 atau 14 putaran terlalu rendah, satu-satunya perubahan yang diperlukan untuk meningkatkan algoritma adalah dengan mengubah batas pada pengulangan.

Sebuah tanda kepercayaan adalah bahwa pemerintah AS telah menyetujui AES untuk melindungi dokumen rahasia Rahasia dan Top Secret. Ini adalah pertama kalinya Amerika Serikat menyetujui penggunaan algoritma komersial yang berasal dari luar pemerintah (dan lebih jauh lagi, di luar Amerika Serikat) untuk mengenkripsi data rahasia.

Namun, kami tidak bisa berpuas diri. Tidak ada yang bisa memprediksi sekarang batasan apa yang mungkin diidentifikasi oleh cryptanalyst di masa depan. Untungnya, ahli kriptologi berbakat terus menyelidiki algoritma yang lebih kuat yang akan dapat menggantikan AES ketika menjadi usang. Saat ini, AES tampaknya merupakan peningkatan yang signifikan dari DES, dan dapat ditingkatkan secara alami jika perlu. DES masih digunakan secara luas, tetapi AES juga diadopsi secara luas, terutama untuk aplikasi baru.

2.5.7 Kriptografi Kunci Publik

Sejauh ini, kita telah melihat algoritma enkripsi dari sudut pandang membuat pengacakan mudah bagi pengirim (sehingga enkripsi cepat dan sederhana) dan dekripsi mudah bagi penerima tetapi tidak untuk penyusup. Tetapi pandangan fungsional untuk mengubah plaintext menjadi ciphertext ini hanyalah sebagian dari gambarannya. Kita juga harus mencari cara untuk mendistribusikan kunci enkripsi. Kami telah mencatat betapa bergunanya kunci dalam menghalangi penyusup, tetapi kuncinya harus tetap rahasia agar efektif. Algoritma enkripsi yang telah kami sajikan sejauh ini disebut algoritma simetris atau kunci rahasia. Dua algoritma simetris yang paling banyak digunakan, DES dan AES, beroperasi dengan cara yang sama: Dua pengguna memiliki salinan dari: kunci yang sama. Satu pengguna menggunakan algoritma untuk mengenkripsi beberapa plaintext di bawah kunci, dan pengguna lain menggunakan kebalikan dari algoritma dengan kunci yang sama untuk mendekripsi ciphertext. Inti dari masalah ini adalah bahwa semua kekuatan enkripsi bergantung pada kerahasiaan kunci.

Pada tahun 1976, Whitfield Diffie dan Martin Hellman menemukan kriptografi kunci publik, jenis enkripsi baru. Dengan sistem enkripsi kunci publik, setiap pengguna memiliki dua kunci, salah satunya tidak harus dirahasiakan. Meskipun berlawanan dengan intuisi, sebenarnya sifat publik dari kunci tersebut tidak mengkompromikan kerahasiaan sistem. Sebaliknya, dasar untuk enkripsi kunci publik adalah untuk membiarkan kunci tersebut dibocorkan tetapi untuk menjaga rahasia teknik dekripsi. Kriptosistem kunci publik mencapai tujuan ini dengan menggunakan dua kunci: satu untuk mengenkripsi dan yang lainnya untuk mendekripsi. Meskipun kunci-kunci ini diproduksi dalam pasangan yang berhubungan secara matematis, orang luar secara efektif tidak dapat menggunakan satu kunci untuk menurunkan yang lain.

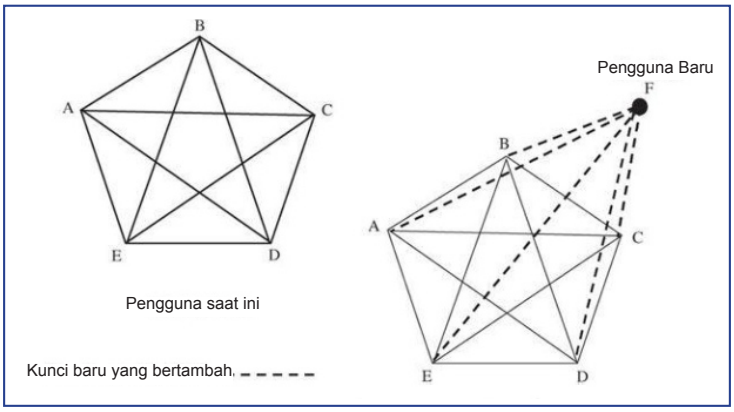
Di bagian ini, kita melihat cara untuk mengizinkan kunci menjadi publik tetapi tetap melindungi pesannya. Kami juga fokus pada algoritme RSA, sistem kunci publik kelas komersial yang populer. Algoritma lain, seperti kriptosistem kurva eliptik, dan algoritma El Gamal beroperasi dengan cara yang sama (walaupun matematika dasarnya sangat berbeda). Kami berkonsentrasi pada RSA karena banyak aplikasi yang menggunakannya. Kami juga menyajikan skema matematika di mana dua pengguna dapat bersama-sama membuat kunci enkripsi rahasia tanpa memiliki rahasia sebelumnya.

Alasan

Mengapa membuat publik kunci diinginkan? Dengan sistem kunci simetris konvensional, setiap pasangan pengguna membutuhkan kunci yang terpisah. Tetapi dengan sistem kunci publik, siapa pun yang menggunakan satu kunci publik dapat mengirim pesan rahasia ke pengguna, dan pesan tersebut tetap terlindungi secara memadai agar tidak dibaca oleh pencegat. Mari kita selidiki mengapa demikian.

Ingat bahwa secara umum, sistem n-pengguna memerlukan n * (n – 1)/2 kunci, dan setiap pengguna harus melacak dan mengingat kunci untuk setiap pengguna lain yang ingin berkomunikasi dengannya. Seiring bertambahnya jumlah pengguna, jumlah kunci meningkat dengan cepat, seperti yang ditunjukkan pada Gambar 2-23. Menentukan dan mendistribusikan kunci ini adalah masalah. Masalah yang lebih serius adalah menjaga keamanan untuk kunci yang sudah didistribusikan—kami tidak

dapat mengharapkan pengguna untuk mengingat begitu banyak kunci. Lebih buruk lagi, kehilangan atau eksposur kunci satu pengguna memerlukan pengaturan pasangan kunci baru dengan masing-masing koresponden pengguna tersebut.



Gambar 2-23 Ledakan dalam Jumlah pengguna Kunci Baru

Karakteristik

Kita dapat mengurangi masalah proliferasi kunci dengan menggunakan pendekatan kunci publik. Dalam kunci publik atau sistem enkripsi asimetris, setiap pengguna memiliki dua kunci: kunci publik dan kunci pribadi. Pengguna dapat dengan bebas mempublikasikan kunci publik karena setiap kunci hanya melakukan enkripsi atau dekripsi, tetapi tidak keduanya. Kunci beroperasi sebagai kebalikan, artinya satu kunci membatalkan enkripsi yang disediakan oleh kunci lainnya. Tetapi menyimpulkan satu kunci dari yang lain secara efektif tidak mungkin.

Untuk melihat caranya, biarkan k_{PRIV} menjadi kunci pribadi pengguna, dan biarkan kPUB menjadi kunci publik yang sesuai. Kemudian, plaintext terenkripsi menggunakan kunci publik didekripsi dengan penerapan kunci pribadi; kita menulis hubungan sebagai :

$$P = D(k_{PRIV}, E(k_{PUB}, P))$$

Artinya, pengguna dapat memecahkan kode dengan kunci pribadi apa yang telah dienkripsi orang lain dengan kunci publik yang sesuai. Selanjutnya, dengan beberapa algoritma enkripsi kunci publik, termasuk RSA, kami memiliki hubungan ini:

$$P = D(k_{PUB}, E(k_{PRIV}, P))$$

Dengan kata lain, pengguna dapat mengenkripsi pesan dengan kunci pribadi, dan pesan hanya dapat diungkapkan dengan kunci publik yang sesuai.

Kedua properti ini memberi tahu kita bahwa kunci publik dan pribadi dapat diterapkan dalam urutan mana pun. Secara khusus, fungsi dekripsi D dapat diterapkan pada argumen apa pun sehingga kita dapat mendekripsi sebelum mengenkripsi. Dengan enkripsi konvensional, kita jarang berpikir untuk mendekripsi sebelum mengenkripsi. Tetapi konsepnya masuk akal dengan kunci publik, di mana itu berarti menerapkan transformasi pribadi terlebih dahulu dan kemudian publik.

Kami telah mencatat bahwa masalah utama dengan enkripsi simetris adalah banyaknya kunci yang harus disimpan dan dilacak oleh satu pengguna. Dengan kunci publik, hanya dua kunci yang

diperlukan per pengguna: satu publik dan satu pribadi. Mari kita lihat apa perbedaannya dalam jumlah kunci yang dibutuhkan. Misalkan kita memiliki tiga pengguna, B, C, dan D, yang harus menyampaikan pesan yang dilindungi ke pengguna A dan juga satu sama lain. Karena setiap pasangan pengguna yang berbeda membutuhkan sebuah kunci, setiap pengguna akan membutuhkan tiga kunci yang berbeda; misalnya, A akan membutuhkan kunci untuk B, kunci untuk C, dan kunci untuk D. Tetapi menggunakan enkripsi kunci publik, masing-masing B, C, dan D dapat mengenkripsi pesan untuk A dengan menggunakan kunci publik A. Jika B telah mengenkripsi pesan menggunakan kunci publik A, C tidak dapat mendekripsinya, bahkan jika C tahu pesan tersebut dienkripsi dengan kunci publik A. Menerapkan kunci publik A dua kali, misalnya, tidak akan mendekripsi pesan. (Kami berasumsi, tentu saja, bahwa kunci pribadi A tetap rahasia.) Dengan demikian, jumlah kunci yang dibutuhkan dalam sistem kunci publik hanya dua per pengguna.

2.5.8 Algoritma Rivest–Shamir–Adelman (RSA)

Sistem kriptografi Rivest–Shamir–Adelman (RSA) adalah sistem kunci publik. Berdasarkan masalah sulit yang mendasari dan dinamai menurut tiga penemunya (Ronald Rivest, Adi Shamir, dan Leonard Adleman), algoritma ini diperkenalkan pada tahun 1978. Cryptanalysts telah melakukan RSA untuk cryptanalysis yang ekstensif, tetapi mereka tidak menemukan kekurangan yang serius. Dua kunci yang digunakan dalam RSA, *d* dan *e*, digunakan untuk dekripsi dan enkripsi. Mereka sebenarnya dapat dipertukarkan: Keduanya dapat dipilih sebagai kunci publik, tetapi yang satu telah dipilih, yang lain harus dirahasiakan. Untuk mempermudah, kita sebut kunci enkripsi e dan kunci dekripsi *d*. Kami menyatakan plaintext sebagai *P* dan ciphertext yang sesuai sebagai C. C = RSA(P,e). Juga, karena sifat algoritme RSA, kunci dapat diterapkan dalam salah satu urutan:

$$P = E(D(P)) = D(E(P))$$

atau

$$P = RSA(RSA(P, e), d) = RSA(RSA(P, d), e)$$

(Anda dapat menganggap E dan D sebagai dua fungsi yang saling melengkapi, yang masing-masing dapat "membatalkan" efek yang lain.)

RSA memang memiliki properti yang disayangkan bahwa kuncinya panjang: 256 bit dianggap sebagai panjang minimum yang dapat digunakan, tetapi dalam sebagian besar konteks, para ahli lebih memilih kunci pada urutan 1000 hingga 2000 bit. Enkripsi di RSA dilakukan dengan eksponensial, menaikkan setiap blok plaintext ke pangkat; bahwa kekuatan adalah kuncinya e. Berbeda dengan substitusi cepat dan transposisi algoritma simetris, eksponensial sangat memakan waktu di komputer. Lebih buruk lagi, waktu untuk mengenkripsi meningkat secara eksponensial saat eksponen (kunci) bertambah panjang. Dengan demikian, RSA jauh lebih lambat dari DES dan AES.

Algoritme enkripsi didasarkan pada masalah mendasar dari memfaktorkan bilangan besar dalam himpunan berhingga yang disebut bidang. Sejauh ini, belum ada yang menemukan jalan pintas atau cara mudah untuk memfaktorkan bilangan besar dalam suatu bidang. Dalam makalah yang sangat teknis namun sangat bagus, Dan Boneh [BON99] meninjau semua serangan kriptanalitik yang diketahui pada RSA dan menyimpulkan bahwa tidak ada yang signifikan. Karena masalah faktorisasi telah terbuka selama beberapa dekade, sebagian besar kriptografer menganggap masalah ini sebagai dasar yang kuat untuk sistem kripto yang aman.

Untuk meringkas, dua algoritma simetris DES dan AES menyediakan enkripsi solid dari blok 64 hingga 256 bit data. Algoritma asimetris RSA mengenkripsi blok dengan berbagai ukuran. DES dan AES secara substansial lebih cepat daripada RSA, dengan faktor 10.000 atau lebih, dan operasi primitif mereka yang agak sederhana telah dibangun ke dalam beberapa chip komputer, membuat enkripsi mereka bahkan lebih efisien daripada RSA. Oleh karena itu, orang cenderung menggunakan DES dan AES sebagai pekerjaan utama kriptografi, dan memesan RSA yang lebih lambat untuk penggunaan terbatas di mana ia unggul.

Karakteristik algoritma kunci rahasia (simetris) dan kunci publik (asimetris) dibandingkan pada Tabel 2-13.

Tabel 2-13 Perbandingan Kunci Rahasia dan Enkripsi Kunci Publik

	Kunci Rahasia (Symmetric Key)	Kunci Publik (Asymmetric)
Jumlah Kunci	1	2
Ukuran Kunci (bits)	Tergantiung dari algoritmanya : 56 -112 (DES); 128 - 256 (AES)	Tak terbatas : umumnya tidak lenuh dari 256; 1000 - 2000 saat ini dianggap sebagai yang sering dipakai
Proteksi Kunci	Harus di dirahasiakan	Satu kunci harus dirahasiakan; yang satu boleh diekspos
Penggunaan terbaik	para pekerja di bidang cryptographic, kerahasiaan dan integritas data, dari satu karakter ke blok data, messages dan file	Pertukaran kunci, auntentifikasi dan pendaftaran
Distribusi Kunci	Harus ***out of band*** (menghubungkan komputer secara langsung dengan port console atau dengan port auxiliary (AUX) dari router yang akan dikonfigurasikan	Kunci publik bisa digunalan untuk membuka kunci lain
Kecepatan	Cepat	Lambat, biasanya dengan faktor hingga 10.000 lebih lambat dari algoritma simetrik

2.5.9 Kriptografi Kunci Publik untuk Bertukar Kunci Rahasia

Algoritma enkripsi saja bukanlah jawaban untuk kebutuhan enkripsi semua orang. Meskipun enkripsi mengimplementasikan saluran komunikasi yang dilindungi, enkripsi juga dapat digunakan untuk tugas lain. Faktanya, menggabungkan enkripsi simetris dan asimetris sering kali memanfaatkan fitur terbaik masing-masing.

Misalkan Anda perlu mengirim pesan yang dilindungi ke seseorang yang tidak Anda kenal dan tidak mengenal Anda. Situasi ini lebih umum daripada yang mungkin Anda pikirkan. Misalnya, Anda mungkin ingin mengirimkan pengembalian pajak penghasilan Anda kepada pemerintah. Anda ingin informasi tersebut dilindungi, tetapi Anda belum tentu mengenal orang yang menerima informasi tersebut. Demikian pula, Anda mungkin ingin membeli dari situs belanja, bertukar email pribadi (terenkripsi), atau mengatur dua host untuk membuat saluran yang dilindungi. Masing-masing situasi ini bergantung pada kemampuan untuk menukar kunci enkripsi sedemikian rupa sehingga

tidak ada orang lain yang dapat mencegatnya. Masalah dua pihak yang sebelumnya tidak dikenal bertukar kunci kriptografi adalah sulit dan penting. Memang, masalahnya hampir melingkar: Untuk membuat sesi terenkripsi, Anda memerlukan sarana terenkripsi untuk bertukar kunci.

Kriptografi kunci publik dapat membantu. Karena kunci asimetris berpasangan, setengah dari pasangan dapat diekspos tanpa mengorbankan setengah lainnya. Faktanya, Anda mungkin menganggap bagian publik dari pasangan kunci benar-benar publik—diposting di situs web publik, terdaftar di direktori publik yang mirip dengan daftar telepon, atau dikirim secara terbuka dalam pesan email. Itulah keindahan kriptografi kunci publik: Selama kunci privat tidak diungkapkan, kunci publik dapat dibuka tanpa mengorbankan keamanan enkripsi.

Protokol Pertukaran Kunci

Misalkan pengirim, Nana, dan penerima, Anto, keduanya memiliki pasangan kunci asimetris untuk algoritma enkripsi umum. Kami menyatakan setiap fungsi enkripsi kunci publik sebagai *E(k, X)*, yang berarti melakukan fungsi enkripsi kunci publik pada *X* dengan menggunakan kunci *k*. Panggil kunci $k_{PRIV\text{-}A}$, $k_{PUB\text{-}A}$, $k_{PRIV\text{-}B}$, dan $k_{PUB\text{-}B}$, masing-masing untuk kunci privat dan publik Nana dan Anto.

Masalah yang ingin kami pecahkan adalah agar Nana dan Anto dapat membuat kunci enkripsi rahasia (algoritma simetris) yang hanya mereka yang tahu. Solusi paling sederhana adalah Nana memilih kunci simetris K, dan mengirim $E(k_{PRIV\text{-}A}, K)$ ke Anto. Anto mengambil kunci publik Nana, menghapus enkripsi, dan mendapatkan *K*.

Namun, analisis ini cacat. Bagaimana pengirim mengetahui kunci publik benar-benar milik penerima yang dituju? Pertimbangkan, misalnya, skenario berikut. Misalkan Nana dan Anto tidak memiliki papan buletin yang nyaman. Jadi, Nana hanya meminta kuncinya pada Anto. Simak skenario percakaapan yang terjadi berikut ini :

1. Nana berkata: Anto, tolong kirimkan saya kunci publik Anda.
2. Anto menjawab: Ini, Nana; ini adalah kunci publik saya.
3. Nana menjawab: Terima kasih. Saya telah membuat kunci simetris untuk kita gunakan untuk pertukaran ini. Saya mengirimi Anda kunci simetris yang dienkripsi di bawah kunci publik Anda.

Dalam subversi yang ditunjukkan pada Gambar 2-25, kami memasukkan penyerang, Revan, ke dalam komunikasi ini.

1. Nana berkata: Anto, tolong kirimkan saya kunci publik Anda.

1a. Revan mencegat pesan dan membuat pesan baru untuk Anto, yang mengaku berasal dari Nana tetapi dengan alamat pengirim Revan, meminta kunci publik Anto.

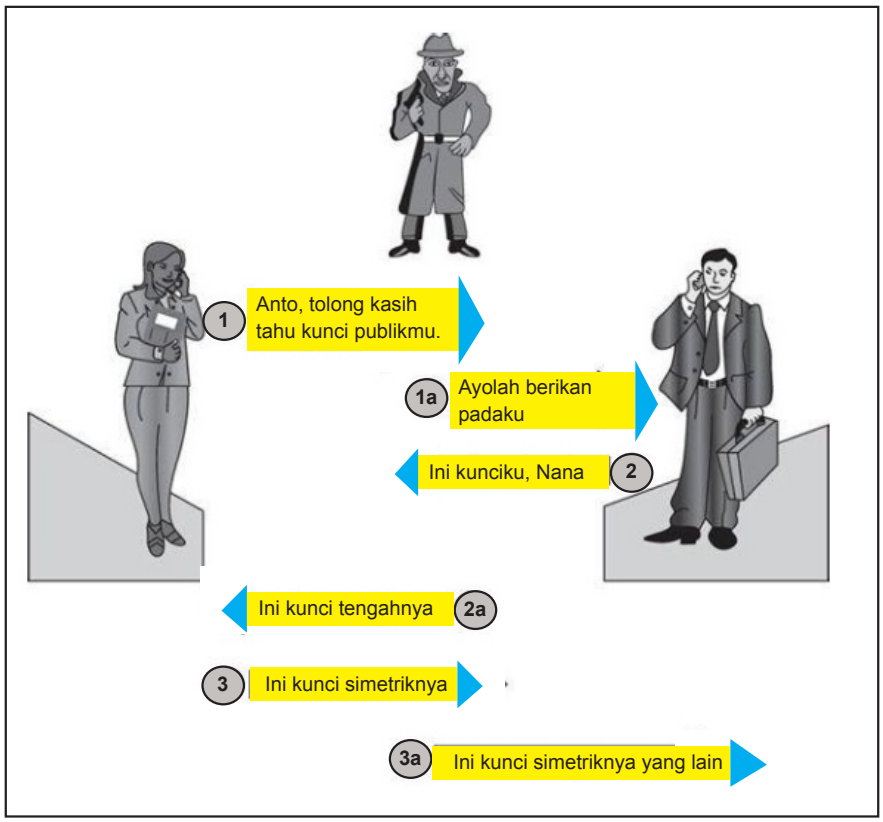
2. Anto menjawab: Ini, Nana; ini adalah kunci publik saya. (Karena alamat pengirim di langkah 1a, balasan ini ditujukan ke Revan.)

2a. Revan memegang kunci publik Anto dan mengirimkan kunci publik Revan sendiri ke Nana, menuduhnya dari Anto.

3. Nana menjawab: Terima kasih. Saya telah membuat kunci simetris untuk kita gunakan untuk pertukaran ini. Saya mengirimi Anda kunci simetris yang dienkripsi di bawah kunci publik Anda.

3a. Revan mencegat pesan ini dan memperoleh serta menahan kunci simetris yang telah dihasilkan Nana.

3b. Revan membuat kunci simetris baru dan mengirimkannya ke Anto, dengan pesan yang konon dari Nana: Terima kasih. Saya telah membuat kunci simetris untuk kita gunakan untuk pertukaran ini. Saya mengirimi Anda kunci simetris yang dienkripsi di bawah kunci publik Anda.



Gambar 2-25 Protokol Pertukaran Kunci dengan Pria di Tengah

Singkatnya, Revan sekarang memegang dua kunci enkripsi simetris, masing-masing dibagikan dengan Nana dan Anto. Revan tidak hanya bisa diam-diam mendapatkan semua simpang susun mereka, tetapi Nana dan Anto tidak dapat berkomunikasi dengan aman satu sama lain karena tidak ada yang berbagi kunci satu sama lain.

Mulai saat ini, semua komunikasi melewati Revan. Memiliki kedua kunci simetris, Revan dapat mendekripsi apa pun yang diterima, memodifikasinya, mengenkripsinya di bawah kunci lain, dan mengirimkan versi yang dimodifikasi ke pihak lain. Baik Nana maupun Anto tidak menyadari peralihan itu. Serangan ini adalah jenis kegagalan *man-in-the-middle*, di mana pihak ketiga yang tidak berwenang menengahi aktivitas yang dianggap eksklusif antara dua orang. Lihat Studi Kasus 2-15 untuk contoh serangan man-in-the-middle di dunia nyata.

Istilah *man in the middle*, terminologi ini sangat seksis. Bahkan jika kami menyebut serangan ini sebagai orang-di-tengah atau penyusup-di-tengah dalam buku ini, Anda hanya akan menemukan istilah man-in-the-middle yang digunakan oleh penulis lain, yang juga menggunakan istilah seperti *man- in-the-browser* dan *man-in-the-phone*. Jadi, sayangnya kita terjebak dengan istilah konvensional.

Man in The Middle attack atau MiTM merupakan istilah yang digunakan ketika hacker memposisikan dirinya di antara percakapan dua belah pihak seperti user dan website. Serangan ini dilakukan secara diam-diam sehingga korban tidak menyadari bahwa percakapan atau komunikasi yang dilakukan sedang diamati oleh hacker. Agar lebih mudah dipahami, Man in The Middle juga dapat digambarkan seperti seorang pekerja pos jahat yang dapat membaca dan mengubah surat sebelum diteruskan kepada penerima sah.

Studi Kasus 2-15 : Aspidistra, Manusia Perang Dunia II di Tengah

Selama Perang Dunia II Inggris menggunakan serangan man-in-the-middle untuk menipu pilot dan warga sipil Jerman. Aspidistra, nama tanaman hias biasa yang juga dikenal sebagai tanaman besi tuang karena tampaknya kemampuannya untuk hidup selamanya, juga merupakan nama yang diberikan kepada pemancar radio raksasa yang dibeli oleh Kantor Perang Inggris dari RCA pada tahun 1942. Pemancar disiarkan pada 500 kW daya, sepuluh kali daya yang diizinkan ke stasiun AS mana pun pada saat itu, yang berarti Aspidistra dapat mengirimkan sinyal dari Inggris ke Jerman.

Bagian dari operasi Aspidistra adalah untuk menipu pilot Jerman dengan menyiarkan arah palsu (darat, pergi ke sini, berbalik). Meskipun pilot juga menerima instruksi penerbangan yang valid dari pengontrol mereka sendiri, obrolan tambahan ini membingungkan mereka dan dapat mengakibatkan penerbangan yang tidak perlu dan kehilangan waktu. Bagian dari serangan ini hanyalah serangan peniruan identitas.

Stasiun radio Jerman tertentu di daerah sasaran dimatikan untuk mencegahnya menjadi suar yang dapat digunakan oleh pesawat Sekutu untuk menerima sinyal; pengebom akan mengikuti sinyal dan menghancurkan antena dan pemancar di dekatnya jika stasiun-stasiun itu terus-menerus menyiarkan. Ketika sebuah stasiun dimatikan, Inggris langsung mengudara menggunakan Aspidistra pada frekuensi yang sama dengan stasiun yang baru saja ditutup oleh Jerman. Mereka menyalin dan menyiarkan ulang program dari stasiun Jerman lain, tetapi mereka menyelingi pesan propaganda yang dapat menurunkan moral warga Jerman dan melemahkan dukungan untuk upaya perang.

Jerman mencoba untuk melawan siaran palsu dengan menasihati pendengar bahwa musuh sedang mentransmisikan dan menasihati penonton untuk mendengarkan pengumuman siaran resmi Jerman — yang, tentu saja, disalin dan disiarkan sendiri oleh Inggris. (Detail lebih lanjut dan gambar ada di http://www.qsl.net/g0crw/Special%20Events/Aspidistra2.htm, dan http://bobrowen.com/nymas/radioproppaper.pdf.)

Protokol Pertukaran Kunci yang Direvisi

Ingatlah bahwa kita memulai diskusi ini dengan serangan man-in-the-middle terhadap protokol pertukaran kunci sederhana. Protokol yang salah adalah

1. A mengatakan: B, tolong kirimkan saya kunci publik Anda.
2. B menjawab: Di sini, A; ini adalah kunci publik saya.
3. A menjawab: Terima kasih. Saya telah membuat kunci simetris untuk kita gunakan untuk pertukaran ini. Saya mengirimi Anda kunci simetris yang dienkripsi di bawah kunci publik Anda.

Pada langkah 2, penyusup mencegat kunci publik B dan melewati kunci publik si penyusup. Penyusup dapat digagalkan jika A dan B bertukar setengah kunci sekaligus. Setengah kunci tidak berguna bagi penyusup karena tidak cukup untuk mengenkripsi atau mendekripsi apa pun. Mengetahui setengah dari kunci tidak secara material meningkatkan kemampuan penyusup untuk memecahkan enkripsi di masa depan.

Rivest dan Shamir (Rivest:84) telah menyusun protokol yang solid sebagai berikut.

1. Nana mengirimkan kunci publiknya ke Anto.
2. Anto mengirimkan kunci publiknya ke Nana.
3. Nana membuat kunci simetris, mengenkripsinya menggunakan kunci publik Anto, dan mengirimkan setengah dari hasilnya ke Anto. (Catatan: setengah dari hasilnya mungkin n/2 bit pertama, semua bit bernomor ganjil, atau bentuk lain yang disepakati.)
4. Anto menjawab Nana bahwa dia menerima hasil sebagian (yang tidak dapat dia tafsirkan saat ini, jadi dia hanya mengkonfirmasi bahwa dia menerima beberapa bagian). Anto mengenkripsi nomor acak apa pun dengan kunci pribadinya dan mengirimkan setengah bit ke Nana.
5. Nana mengirimkan setengah lainnya dari hasil terenkripsi ke Anto.
6. Anto mengumpulkan dua bagian dari hasil Nana, mendekripsinya menggunakan kunci pribadinya dan dengan demikian memperoleh kunci simetris bersama. Anto mengirimkan separuh nomor acak terenkripsi lainnya kepada Nana.
7. Nana mengumpulkan dua bagian nomor acak Anto, mendekripsinya menggunakan kunci pribadinya, mengekstrak nomor acak Anto, mengenkripsinya menggunakan kunci simetris yang sekarang dibagikan, dan mengirimkannya ke Anto.
8. Anto mendekripsi transmisi Nana dengan kunci simetris dan membandingkannya dengan nomor acak yang dipilihnya pada langkah 6. Kecocokan mengkonfirmasi validitas pertukaran.

Untuk melihat mengapa protokol ini bekerja, lihat langkah 3. Revan, si penyusup, pasti dapat mencegat kedua kunci publik di langkah 1 dan 2 dan mengganti miliknya sendiri. Namun, pada langkah 3 Revan tidak dapat mengambil setengah dari hasil, mendekripsi menggunakan kunci pribadinya, dan mengenkripsi ulang di bawah kunci Anto. Bit tidak dapat didekripsi satu per satu dan dipasang kembali.

Pada langkah 4 Anto memilih nomor acak apa pun, yang kemudian dikembalikan Nana kepada Anto untuk menunjukkan bahwa dia telah berhasil menerima nilai terenkripsi yang dikirim Anto. Nilai acak seperti itu disebut nonce, nilai yang tidak berarti dalam dirinya sendiri, untuk menunjukkan aktivitas (keaktifan) dan orisinalitas (bukan replay). Dalam beberapa protokol, penerima mendekripsi nonce, menambahkan 1 ke dalamnya, mengenkripsi ulang hasilnya, dan mengembalikannya. Di lain

waktu nonce menyertakan tanggal, waktu, atau nomor urut untuk menunjukkan bahwa nilainya saat ini. Konsep ini digunakan dalam pertukaran komputer-ke-komputer yang tidak memiliki beberapa karakteristik interaksi manusia.

2.4.10 Keaslian

Masalah orang di tengah dapat diselesaikan dengan cara lain: Nana harus mengirim ke Anto

$$E(k_{PUB-B}, E(k_{PRIV-A}, K))$$

Fungsi ini memastikan bahwa hanya Anto, menggunakan k_{PRIV-B}, yang dapat menghapus enkripsi yang diterapkan dengan kPUB-B, dan Anto tahu bahwa hanya Nana yang dapat menerapkan k_{PRIV-A} yang dihapus Anto dengan k_{PUB-A}.

Kita dapat memikirkan pertukaran ini dalam hal kunci dan segel. Siapa pun dapat memasukkan surat ke dalam kotak surat yang terkunci (melalui slot surat), tetapi hanya pemegang kunci yang dapat mengeluarkannya. Di masa lalu, orang-orang penting memiliki segel yang akan mereka tempelkan ke dalam lilin cair pada sebuah surat; jejak segel menunjukkan keaslian, tetapi siapa pun dapat memecahkan segel dan membaca surat itu. Menyatukan kedua bagian ini, sebuah surat yang disegel di dalam kotak surat yang terkunci menegakkan keaslian pengirim (stempel) dan kerahasiaan penerima (kotak surat yang terkunci).

Jika Nana ingin mengirim sesuatu yang dilindungi ke Anto (seperti nomor kartu kredit atau satu set catatan medis), maka pertukaran bekerja seperti ini. Nana menyegel informasi yang dilindungi dengan kunci pribadinya sehingga hanya dapat dibuka dengan kunci publik Nana. Langkah ini memastikan keaslian: hanya Nana yang dapat menerapkan enkripsi yang dibalikkan dengan kunci publik Nana. Nana kemudian mengunci informasi tersebut dengan kunci publik Anto. Langkah ini menambah kerahasiaan karena hanya kunci pribadi Anto yang dapat mendekripsi data yang dienkripsi dengan kunci publik Anto. Anto dapat menggunakan kunci pribadinya untuk membuka kotak surat (sesuatu yang hanya bisa dia lakukan) dan menggunakan kunci publik Nana untuk memverifikasi segel dalam (membuktikan bahwa paket itu berasal dari Nana).

Jadi, seperti yang telah kita lihat, fungsi kriptografi asimetris adalah sarana yang ampuh untuk bertukar kunci kriptografi antara orang-orang yang tidak memiliki hubungan sebelumnya. Fungsi kriptografi asimetris lambat, tetapi hanya digunakan sekali, untuk bertukar kunci simetris. Selanjutnya, jika kunci yang dipertukarkan adalah untuk sistem enkripsi simetris seperti AES atau DES, panjang kuncinya relatif pendek, hingga 256 bit untuk AES atau 64 untuk DES. Bahkan jika kita menggunakan bentuk AES yang diperluas dengan panjang kunci 1000 bit, kecepatan lambat kriptografi kunci publik tidak akan menjadi masalah yang signifikan karena hanya dilakukan sekali, untuk membuat kunci bersama.

Kriptografi asimetris juga berguna untuk konstruksi keamanan penting lainnya: tanda tangan digital. Tanda tangan manusia pada dokumen kertas adalah pengesahan yang kuat: itu menandakan kesepakatan (Anda setuju dengan persyaratan dalam dokumen yang Anda tandatangani) dan pemahaman (Anda tahu apa yang Anda tanda tangani). Orang-orang menerima tanda tangan

tertulis sebagai pengganti konfirmasi langsung. Kami ingin konstruksi yang sama kuatnya untuk mengkonfirmasi dokumen elektronik. Untuk membangun tanda tangan digital, kami memperkenalkan kode integritas, sertifikat kunci, dan, terakhir, tanda tangan itu sendiri.

Kode Pendeteksi Kesalahan

Komunikasi sangat rentan terhadap kesalahan dalam transmisi. Anda mungkin telah memperhatikan bahwa terkadang percakapan telepon seluler akan melewatkan atau mendistorsi sebagian kecil percakapan, dan sinyal televisi terkadang menunjukkan masalah yang biasa disebut kebisingan. Dalam Studi Kasus ini, penerimaan yang lengkap dan akurat tidak penting selama kebisingannya relatif kecil atau jarang. Anda selalu dapat meminta mitra telepon Anda untuk mengulangi kalimat, dan gol kemenangan di televisi selalu disiarkan ulang berkali-kali.

Dengan data penting, bagaimanapun, kita memerlukan beberapa cara untuk menentukan bahwa transmisi lengkap dan utuh. Matematikawan dan insinyur telah merancang formula yang disebut deteksi kesalahan dan kode koreksi untuk membuat kesalahan transmisi terlihat dan untuk melakukan perbaikan kecil.

Kode pendeteksi kesalahan memiliki banyak nama, seperti kode hash, intisari pesan, checksum, pemeriksaan integritas, deteksi kesalahan dan kode koreksi, dan tes redundansi. Meskipun istilah-istilah ini memiliki perbedaan arti yang baik, tujuan dasar dari semuanya adalah untuk menunjukkan bahwa sebuah blok data telah dimodifikasi. Kalimat itu diucapkan dengan hati-hati: Intisari pesan akan (kadang-kadang) memberi sinyal bahwa konten telah berubah, tetapi kurang solid dalam menunjukkan tidak ada modifikasi, meskipun itulah yang sebenarnya kita inginkan. Kami ingin sesuatu untuk menunjukkan tidak ada gangguan—berbahaya atau tidak; kita mendapatkan sesuatu yang biasanya menunjukkan gangguan.

Rudi menulis surat, membuat dan menyimpan fotokopi, dan mengirimkan aslinya kepada Amanda. Sepanjang jalan, Samuel mencegat surat itu dan membuat perubahan; menjadi pemalsu yang terampil, Samuel menipu Amanda. Hanya ketika Amanda dan Rudi bertemu dan membandingkan (dimodifikasi) asli mereka mendeteksi perubahan.

Lain halnya jika Rudi dan Amanda mencurigai seorang pemalsu sudah dekat. Rudi dengan hati-hati menghitung huruf-huruf dalam dokumennya, menghitung 1 untuk a, 2 untuk *a b*, dan seterusnya hingga 26 untuk *a z*. Dia menambahkan nilai-nilai itu dan menulis jumlahnya dalam angka-angka kecil di bagian bawah surat itu.

Ketika Teresa menerima surat itu, dia melakukan perhitungan yang Rudia dan membandingkan hasilnya dengan yang tertulis di bawah. Tiga Studi Kasus muncul:

- terhitung tidak setuju, dalam hal ini Amanda menduga ada perubahan
- tidak ada hitungan, dalam hal ini Amanda menduga Rudi malas atau pelupa atau pemalsu mengabaikan kode mereka
- Hitungan Teresa Rudia dengan yang tertulis di bawah

Studi Kasus terakhir adalah yang paling bermasalah. Amanda mungkin menyimpulkan dengan lega bahwa tidak ada perubahan. Namun, seperti yang mungkin sudah Anda tentukan, dia mungkin tidak

berpikir dengan benar. Samuel mungkin menangkap kode dan menghitung dengan benar jumlah baru agar sesuai dengan modifikasi. Lebih buruk lagi, mungkin perubahan Samuel terjadi untuk menghindari deteksi. Misalkan Samuel menghapus satu huruf c (nilai=3) dan menggantinya dengan tiga salinan huruf a (nilai=1+1+1=3); jumlahnya Rudia, atau jika Samuel hanya mengubah huruf, jumlahnya tetap Rudia, karena tidak peka terhadap urutan.

Semua masalah ini muncul karena kode adalah fungsi banyak-ke-satu: dua atau lebih input menghasilkan output yang sama. Dua input yang menghasilkan output yang sama disebut collision. Faktanya, semua intisari pesan adalah fungsi banyak-ke-satu, dan dengan demikian ketika mereka melaporkan perubahan, satu memang terjadi, tetapi ketika mereka melaporkan tidak ada perubahan, itu hanya mungkin — tidak pasti — bahwa tidak ada yang terjadi karena kemungkinan tabrakan.

Tabrakan biasanya tidak menjadi masalah karena dua alasan. Pertama, mereka jarang terjadi. Jika plaintext direduksi menjadi 64-bit digest, kami berharap kemungkinan tabrakan menjadi 1 dalam 264, atau sekitar 1 dalam 1019, sangat tidak mungkin, memang. Lebih penting lagi, fungsi digest tidak dapat diprediksi, jadi dengan satu input, menemukan input kedua yang menghasilkan output yang sama tidak mungkin dilakukan. Jadi, dengan fungsi digest yang baik, tabrakan jarang terjadi, dan kami tidak dapat menyebabkan atau memprediksinya.

Kita dapat menggunakan pendeteksi kesalahan dan kode koreksi kesalahan untuk menjaga dari modifikasi data. Kode deteksi dan koreksi adalah prosedur atau fungsi yang diterapkan pada blok data; Anda mungkin akrab dengan satu jenis kode pendeteksi: paritas. Kode-kode ini berfungsi sesuai namanya: Kode pendeteksi kesalahan mendeteksi ketika kesalahan telah terjadi, dan kode koreksi kesalahan sebenarnya dapat memperbaiki kesalahan tanpa memerlukan salinan data asli. Kode kesalahan dihitung dan disimpan dengan aman pada data asli yang dianggap utuh; nanti siapa pun dapat menghitung ulang kode kesalahan dan memeriksa apakah hasil yang diterima sesuai dengan nilai yang diharapkan. Jika nilainya tidak cocok, pasti telah terjadi perubahan; jika nilainya cocok, kemungkinan—tetapi tidak pasti—tidak ada perubahan yang terjadi.

Pemeriksaan Paritas (Parity Check)

Kode deteksi kesalahan yang paling sederhana adalah pemeriksaan paritas. Bit tambahan, yang kita sebut sidik jari, ditambahkan ke grup bit data yang ada, tergantung pada jumlahnya. Dua jenis paritas disebut genap dan ganjil. Dengan paritas genap sidik jari adalah 0 jika jumlah bit data genap, dan 1 jika jumlahnya ganjil; yaitu, bit paritas diatur sehingga jumlah semua bit data ditambah bit paritas genap. Paritas ganjil adalah sama kecuali jumlah keseluruhan ganjil. Misalnya, aliran data 01101101 akan memiliki bit paritas genap 1 (dan bit paritas ganjil 0) karena 0+1+1+0+1+1+0+1 = 5 + 1 = 6 (atau 5 + 0 = 5 untuk paritas ganjil).

Satu bit paritas dapat mengungkapkan modifikasi satu bit. Namun, paritas tidak mendeteksi kesalahan dua bit—Studi Kasus di mana dua bit dalam grup diubah. Satu bit paritas dapat mendeteksi semua perubahan bit tunggal, serta perubahan tiga, lima dan tujuh bit. Tabel 2-14 menunjukkan beberapa contoh perubahan yang terdeteksi dan tidak terdeteksi. Bit yang diubah (setiap baris menunjukkan perubahan dari nilai asli 00000000) dicetak tebal, digarisbawahi; tabel menunjukkan apakah paritas benar mendeteksi bahwa setidaknya satu perubahan terjadi.

TABEL 2-14 Perubahan yang Dideteksi oleh Paritas

Original Data	Parity Bit	Modified Data	Modification Detected?
0 0 0 0 0 0 0 0	1	0 0 0 0 0 0 0 1	Yes
0 0 0 0 0 0 0 0	1	1 0 0 0 0 0 0 0	Yes
0 0 0 0 0 0 0 0	1	1 0 0 0 0 0 0 1	No
0 0 0 0 0 0 0 0	1	0 0 0 0 0 0 1 1	No
0 0 0 0 0 0 0 0	1	0 0 0 0 0 1 1 1	Yes
0 0 0 0 0 0 0 0	1	0 0 0 0 1 1 1 1	No
0 0 0 0 0 0 0 0	1	0 1 0 1 0 1 0 1	No
0 0 0 0 0 0 0 0	1	1 1 1 1 1 1 1 1	No

Mendeteksi angka ganjil dari bit yang diubah mengarah ke tingkat deteksi perubahan sekitar 50 persen, yang hampir tidak cukup baik untuk tujuan kita. Kita dapat meningkatkan kecepatan ini dengan lebih banyak bit paritas (misalnya menghitung bit paritas kedua dari bit 1, 3, 5, dan 7), tetapi lebih banyak bit paritas meningkatkan ukuran sidik jari; setiap kali kami menambah ukuran sidik jari, kami juga meningkatkan ukuran penyimpanan sidik jari ini.

Paritas hanya menandakan bahwa sedikit telah diubah; itu tidak mengidentifikasi bit mana yang telah diubah, apalagi kapan, bagaimana, atau oleh siapa. Pada perangkat penyimpanan perangkat keras, kode yang disebut pemeriksaan redundansi siklik mendeteksi kesalahan dalam perekaman dan pemutaran. Beberapa kode yang lebih kompleks, yang dikenal sebagai kode koreksi kesalahan, dapat mendeteksi kesalahan beberapa bit (dua atau lebih bit diubah dalam kelompok data) dan mungkin dapat menunjukkan dengan tepat bit yang diubah (yang merupakan bit yang akan diatur ulang untuk memperbaiki modifikasi). Ukuran sidik jari, tingkat deteksi kesalahan, dan koreksi membawa kita ke kode yang lebih kuat.

Kode Hash

Di sebagian besar file, elemen atau komponen file tidak terikat bersama dengan cara apa pun. Artinya, setiap byte atau bit atau karakter tidak tergantung satu sama lain dalam file. Kurangnya pengikatan ini berarti bahwa mengubah satu nilai memengaruhi integritas file tetapi satu perubahan itu dapat dengan mudah tidak terdeteksi.

Apa yang ingin kami lakukan adalah memasang segel atau pelindung di sekitar file sehingga kami dapat mendeteksi ketika segel telah rusak dan dengan demikian mengetahui bahwa ada sesuatu yang telah diubah. Gagasan ini mirip dengan penggunaan segel lilin pada surat-surat di abad pertengahan; jika lilin itu pecah, penerima akan tahu bahwa seseorang telah memecahkan segelnya dan membaca pesan di dalamnya. Dengan cara yang sama, kriptografi dapat digunakan untuk menyegel file, membungkusnya sehingga setiap perubahan menjadi jelas. Salah satu teknik untuk memberikan segel adalah menghitung fungsi, kadang-kadang disebut hash atau checksum atau intisari pesan dari file.

Kode antara Rudi dan Amanda adalah kode hash. Kode hash sering digunakan dalam komunikasi di mana kesalahan transmisi dapat mempengaruhi integritas data yang dikirimkan. Dalam Studi Kasus tersebut nilai kode ditransmisikan dengan data. Apakah data atau nilai kode dirusak, penerima mendeteksi beberapa masalah dan hanya meminta transmisi ulang blok data.

Protokol semacam itu memadai dalam Studi Kasus kesalahan yang tidak disengaja tetapi tidak dimaksudkan untuk menangani musuh khusus. Jika Fagin mengetahui algoritma fungsi deteksi kesalahan, maka ia dapat mengubah konten dan memperbaiki nilai deteksi agar sesuai. Jadi, ketika musuh jahat mungkin terlibat, komunikasi yang aman membutuhkan bentuk intisari pesan yang lebih kuat.

Fungsi Hash Satu Arah

Sebagai langkah pertama dalam mengalahkan Samuel, kita harus mencegahnya bekerja mundur dari nilai intisari untuk melihat input apa yang mungkin mengarah pada hasil tersebut. Misalnya, beberapa enkripsi bergantung pada fungsi yang mudah dipahami tetapi sulit untuk dihitung. Sebagai contoh sederhana, perhatikan fungsi kubus, y = x3. Menghitung x3 dengan tangan, dengan pensil dan kertas, atau dengan kalkulator tidaklah sulit. Tetapi fungsi invers, , jauh lebih sulit untuk dihitung. Dan fungsi y = x2 tidak memiliki fungsi invers karena ada dua kemungkinan untuk : + x dan x. Fungsi seperti ini, yang jauh lebih mudah dihitung daripada kebalikannya, disebut fungsi satu arah.

Deteksi Perubahan File

Fungsi satu arah dapat berguna dalam membuat algoritma pendeteksian perubahan. Fungsi harus bergantung pada semua bit file yang disegel, sehingga setiap perubahan bahkan satu bit pun akan mengubah hasil checksum. Nilai checksum disimpan dengan file. Kemudian, setiap kali seseorang mengakses atau menggunakan file tersebut, sistem akan menghitung ulang checksum tersebut. Jika checksum yang dihitung cocok dengan nilai yang disimpan, file tersebut kemungkinan akan utuh.

Properti satu arah menjaga dari modifikasi berbahaya: Penyerang tidak dapat "membatalkan" fungsi untuk melihat file aslinya, jadi tidak ada cara sederhana untuk menemukan serangkaian perubahan yang menghasilkan nilai fungsi yang sama. (Jika tidak, penyerang dapat menemukan modifikasi yang tidak terdeteksi yang juga memiliki dampak berbahaya.)
Tripwire [KIM98] adalah program utilitas yang melakukan pemeriksaan integritas pada file. Dengan Tripwire, administrator sistem menghitung hash dari setiap file dan menyimpan nilai hash ini di tempat yang aman, biasanya offline. Kemudian administrator menjalankan kembali Tripwire dan membandingkan nilai hash baru dengan yang sebelumnya.

Checksum Kriptografi

Modifikasi berbahaya harus ditangani dengan cara yang juga mencegah penyerang memodifikasi mekanisme deteksi kesalahan serta bit data itu sendiri. Salah satu cara untuk mengatasinya adalah dengan menggunakan teknik yang menyusutkan dan mengubah data sesuai dengan nilai bit data.

Sebuah checksum kriptografi adalah fungsi kriptografi yang menghasilkan sebuah checksum. Ini adalah fungsi intisari menggunakan kunci kriptografik yang mungkin hanya diketahui oleh pencetus dan penerima data yang tepat. Kriptografi mencegah penyerang mengubah blok data (plaintext)

dan juga mengubah nilai checksum (ciphertext) agar sesuai. Penyerang tentu saja dapat mengubah plaintext, tetapi penyerang tidak memiliki kunci untuk menghitung ulang checksum. Salah satu contoh checksum kriptografi adalah pertama-tama menggunakan fungsi checksum nonkriptografis untuk memperoleh intisari n-bit dari data sensitif. Kemudian terapkan algoritma enkripsi simetris apa pun ke intisari. Tanpa kunci penyerang tidak dapat menentukan nilai checksum yang disembunyikan oleh enkripsi.

Dua kegunaan utama dari checksum kriptografi adalah perlindungan kode-tamper dan perlindungan integritas pesan dalam perjalanan. Perlindungan gangguan kode diimplementasikan dengan cara yang baru saja kami jelaskan untuk mendeteksi perubahan pada file. Demikian pula, checksum pada data dalam komunikasi mengidentifikasi data yang telah diubah dalam transmisi, baik secara sengaja maupun tidak sengaja. Pemerintah AS mendefinisikan Secure Hash Standard or Algorithm (SHS atau SHA), sebenarnya kumpulan algoritme, untuk menghitung checksum.

Checksum adalah tindakan pencegahan yang penting untuk mendeteksi modifikasi. Di bagian ini kami menerapkannya pada masalah mendeteksi modifikasi berbahaya pada program yang disimpan di disk, tetapi teknik yang sama berlaku untuk melindungi dari perubahan data, seperti yang akan kami tunjukkan nanti dalam buku ini.

Algoritme kriptografi yang kuat, seperti untuk DES atau AES, sangat sesuai untuk menyegel nilai, karena orang luar tidak akan mengetahui kuncinya dan dengan demikian tidak akan dapat mengubah nilai yang disimpan agar sesuai dengan data yang sedang dimodifikasi. Untuk aplikasi dengan ancaman rendah, algoritma yang lebih sederhana daripada DES atau AES dapat digunakan. Dalam skema enkripsi blok, chaining berarti menghubungkan setiap blok ke nilai blok sebelumnya (dan karena itu ke semua blok sebelumnya), misalnya, dengan menggunakan OR eksklusif untuk menggabungkan blok sebelumnya yang dienkripsi dengan yang sekarang. Checksum kriptografi file bisa menjadi blok terakhir dari enkripsi berantai file karena blok itu akan bergantung pada semua blok lainnya.

Seperti yang akan kita lihat nanti dalam bab ini, teknik-teknik ini membahas non-alterability dan non-reusability yang diperlukan dalam tanda tangan digital. Perubahan atau penggunaan kembali mungkin akan ditandai oleh checksum sehingga penerima dapat mengetahui bahwa ada sesuatu yang salah.

2.5.11 Tanda Tangan (*Digital Signature*)

Teknik paling ampuh untuk menunjukkan keaslian adalah tanda tangan digital. Seperti mitranya di atas kertas, tanda tangan digital adalah cara seseorang atau organisasi dapat membubuhkan pola bit ke file sedemikian rupa sehingga menyiratkan konfirmasi, berkaitan dengan file itu saja, tidak dapat dipalsukan, dan menunjukkan keasliannya. Kami menginginkan sarana di mana satu pihak dapat menandatangani sesuatu dan, seperti di atas kertas, tanda tangan itu tetap berlaku selama berhari-hari, berbulan-bulan, bertahun-tahun—tanpa batas. Selanjutnya, tanda tangan harus meyakinkan semua orang yang mengakses file tersebut. Tentu saja, seperti kebanyakan kondisi yang melibatkan metode digital, peringatannya adalah bahwa jaminan dibatasi oleh keterampilan dan energi yang diasumsikan dari siapa pun yang akan mencoba mengalahkan jaminan.

Tanda tangan digital sering menggunakan kriptografi kunci publik atau asimetris. Seperti yang baru saja Anda lihat, protokol kunci publik berguna untuk pertukaran kunci kriptografi antara dua pihak yang tidak memiliki dasar kepercayaan lain. Sayangnya, protokol kriptografi kunci publik melibatkan beberapa urutan pesan dan balasan, yang dapat memakan waktu jika salah satu pihak tidak segera tersedia untuk membalas permintaan terbaru. Akan berguna untuk memiliki teknik di mana satu pihak dapat dengan andal menghitung beberapa langkah protokol dan meninggalkannya di tempat yang aman sehingga protokol dapat dijalankan meskipun hanya satu pihak yang aktif. Situasi ini mirip dengan perbedaan antara teller bank dan ATM. Anda dapat memperoleh uang tunai, melakukan setoran atau pembayaran, atau memeriksa saldo Anda karena bank telah menetapkan langkah-langkah untuk ATM untuk menangani aktivitas sederhana tersebut 24 jam sehari, bahkan jika bank tidak buka. Tetapi jika Anda memerlukan cek bersertifikat atau mata uang asing, Anda mungkin perlu berinteraksi langsung dengan agen bank.
Di bagian ini kami mendefinisikan tanda tangan digital dan membandingkan propertinya dengan tanda tangan tulisan tangan di atas kertas. Kami kemudian menjelaskan infrastruktur seputar tanda tangan digital yang memungkinkan mereka dikenali dan valid tanpa batas.

Komponen dan Karakteristik Tanda Tangan

Tanda tangan digital hanyalah objek biner yang terkait dengan file. Tetapi jika kita ingin tanda tangan itu memiliki kekuatan tanda tangan berbasis kertas, kita perlu memahami sifat-sifat tanda tangan manusia. Hanya dengan begitu kami dapat menyatakan persyaratan untuk versi digital kami.

Properti Tanda Tangan Berbasis Kertas yang Aman

Pertimbangkan situasi khas yang sejajar dengan kebutuhan manusia biasa: perintah untuk mentransfer dana dari satu orang ke orang lain. Dengan kata lain, kami ingin dapat mengirimkan cek elektronik yang setara dengan cek terkomputerisasi. Kami memahami sifat-sifat transaksi ini untuk pemeriksaan kertas konvensional:

- Cek adalah benda berwujud yang memberikan otorisasi transaksi keuangan.
- Tanda tangan pada cek menegaskan keaslian karena (mungkin) hanya penandatangan yang sah yang dapat membuat tanda tangan tersebut.
- Dalam Studi Kasus dugaan pemalsuan, pihak ketiga dapat dipanggil untuk menilai keasliannya.
- Setelah cek dicairkan, cek tersebut dibatalkan sehingga tidak dapat digunakan kembali.
- Pemeriksaan kertas tidak dapat diubah. Atau, sebagian besar bentuk perubahan mudah dideteksi.

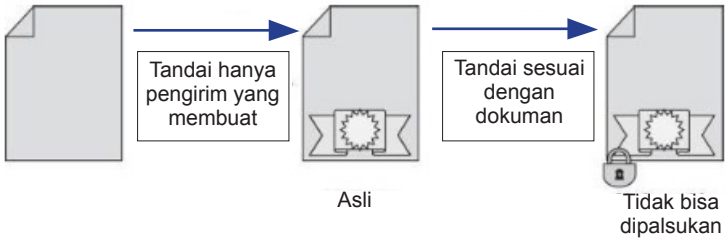
Bertransaksi bisnis dengan cek tergantung pada objek nyata dalam bentuk yang ditentukan. Tetapi objek berwujud tidak ada untuk transaksi di komputer. Oleh karena itu, otorisasi pembayaran dengan komputer memerlukan model yang berbeda. Mari kita pertimbangkan persyaratan situasi seperti itu, dari sudut pandang bank dan pengguna.

Sifat Tanda Tangan Digital

Misalkan Sheila mengirimi banknya pesan yang mengizinkannya untuk mentransfer $100 ke Rob. Bank Sheila harus dapat memverifikasi dan membuktikan bahwa pesan tersebut benar-benar berasal dari Sheila jika dia kemudian menolak mengirim pesan tersebut. (Properti ini disebut non-

repudiation.) Bank juga ingin mengetahui bahwa pesan tersebut sepenuhnya milik Sheila, bahwa pesan tersebut tidak diubah selama ini. Sementara itu, Sheila ingin memastikan bahwa banknya tidak dapat memalsukan pesan seperti itu. (Properti ini disebut keaslian.) Kedua belah pihak ingin memastikan bahwa pesan tersebut baru, bukan penggunaan kembali dari pesan sebelumnya, dan bahwa pesan tersebut tidak diubah selama transmisi. Menggunakan sinyal elektronik sebagai pengganti kertas memperumit proses ini.

Tanda tangan digital adalah protokol yang menghasilkan efek yang sama dengan tanda tangan asli: Ini adalah tanda yang hanya dapat dibuat oleh pengirim tetapi dapat dikenali dengan mudah oleh orang lain sebagai milik pengirim. Sama seperti tanda tangan asli, tanda tangan digital mengkonfirmasi persetujuan terhadap sebuah pesan.



Gambar 2-26 Persyaratan Tanda Tangan Digital

Tanda tangan digital harus memenuhi dua syarat utama:

- Itu harus tidak bisa ditempa. Jika orang S menandatangani pesan M dengan tanda tangan Sig(S,M), tidak ada orang lain yang dapat menghasilkan pasangan [M,Sig(S,M)].
- Itu harus asli. Jika seseorang R menerima pasangan [M, Sig(S,M)] konon dari S, R dapat memeriksa bahwa tanda tangan tersebut benar-benar dari S. Hanya S yang dapat membuat tanda tangan ini, dan tanda tangan tersebut melekat erat pada M.

Kedua persyaratan ini, yang ditunjukkan pada Gambar 2-26, merupakan rintangan utama dalam transaksi komputer. Dua properti lagi, juga diambil dari paralel dengan lingkungan berbasis kertas, diinginkan untuk transaksi yang diselesaikan dengan bantuan tanda tangan digital:

- Tidak dapat diubah. Setelah ditransmisikan, M tidak dapat diubah oleh S, R, atau pencegat.
- Tidak dapat digunakan kembali. Pesan sebelumnya yang disajikan lagi akan langsung terdeteksi oleh R.

Untuk melihat cara kerja tanda tangan digital, pertama-tama kami menyajikan mekanisme yang memenuhi dua persyaratan pertama. Kami kemudian menambahkan solusi itu untuk memenuhi persyaratan lainnya. Kami mengembangkan tanda tangan digital dalam potongan-potongan: pertama membangun sepotong untuk mengatasi perubahan, kemudian menjelaskan cara untuk memastikan keaslian, dan akhirnya mengembangkan struktur untuk membangun identitas. Akhirnya semua bagian ini mengikat bersama dalam kerangka konseptual sederhana.

Kami baru saja menjelaskan bagian-bagian untuk tanda tangan digital: kriptografi kunci publik dan intisari pesan yang aman. Kedua bagian ini secara teknis cukup untuk membuat tanda tangan digital, tetapi tidak membahas keasliannya. Untuk itu, diperlukan suatu struktur yang mengikat identitas pengguna dan kunci publik dengan cara yang dapat dipercaya. Struktur seperti itu disebut sertifikat. Terakhir, kami menghadirkan infrastruktur untuk mentransmisikan dan memvalidasi sertifikat.

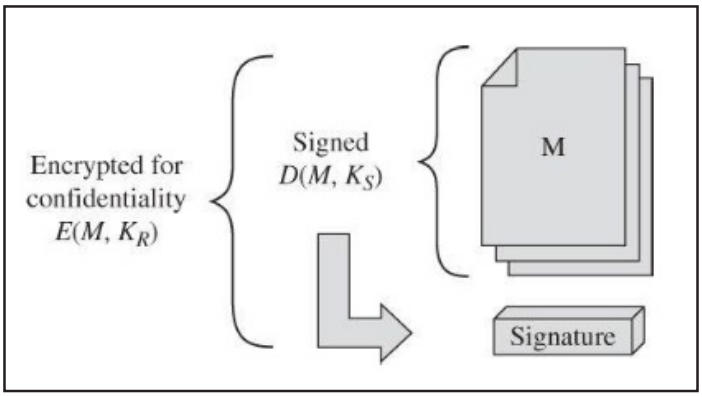
Kunci Publik untuk Tanda Tangan

Sistem enkripsi kunci publik cocok untuk penandatanganan. Untuk notasi sederhana, mari kita asumsikan bahwa enkripsi kunci publik untuk pengguna *U* diakses melalui $E(M,K_U)$ dan transformasi kunci privat untuk *U* ditulis sebagai $D(M,K_U)$. Kita dapat menganggap *E* sebagai transformasi privasi (karena hanya *U* yang dapat mendekripsinya) dan *D* sebagai transformasi keaslian (karena hanya *U* yang dapat memproduksinya). Ingat, bagaimanapun, bahwa di bawah beberapa algoritma asimetris seperti RSA, *D* dan *E* adalah komutatif dan salah satu dapat diterapkan untuk pesan apapun. Jadi,

$$D(E(M, K_U), K_U) = M = E(D(M, K_U), K_U)$$

Jika *S* ingin mengirim *M* ke *R, S* menggunakan transformasi otentisitas untuk menghasilkan $D(M, K_S)$. S kemudian mengirim $D(M, K_S)$ ke *R*. *R* menerjemahkan pesan dengan transformasi kunci publik S, menghitung $E(D(M, K_S), K_S) = M$. Karena hanya S yang dapat membuat pesan yang masuk akal di bawah $E(-,K_S)$, pesan harus benar-benar berasal dari S. Pengujian ini memenuhi syarat keaslian.

R akan menghemat $D(M, K_S)$. Jika S kemudian menyatakan bahwa pesan itu palsu (bukan benar-benar dari *S), R* cukup menunjukkan *M* dan $D(M, K_S)$. Siapapun dapat memverifikasi bahwa karena $D(M, K_S)$ ditransformasikan ke M dengan transformasi kunci publik *S* — tetapi hanya *S* yang dapat menghasilkan $D(M, K_S)$ — maka $D(M, K_S)$ harus dari *S*. Tes ini memenuhi persyaratan yang tidak dapat dipalsukan.



Gambar 2-27 Penggunaan Dua Kunci dalam Tanda Tangan Digital Asimetris

Ada pendekatan lain untuk penandatanganan; beberapa menggunakan enkripsi simetris, yang lain menggunakan asimetris. Pendekatan yang ditunjukkan di sini mengilustrasikan bagaimana protokol dapat mengatasi persyaratan untuk ketidakterbatasan dan keaslian. Untuk menambah kerahasiaan, *S* menerapkan $E(M, K_R)$ seperti yang ditunjukkan pada Gambar 2-27.

Potongan-potongan ini, fungsi hash, kriptografi kunci publik, dan protokol, memberi kita potongan teknis tanda tangan digital. Namun, kami juga membutuhkan satu komponen nonteknis. Penanda tangan *S* kami pasti dapat melakukan protokol untuk menghasilkan tanda tangan digital, dan siapa pun yang memiliki kunci publik *S* dapat menentukan bahwa tanda tangan itu memang berasal dari *S*. Tapi siapa *S*? Kami tidak memiliki cara yang dapat diandalkan untuk mengasosiasikan manusia tertentu dengan kunci publik itu. Bahkan jika seseorang mengatakan "kunci publik ini milik *S*," atas dasar apa kita percaya pernyataan itu? Ingat serangan *man-in-the-middle* di awal bab ini ketika Nana

dan Anto ingin membuat kunci rahasia bersama? Selanjutnya kita mengeksplorasi cara membuat pengikatan yang dapat dipercaya antara kunci publik dan identitas.

3.5.12 Kepercayaan

Masalah utama perdagangan digital adalah kepercayaan: Bagaimana Anda tahu bahwa halaman web Microsoft benar-benar milik Microsoft, misalnya? Bagian ini lebih sedikit tentang teknologi dan lebih banyak tentang aspek kepercayaan manusia, karena kepercayaan itu mendasari seluruh konsep tanda tangan digital.

Dalam kehidupan nyata, Anda mungkin memercayai teman dekat dengan cara yang tidak Anda percayai kenalan baru. Seiring waktu kepercayaan Anda pada seseorang dapat tumbuh dengan pengalaman Anda, tetapi dapat menurun jika orang itu mengkhianati Anda. Anda mencoba seseorang, dan, tergantung pada hasilnya, Anda menambah atau mengurangi tingkat kepercayaan Anda. Pengalaman-pengalaman ini membangun kerangka kepercayaan pribadi.

Halaman web dapat diganti dan dipalsukan tanpa peringatan. Sampai batas tertentu, Anda menganggap halaman itu asli jika tidak ada yang tampak tidak biasa, jika konten di situs tampak kredibel atau setidaknya masuk akal, dan jika Anda tidak menggunakan situs untuk keputusan penting. Jika situs tersebut milik bank Anda, Anda dapat memverifikasi bahwa URL tersebut terlihat asli. Beberapa situs, terutama lembaga keuangan, sudah mulai membiarkan setiap pelanggan memilih gambar keamanan, misalnya, mobil sport merah panas atau tengara ikonik; pengguna diperingatkan untuk memasukkan informasi sensitif hanya jika mereka melihat gambar pribadi yang mereka pilih sebelumnya.

Dalam lingkungan komersial, jenis institusi tertentu berkonotasi kepercayaan. Anda dapat mempercayai (pejabat di) organisasi pendidikan, keagamaan, atau sosial tertentu. Perusahaan besar dan mapan seperti bank, perusahaan asuransi, rumah sakit, dan produsen besar telah mengembangkan ukuran kepercayaan. Usia sebuah institusi juga menginspirasi kepercayaan. Memang, kepercayaan adalah dasar dari gagasan branding, di mana Anda mempercayai kualitas sesuatu karena Anda tahu mereknya. Seperti yang akan Anda lihat segera, kepercayaan pada entitas yang diakui seperti itu merupakan komponen penting dalam tanda tangan digital.

Membangun Kepercayaan Antar Orang

Sebagai manusia, kita membangun kepercayaan sepanjang waktu dalam interaksi kita sehari-hari dengan orang lain. Kami mengidentifikasi orang yang kami kenal dengan mengenali suara, wajah, atau tulisan tangan mereka. Di lain waktu, kami menggunakan afiliasi untuk menyampaikan kepercayaan. Misalnya, jika orang asing menelepon kami dan kami mendengar, "Saya mewakili pemerintah daerah ..." atau "Saya menelepon atas nama badan amal ini. …" atau “Saya menelepon dari sekolah/rumah sakit/polisi tentang ibu/ayah/anak laki-laki/anak perempuan/kakak/adik Anda …,” kita mungkin memutuskan untuk mempercayai si penelepon meskipun kita tidak mengenalnya. Bergantung pada sifat panggilan, kami dapat memutuskan untuk mempercayai afiliasi penelepon atau mencari verifikasi independen. Misalnya, kita dapat memperoleh nomor afiliasi dari direktori telepon dan hubungi kembali pihak tersebut. Atau kita dapat mencari informasi tambahan dari penelepon, seperti “Jaket warna apa yang dia kenakan?” atau “Siapa presiden organisasi Anda?” Jika kami

memiliki tingkat kepercayaan yang rendah, kami bahkan dapat bertindak untuk mengecualikan orang luar, seperti dalam "Saya akan mengirimkan cek langsung ke badan amal Anda daripada memberi Anda nomor kartu kredit saya."

Untuk setiap interaksi ini, kita memiliki apa yang kita sebut sebagai “ambang kepercayaan”, suatu tingkat di mana kita bersedia mempercayai individu yang tidak dikenal. Ambang batas ini juga ada dalam interaksi komersial. Ketika Perusahaan Manufaktur Acorn mengirimkan pesanan kepada Perusahaan Baja Besar untuk 10.000 lembar baja, untuk dikirim dalam waktu seminggu dan dibayar dalam waktu sepuluh hari, kepercayaan berlimpah. Pesanan dicetak pada formulir Acorn, ditandatangani oleh seseorang yang diidentifikasi sebagai Helene Smudge, Agen Pembelian. Big Steel mungkin mulai mempersiapkan baja bahkan sebelum menerima uang dari Acorn. Big Steel dapat memeriksa peringkat kredit Acorn untuk memutuskan apakah akan mengirimkan pesanan tanpa pembayaran terlebih dahulu. Jika curiga, Big Steel mungkin menelepon Acorn dan meminta untuk berbicara dengan Ms. Smudge di departemen pembelian. Tetapi kemungkinan besar Big Steel akan benar-benar mengirimkan barang tanpa mengetahui siapa Nona Smudge, apakah dia sebenarnya agen pembelian, apakah dia berwenang untuk berkomitmen pada pesanan sebesar itu, atau bahkan apakah tanda tangan itu benar-benar miliknya. Terkadang transaksi seperti ini terjadi melalui faks, sehingga Big Steel bahkan tidak memiliki tanda tangan asli di arsipnya. Dalam Studi Kasus seperti ini, yang terjadi setiap hari, kepercayaan didasarkan pada penampilan keaslian (seperti formulir yang dicetak dan ditandatangani), informasi luar (seperti laporan kredit), dan urgensi (permintaan Acorn agar baja dikirim dengan cepat).

Membangun Kepercayaan Secara Elektronik

Agar komunikasi elektronik berhasil, kita harus mengembangkan cara yang sama bagi dua pihak untuk membangun kepercayaan tanpa harus bertemu. Benang merah dalam interaksi pribadi dan bisnis kita adalah kemampuan untuk memiliki seseorang atau sesuatu yang menjamin keberadaan dan integritas salah satu atau kedua belah pihak. Polisi, Kamar Dagang, atau Better Business Bureau menjamin keaslian penelepon. Acorn secara tidak langsung menjamin fakta bahwa Ms. Smudge adalah agen pembeliannya dengan mentransfer panggilan kepadanya di departemen pembelian saat Big Steel memanggilnya. Dalam arti tertentu, perusahaan telepon menjamin keaslian suatu pesta dengan mendaftarkan seseorang dalam direktori. Konsep "menjamin" oleh pihak ketiga ini dapat menjadi dasar kepercayaan dalam pengaturan komersial di mana dua pihak tidak saling mengenal.

Masalah kepercayaan yang perlu kita atasi untuk tanda tangan digital adalah keaslian kunci publik. Jika Monique menandatangani dokumen dengan kunci pribadinya, orang lain dapat mendekripsi tanda tangan dengan kunci publiknya untuk memverifikasi bahwa hanya Monique yang dapat menandatanganinya. Satu-satunya masalah adalah dapat memperoleh kunci publik Monique dengan cara yang cukup dapat kita percayai bahwa kunci tersebut benar-benar miliknya, yaitu, bahwa kunci tersebut tidak diedarkan oleh beberapa aktor jahat yang menyamar sebagai Monique. Pada bagian selanjutnya kami menyajikan cara yang dapat dipercaya untuk mengikat kunci publik dengan identitas.

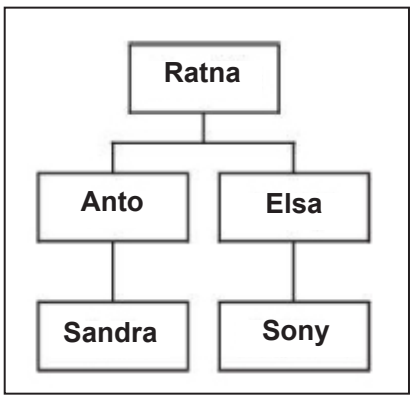
Kepercayaan Berdasarkan Individu Yang Dihormati Secara Umum

Sebuah perusahaan besar mungkin memiliki beberapa divisi, setiap divisi mungkin memiliki beberapa departemen, setiap departemen mungkin memiliki beberapa proyek, dan setiap proyek mungkin memiliki beberapa kelompok tugas (dengan variasi nama, jumlah level, dan tingkat pekerjaan).

kelengkapan hierarki). Eksekutif puncak mungkin tidak tahu dengan nama atau pandangan setiap karyawan di perusahaan, tetapi pemimpin kelompok tugas tahu semua anggota kelompok tugas, pemimpin proyek tahu semua pemimpin kelompok tugas, dan seterusnya. Hirarki ini dapat menjadi dasar kepercayaan di seluruh organisasi.

Untuk melihat caranya, misalkan dua orang bertemu: Sandra dan Sony. Sony mengatakan dia bekerja untuk perusahaan yang sama dengan Sandra. Sandra ingin verifikasi independen yang dia lakukan. Dia mengetahui bahwa Anto dan Elsa adalah dua pemimpin kelompok tugas untuk proyek yang sama (dipimpin oleh Ratna); Sandra bekerja untuk Anto dan Sony untuk Elsa. (Hubungan organisasi ditunjukkan pada Gambar 2-28.) Fakta-fakta ini memberi Sandra dan Sony dasar untuk mempercayai identitas satu sama lain. Rantai verifikasi mungkin seperti ini:

- Sandra bertanya pada Anto siapa Sony.
- Anto bertanya pada Elsa, apakah dia mengenalnya secara langsung, dan jika tidak, dia bertanya pada Ratna.
- (Jika ditanya, Ratna kemudian bertanya kepada Elsa.)
- Elsa menjawab Ratna atau Anto bahwa Sony bekerja untuknya.
- (Ratna memberi tahu Anto, jika dia terlibat.)
- Anto memberitahu Sandra.

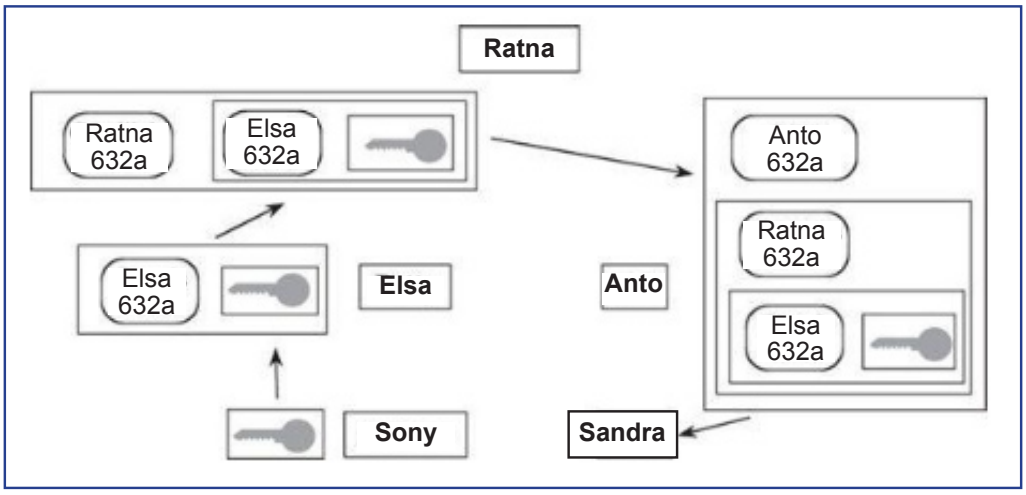


GAMBAR 2-28 Hubungan Kepercayaan

Jika Sony berada dalam kelompok tugas yang berbeda, mungkin perlu untuk naik lebih tinggi di pohon organisasi sebelum titik yang sama ditemukan.

Kita dapat menggunakan proses serupa untuk pertukaran kunci kriptografi, seperti yang ditunjukkan pada Gambar 2-39. Jika Sony dan Sandra ingin berkomunikasi, Sony dapat memberikan kunci publiknya kepada Elsa, yang memberikanya kepada Ratna, lalu Anto, atau langsung kepada Anto, yang memberikanya kepada Sandra Tetapi urutan ini tidak persis seperti yang akan terjadi di kehidupan nyata. Kuncinya mungkin akan disertai dengan catatan yang mengatakan itu dari Sony, mulai dari sedikit kertas kuning hingga bentuk Pernyataan Identitas 947. Dan jika formulir 947

digunakan, maka Elsa juga harus melampirkan formulir 632a Transmittal of Identity, Ratna akan melampirkan 632a lainnya, dan Anto akan melampirkan yang terakhir, seperti yang ditunjukkan pada Gambar 2-39. Rantai formulir 632a ini akan mengatakan, pada dasarnya, "Saya Elsa dan saya menerima kunci ini dan pernyataan identitas terlampir secara pribadi dari seseorang yang saya kenal sebagai Sony," "Saya Ratna dan saya menerima kunci ini dan pernyataan terlampir. identitas dan transmisi identitas pribadi yang dilampirkan dari seseorang yang saya kenal sebagai Elsa," dan seterusnya. Ketika Sandra menerima kuncinya, dia dapat meninjau rantai bukti dan menyimpulkan dengan jaminan yang masuk akal bahwa kunci itu benar-benar berasal dari Sony. Protokol ini adalah cara untuk mendapatkan kunci publik yang diautentikasi, pengikatan sebuah kunci dan identitas yang dapat diandalkan.

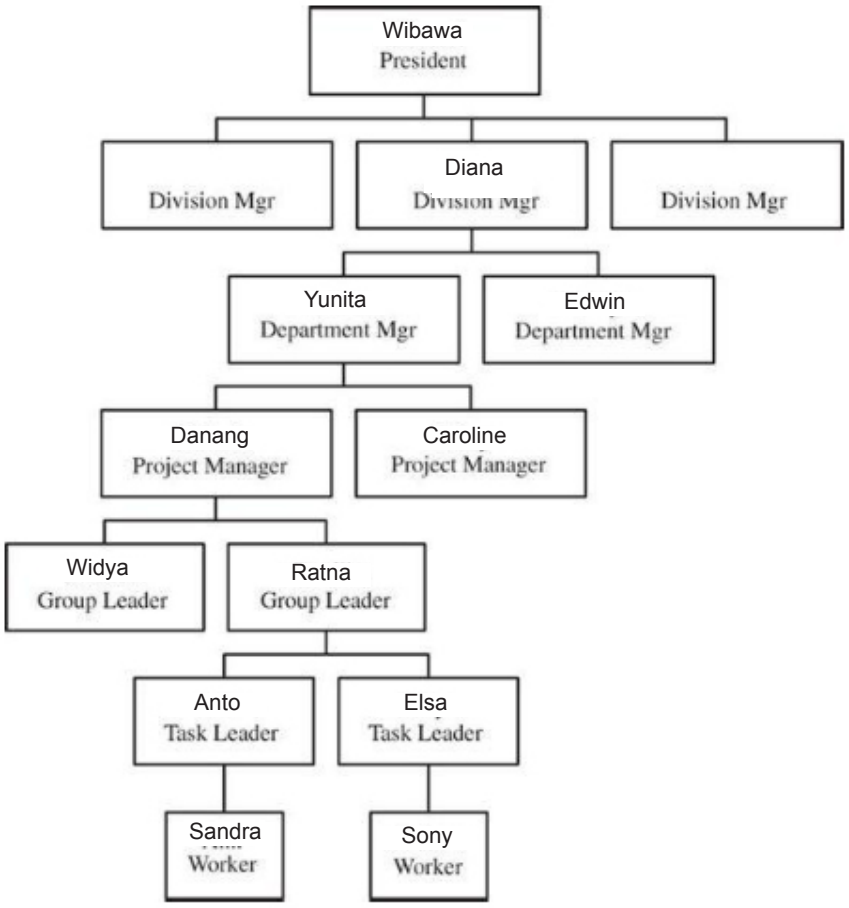


Gambar 2-29 Hubungan Kunci dalam Sertifikat

Model ini bekerja dengan baik dalam sebuah perusahaan karena selalu ada seseorang yang sama untuk setiap dua karyawan, bahkan jika dua karyawan berada di divisi yang berbeda sehingga satu-satunya orang biasa adalah presiden. Namun, prosesnya macet, jika Sandra, Anto, Ratna, Elsa, dan Sony semuanya harus tersedia kapan pun Sandra dan Sony ingin berkomunikasi. Jika Elsa sedang dalam perjalanan bisnis atau Anto sedang sakit, protokolnya terputus-putus. Juga tidak baik jika presiden tidak dapat menyelesaikan pekerjaan yang berarti karena setiap hari disibukkan dengan penanganan formulir 632a.

Untuk mengatasi yang pertama dari masalah ini, Sony dapat meminta rantai formulirnya yang lengkap 632a dari presiden sampai dia. Sony kemudian dapat memberikan salinan set lengkap ini kepada siapa pun di perusahaan yang menginginkan kuncinya. Alih-alih bekerja dari bawah ke titik yang sama, Sony mulai dari atas dan mendokumentasikan rantai lengkapnya. Dia mendapatkan tanda tangan ini setiap kali atasSandraya tersedia, sehingga mereka tidak perlu tersedia ketika dia ingin memberikan kunci publik yang diautentikasi.

Kita dapat menyelesaikan masalah kedua dengan membalikkan prosesnya. Alih-alih mulai dari bawah (dengan anggota tugas) dan bekerja ke atas pohon (presiden), kita mulai dari atas. Dengan demikian, Sony memiliki kunci publik yang telah diautentikasi untuk penggunaan tak terbatas di masa mendatang. Misalkan struktur yang diperluas dari perusahaan hipotetis kami, yang menunjukkan presiden dan tingkat lainnya, seperti yang diilustrasikan pada Gambar 2-30.



Gambar 2-30 Pendelegasian Kepercayaan

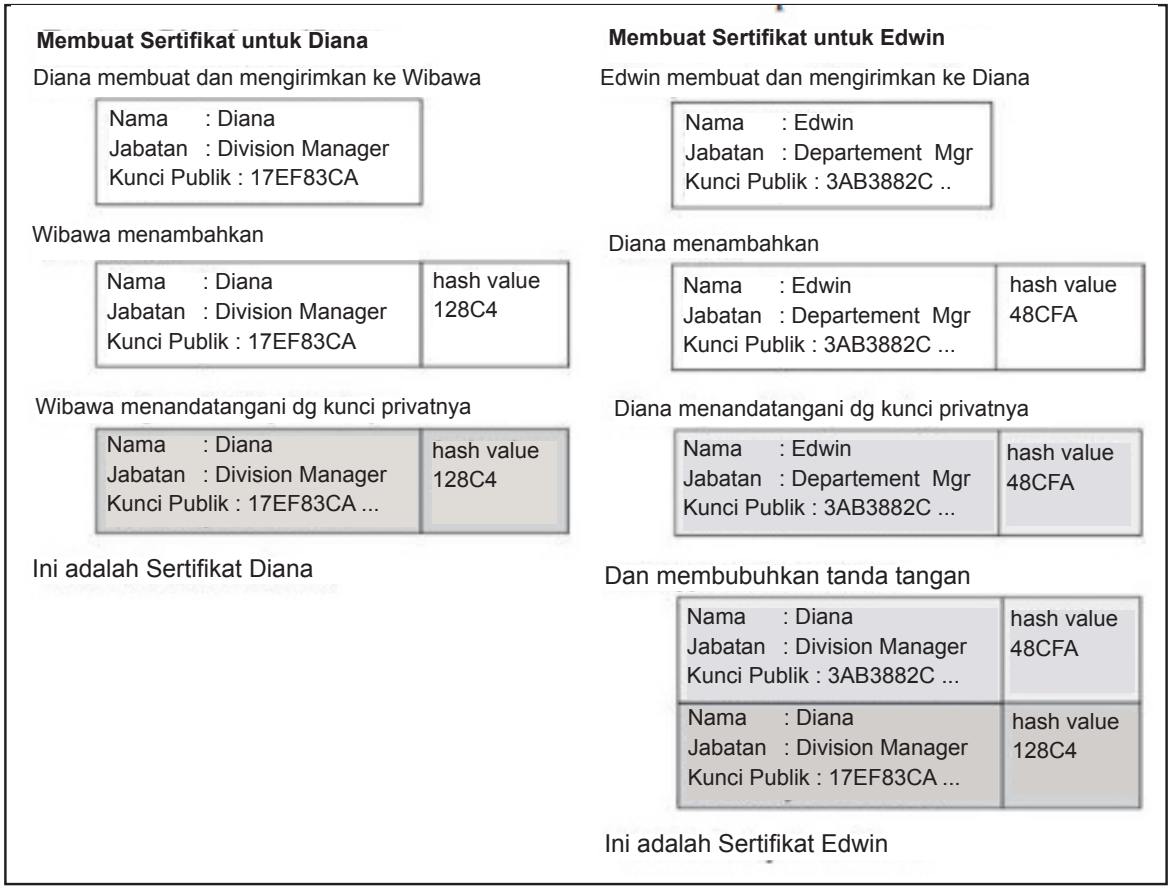
Presiden membuat surat untuk setiap manajer divisi yang mengatakan "Saya Wibawa, presiden, saya membuktikan identitas manajer divisi Diana, yang saya kenal secara pribadi, dan saya percaya Diana untuk membuktikan identitas bawahanya." Setiap manajer divisi melakukan hal yang sama, menyalin surat presiden dengan setiap surat yang dibuat oleh manajer, dan seterusnya. Sony menerima paket surat, dari presiden ke bawah melalui pemimpin kelompok tugasnya, setiap surat dihubungkan dengan nama ke yang berikutnya. Jika setiap karyawan di perusahaan menerima paket seperti itu, dua karyawan mana pun yang ingin bertukar kunci yang diautentikasi hanya perlu membandingkan paket satu sama lain; kedua paket akan memiliki setidaknya Wibawa yang sama, mungkin beberapa manajer lain di bawah Wibawa, dan pada titik tertentu akan menyimpang. Sony dan Sandra, misalnya, dapat membandingkan rantai mereka, menentukan bahwa mereka sama melalui Ratna, dan melacak hanya dari Ratna ke bawah. Sony tahu rantai dari Wibawa ke Ratna asli karena identik dengan rantainya, dan Sandra tahu hal yang sama. Masing-masing tahu sisa rantai itu akurat karena mengikuti garis nama dan tanda tangan yang tidak terputus.

2.5.14 Sertifikat: Identitas Tepercaya dan Kunci Publik

Anda mungkin telah menyimpulkan bahwa proses ini berhasil, tetapi terlalu rumit untuk diterapkan dalam kehidupan nyata; mungkin Anda telah menduga bahwa kami sedang membangun sebuah sistem untuk komputer. Protokol ini diwakili lebih mudah secara elektronik daripada di atas kertas.

Dengan kertas, orang harus menjaga terhadap pemalsuan, untuk mencegah bagian dari satu rantai diganti dan untuk memastikan bahwa kunci publik di bagian bawah terikat pada rantai. Semuanya dapat dilakukan secara elektronik dengan tanda tangan digital dan fungsi hash. .

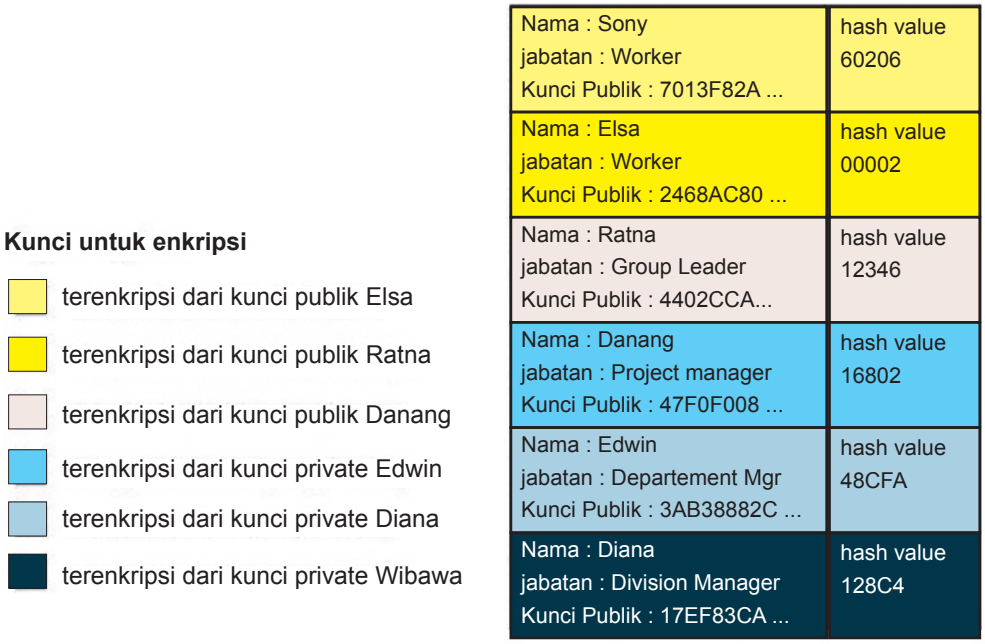
Kunci publik dan identitas pengguna diikat bersama dalam sertifikat, yang kemudian ditandatangani oleh seseorang yang disebut otoritas sertifikat, yang menyatakan keakuratan pengikatan. Dalam contoh kami, perusahaan mungkin menyiapkan skema sertifikat dengan cara berikut. Pertama, Wibawa memilih pasangan kunci publik, memposting bagian publik di mana semua orang di perusahaan dapat mengambilnya, dan mempertahankan bagian pribadi. Kemudian, setiap manajer divisi, seperti Diana, membuat pasangan kunci publiknya, menempatkan kunci publik dalam sebuah pesan bersama dengan identitasnya, dan meneruskan pesan tersebut dengan aman ke Wibawa. Wibawa menandatanganinya dengan membuat nilai hash dari pesan dan kemudian mengenkripsi hash dengan kunci pribadinya. Dengan menandatangani pesan, Wibawa menegaskan bahwa kunci publik (Diana) dan identitas (juga Diana) dalam pesan adalah untuk orang yang sama. Pesan ini disebut sertifikat Diana.



Gambar 2-31 Membuat Sertifikat

Semua manajer departemen Diana membuat pesan dengan kunci publik mereka, masing-masing memiliki hash dan menandatangani, dan mengembalikSandraya. Dia juga menambahkan ke setiap salinan sertifikat yang dia terima dari Wibawa. Dengan cara ini, siapa pun dapat memverifikasi sertifikat manajer dengan memulai dengan kunci publik Wibawa yang terkenal, mendekripsi sertifikat Diana untuk mengambil kunci publik (dan identitasnya), dan menggunakan kunci publik Diana untuk mendekripsi sertifikat manajer. Gambar 2-31 menunjukkan bagaimana sertifikat dibuat untuk Diana

dan salah satu manajernya, Delwyn. Proses ini berlanjut ke bawah hierarki ke Sandra dan Sony. Seperti yang ditunjukkan pada Gambar 2-32, sertifikat Sony benar-benar sertifikat individualnya yang digabungkan dengan semua sertifikat untuk mereka yang berada di atasnya dalam garis presiden.



Gambar 2-32 Hirarki Sertifikat

Penandatanganan Sertifikat Tanpa Hirarki Tunggal

Dalam contoh kami, sertifikat dikeluarkan berdasarkan struktur manajerial. Tetapi kami tidak memerlukan struktur seperti itu dan kami juga tidak harus mengikuti proses yang berbelit-belit untuk menggunakan penandatanganan sertifikat untuk otentikasi. Siapa pun yang dianggap dapat diterima sebagai otoritas dapat menandatangani sertifikat. Misalnya, jika Anda ingin menentukan apakah seseorang menerima gelar dari universitas, Anda tidak akan menghubungi presiden atau rektor melainkan pergi ke kantor catatan atau registrar. Untuk memverifikasi pekerjaan seseorang, Anda dapat bertanya kepada kantor personalia atau direktur sumber daya manusia. Dan untuk memeriksa apakah seseorang tinggal di alamat tertentu, Anda dapat berkonsultasi dengan kantor catatan publik.

Kadang-kadang, orang tertentu ditunjuk untuk membuktikan keaslian atau validitas dokumen atau orang. Misalnya, notaris membuktikan keabsahan tanda tangan (tertulis) pada suatu dokumen. Beberapa perusahaan memiliki petugas keamanan untuk memverifikasi bahwa karyawan memiliki izin keamanan yang sesuai untuk membaca dokumen atau menghadiri rapat. Banyak perusahaan memiliki kantor personalia terpisah untuk setiap lokasi atau setiap lokasi pabrik; petugas personalia menjamin status pekerjaan karyawan di lokasi tersebut. Pejabat atau kepala kantor mana pun dapat menandatangani sertifikat untuk orang-orang yang berada di bawah lingkup mereka dengan kredibel. Hirarki alami ada di masyarakat, dan hierarki yang sama ini dapat digunakan untuk memvalidasi sertifikat.

Satu-satunya masalah dengan hierarki adalah kebutuhan akan kepercayaan dari tingkat atas. Seluruh rantai keaslian aman karena setiap sertifikat berisi kunci yang mendekripsi sertifikat berikutnya, kecuali yang atas. Dalam sebuah perusahaan, karyawan secara alami mempercayai orang di atas. Tetapi jika sertifikat ingin digunakan secara luas dalam perdagangan elektronik, orang harus dapat bertukar sertifikat dengan aman di seluruh perusahaan, organisasi, dan negara.

Internet adalah federasi besar jaringan untuk komunikasi antarpribadi, antarperusahaan, antarorganisasi, dan internasional (serta intraperusahaan, intraorganisasi, dan intranasional). Itu bukan bagian dari pemerintah mana pun, juga bukan milik perusahaan swasta. Hal ini diatur oleh sebuah dewan yang disebut Internet Society. Masyarakat Internet memiliki kekuatan hanya karena para anggotanya, pemerintah dan perusahaan yang membentuk Internet, setuju untuk bekerja sama. Tetapi sebenarnya tidak ada "atas" untuk Internet. Perusahaan yang berbeda, seperti C&W HKT, SecureNet, VeriSign, Baltimore Technologies, Deutsche Telecom, Societá Interbancaria per l'Automatzione di Milano, Entrust, dan Certipostes adalah otoritas sertifikasi root, yang berarti masing-masing adalah otoritas tertinggi yang menandatangani sertifikat. Jadi, alih-alih satu akar dan satu puncak, ada banyak akar, sebagian besar terstruktur di sekitar batas-batas negara.

Mendistribusikan Kunci dan Sertifikat

Sebelumnya dalam bab ini kami memperkenalkan beberapa pendekatan untuk distribusi utama, mulai dari pertukaran langsung hingga distribusi melalui fasilitas distribusi pusat hingga distribusi lanjutan bersertifikat. Tetapi tidak peduli pendekatan apa yang diambil untuk distribusi kunci, masing-masing memiliki kelebihan dan kekurangan.

Poin yang perlu diingat tentang protokol distribusi kunci apa pun termasuk yang berikut:

- Apa batasan operasional yang ada? Misalnya, apakah protokol memerlukan fasilitas yang tersedia secara terus-menerus, seperti pusat distribusi kunci?
- Persyaratan kepercayaan apa yang ada? Siapa dan entitas apa yang harus dipercaya untuk bertindak dengan benar?
- Apa perlindungan terhadap kegagalan? Bisakah orang luar meniru salah satu entitas dalam protokol dan merusak keamanan? Bisakah pihak mana pun dari protokol curang tanpa deteksi?
- Seberapa efisien protokolnya? Sebuah protokol yang membutuhkan beberapa langkah untuk membuat kunci enkripsi yang akan digunakan berkali-kali adalah satu hal; itu cukup lain untuk melalui beberapa langkah yang memakan waktu untuk penggunaan satu kali.
- Seberapa mudah protokol untuk diimplementasikan? Perhatikan bahwa kompleksitas dalam implementasi komputer mungkin berbeda dari penggunaan manual.

Tanda Tangan Digital—Semua Bagian

Menempatkan potongan-potongan ini bersama-sama kita sekarang dapat menguraikan skema tanda tangan digital lengkap. Asumsikan pengguna S ingin menerapkan tanda tangan digital ke file (atau objek data lainnya), memenuhi empat tujuan tanda tangan digital: tidak dapat diubah, otentik, tidak dapat diubah, dan tidak dapat digunakan kembali.

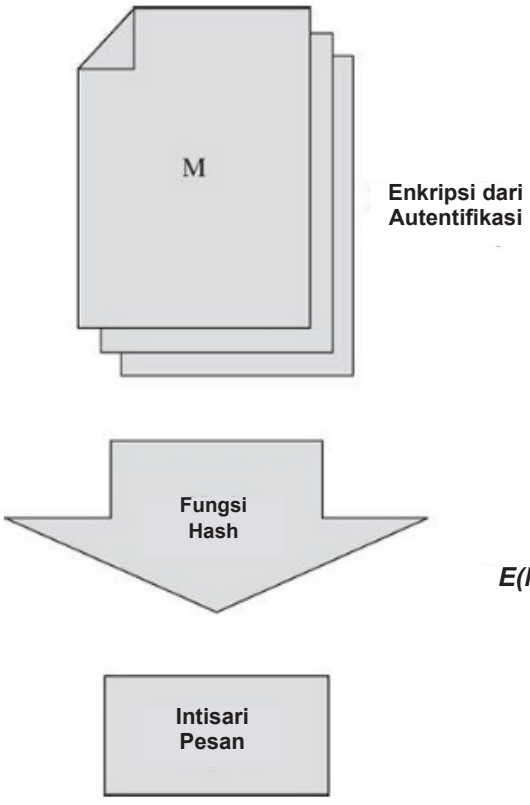
Tanda tangan digital terdiri dari:

- file
- demonstrasi bahwa file tersebut belum diubah
- indikasi siapa yang membubuhkan tanda tangan
- validasi bahwa tanda tangan itu asli, yaitu milik penandatangan
- koneksi tanda tangan ke file

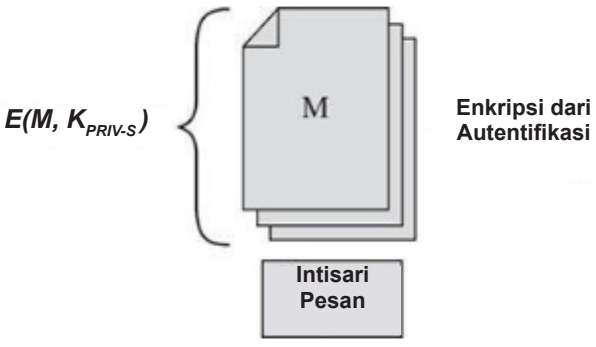
Dengan lima komponen ini kita dapat membuat tanda tangan digital.

Kita mulai dengan file. Jika kita menggunakan kode hash file yang aman untuk menghitung intisari pesan dan menyertakan kode hash tersebut dalam tanda tangan, kode tersebut menunjukkan bahwa file tersebut tidak diubah lagi. Penerima file yang ditandatangani dapat menghitung ulang fungsi hash dan, jika nilai hash cocok, menyimpulkan dengan kepercayaan yang masuk akal bahwa file yang diterima sama dengan yang ditandatangani. Sejauh ini, tanda tangan digital kita terlihat seperti objek pada Gambar 2-33.

Selanjutnya, kami menerapkan kunci enkripsi pribadi penandatangan untuk mengenkripsi intisari pesan. Karena hanya penandatangan yang mengetahui kunci itu, penanda tangan adalah satu-satunya yang dapat menerapkannya. Sekarang objek yang ditandatangani terlihat seperti Gambar 2-34.



Gambar 2-33 Kode Hash untuk Mendeteksi Perubahan



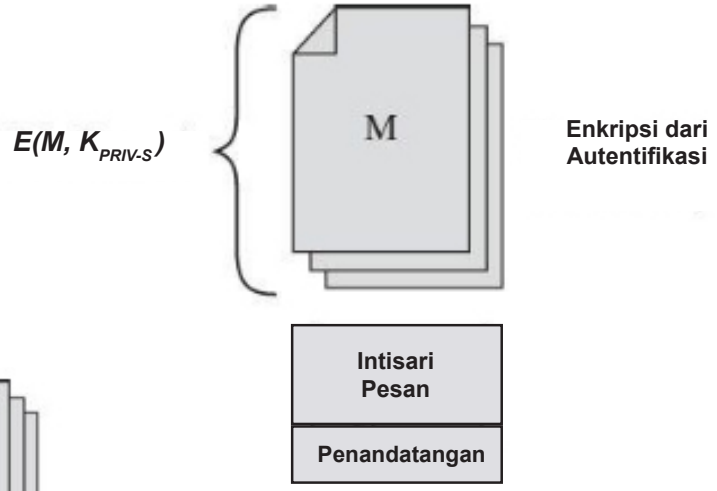
Gambar 2-34 Enkripsi untuk Menunjukkan Keaslian

Satu-satunya bagian lain yang ditambahkan adalah indikasi siapa penandatangannya, sehingga penerima mengetahui kunci publik mana yang digunakan untuk membuka kunci enkripsi, seperti

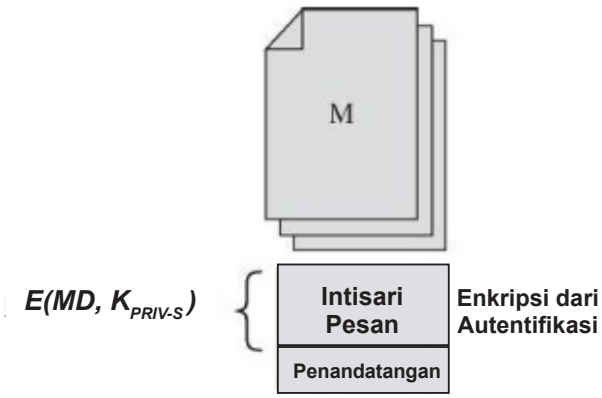
yang ditunjukkan pada Gambar 2-35. Identitas penanda tangan harus berada di luar enkripsi karena jika berada di dalam, identitas tidak dapat diekstraksi.

Dua perkembangan ekstra masih harus ditambahkan. Pertama, tergantung pada ukuran file, objek ini bisa besar, dan enkripsi asimetris lambat, tidak cocok untuk mengenkripsi hal-hal besar. Namun, enkripsi otentikasi S hanya perlu mencakup kode hash yang aman, bukan seluruh file itu sendiri.

Jika file dimodifikasi, itu tidak akan lagi cocok dengan kode hash, sehingga penerima akan tahu untuk tidak mempercayai objek sebagai otentik dari S. Dan jika kode hash dipecah dan dilampirkan ke file yang berbeda, itu tidak akan cocok di sana , antara. Jadi untuk efisiensi kita hanya perlu mengenkripsi nilai hash dengan kunci privat S, seperti yang ditunjukkan pada Gambar 2-36.



Gambar 2-35 Indikasi Penandatangan



Gambar 2-36 Enkripsi Asimetris yang Meliputi Nilai Hash

Kedua, file, bagian data dari objek, diekspos untuk dibaca siapa saja. Jika S menginginkan kerahasiaan, yaitu agar hanya satu penerima yang dapat melihat isi file, S dapat memilih kunci enkripsi simetris, mengenkripsi file, dan menyimpan kunci di bawah kunci enkripsi publik asimetris pengguna U. Penambahan terakhir ini ditunjukkan pada Gambar 2-37.

Kesimpulannya, tanda tangan digital dapat menunjukkan keaslian suatu file, terutama potongan kode. Ketika Anda mencoba menginstal sepotong kode yang ditandatangani, sistem operasi akan memeriksa sertifikat dan file dan memberi tahu Anda jika sertifikat dan hash tidak dapat diterima. Tanda tangan digital, ditambah dengan fungsi hash yang kuat dan enkripsi simetris, adalah cara yang efektif untuk memastikan bahwa file persis seperti yang disimpan oleh pembuatnya untuk diunduh.

Maksud kami dalam bab ini bukanlah untuk melatih korps kriptografer atau kriptologis baru; untuk melakukan itu akan membutuhkan jauh lebih banyak materi daripada yang dapat dimuat dalam buku ini. Sebaliknya, kami ingin Anda mengetahui dan memahami konsep dasar kriptografi sehingga pada bab selanjutnya Anda dapat memahami kesulitan, kekuatan, dan kelemahan dari, misalnya, mengamankan jaringan nirkabel, sinyal jaringan atau membangun komunikasi yang dilindungi antara pengguna browser dan situs web.

Bahan Evaluasi

1. Jelaskan masing-masing dari empat jenis mekanisme kontrol akses berikut dalam hal :
   (a) kemudahan menentukan akses yang diotorisasi selama eksekusi,
   (b) kemudahan penambahan akses untuk subjek baru,
   (c) kemudahan penghapusan akses oleh subjek, dan
   (d) kemudahan membuat objek baru yang dapat diakses oleh semua subjek secara default.
   - daftar kontrol akses per subjek (yaitu, satu daftar untuk setiap subjek memberi tahu semua objek yang dapat diakses oleh subjek tersebut)
   - daftar kontrol akses per-objek (yaitu, satu daftar untuk setiap objek memberi tahu semua subjek yang memiliki akses ke objek itu)
   - matriks kontrol akses
   - kemampuan
2. Misalkan daftar kontrol akses per subjek digunakan. Menghapus objek dalam sistem seperti itu tidak nyaman karena semua perubahan harus dilakukan pada daftar kontrol semua subjek yang memiliki akses ke objek. Sarankan cara alternatif yang lebih murah untuk menangani penghapusan.
3. Kontrol akses file sebagian besar berkaitan dengan dimensi kerahasiaan keamanan. Apa hubungan antara matriks kontrol akses dan integritas objek yang aksesnya dikendalikan?
4. Salah satu fitur dari sistem proteksi berbasis kemampuan adalah kemampuan satu proses untuk mentransfer salinan kemampuan ke proses lain. Jelaskan situasi di mana satu proses harus dapat mentransfer kemampuan ke yang lain.
5. Sarankan skema yang efisien untuk mempertahankan skema perlindungan per pengguna. Artinya, sistem memelihara satu direktori per pengguna, dan direktori itu mencantumkan semua objek yang diizinkan aksesnya oleh pengguna. Desain Anda harus memenuhi kebutuhan sistem dengan 1000 pengguna, di antaranya tidak lebih dari 20 yang aktif setiap saat. Setiap pengguna memiliki rata-rata 200 objek yang diizinkan; ada 50.000 total objek dalam sistem.
6. Hitung waktu serangan tebak kata sandi:
   (a) Jika kata sandi terdiri dari tiga karakter alfabet huruf besar, berapa lama waktu yang dibutuhkan untuk menentukan kata sandi tertentu, dengan asumsi bahwa pengujian satu kata sandi memerlukan 5 detik? Berapa banyak waktu jika pengujian membutuhkan 0,001 detik?
   (b) Berdebat untuk jangka waktu tertentu sebagai titik awal untuk "aman". Artinya, misalkan penyerang berencana menggunakan serangan brute force untuk menentukan kata sandi. Untuk berapa nilai x (jumlah total waktu untuk mencoba kata sandi sebanyak yang diperlukan) penyerang akan menganggap serangan ini sangat lama?

(c) Jika batas waktu antara "tidak aman" dan "aman" adalah x jumlah waktu, berapa lama kata sandi yang aman harus digunakan? Nyatakan dan jelaskan asumsi Anda mengenai kumpulan karakter dari mana kata sandi dipilih dan jumlah waktu yang diperlukan untuk menguji satu kata sandi.

7. Rancang protokol di mana dua pihak yang saling curiga dapat saling mengotentikasi. Protokol Anda harus dapat digunakan saat pertama kali pihak-pihak ini mencoba mengotentikasi satu sama lain.
8. Sebutkan tiga alasan orang mungkin enggan menggunakan biometrik untuk otentikasi. Dapatkah Anda memikirkan cara untuk melawan keberatan tersebut?
9. Tingkat positif palsu dan negatif palsu dapat disesuaikan, dan sering kali saling melengkapi: Menurunkan yang satu menaikkan yang lain. Sebutkan dua situasi di mana negatif palsu secara signifikan lebih serius daripada positif palsu.
10. Di kantor biasa, otentikasi biometrik dapat digunakan untuk mengontrol akses ke karyawan dan pengunjung terdaftar saja. Kami tahu sistem akan memiliki beberapa negatif palsu, beberapa karyawan secara salah menolak akses, jadi kami membutuhkan penggantian manusia, seseorang yang dapat memeriksa karyawan dan mengizinkan akses meskipun otentikasi gagal. Dengan demikian, kita membutuhkan penjaga manusia di pintu untuk menangani masalah, serta perangkat otentikasi; tanpa biometrik kita hanya akan memiliki penjaga. Akibatnya, kami memiliki jumlah personel yang sama dengan atau tanpa biometrik, ditambah kami memiliki biaya tambahan untuk memperoleh dan memelihara sistem biometrik. Jelaskan keuntungan keamanan dalam situasi ini yang membenarkan biaya tambahan.
11. Garis besar desain skema otentikasi yang "belajar." Skema otentikasi akan dimulai dengan informasi primitif tertentu tentang pengguna, seperti nama dan kata sandi. Saat penggunaan sistem komputasi berlanjut, sistem otentikasi akan mengumpulkan informasi seperti bahasa pemrograman yang umum digunakan; tanggal, waktu, dan lama sesi komputasi; dan penggunaan sumber daya yang khas. Tantangan otentikasi akan menjadi lebih individual karena sistem mempelajari lebih banyak informasi tentang pengguna.
    - Desain Anda harus menyertakan daftar banyak informasi tentang pengguna yang dapat dikumpulkan oleh sistem. Sistem diperbolehkan untuk meminta pengguna yang diautentikasi untuk informasi tambahan tertentu, seperti buku favorit, untuk digunakan dalam tantangan berikutnya.
    - Desain Anda juga harus mempertimbangkan masalah penyajian dan validasi tantangan ini: Apakah calon pengguna menjawab pertanyaan benar-salah atau pilihan ganda? Apakah sistem menginterpretasikan prosa bahasa alami?
12. Bagaimana kata sandi disimpan di komputer pribadi Anda?
13. Jelaskan situasi di mana kata sandi yang lemah tetapi mudah digunakan mungkin memadai.
14. Sebutkan tiga pertanyaan otentikasi (tetapi bukan jawaban) yang dapat diminta oleh perusahaan kartu kredit Anda untuk mengotentikasi Anda melalui telepon. Pertanyaan Anda harus pertanyaan yang penipu tidak bisa dengan mudah mendapatkan jawabannya. Seberapa sulit bagi Anda untuk memberikan jawaban yang benar (misalnya, Anda harus mencari sesuatu atau Anda harus melakukan perhitungan aritmatika cepat)?
15. Jika Anda lupa kata sandi untuk sebuah situs web dan Anda mengklik [Lupa kata sandi saya], terkadang perusahaan mengirimi Anda kata sandi baru melalui email tetapi terkadang mengirim Anda kata sandi lama Anda melalui email. Bandingkan dua Studi Kasus ini dalam hal kerentanan pemilik situs web.

16. Mengalahkan otentikasi mengikuti paradigma metode-kesempatan-motif yang dijelaskan dalam Bab 1. Diskusikan bagaimana ketiga faktor ini berlaku untuk serangan otentikasi.
17. Sarankan sumber dari beberapa nomor tak terduga yang sangat panjang. Sumber Anda harus sesuatu yang dapat diakses oleh pengirim dan penerima dengan mudah tetapi tidak jelas bagi orang luar dan tidak ditransmisikan secara langsung dari pengirim ke penerima.
18. Apa risikonya jika pemerintah Amerika Serikat memilih sistem kripto untuk penggunaan komersial secara luas (baik di dalam maupun di luar Amerika Serikat). Bagaimana pengguna dari luar Amerika Serikat dapat mengatasi beberapa atau semua risiko ini?
19. Jika masa manfaat DES adalah sekitar 20 tahun (1977–1999), berapa lama perkiraan masa manfaat AES? Jelaskan jawaban Anda.
20. Manusia dikatakan sebagai mata rantai terlemah dalam sistem keamanan apa pun. Berikan contoh untuk masing-masing hal berikut:
    (a) situasi di mana kegagalan manusia dapat menyebabkan kompromi data terenkripsi
    (b) situasi di mana kegagalan manusia dapat menyebabkan kompromi identifikasi dan otentikasi
    (c) situasi di mana kegagalan manusia dapat menyebabkan kompromi kontrol akses access
21. Mengapa ahli kriptologi merekomendasikan untuk mengubah kunci enkripsi dari waktu ke waktu? Apakah ini alasan yang sama para pakar keamanan merekomendasikan untuk mengubah kata sandi dari waktu ke waktu? Bagaimana seseorang dapat menentukan seberapa sering untuk mengubah kunci atau kata sandi?
22. Jelaskan mengapa tabrakan hash terjadi. Artinya, mengapa harus selalu ada dua plainteks berbeda yang memiliki nilai hash yang sama?
23. Apa properti dari fungsi hash yang berarti bahwa tabrakan bukan masalah keamanan? Artinya, mengapa penyerang tidak dapat memanfaatkan tabrakan dan mengubah plaintext yang mendasarinya ke bentuk lain yang nilainya bertabrakan dengan nilai hash dari plaintext asli?
24. Apakah PKI melakukan enkripsi? Jelaskan jawabanmu.
25 Apakah PKI menggunakan enkripsi simetris atau asimetris? Jelaskan jawabanmu.
26. Haruskah PKI didukung di firewall (artinya sertifikat akan disimpan di firewall dan firewall akan mendistribusikan sertifikat sesuai permintaan)? Jelaskan jawabanmu.
27. Mengapa PKI membutuhkan sarana untuk membatalkan atau membatalkan sertifikat? Mengapa PKI tidak cukup berhenti mengedarkan sertifikat setelah menjadi tidak sah?
28. Beberapa orang berpikir bahwa otoritas sertifikat untuk PKI harus pemerintah, tetapi yang lain berpikir otoritas sertifikat harus badan swasta, seperti bank, perusahaan, atau sekolah. Apa kelebihan dan kekurangan masing-masing pendekatan?
29. Jika Anda tinggal di negara A dan menerima sertifikat yang ditandatangani oleh pemerintah otoritas sertifikat di negara B, kondisi apa yang menyebabkan Anda mempercayai tanda tangan itu sebagai otentik?
30. Sertifikat berisi identitas, kunci publik, dan tanda tangan yang membuktikan bahwa kunci publik milik identitas tersebut. Bidang lain yang mungkin ada termasuk organisasi (misalnya, universitas, perusahaan, atau pemerintah) yang identitas milik dan mungkin suborganisasi (perguruan tinggi, departemen, program, cabang, kantor). Apa tujuan keamanan yang dilayani oleh bidang lain ini, jika ada? Jelaskan jawabanmu.

Bab 3

Jaringan Komputer

Bab ini akan membahas tentang:

Kerentanan :

- Ancaman dalam jaringan: penyadapan, modifikasi, addressing
- Jaringan nirkabel: intersepsi, asosiasi, WEP, WPA
- Penolakan layanan dan penolakan layanan terdistribusi

Perlindungan :

- Kriptografi untuk jaringan: SSL, IPsec, jaringan pribadi virtual
- Firewall
- Deteksi penyusupan dan sistem perlindungan
- Mengelola keamanan jaringan, informasi keamanan, dan manajemen acara

Dalam kehidupan sehari-hari di era digital seperti saat ini kebutuhan manusia akan komputerisasi memang tidak dapat dielakkan. Komputerisasi menjadi kebutuhan utama dalam mendukung berbagai aktivitas, baik aktivitas individu manusia maupun aktivitas industri dan bisnis.

Seperti yang kita semua tahu, banyak komputasi saat ini melibatkan interaksi jarak jauh dengan orang, komputer, proses, dan bahkan pelaku kejahatan. Anda dapat bersembunyi di kamar Anda dan tidak pernah memberikan apa pun atau menerima apa pun dari luar, tetapi itu akan sangat membatasi apa yang dapat Anda lakukan. Jadi, beberapa tingkat konektivitas eksternal hampir tak terelakkan bagi sebagian besar pengguna komputer, dan pertanyaannya adalah bagaimana melakukannya dengan keamanan yang wajar.

Jaringan komputer adalah sebuah sistem yang terdiri dari dua atau lebih komputer yang saling terhubung satu sama lain melalui media transmisi atau media komunikasi sehingga dapat saling berbagi data, aplikasi maupun berbagi perangkat keras komputer. Istilah tersebut juga dapat diartikan sebagai kumpulan sejumlah terminal komunikasi yang terdiri dari dua komputer atau lebih yang saling terhubung. Tujuan dibangunnya jaringan antar komputer adalah agar informasi/ data yang dibawa pengirim (transmitter) dapat sampai kepada penerima (receiver) dengan tepat dan akurat.

Tetapi segera setelah Anda memutuskan untuk terhubung ke titik di luar batas keamanan Anda, koneksi itu meninggalkan zona perlindungan Anda, dan Anda berisiko orang lain akan membaca, mengubah, dan bahkan menghapus komunikasi Anda. Dalam bab ini kami mempertimbangkan keamanan di jaringan jarak jauh.

Bab ini mencakup dua sisi keamanan jaringan: ancaman dan penanggulangan. Jadi, kami telah membagi bab ini menjadi dua bagian untuk membantu Anda menemukan dan mencerna topik dan untuk menyoroti perbedaan di antara bidang-bidang ini. Tentu saja, kedua bagian saling memperkuat, dan keduanya diperlukan untuk pemahaman yang benar tentang keamanan dalam jaringan.

Dalam bab ini kita mulai dengan tinjauan singkat tentang istilah dan konsep jaringan. Setelah latar belakang itu, kami membuka bagian pertama bab ini: ancaman. Seperti yang Anda lihat, ancaman terhadap jaringan berasal dari empat jenis ancaman dasar yang kami perkenalkan di Bab 1: intersepsi, modifikasi, fabrikasi atau penyisipan, dan interupsi. Kami memeriksa ancaman ini dalam konteks jaringan nirkabel, sebuah teknologi yang telah menjadi populer melalui hotspot WiFi di kedai kopi, kampus universitas, bandara, dan lingkungan perusahaan; banyak dari titik akses ini gratis. Tetapi ketika Anda terhubung ke titik akses gratis, keamanan apa yang Anda miliki? (Pertanyaan serupa muncul dengan komputasi awan, seperti yang kita bahas di Bab 8 tentang topik itu.) Selanjutnya kita membahas serangan penolakan layanan, di mana penggunaan jaringan pengguna yang sah sangat dibatasi atau bahkan terputus; jenis serangan ini unik untuk jaringan.

Bagian kedua dari bab ini menyajikan tiga cara penting untuk melawan ancaman terhadap jaringan. Kami meninjau kembali kerja keras kami, kriptografi, menunjukkan bagaimana ia dapat melindungi kerahasiaan dan integritas dalam komunikasi jaringan. Kemudian, kami memperkenalkan dua teknologi yang dapat membantu melindungi pengguna dari bahaya jaringan: firewall dan sistem deteksi dan perlindungan intrusi. Kami menyimpulkan bab dengan teknik dan teknologi untuk mengelola keamanan jaringan.

3.1 Konsep Jaringan

Jaringan sedikit lebih rumit daripada instalasi komputasi lokal. Untuk meremehkan, kita dapat menganggap lingkungan lokal sebagai satu set komponen — komputer, printer, perangkat penyimpanan, dan sebagainya — dan kabel. Sebuah kawat adalah titik ke titik, dengan dasarnya tidak ada kebocoran antara titik akhir, meskipun penyadapan memungkinkan siapa pun dengan akses ke kawat untuk mencegat, memodifikasi, atau bahkan memblokir transmisi. Di lingkungan lokal, kabel fisik sering diamankan secara fisik atau mungkin secara visual sehingga penyadapan bukan masalah besar. Dengan komunikasi jarak jauh, konsep kabel yang sama berlaku, tetapi kabel berada di luar kendali dan perlindungan pengguna, sehingga merusak transmisi merupakan ancaman serius. Sifat ancaman itu sebagian bergantung pada media "kabel" ini, yang sebenarnya bisa berupa kawat logam, serat kaca, atau sinyal elektromagnetik seperti komunikasi radio. Dalam sekejap kita melihat berbagai jenis media komunikasi.

Mengembalikan perhatian kita ke lingkungan lokal dengan kabel untuk setiap pasangan perangkat, untuk mengirim data dari satu perangkat ke perangkat lain pengirim cukup menggunakan kabel satu ke tujuan. Dengan jaringan jarak jauh, biasanya pengirim tidak memiliki satu kabel untuk setiap

penerima yang memungkinkan, karena jumlah kabel akan menjadi tidak dapat diatur. Sebagai gantinya, seperti yang mungkin Anda ketahui, pengirim mendahului data dengan apa yang pada dasarnya adalah label surat, sebuah tag yang menunjukkan ke mana (dan seringkali dari mana) untuk mengirimkan data. Di berbagai titik di sepanjang jalur transmisi, perangkat memeriksa label untuk menentukan apakah perangkat tersebut adalah penerima yang dituju dan, jika tidak, bagaimana meneruskan data agar lebih dekat ke tujuan. Pemrosesan label ini disebut perutean. Perutean diimplementasikan oleh komputer dan, seperti yang telah Anda ketahui, program komputer rentan terhadap kegagalan yang tidak disengaja dan berbahaya. Di bagian ini kami juga mempertimbangkan beberapa ancaman yang rentan terhadap perutean.

3.1.1 Media Transmisi Jaringan

Ketika item data meninggalkan lingkungan yang dilindungi, orang lain di sepanjang jalan dapat melihat atau mencegat data; istilah lain yang digunakan adalah eavesdrop, wiretap, atau sniff. Jika Anda meneriakkan sesuatu kepada teman yang agak jauh, Anda sadar bahwa orang-orang di sekitar Anda dapat mendengar apa yang Anda katakan. Hal yang sama berlaku dengan data, yang dapat dicegat baik dari jarak jauh, di seluruh jaringan area luas, dan secara lokal, di jaringan area lokal (LAN). Komunikasi data berjalan baik melalui kabel atau nirkabel, keduanya rentan, dengan berbagai tingkat kemudahan serangan. Sifat intersepsi tergantung pada mediumnya, yang akan kita uraikan selanjutnya. Saat Anda membaca penjelasan ini, pikirkan juga tentang modifikasi dan serangan pemblokiran, yang akan kami jelaskan secara singkat.

Kabel

Pada tingkat paling lokal, semua sinyal dalam Ethernet atau LAN lain tersedia di kabel untuk dicegat oleh siapa saja. Setiap konektor LAN (seperti papan komputer) memiliki alamat unik, yang disebut alamat MAC (untuk Kontrol Akses Media); setiap papan dan drivernya diprogram untuk memberi label semua paket dari host-nya dengan alamat uniknya (sebagai "alamat pengirim") dan untuk mengambil hanya paket-paket yang dialamatkan ke host-nya dari internet.

Packet Sniffing

Sniffing adalah tindak kejahatan penyadapan yang dilakukan menggunakan jaringan internet dengan tujuan utama untuk mengambil data dan informasi sensitive secara illegal. Cara kerja sniffing adalah ketika Anda terhubung ke jaringan yang bersifat public, saat Anda melakukan proses transfer data dari client server dan sebaliknya. Karena data yang mengalir pada client dan server yang bersifat bolak-balik, sniffing ini akan menangkap paket-paket yang dikirimkan dengan cara illegal menggunakan tools pembantu. Sniffing bekerja pada segmen data di layer transport dengan cara menyisipkan program jahat pada computer korban, dimana nanti program jakat tersebut akan melakukan proses sniffing sehingga data-data sensitive yang ada pada koputer korban dapat terbaca oleh sniffer.

Menghapus hanya paket-paket yang ditujukan ke host tertentu sebagian besar merupakan masalah kesopanan; ada sedikit untuk menghentikan program dari memeriksa setiap paket saat berjalan. Perangkat yang disebut packet sniffer mengambil semua paket di LAN-nya. Atau, salah satu kartu antarmuka dapat diprogram ulang untuk memiliki alamat MAC yang dianggap unik dari kartu lain yang ada di LAN sehingga dua kartu yang berbeda akan mengambil paket untuk satu alamat. (Untuk menghindari deteksi, kartu nakal harus meletakkan kembali salinan bersih dari paket yang

telah dicegat.) Untungnya (untuk saat ini), LAN kabel biasanya hanya digunakan di lingkungan yang cukup ramah, sehingga serangan semacam ini terjadi jarang.

Radiasi

Penyerang yang cerdik dapat memanfaatkan properti kawat dan dapat membaca paket tanpa manipulasi fisik. Kawat biasa (dan banyak komponen elektronik lainnya) memancarkan radiasi. Dengan proses yang disebut induktansi, penyusup dapat mengetuk kabel dan membaca sinyal yang dipancarkan tanpa melakukan kontak fisik dengan kabel; pada dasarnya, penyusup menempatkan antena di dekat kabel dan mengambil radiasi elektromagnetik dari sinyal yang melewati kabel. (Baca Studi Kasus 3-1 untuk beberapa contoh intersepsi radiasi tersebut.) Sinyal induktansi kabel hanya menempuh jarak pendek, dan mereka dapat diblokir oleh bahan konduktif lainnya, sehingga penyerang dapat menggagalkan induktansi dengan membungkus kabel di lebih banyak kawat dan mungkin mengirim sinyal lain yang membingungkan melalui kabel yang dibungkus. Peralatan yang dibutuhkan untuk menangkap sinyal tidak mahal dan mudah didapat, sehingga ancaman induktansi menjadi perhatian serius untuk jaringan berbasis kabel. Agar serangan berhasil, penyusup harus cukup dekat dengan kabel; oleh karena itu, bentuk serangan ini terbatas pada situasi dengan akses fisik.

Studi Kasus 3-1 : Radiasi Elektromagnetik

Kebocoran elektromagnetik dari perangkat elektronik adalah fenomena yang diketahui yang telah dipelajari selama beberapa dekade. Pakar militer khawatir tentang kemampuan musuh untuk mencegat informasi sensitif dari sumber seperti impuls listrik yang dihasilkan saat tombol ditekan pada keyboard atau radiasi magnetik dari sirkuit yang menampilkan Gambar di layar video. Untuk mencegat data tersebut memerlukan peralatan elektronik canggih yang mampu mendeteksi perubahan kecil pada sinyal tingkat rendah; akibatnya, teknik ini berlaku terutama untuk situasi bernilai sangat tinggi, seperti situasi militer.

Karena militer adalah target utama yang terkena dampak, banyak penelitian di bidang ini tidak dipublikasikan. Dua peneliti Ukraina, N.N. Gorobets dan A.V. Trivaylo, telah menerbitkan [GOR09] hasil dari beberapa studi publik baru-baru ini di bidang ini.

Mereka mempertimbangkan teknologi saat ini: layar panel datar. Kebijaksanaan konvensional adalah bahwa tampilan tabung sinar katoda (CRT) gaya lama memancarkan sinyal yang dapat dideteksi tetapi tampilan kristal cair panel datar (LCD) yang lebih baru "aman." Sebaliknya, para peneliti melaporkan, karakteristik teknis tertentu dari antarmuka dan tampilan dapat membuat LCD lebih mudah dikompromikan daripada CRT. Para peneliti menyajikan contoh yang menunjukkan intersepsi data uji dari jarak 10 meter (30 kaki), dua kantor jauhnya.

Mereka juga melaporkan eksperimen yang melibatkan keyboard. Dengan menggunakan teknik yang berbeda, Gorobets dan Trivaylo memulihkan sinyal keyboard dari jarak 5 hingga 8 meter (kira-kira 15 hingga 25 kaki).

Jarak ini cukup kecil sehingga komputer di sebagian besar kantor, laboratorium, atau instalasi pemerintah mungkin tidak berisiko besar disadap oleh pihak luar. Pada jarak tersebut, penyerang harus berada tepat di luar gedung (di lokasi yang agak terbuka) atau di seberang aula, kedua lokasi yang mengundang pertanyaan seperti "Apa yang kamu lakukan?" Namun, orang-orang di kedai kopi, ruang tunggu, bahkan kamar hotel dan fasilitas konferensi harus menyadari bahwa privasi sinyal komputer mereka tidak terjamin. Apakah orang yang duduk di meja sebelah menjelajahi Web atau mencegat penekanan tombol Anda? Kita tidak boleh mengabaikan potensi kerentanan penyadapan di kejauhan.

Penyambungan Kabel

Jika penyerang tidak cukup dekat untuk mengambil keuntungan dari induktansi, maka tindakan yang lebih bermusuhan mungkin diperlukan. Cara termudah untuk memotong kabel adalah dengan memotong langsung. Jika kabel terputus, semua layanan di atasnya berhenti. Sebagai bagian dari perbaikan, penyerang dapat menyambungkan kabel sekunder yang kemudian menerima salinan semua sinyal di sepanjang kabel primer. Pencegat bisa sedikit kurang jelas tetapi masih mencapai tujuan yang sama. Misalnya, penyerang mungkin dengan hati-hati mengekspos beberapa konduktor luar, menyambungkannya, kemudian dengan hati-hati mengekspos beberapa konduktor dalam dan menyambungkannya. Kedua operasi ini mengubah resistansi, yang disebut impedansi, dari kabel. Dalam kasus pertama, perbaikan itu sendiri mengubah impedansi, dan perubahan impedansi dapat dijelaskan (atau disembunyikan) sebagai bagian dari perbaikan. Dalam kasus kedua, sedikit rekayasa sosial dapat menjelaskan perubahan tersebut. ("Halo, ini Matt, teknisi Bignetworks. Kami sedang mengganti beberapa peralatan di pihak kami, jadi Anda mungkin melihat perubahan impedansi.")

Beberapa LAN memiliki perangkat tetap yang jarang berubah; dengan LAN lain, orang cukup sering menambah dan menghapus perangkat sehingga perubahan bukanlah peristiwa yang luar biasa. Di kantor, karyawan menyalakan stasiun kerja yang telah dimatikan pada malam hari, karyawan yang berkunjung menghubungkan laptop ke jaringan, dan teknisi menambah dan menghapus peralatan pemantauan untuk memelihara jaringan. Menambahkan satu perangkat lagi mungkin tidak diketahui. Penyerang hanya perlu menemukan titik koneksi jaringan yang tidak digunakan dan menyambungkannya.

Cara lain untuk mencegat dari LAN adalah dengan menemukan lemari kabel atau panel, tempat di mana semua kabel jaringan berkumpul dan dari mana administrator jaringan dapat mengkonfigurasi ulang topologi LAN, misalnya, dengan merutekan satu set kabel melalui sakelar untuk membuat subnet terpisah. Dengan perangkat yang disebut sniffer, seseorang dapat terhubung dan mencegat semua lalu lintas di jaringan; sniffer dapat menangkap dan menyimpan data atau meneruskannya ke jaringan lain.

Sinyal pada jaringan digandakan, artinya lebih dari satu sinyal ditransmisikan pada waktu tertentu. Misalnya, dua sinyal analog (suara) dapat digabungkan, seperti dua nada dalam akor musik, dan dua sinyal digital dapat digabungkan dengan interleaving, seperti kartu remi yang dikocok. LAN membawa paket yang berbeda, tetapi data pada WAN mungkin sangat banyak dimultipleks saat meninggalkan host pengirimnya. Jadi, penyadap pada WAN harus mampu tidak hanya untuk

mencegat komunikasi yang diinginkan tetapi juga untuk mengekstraknya dari yang lain dengan yang multiplexing. Meskipun hal ini dapat dilakukan, upaya yang terlibat berarti bahwa teknik tersebut akan digunakan dengan hemat.

Serat Optik

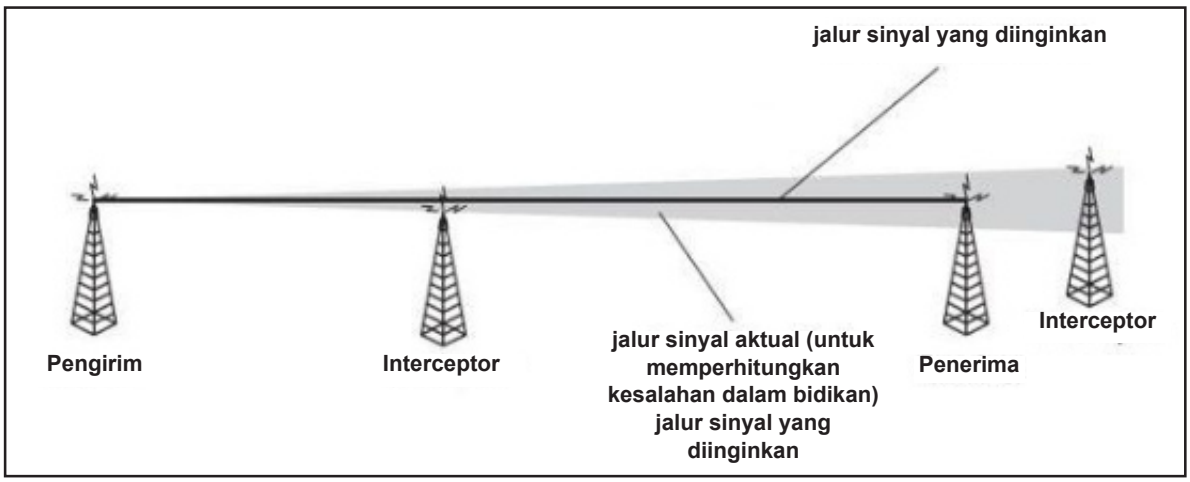
Serat optik menawarkan dua keunggulan keamanan yang signifikan dibandingkan media transmisi lainnya. Pertama, seluruh jaringan optik harus disetel dengan hati-hati setiap kali sambungan baru dibuat. Oleh karena itu, tidak ada yang bisa menyadap sistem optik tanpa deteksi. Memotong hanya satu serat dalam satu bundel akan merusak keseimbangan dalam jaringan. Kedua, serat optik membawa energi cahaya, bukan listrik. Cahaya tidak menciptakan medan magnet seperti halnya listrik. Oleh karena itu, keran induktif tidak mungkin dilakukan pada kabel serat optik.

Hanya menggunakan serat, bagaimanapun, tidak menjamin keamanan, seperti halnya hanya menggunakan enkripsi. Repeater, splices, dan tap di sepanjang kabel adalah tempat di mana data mungkin tersedia dengan lebih mudah daripada di kabel fiber itu sendiri. Sambungan dari peralatan komputasi ke serat juga dapat menjadi titik penetrasi. Dengan sendirinya, serat jauh lebih aman daripada kabel, tetapi juga memiliki kerentanan.

Kabel fisik dengan demikian rentan terhadap berbagai ancaman intersepsi. Tetapi melakukan intrusi semacam itu membutuhkan akses fisik ke salah satu kabel yang membawa komunikasi yang menarik, bukan prestasi kecil. Dalam banyak kasus, menarik data dari udara lebih mudah, seperti yang akan kami jelaskan selanjutnya.

Gelombang Mikro

Sinyal gelombang mikro tidak dibawa sepanjang kabel; mereka disiarkan melalui udara, membuat mereka lebih mudah diakses oleh orang luar. Microwave adalah teknologi line-of-sight; penerima harus berada di jalur yang tidak diblokir dengan sinyal pengirim. Biasanya, sinyal pemancar difokuskan pada penerima yang sesuai karena penerimaan gelombang mikro membutuhkan ruang yang jelas antara pengirim dan penerima. Jalur sinyal cukup lebar, untuk memastikan mengenai penerima, seperti yang ditunjukkan pada Gambar 3-1.



Gambar 3-1 Intersepsi Transmisi Gelombang Mikro

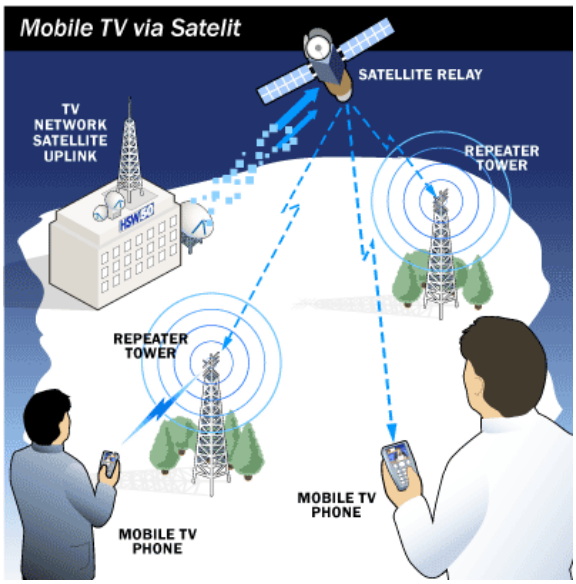
Dari sudut pandang keamanan, petak luas adalah undangan untuk kerusakan. Seseorang tidak hanya dapat mencegat transmisi gelombang mikro dengan mengganggu garis pandang antara pengirim dan penerima, seseorang juga dapat mengambil seluruh transmisi dari antena yang terletak dekat tetapi sedikit di luar titik fokus langsung.

Sinyal gelombang mikro biasanya tidak dilindungi atau diisolasi untuk mencegah intersepsi. Microwave, oleh karena itu, merupakan media yang tidak aman karena sinyalnya sangat terbuka. Namun, karena volume lalu lintas yang besar yang dibawa oleh tautan gelombang mikro, pencegat tidak mungkin memisahkan transmisi individu dari semua transmisi lain yang disisipkan dengannya. Sambungan gelombang mikro milik pribadi, yang hanya membawa komunikasi untuk satu organisasi, tidak terlindungi dengan baik oleh volume.

Sinyal gelombang mikro membutuhkan keselarasan yang benar-benar terlihat, sehingga penggunaannya terbatas di medan berbukit. Plus, karena kelengkungan bumi mengganggu transmisi, sinyal gelombang mikro harus diambil dan diulang untuk menjangkau jarak jauh, yang memperumit komunikasi jarak jauh, misalnya, di atas lautan. Solusi untuk keterbatasan ini, secara mengejutkan, adalah satelit.

Komunikasi Satelit

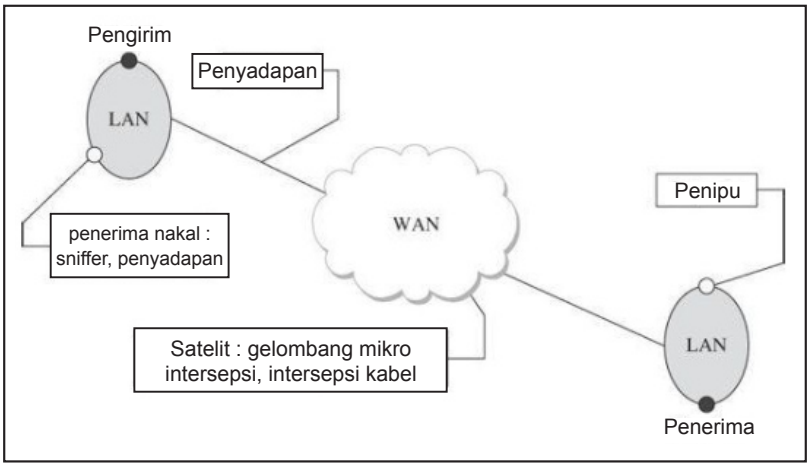
Sinyal dapat dipantulkan dari satelit: dari bumi ke satelit dan kembali ke bumi lagi. Pengirim dan penerima adalah titik tetap; pengirim mengirimkan sinyal ke area yang luas di mana satelit berada, dan satelit menyiarkan ulang sinyal tersebut ke radius tertentu di sekitar penerima. Satelit berada di orbit pada tingkat yang sinkron dengan orbit bumi, sehingga tampak berada pada titik tetap relatif terhadap bumi.



Gambar 3-2 Komunikasi Satelit (http://pacebuk.blogspot.com/)

Transmisi ke satelit dapat mencakup area yang luas di sekitar satelit karena tidak ada tempat lain di dekatnya untuk mengambil sinyal. Namun, saat kembali ke bumi, radius penyebaran yang luas, yang disebut jejak siaran, memungkinkan antena apa pun dalam jangkauan untuk mendapatkan sinyal tanpa deteksi, seperti yang ditunjukkan pada Gambar 3-2. Satelit yang berbeda memiliki karakteristik yang berbeda, tetapi beberapa sinyal dapat dicegat di area yang lebarnya beberapa ratus mil dan

panjangnya seribu mil. Oleh karena itu, potensi intersepsi dengan berada di jalur sinyal bahkan lebih besar dibandingkan dengan sinyal gelombang mikro. Namun, karena komunikasi satelit umumnya sangat multipleks, risikonya kecil bahwa salah satu komunikasi akan dicegat.



Gambar 3-3 Komunikasi Terbuka

Singkatnya, lalu lintas jaringan tersedia untuk pencegat di banyak titik. Gambar 3-3 mengilustrasikan bagaimana komunikasi diekspos dari asalnya ke tujuannya. Kami merangkum kekuatan dan kelemahan media komunikasi yang berbeda dalam Tabel 3-1.

Tabel 3-1 Kekuatan dan Kelemahan Media Komunikasi

Media	Kelebihan	Kelemahan
Wired (Kabel)	• banyak digunakan • biaya pemasangan dan pemeliharaan murah • Bandwith lebih tinggi.	• rentan terhadap emanasi • rentan terhadap penyadapan • Kalau kabel putus maka tidak akan bisa disambung kembali.
Kabel Fiber Optik	• bebas dari gangguan sinyal elektromagnetik, sinyal radio, serta mempunyai ketahanan yang cukup kuat • sulit untuk disadap • mengantarkan data dengan kapasitas besar serta jarak transmisi yang sangat jauh	• berpotensi terpapar pada titik koneksi • harganya yang cukup tinggi • biaya besar pada saat pemasangan dan perawatan
Gelombang mikro	• Sinyal kuat dan tidak terpengaruh cuaca • Perambatannya relatif cepat, (dikarenakan gelombangnya yang pendek) • Efisiensi biaya, biasanya biayanya cukup murah • Mudah untuk diimplementasikan dan diinstalasi	• Mudah terinferensi, atau mudah terganggu oleh gelombang radio lainnya, bahkan oleh dua sistem transmisi gelombang mikro yang berhampiran sekalipun juga bisa terganggu. • Dalam anggota transmisi selang pemancar dan penerima sinyal diharuskan berada dalam garis pandang serta tidak berada penghalang
Wireless (WiFi)	• Tidak membutuhkan instalasi kabel yang panjang • Cocok untuk digunakan pada jaringan yang luas cakupan areanya • Kecepatan transfer data yang cenderung stabil • Kompatibel hampir dengan berbagai jenis perangkat, seperti smartphone dan laptop	• Instalasi dan pengembangan yang biayanya cukup mahal • Kekuatan sinyal sangat tergantung dengan kondisi cuaca dan lingkungan fisik

Satelit	• Cakupan yang luas. Bisa satu Negara, satu wilayah, satu daerah ataupun satu benua, • Bandwith yang tersedia cukup lebar, • Independen dari infrastruktur terrestrial, • Instalasi jaringan segmen bumi yang cepat	• Dalam hal keamanan, yaitu transmisi data sangat mudah ditangkap karena berjalan melalui udara terbuka. • rentan terhadap pengaruh atmosfir • delay propagasi besar

Dari sudut pandang keamanan, Anda harus berasumsi bahwa semua hubungan komunikasi antara node jaringan dapat terputus. Seperti yang ditunjukkan Studi Kasus 3-2, bahkan bola mata pun dapat melewatkan data secara tidak sengaja. Untuk alasan ini, pengguna jaringan komersial menggunakan enkripsi untuk melindungi kerahasiaan komunikasi mereka, seperti yang akan kami tunjukkan nanti dalam bab ini. Komunikasi jaringan lokal dapat dienkripsi, meskipun untuk alasan kinerja mungkin lebih baik untuk melindungi koneksi lokal dengan keamanan fisik dan administratif yang kuat.

Mengganggu atau mencegat dari media komunikasi hanyalah salah satu cara untuk menyerang jaringan. Ancaman integritas dan ketersediaan juga berlaku. Pengalamatan dan perutean juga merupakan titik kerentanan yang bermanfaat. Pada bagian berikutnya kami menyajikan konsep pengalamatan jaringan dasar yang memiliki implikasi keamanan.

3.1.2 Lapisan Protokol

Protokol yaitu sebuah aturan atau standar yang mengatur atau mengijinkan terjadinya hubungan, komunikasi, dan perpindahan data selang dua atau semakin titik komputer. Protokol bisa dilaksanakan pada perangkat keras, perangkat lunak atau kombinasi dari keduanya. Pada tingkatan yang terendah, protokol mendefinisikan koneksi perangkat keras.

Protokol perlu diutamakan pada penggunaan standar teknis, kepada menspesifikasi bagaimana mendirikan komputer atau menghubungkan peralatan perangkat keras. Protokol secara umum dipakai pada komunikasi real-time dimana standar dipakai kepada mengatur struktur dari informasi kepada penyimpanan jangka panjang.

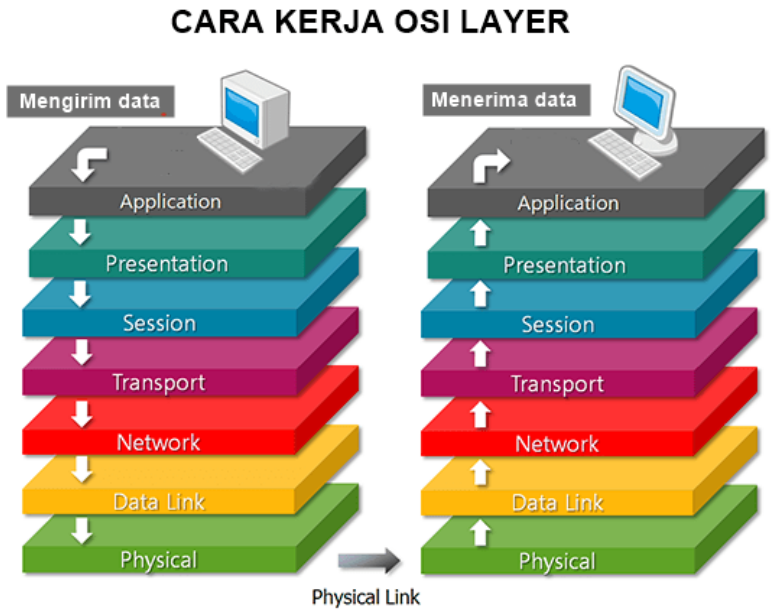
Komunikasi jaringan dilakukan melalui konsep virtual yang disebut model Open System Interconnection (atau OSI). Model tujuh lapis ini dimulai dengan aplikasi yang menyiapkan data untuk dikirim melalui jaringan. Data bergerak ke bawah melalui lapisan, diubah dan dikemas ulang; di lapisan bawah, informasi kontrol ditambahkan di header dan trailer. Akhirnya, data siap untuk melakukan perjalanan pada media fisik, seperti kabel atau melalui udara pada microwave atau link satelit.

Model referensi jaringan terbuka OSI atau OSI Reference Model for open networking adalah sebuah model arsitektural jaringan yang dikembangkan oleh badan International Organization for Standardization (ISO) di Eropa pada tahun 1977. OSI sendiri merupakan singkatan dari Open System Interconnection. Model ini disebut juga dengan model "Model tujuh lapis OSI" (OSI seven layer model).

Sebelum munculnya model referensi OSI, sistem jaringan komputer sangat tergantung kepada pemasok (vendor). OSI berupaya membentuk standar umum jaringan komputer untuk menunjang interoperatibilitas antar pemasok yang berbeda. Dalam suatu jaringan yang besar biasanya terdapat

banyak protokol jaringan yang berbeda. Tidak adanya suatu protokol yang sama, membuat banyak perangkat tidak bisa saling berkomunikasi.

Di sisi penerima, data masuk ke bagian bawah model dan naik melalui lapisan di mana informasi kontrol diperiksa dan dihapus, dan data diformat ulang. Akhirnya, data tiba di aplikasi di lapisan atas model untuk penerima. Komunikasi ini ditunjukkan pada Gambar 3-4.



Gambar 3-4 Model OSI (www.nesabamedia.com)

3.1.3 Addressing dan Routing

Jika data akan pergi dari titik A ke B, harus ada beberapa jalur antara dua titik ini. Salah satu cara, jelas, adalah kabel koneksi langsung. Dan untuk transfer volume tinggi yang sering antara dua titik yang diketahui, tautan khusus memang digunakan. Sebuah perusahaan dengan dua kantor di sisi kota yang berlawanan mungkin mendapatkan koneksi pribadinya sendiri. Namun, koneksi pribadi ini menjadi satu titik kegagalan, karena jika saluran itu gagal karena alasan apa pun, kedua kantor kehilangan konektivitas, dan koneksi yang solid adalah alasan keseluruhan untuk saluran pribadi.

Jelas, koneksi langsung hanya berfungsi untuk sejumlah kecil pihak. Tidak mungkin bagi setiap pengguna Internet untuk memiliki kabel khusus untuk setiap pengguna lainnya. Untuk alasan keandalan dan ukuran, Internet dan sebagian besar jaringan lain menyerupai jaring, dengan data didorong di sepanjang jalur dari sumber ke tujuan.

Protokol

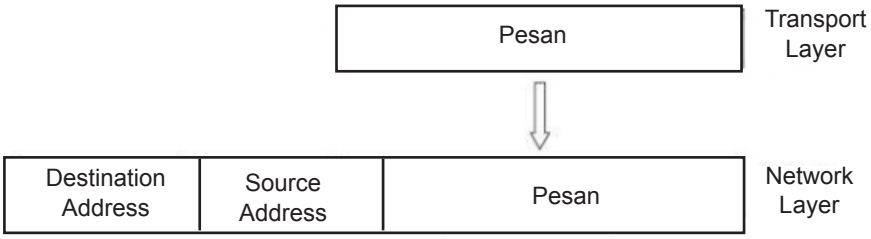
Ketika kita menggunakan jaringan, media komunikasi biasanya transparan kepada kita. Artinya, kebanyakan dari kita tidak tahu apakah komunikasi kita dilakukan melalui kawat tembaga, serat optik, satelit, gelombang mikro, atau kombinasi lainnya. Bahkan, media komunikasi dapat berubah dari satu transmisi ke transmisi berikutnya. Ambiguitas ini sebenarnya adalah fitur positif dari sebuah

jaringan: independensinya. Artinya, komunikasi dipisahkan dari media komunikasi yang sebenarnya. Independensi dimungkinkan karena kami telah menetapkan protokol yang memungkinkan pengguna untuk melihat jaringan pada tingkat komunikasi abstrak yang tinggi (melihatnya dari segi pengguna dan data); rincian bagaimana komunikasi dicapai tersembunyi di dalam perangkat lunak dan perangkat keras di kedua ujungnya. Perangkat lunak dan perangkat keras memungkinkan kami untuk mengimplementasikan jaringan menurut tumpukan protokol, arsitektur berlapis untuk komunikasi; kami mengGambarkan model protokol OSI sebelumnya dalam bab ini. Setiap lapisan dalam tumpukan sangat mirip dengan bahasa untuk mengkomunikasikan informasi yang relevan pada lapisan itu.

Protokol adalah bahasa atau seperangkat konvensi untuk bagaimana dua komputer akan berinteraksi. Protokol sederhana menyelesaikan transfer email. Pada dasarnya komputer pengirim menghubungi penerima dan mengatakan "Saya punya email untuk pengguna Anda Dmitri." Komputer penerima membalas untuk menerima transfer, pengirim mengirimkannya dan kemudian mengirimkan pemberitahuan penyelesaian untuk menunjukkan akhir transfer. Tentu saja, ikhtisar ini menghilangkan detail penting.

Addressing

Tetapi bagaimana pengirim menghubungi penerima? Misalkan pesan Anda ditujukan ke yourfriend@somewhere.net. Notasi ini berarti bahwa “somewhere.net” adalah nama host tujuan (atau lebih tepatnya, jaringan tujuan). Pada lapisan jaringan, perangkat keras yang disebut router sebenarnya mengirim pesan dari jaringan Anda ke router di jaringan di suatu tempat.net. Lapisan jaringan menambahkan dua header untuk menunjukkan alamat komputer Anda sebagai sumber dan alamat where.net sebagai tujuan. Logikanya, pesan Anda siap untuk berpindah dari mesin Anda ke router Anda ke router teman Anda ke komputer teman Anda. (Faktanya, di antara dua router mungkin ada banyak router lain di jalur melalui jaringan dari Anda ke teman Anda.) Bersama-sama, struktur lapisan jaringan dengan alamat tujuan, alamat sumber, dan data disebut apacket. Transformasi protokol lapisan jaringan dasar ditunjukkan pada Gambar 3-5.



Gambar 3-5 Transformasi Lapisan Jaringan

Pesan harus berjalan dari komputer Anda ke router Anda. Setiap komputer yang terhubung ke jaringan memiliki kartu antarmuka jaringan (NIC) dengan alamat fisik yang unik, yang disebut alamat MAC (untuk Media Access Control). Pada tingkat tautan data, dua header lagi ditambahkan, satu untuk alamat NIC komputer Anda (MAC sumber) dan satu lagi untuk alamat NIC router Anda. Struktur lapisan data-link dengan MAC tujuan, MAC sumber, dan data disebut bingkai. Setiap NIC menempatkan data ke media komunikasi ketika memiliki data untuk dikirim dan diambil dari jaringan frame tersebut dengan alamatnya sendiri sebagai alamat tujuan.

Di sisi penerima (tujuan), proses ini dilakukan secara terbalik: Kartu NIC menerima bingkai yang ditujukan untuknya. Lapisan jaringan penerima memeriksa apakah paket benar-benar ditujukan padanya. Paket mungkin tidak tiba sesuai urutan pengirimannya (karena penundaan jaringan atau perbedaan jalur melalui jaringan), sehingga lapisan sesi mungkin harus menyusun ulang paket. Lapisan presentasi menghilangkan kompresi dan mengatur tampilan yang sesuai untuk komputer tujuan. Akhirnya, lapisan aplikasi memformat dan mengirimkan data sebagai satu kesatuan yang lengkap.

Pelapisan dan koordinasi banyak pekerjaan, dan setiap lapisan protokol melakukan bagiannya sendiri. Tetapi pekerjaan ini sepadan dengan usaha karena lapisan yang berbeda memungkinkan Outlook berjalan pada PC IBM pada jaringan Ethernet di Washington D.C. untuk berkomunikasi dengan pengguna yang menjalankan Eudora pada komputer Apple melalui koneksi dial-up di Praha. Selain itu, pemisahan berdasarkan lapisan membantu staf jaringan memecahkan masalah saat terjadi kesalahan.

Routing

Kami masih belum menjawab pertanyaan bagaimana data didapat dari NIC sumber ke tujuan. Internet memiliki banyak perangkat yang disebut router, yang tujuannya adalah untuk mengarahkan ulang paket dalam upaya untuk membawa mereka lebih dekat ke tujuan mereka. Protokol perutean rumit, tetapi pada dasarnya ketika router menerima paket, ia menggunakan Tabel untuk menentukan jalur tercepat ke tujuan dan meneruskan paket ke langkah berikutnya di jalur itu. Router berkomunikasi dengan router tetangga untuk memperbarui status konektivitas dan arus lalu lintas; dengan pembaruan ini, router terus memperbarui Tabel langkah terbaik berikutnya.
Router mengarahkan lalu lintas pada jalur yang mengarah ke tujuan.

Ports

Seperti yang baru saja kami jelaskan, data tidak secara ajaib masuk ke komputer atau dieksekusi sendiri; beberapa program aktif pada komputer penerima harus menerima data dan menyimpan atau memprosesnya. Beberapa program meminta data, seperti kotak yang meminta nama dan kata sandi, tetapi di lain waktu data tersebut datang dari jaringan dan harus diarahkan ke program yang akan menanganinya. Contoh kasus terakhir ini adalah email masuk: Email baru dapat dikirim kapan saja, jadi program layanan yang berjalan di komputer harus siap menerima email dan meneruskannya ke klien email pengguna seperti Microsoft Outlook atau Mozilla Thunderbird . Layanan semacam itu terkadang disebut daemon; misalnya, daemon yang siap menerima surat masuk bernama popd, daemon yang mendukung fungsi penerimaan surat Protokol Kantor Pos.

Banyak layanan umum terikat pada port yang disepakati, yang pada dasarnya hanyalah angka untuk mengidentifikasi layanan yang berbeda; nomor port tujuan diberikan di header setiap paket atau unit data. Port 0 hingga 4095 disebut port terkenal dan menurut konvensi terkait dengan layanan tertentu. Misalnya, email masuk sering dikirim dengan Post Office Protocol (POP), dan server POP biasanya terikat ke port 110. Server POP adalah program yang menerima dan menyimpan email masuk dan menunggu klien untuk meminta email yang telah sudah diterima dan antri. Klien menghubungi server, mengirim ke port 110 paket yang meminta pembentukan sesi; dengan respons server, klien dan server bernegosiasi untuk mentransfer email dari server.

Tinjauan atau tinjauan jaringan ini dengan sendirinya menghilangkan detail penting. Tujuan kami hanya untuk memastikan Anda mengetahui beberapa istilah dan konsep dasar jaringan sehingga kami dapat memeriksa masalah keamanan dalam jaringan.

Untuk topik ketiga dari bagian ini kami mengeksplorasi ketersediaan, atau kerugiannya, dalam kelas serangan yang dikenal sebagai penolakan layanan. Karena konektivitas atau akses merupakan aspek penting dari komputasi jaringan, segala sesuatu yang sangat membatasi penggunaan jaringan meniadakan seluruh tujuan jaringan. Dengan demikian, penyerang yang dapat menolak layanan kepada pengguna menyebabkan kerugian serius. Serangan penolakan layanan adalah contoh pertama dalam buku ini di mana ketersediaan, bukan kerahasiaan atau integritas, adalah fitur keamanan yang dominan.

Saat kami menyajikan tiga jenis ancaman ini, kami juga mengisyaratkan kontrol dan penanggulangannya. Namun, Bagian II dari bab ini adalah tempat kami menyajikan kontrol dan penanggulangan ini secara mendalam.

Bagian I—Perang terhadap Jaringan: Serangan Keamanan Jaringan

Pada bagian ini kami membahas tiga jenis ancaman dan kerentanan keamanan: Pertama, kami mempertimbangkan kegagalan kerahasiaan dan integritas versi jaringan. Dalam jaringan, hilangnya kerahasiaan sering disebut penyadapan (bahkan ketika tidak ada kabel fisik yang terlibat), dan hilangnya integritas berada di bawah judul korupsi data yang luas. Meskipun metode serangannya mungkin baru, hilangnya kerahasiaan atau integritas harus diketahui dari bab-bab sebelumnya di mana kita mengeksplorasi kegagalan ini dalam program, browser, dan sistem operasi.

Ancaman kedua yang kami tangani melibatkan jaringan nirkabel. Di sini juga, bahaya utama adalah kerahasiaan atau integritas data pengguna. Berbeda dengan jaringan area luas bersama, jaringan nirkabel adalah sesuatu yang dapat dikontrol oleh pengguna. Dalam jaringan bersama yang besar, dikelola secara terpusat, pengguna memiliki sedikit kendali atas jenis layanan keamanan yang tersedia. Sebaliknya, jaringan nirkabel lebih kecil, mungkin tidak ada manajemen atau pemantauan aktif, dan pihak yang berbagi jaringan cukup lokal. Akibatnya, pengguna memiliki peran yang lebih langsung dalam keamanan.

3.2 Ancaman Terhadap Komunikasi Jaringan

Sekarang kita melihat ancaman keamanan dalam jaringan. Dari uraian k tentang ancaman di Bab 1, Anda mungkin ingat empat jenis potensi bahaya:

- intersepsi, atau melihat tanpa izin
- modifikasi, atau perubahan tidak sah
- fabrikasi, atau kreasi yang tidak sah
- gangguan, atau mencegah akses resmi

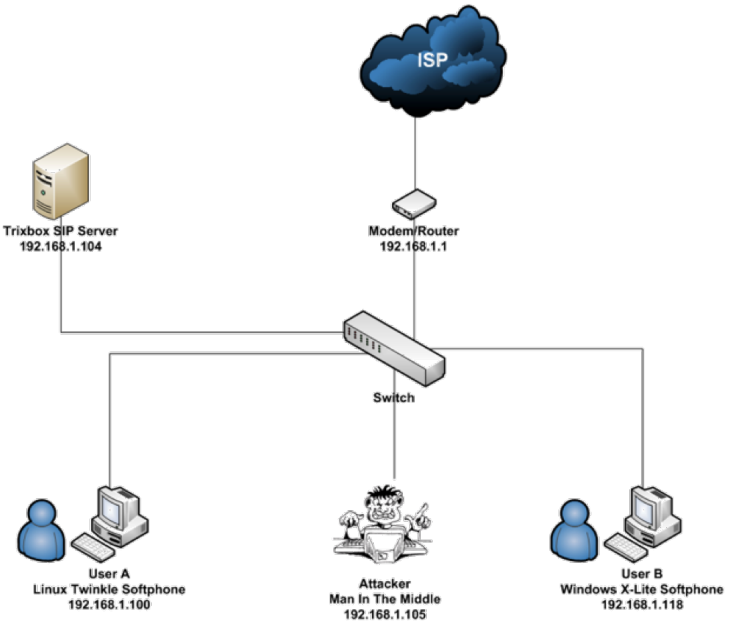
Keempat jenis ini berlaku untuk jaringan, meskipun terminologinya sedikit berbeda. Intersepsi kadang-kadang disebut penyadapan atau penyadapan, modifikasi dan fabrikasi biasanya dikenal

dengan istilah kegagalan integritas yang lebih umum, dan gangguan dalam jaringan adalah penolakan layanan. Selanjutnya kami mempertimbangkan masalah keamanan jaringan ini. Kami juga membahas pemindaian port: pengintaian sebelum serangan.

3.2.1 Intersepsi: Eavesdropping dan Penyadapan

Analis keamanan terkadang menggunakan konsep perimeter keamanan, garis virtual yang mengelilingi serangkaian sumber daya komputasi yang dilindungi. Anda mungkin menganggap perimeter keamanan mencakup lokasi fisik, seperti rumah, sekolah, kantor, atau toko. Tentu saja, jalur-jalur ini tidak benar-benar ada, dan untuk banyak penggunaan jaringan, Anda perlu memperluas akses Anda di luar zona yang dilindungi. Tetapi karena Anda kehilangan kendali atas peralatan (kabel, perangkat jaringan, server) di luar zona Anda, kemampuan Anda untuk mengamankan data menjadi terbatas.

Eavesdropping adalah aktifitas mendengarkan (*listening*) terhadap konversasi yg dilakukan pihak lain dengan tidak diketahui oleh pihak tersebut. Umumnya dapat terjadi pada media Telepon, Email, Instant Messaging, dan media komunikasi lainnya Tujuannya untuk mencuri data yang tidak terenkripsi dan dikirimkan melalui jaringan data. Seperti yang digambarkan pada Gambar 3.6



Gambar 3-6 Mekansime Eavesdropping

Penyadapan adalah nama yang diberikan untuk penyadapan data, seringkali terselubung dan tidak sah. Seperti yang dijelaskan Studi Kasus 3-2, bahkan pintu belakang yang ditujukan hanya untuk penyadapan resmi pengadilan dapat disalahgunakan. Nama penyadapan mengacu pada mekanisme asli, yang merupakan perangkat yang dilampirkan ke kawat untuk memisahkan jalur kedua yang akan diikuti oleh data selain jalur utama. Sekarang, tentu saja, medianya berkisar dari kawat tembaga hingga kabel serat dan sinyal radio, dan cara penyadapan bergantung pada medianya.

Studi Kasus 3-2 : Intersepsi yang Tidak Diinginkan

Penyedia telekomunikasi bekerja sama dengan pemerintah dalam apa yang disebut penyadapan yang sah. Setiap kali pengadilan mengizinkan penyadapan di telepon atau komunikasi data, agen pemerintah bekerja sama dengan penyedia layanan untuk memasang penyadapan. Perangkat keras dan perangkat lunak telekomunikasi modern menyertakan fitur khusus untuk mengimplementasikan penyadapan ini seiring dengan berkembangnya teknologi. Bahkan komunikasi suara sekarang sering ditransmisikan secara digital, menggunakan router dan protokol perutean seperti yang digunakan untuk jaringan data di Internet.

Pada konferensi keamanan Black Hat pada bulan Februari 2010, peneliti keamanan IBM Tom Cross mempresentasikan sebuah makalah di mana ia mengungkapkan masalah teknis dan prosedural dengan router Cisco yang memengaruhi intersepsi yang sah. Router Cisco telah rentan terhadap kelemahan keamanan yang pertama kali diumumkan pada tahun 2008: Cacat tersebut dapat memungkinkan akses yang tidak diautentikasi ke router. Meskipun patch telah dirilis, tidak semua router jaringan telekomunikasi telah diperbarui. Lebih lanjut, Cross mengatakan peralatan Cisco tidak melacak upaya login yang gagal atau memberi tahu administrator, membuat perangkat rentan terhadap serangan tebak kata sandi otomatis, dan tidak ada audit yang dihasilkan dari penggunaan fungsi intersep.

Karena itu, seorang karyawan ISP berpotensi memantau penyadapan penyadapan yang sah secara sah dan memperingatkan subjek bahwa mereka sedang diawasi, menurut Cross. Demikian pula, seorang karyawan dapat membuat penyadapan yang tidak sah terhadap pelanggan mana pun, dan ISP tidak akan memiliki jejak audit untuk mendeteksi penyadapan.

Cross menunjukkan bahwa Cisco adalah satu-satunya vendor perangkat keras utama yang merilis untuk pengawasan publik desain produknya untuk fungsi intersep yang sah; dia kemudian mengatakan bahwa karena perusahaan lain belum mempublikasikan desain mereka, tidak ada yang bisa memastikan apakah produk mereka aman.

Kerentanan di Kasus ini diutarakan secara tentatif: "bisa berpotensi ..." dan "bisa membangun ..." Pilihan kata itu menunjukkan bahwa kita tidak tahu apakah peralatan Cisco telah disusupi dengan cara itu.

Namun, peralatan Vodafone Yunani, penyedia telekomunikasi lain, dikompromikan dengan serangan serupa, yang melibatkan komunikasi pejabat tinggi Yunani, seperti yang dilaporkan oleh Vassilis Pervalakis dan Diomidis Spinellis [PRE07]. Vodafone menggunakan peralatan yang diproduksi oleh Ericsson, produsen Swedia. Dapat dimengerti, Ericsson ingin membuat satu model perangkat keras dan perangkat lunak yang dapat dijual di banyak pasar, sehingga mencakup kode untuk menerapkan intersep yang sah tetapi menonaktifkan kode untuk pelanggan yang tidak membeli

paket itu. Vodafone tidak mendapatkan pengaya itu. Pada tahun 2003, Ericsson meningkatkan perangkat lunak dan secara tidak sengaja mengaktifkan kode intersep dalam versi yang dikirimkan ke Vodafone. Meskipun kodenya ada di sana, kode itu tidak muncul di antarmuka tingkat pengguna, sehingga karyawan Vodafone tidak memiliki cara untuk mengaksesnya, apalagi mengetahui bahwa mereka memilikinya. Situasi ini adalah contoh sempurna dari pintu jebakan yang dijelaskan di Bab 3, dengan semua peringatan yang kami sampaikan di sana juga. Dan di bab yang sama kami menentang keamanan melalui ketidakjelasan—berharap tidak ada yang akan menemukan sesuatu jika Anda tidak mempublikasikannya.

Agen tidak dikenal memasang tambalan di perangkat lunak sakelar Ericsson yang mengaktifkan kode intersepsi yang tidak aktif. Selanjutnya, agen melakukannya dengan cara yang tidak menghasilkan log audit, meskipun intersepsi dan penyalinan panggilan biasanya membuat entri audit untuk catatan penegakan hukum. Modifikasi kode ini dibuat dengan hati-hati agar tidak ditemukan.
Dengan kode ini, orang luar yang tidak dikenal dapat mendengarkan semua percakapan telepon seluler dari sekitar 100 pejabat politik, termasuk perdana menteri, istrinya, dan beberapa menteri kabinet.

Skema ini akhirnya terungkap pada tahun 2004 hanya ketika Ericsson mendistribusikan patch perangkat lunak lain; karena interaksi dengan perangkat lunak jahat, pesan kesalahan dalam perangkat lunak sakelar nyata tidak berhasil dikirim, yang memicu kondisi peringatan. Saat menyelidiki sumber peringatan tersebut, teknisi Ericsson menemukan tambahan tersebut.

Pelajaran dari kompromi itu adalah bahwa pintu belakang, bahkan tanpa dokumen, dapat ditemukan dan dieksploitasi. Keamanan melalui ketidakjelasan bukanlah tindakan pencegahan yang efektif.

Pengguna umumnya memiliki sedikit kendali atas perutean sinyal. Dengan sistem telepon, misalnya, panggilan dari New York ke Sydney mungkin melakukan perjalanan ke barat melalui satelit, mentransfer ke kabel bawah laut, dan mencapai tujuan akhir dengan kabel konvensional. Sepanjang jalan, sinyal bisa melewati berbagai negara, serta wilayah internasional lautan dan langit. Hal yang sama berlaku untuk komunikasi digital jaringan, yang menggunakan beberapa sumber daya yang sama dengan telepon. Sinyal dapat melakukan perjalanan melalui daerah yang tidak bersahabat dan daerah yang penuh dengan pesaing. Sepanjang jalan mungkin orang-orang dengan metode, peluang, dan motif untuk mendapatkan data Anda. Dengan demikian, jaringan area luas bisa jauh lebih berisiko daripada jaringan lokal yang terkontrol dengan baik.

Enkripsi adalah tindakan pencegahan terkuat dan paling umum digunakan terhadap intersepsi, meskipun keamanan fisik (melindungi jalur komunikasi itu sendiri), jalur khusus, dan perutean terkontrol (memastikan bahwa komunikasi hanya berjalan di sepanjang jalur tertentu) juga memiliki peran. Kami memeriksa enkripsi untuk komunikasi nanti dalam bab ini.

3.2.2 Apa yang Membuat Jaringan Rentan terhadap Intersepsi?

Pengguna rumahan yang terisolasi atau kantor yang berdiri sendiri dengan beberapa karyawan tidak mungkin menjadi target banyak serangan. Tetapi tambahkan jaringan ke dalam campuran dan risikonya meningkat tajam. Pertimbangkan bagaimana jaringan berbeda dari lingkungan yang berdiri sendiri.

Anonimitas

Seorang penyerang dapat melakukan serangan dari jarak ribuan mil dan tidak pernah melakukan kontak langsung dengan sistem, administratornya, atau penggunanya. Dengan demikian, penyerang potensial aman di balik perisai elektronik. Serangan tersebut dapat melewati banyak host lain dalam upaya untuk menyamarkan asal serangan. Dan otentikasi komputer-ke-komputer tidak sama untuk komputer seperti halnya untuk manusia. Seperti yang diilustrasikan oleh Studi Kasus 3-3, otentikasi terdistribusi yang aman membutuhkan pemikiran dan perhatian terhadap detail.

Studi Kasus 3-3 : Kegagalan Otentikasi Terdistribusi

Otentikasi harus ditangani dengan hati-hati dan benar dalam jaringan karena jaringan melibatkan otentikasi tidak hanya orang tetapi juga proses, server, dan layanan yang hanya terkait secara longgar dengan seseorang. Dan untuk jaringan, proses otentikasi dan database sering didistribusikan untuk kinerja dan keandalan. Pertimbangkan skema otentikasi yang diterapkan Microsoft untuk sistem operasi Windows-nya pada tahun 2000. Pada Windows NT 4.0, sistem Microsoft pertama yang mendukung komputasi terdistribusi skala besar, basis data otentikasi didistribusikan di antara beberapa pengontrol domain. Setiap pengontrol domain ditetapkan sebagai pengontrol utama atau cadangan. Semua perubahan pada basis data otentikasi harus dilakukan pada pengontrol domain primer (tunggal); perubahan kemudian direplikasi dari pengontrol domain utama ke cadangan. Pendekatan ini berarti perubahan secara konsisten dikendalikan dan diimplementasikan pada satu titik pengontrol utama. Tentu saja, pengontrol tunggal ini juga menjadi satu titik kegagalan dan potensi kemacetan kinerja untuk domain tersebut.

Pada Windows 2000, konsep pengontrol domain utama dan cadangan ditinggalkan. Sebagai gantinya, jaringan melihat pengontrol sebagai pohon yang sama di hutan, di mana pengontrol domain dapat memperbarui database otentikasi. Skema ini mencerminkan gagasan Microsoft bahwa sistem itu "multimaster": Hanya satu pengontrol yang bisa menjadi master pada waktu tertentu, tetapi pengontrol apa pun bisa menjadi master. Setelah perubahan dibuat ke master, mereka secara otomatis direplikasi ke pengontrol domain yang tersisa di hutan.

Pendekatan ini lebih fleksibel dan kuat daripada pendekatan primer-sekunder karena memungkinkan pengontrol mana pun untuk mengambil alih—terutama berguna jika satu atau lebih pengontrol gagal atau tidak berfungsi karena alasan tertentu. Tetapi pendekatan multimaster menimbulkan masalah baru: Karena pengontrol domain mana pun dapat memulai perubahan pada basis data autentikasi, peretas mana pun

yang dapat mendominasi pengontrol domain tunggal dapat mengubah basis data autentikasi. Dan, yang lebih buruk, perubahan yang salah itu kemudian direplikasi di seluruh hutan yang tersisa. Secara teoritis, peretas dapat mengakses apa pun di hutan yang mengandalkan Windows 2000 untuk otentikasi.

Ketika kita memikirkan penyerang, kita biasanya memikirkan ancaman dari luar sistem. Namun pada kenyataannya pendekatan multimaster bisa menggoda orang di dalam sistem juga. Administrator domain di domain mana pun di hutan dapat mengakses pengontrol domain dalam domain tersebut. Berkat multimaster, administrator domain juga dapat memodifikasi basis data autentikasi untuk mengakses hal lain di hutan.

Untuk alasan ini, administrator sistem harus mempertimbangkan bagaimana mereka mendefinisikan domain dan pemisahannya dalam jaringan. Kalau tidak, mereka bisa memunculkan skenario yang menakutkan tapi mungkin. Misalnya, anggaplah satu administrator domain adalah apel yang buruk. Dia mencari cara untuk memodifikasi database otentikasi untuk menjadikan dirinya administrator untuk seluruh hutan. Kemudian dia dapat mengakses data apa pun di hutan, mengaktifkan layanan untuk beberapa pengguna, dan mematikan layanan untuk pengguna lain.

Titik Serangan

Sistem komputasi sederhana adalah unit mandiri. Kontrol akses pada satu mesin menjaga kerahasiaan data pada prosesor tersebut. Namun, ketika file disimpan di host jaringan yang jauh dari pengguna, data atau file itu sendiri dapat melewati banyak host untuk sampai ke pengguna. Administrator satu host dapat menerapkan kebijakan keamanan yang ketat, tetapi administrator tersebut tidak memiliki kendali atas host lain dalam jaringan. Dengan demikian, pengguna harus bergantung pada mekanisme kontrol akses di masing-masing sistem ini. Serangan dapat datang dari host mana pun ke host mana pun, sehingga jaringan besar menawarkan banyak titik kerentanan.

Berbagi (*Sharing*)

Karena jaringan memungkinkan berbagi sumber daya dan beban kerja, sistem jaringan membuka akses potensial ke lebih banyak pengguna daripada komputer tunggal. Mungkin lebih buruk, akses diberikan ke lebih banyak sistem, sehingga kontrol akses untuk sistem tunggal mungkin tidak memadai dalam jaringan.

Kompleksitas Sistem

Keamanan yang andal sulit, jika bukan tidak mungkin, pada sistem operasi besar, terutama yang tidak dirancang khusus untuk keamanan. Jaringan menggabungkan dua atau lebih sistem operasi yang mungkin berbeda. Oleh karena itu, sistem operasi/kontrol jaringan cenderung lebih kompleks daripada sistem operasi untuk sistem komputasi tunggal. Selain itu, komputer laptop biasa saat ini memiliki daya komputasi yang lebih besar daripada banyak komputer kantor dalam dua dekade terakhir.

Penyerang dapat menggunakan kekuatan ini untuk mendapatkan keuntungan dengan menyebabkan komputer korban melakukan bagian dari perhitungan serangan. Dan karena rata-rata komputer

sangat kuat, sebagian besar pengguna tidak tahu apa yang sebenarnya dilakukan komputer mereka setiap saat: Proses apa yang aktif di latar belakang saat Anda memainkan Invaders from Mars? Kompleksitas ini mengurangi kepercayaan pada keamanan jaringan.

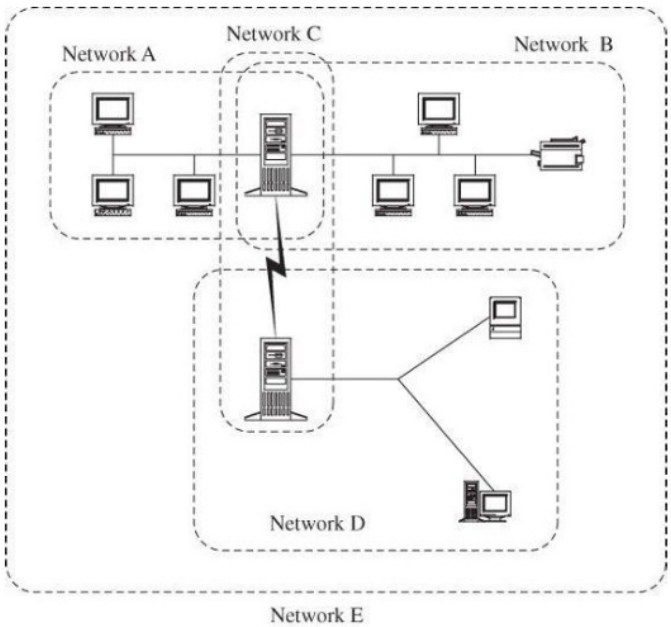
Perimeter Tidak Diketahui

Perluasan jaringan juga menyiratkan ketidakpastian tentang batas jaringan. Satu host mungkin merupakan node pada dua jaringan yang berbeda, sehingga sumber daya pada satu jaringan dapat diakses oleh pengguna jaringan lain juga. Meskipun aksesibilitas yang luas merupakan keuntungan, kelompok yang tidak diketahui atau tidak terkendali dari pengguna yang mungkin berbahaya ini merupakan kerugian keamanan.

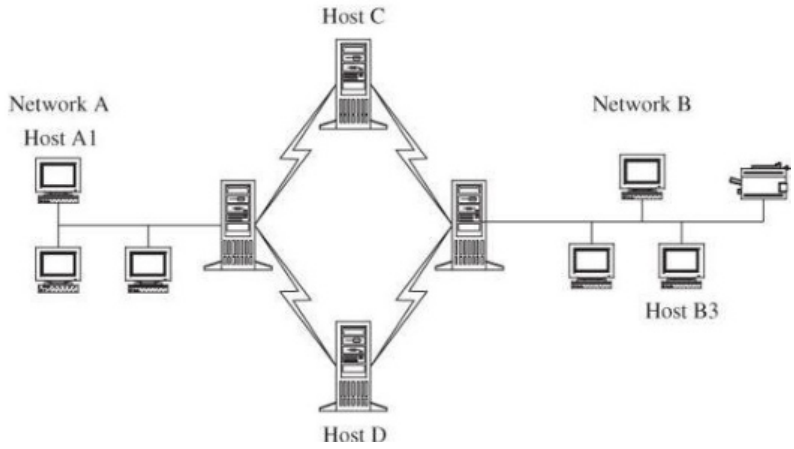
Masalah serupa terjadi ketika host baru dapat ditambahkan ke jaringan. Setiap node jaringan harus dapat bereaksi terhadap kemungkinan kehadiran host baru yang tidak dapat dipercaya. Gambar 3-7 menunjukkan masalah dalam mendefinisikan batas-batas jaringan. Perhatikan, misalnya, bahwa pengguna di host di jaringan D mungkin tidak menyadari koneksi potensial dari pengguna jaringan A dan B. Dan host di tengah jaringan A dan B sebenarnya milik A, B, C, dan E. Jika jaringan ini memiliki aturan keamanan yang berbeda, aturan apa yang menjadi subjek host tersebut?

Jalur Tidak Diketahui

Gambar 3-8 mengilustrasikan bahwa mungkin ada banyak jalur dari satu host ke host lainnya. Misalkan pengguna di host A1 ingin mengirim pesan ke pengguna di host B3. Pesan itu mungkin dirutekan melalui host C atau D sebelum tiba di host B3. Host C dapat memberikan keamanan yang dapat diterima, tetapi D tidak. Pengguna jaringan jarang memiliki kendali atas perutean pesan mereka. Ketidakmampuan untuk mengontrol angka perutean dalam intersepsi sinyal ponsel, seperti yang dijelaskan di Studi Kasus 3-4.



Gambar 3-7 Batas Jaringan Tidak Jelas



Gambar 3-8 Beberapa Jalur Perutean

Studi Kasus 3-4 : Percakapan dengan Handphone

Telepon seluler jauh lebih rumit daripada yang kadang-kadang kita bayangkan. Dengan telepon rumah, pada dasarnya Anda memiliki satu kabel yang menghubungkan telepon Anda dengan kantor peralihan telepon lokal, sehingga sebagian besar telepon hanyalah elektronik untuk mengubah suara audio menjadi sinyal elektronik dan kembali lagi. Telepon seluler melakukan itu, ditambah lagi mereka harus mengatur koneksi ke jaringan seluler. Berbeda halnya dengan komunikasi darat, saat ponsel bergerak (dan terkadang bahkan saat tidak), perangkat terus mencari sinyal berbeda untuk dihubungkan.

Pada konferensi Defcon 18 2010 di Las Vegas, Nevada, peneliti keamanan Chris Paget mendemonstrasikan menara GSM buatannya sendiri, dan meyakinkan hingga 30 peserta tanpa disadari dengan ponsel untuk terhubung ke sistemnya. Bagian-bagiannya berharga sekitar $ 1500, dan dia menggunakan laptop biasa yang menjalankan aplikasi open source yang pada dasarnya mengubah laptop menjadi stasiun pangkalan GSM. Ponsel akan mencoba mengasosiasikan sinyal terkuat yang dapat ditemukannya; kedekatan membantunya memenuhi tujuan itu. Pengguna tidak menyadari ketika ponsel membuat atau mengubah hubungannya dengan penyedia.

Amerika Serikat memiliki undang-undang yang melarang penyadapan percakapan telepon, jadi Paget berhati-hati untuk mengumumkan niat dan aktivitasnya kepada para hadirin. Dia juga dengan hati-hati menghilangkan semua jejak panggilan telepon yang ditangani sistemnya untuk menjaga privasi pelanggannya. (Namun, sebagian besar penyerang tidak akan begitu sopan.) Untuk tujuan demonstrasi, dia hanya mencegat panggilan keluar dan memainkan pesan peringatan kepada penelepon. Mungkin yang paling menarik, sistemnya memaksa ponsel yang terhubung untuk menggunakan protokol 2G yang lebih lama; Paget juga mengatakan sistemnya, dalam kemampuan negosiasi dengan ponsel, bisa memaksa ponsel untuk tidak menggunakan enkripsi (yang tentu saja memfasilitasi intersepsi).

Tujuan Paget untuk demonstrasi adalah untuk menunjukkan betapa mudahnya penyerang dapat mencegat komunikasi di jaringan seluler. “Masalah utamanya adalah GSM rusak. Anda memiliki 3G dan semua protokol selanjutnya dengan masalah untuk GSM yang telah dikenal selama beberapa dekade. Sudah waktunya kita melanjutkan, ” kata Paget.

Orang-orang memang pindah … tapi tidak seperti yang dimaksud Paget. Pada bulan Desember 2010, dua peneliti di Chaos Computer Club Congress di Berlin, Jerman, menunjukkan kemampuan mereka untuk mencegat panggilan GSM. Karsten Nohl dan Sylvain Munaut menggunakan ponsel murah Motorola untuk mencegat panggilan yang sedang berlangsung. Ponsel Motorola berisi firmware yang mudah diganti, mengubah ponsel menjadi pencegat yang menerima semua lalu lintas dalam jangkauan. Dari alam semesta itu, mereka dapat mengisolasi komunikasi telepon mana pun. Dengan menggunakan Tabel besar kunci enkripsi yang telah dibuat sebelumnya, mereka menentukan kunci khusus yang digunakan untuk aliran komunikasi itu dan akhirnya mencegat teks biasa dari seluruh percakapan.

Dengan demikian, jaringan berbeda secara signifikan dari lingkungan lokal yang berdiri sendiri. Karakteristik jaringan secara signifikan meningkatkan risiko keamanan.

3.2.3 Data Corruption : Modifikasi dan Fabrikasi

Penyadapan tentu saja merupakan ancaman yang signifikan, dan merupakan inti dari insiden besar pencurian rahasia dagang atau spionase. Tetapi intersepsi adalah ancaman pasif: Komunikasi berjalan normal, kecuali bahwa pihak ketiga yang tersembunyi juga mendengarkan.

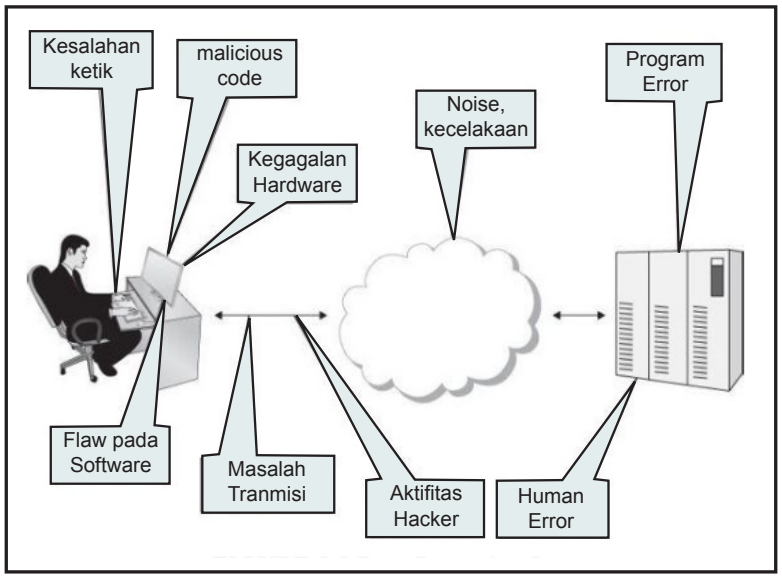
Jika Anda ingat dari Bab 1, modifikasi dan fabrikasi juga merupakan masalah keamanan komputer, dan juga berlaku untuk jaringan. Ancamannya adalah bahwa komunikasi akan berubah selama transmisi. Terkadang tindakan tersebut melibatkan modifikasi data dalam perjalanan; di lain waktu itu memerlukan pembuatan konten baru atau mengulangi komunikasi yang sudah ada. Ketiga serangan ini masing-masing disebut modifikasi, penyisipan, dan pemutaran ulang. Serangan tersebut dapat berbahaya atau tidak, diinduksi atau dari penyebab alami.

Orang sering menerima data yang salah atau rusak: salah mengeja nama, kesalahan ketik yang jelas, entri yang salah pada daftar. Jika Anda menonton teks tertutup secara real-time di televisi, terkadang Anda melihat teks normal berubah menjadi omong kosong dan kemudian kembali normal setelah beberapa saat. Kesalahan seperti ini terjadi, dan kami menghubungi seseorang untuk meminta koreksi jika masalahnya serius atau mengabaikannya jika tidak. Kesalahan sering terjadi sehingga terkadang kita tidak menyadarinya.

Pada Gambar 3-9 kami mengingatkan Anda tentang beberapa sumber korupsi data; kami sebelumnya telah menjelaskan sebagian besar penyebab ini. Anda harus ingat bahwa kerusakan data dapat disengaja atau tidak disengaja, dari sumber yang berbahaya atau tidak berbahaya, dan diarahkan atau tidak disengaja. Korupsi data dapat terjadi selama entri data, dalam penyimpanan, selama

penggunaan dan komputasi, dalam perjalanan, dan pada keluaran dan pengambilan. Pada bagian ini kita tertarik pada korupsi sebagai bagian dari interaksi jaringan.

Kadang-kadang modifikasi terang-terangan, membuatnya mudah terlihat bahwa perubahan telah terjadi (misalnya, penghapusan lengkap, yang dapat dideteksi oleh program, atau penggantian teks dengan data biner, yang akan terlihat oleh pembaca manusia). Di lain waktu perubahannya tidak kentara, seperti perubahan satu bit, yang memungkinkan pemrosesan berlanjut, meskipun mungkin menghasilkan hasil yang salah.



Gambar 3-9 Sumber Data Corruption

Media komunikasi diketahui rentan terhadap korupsi data. Faktor sederhana seperti cuaca dan pepohonan dapat mengganggu transmisi bersih. Untuk alasan ini, protokol komunikasi menyertakan fitur untuk memeriksa dan memperbaiki, setidaknya beberapa, kesalahan dalam transmisi. Paket protokol TCP/IP (yang akan kami jelaskan nanti dalam bab ini), digunakan untuk sebagian besar komunikasi data Internet. TCP/IP memiliki fitur ekstensif untuk memastikan bahwa penerima mendapatkan aliran data yang lengkap, benar, dan tertata dengan baik, meskipun ada kesalahan selama transmisi.

Pada bagian ini kami menjelaskan beberapa kegagalan modifikasi yang rentan terhadap komunikasi.

Sequencing

Serangan atau masalah pengurutan melibatkan permutasi urutan data. Paling sering ditemukan dalam komunikasi jaringan, kesalahan pengurutan terjadi ketika fragmen aliran data berikutnya tiba sebelum yang sebelumnya: Paket 2 tiba sebelum paket 1.

Kesalahan pengurutan sebenarnya cukup umum dalam lalu lintas jaringan. Karena unit data dirutekan menurut informasi perutean yang tersedia, ketika paket 1 dikirim, rute terbaik, yang mana rute yang dipilih, melewati node C. Selanjutnya router mengetahui node C tidak lagi optimal, sehingga ketika paket 2 akan dikirim, routing melalui node D. Rute kedua memang lebih unggul, sehingga paket 2

tiba sebelum paket 1. Kemacetan, gangguan jaringan, peralatan yang rusak atau gagal, dan masalah kinerja dapat dengan mudah menyebabkan kesulitan kecepatan seperti ini.

Protokol jaringan seperti TCP suite memastikan urutan lalu lintas yang tepat. Namun, program aplikasi tidak selalu mendeteksi atau memperbaiki masalah pengurutan dalam aliran data. Misalnya, jika aplikasi menangani input dari banyak klien bersamaan di situs belanja, aplikasi harus memastikan bahwa pesanan individu dibuat dengan benar, terlepas dari urutan kedatangan pesanan.

Substitution

Serangan substitusi adalah penggantian satu bagian dari aliran data dengan yang lain. Substitusi nonmalicious dapat terjadi jika kerusakan perangkat keras atau perangkat lunak menyebabkan dua aliran data menjadi kusut, sehingga bagian dari satu aliran dipertukarkan dengan aliran lainnya.

Kesalahan substitusi dapat terjadi dengan kabel yang berdekatan atau komunikasi paralel multipleks dalam jaringan; kadang-kadang, gangguan, yang disebut crosstalk. memungkinkan data mengalir ke jalur yang berdekatan. Kabel logam lebih rentan terhadap crosstalk dari kabel yang berdekatan daripada serat optik. Crossover dalam komunikasi multiplexing terjadi jika pemisahan antar subchannel tidak memadai. Substitusi yang diinduksi perangkat keras seperti itu jarang terjadi.

Penyerang jahat dapat melakukan serangan substitusi dengan menyambungkan bagian dari satu komunikasi ke komunikasi lainnya. Dengan demikian, Amy dapat memperoleh salinan dari dua komunikasi, satu untuk mentransfer $100 ke Amy, dan yang kedua untuk mentransfer $100.000 ke Bill, dan Amy dapat menukar dua jumlah atau dua tujuan. Serangan substitusi semacam ini paling mudah dilakukan dengan komunikasi terformat. Jika Amy tahu, misalnya, bahwa byte 24-31 mewakili nomor akun, dia tahu cara merumuskan pesan baru yang mengalihkan uang ke akunnya.

Penanggulangan yang jelas terhadap serangan substitusi adalah enkripsi, mencakup seluruh pesan (membuat penyerang sulit melihat bagian mana yang akan diganti) atau membuat pemeriksaan integritas (membuat modifikasi lebih jelas).

Tidak semua serangan substitusi berbahaya, seperti yang dijelaskan pada contoh Studi Kasus 3-5.

Studi Kasus 3-5 : Donor Pengganti

Layanan Kesehatan Nasional Inggris (NHS) memelihara database donor organ potensial di Inggris. Menurut sebuah artikel di The Register 12 April 2010, bidang status donor organ salah dimasukkan untuk orang-orang yang mendaftarkan preferensi donasi organ mereka saat mengajukan permohonan SIM. Sekitar 400.000 bidang data dikoreksi oleh NHS dan 300.000 orang lainnya harus dihubungi untuk menentukan nilai yang benar.

Menurut tinjauan berikutnya [DUF10], kesalahan muncul pada tahun 1999 dan tidak diketahui sejak saat itu hingga 2010. NHS menerima data dari tiga sumber: rumah sakit, dokter, dan kantor SIM. Saat mengajukan permohonan SIM atau mendaftar

ke dokter atau rumah sakit, pemohon dapat menandai kotak yang menyebutkan organ mana, jika ada, yang ingin disumbangkan pemohon setelah kematian. Catatan yang dikirimkan ke NHS dari sumber mana pun berisi data identifikasi dan tujuh digit nomor berkode 1 untuk tidak dan 2 untuk ya. Namun, urutan organ yang tercantum pada aplikasi lisensi berbeda dari urutan penggunaan dua sumber lainnya, yang ditangani dengan benar oleh perangkat lunak sebelum tahun 1999. Dalam peningkatan perangkat lunak pada tahun 1999, semua input salah diproses dengan urutan yang sama.

Tinjauan setelah penemuan kesalahan merekomendasikan prosedur pengujian yang ditingkatkan, pemberitahuan kepada semua pihak yang terpengaruh setiap kali perubahan pemrograman akan diterapkan, dan audit sistem secara berkala, termasuk validasi catatan sampel.

Insersi

Serangan penyisipan, yang hampir merupakan bentuk substitusi, adalah serangan di mana nilai data dimasukkan ke dalam aliran. Seorang penyerang bahkan tidak perlu merusak skema enkripsi untuk memasukkan data yang tampak otentik; selama penyerang tahu persis di mana harus menyelipkan data, bagian baru dienkripsi di bawah kunci yang sama dengan komunikasi lainnya.

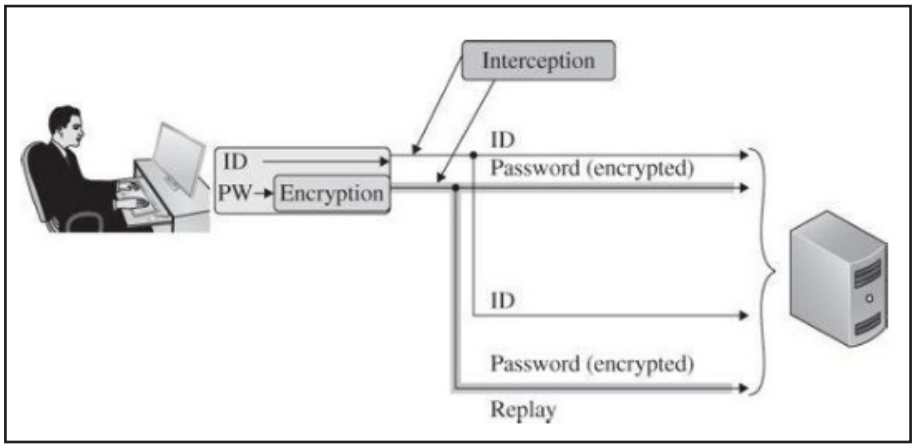
Replay

Dalam serangan replay, data yang sah dicegat dan digunakan kembali, umumnya tanpa modifikasi. Serangan replay berbeda dari serangan penyadapan (di mana konten data diperoleh tetapi tidak digunakan kembali) dan serangan man-in-the-middle (di mana konten dimodifikasi untuk menipu dua ujung agar percaya bahwa mereka berkomunikasi secara langsung).

Dalam kehidupan nyata, bank mencegah seseorang menyetor cek yang sama dua kali dengan menandai cek fisik, tetapi dengan setoran elektronik, di mana deposan mengambil Gambar cek dengan smartphone, mencegah penggunaan kembali lebih sulit. Contoh klasik serangan replay melibatkan transaksi keuangan dengan cara berikut. Pedagang yang tidak bermoral memproses kartu kredit atau transfer dana atas nama pengguna dan kemudian, melihat bahwa transfer berhasil, mengirimkan kembali transaksi lain atas nama pengguna.

Dengan serangan replay, pencegat tidak perlu mengetahui konten atau format transmisi; pada kenyataannya, serangan replay dapat berhasil pada data terenkripsi tanpa mengubah atau merusak enkripsi. Misalkan pedagang memiliki terminal kartu kredit dengan enkripsi bawaan, sehingga nomor kartu pengguna, mungkin PIN, jumlah transaksi, dan pengenal pedagang terikat menjadi satu pesan, dienkripsi, dan dikirim ke pusat pemrosesan kredit. Bahkan tanpa melanggar enkripsi, pedagang yang menyadap saluran komunikasi dapat mengulangi pesan transaksi yang sama untuk transfer kedua dengan jumlah yang sama. Tentu saja, dua transaksi yang identik dengan satu pedagang akan terlihat dan wajar bagi klien untuk disengketakan, dan keuntungan bersih dari pengulangan satu pembelian kredit akan relatif kecil. Namun demikian, kemungkinan pengulangan transaksi akan menjadi kerentanan.

Serangan replay juga dapat digunakan dengan kredensial otentikasi. Mengirim identitas dan kata sandi secara jelas merupakan kelemahan yang jelas, tetapi mengirimkan identitas secara jelas tetapi dengan kata sandi terenkripsi juga lemah, seperti yang ditunjukkan pada Gambar 3-10. Jika penyerang dapat memasukkan kata sandi terenkripsi ke dalam jalur komunikasi, maka penyerang dapat menyamar sebagai pengguna yang valid tanpa mengetahui kata sandinya.



Gambar 3-10 Kegagalan Kata Sandi Terenkripsi

Contoh serupa melibatkan cookie untuk otentikasi. Program email yang berjalan di dalam browser (seperti Gmail, Yahoo mail, dan Hotmail) terkadang mengidentifikasi dan mengautentikasi dengan cookie sehingga pengguna tidak perlu berulang kali mengetik pengenal dan kata sandi untuk membuka email. Jika penyerang dapat mencegat cookie yang dikirim ke (atau mengekstrak cookie yang disimpan oleh) browser korban, maka mengembalikan cookie yang sama dapat memungkinkan penyerang membuka sesi email dengan identitas korban. Pertukaran login dan kata sandi dapat dienkripsi dengan aman dan begitu juga konten cookie. Agar serangan ini berhasil, layanan email jarak jauh hanya perlu menerima salinan cookie-nya sendiri sebagai kredensial login yang valid.

Serangan ulangan dilawan dengan nomor urut. Pengirim memberikan setiap komunikasi nomor urut, yang dapat unik untuk satu penerima (pesan 1 ke James, pesan 2 ke James, dan sebagainya) atau satu urutan penomoran untuk semua pesan (pesan 1, pesan 2, pesan 3, di mana 1 pergi ke James, 2 ke Klara, dan 3 ke Lars). Setiap penerima menyimpan nomor pesan terakhir yang diterima dan memeriksa setiap pesan yang masuk untuk memastikan bahwa jumlahnya lebih besar dari pesan yang diterima sebelumnya.

Physical Replay

Akhirnya, untuk contoh fisik, pikirkan kamera keamanan yang memantau ruang, misalnya, pintu ke brankas bank. Penjaga di ruang kendali jarak jauh menonton monitor video untuk mendeteksi akses tidak sah ke pintu. Seorang penyerang dapat memberi makan Gambar yang tidak bersalah ke monitor. Para penjaga dibiarkan melihat Gambar yang tidak bersalah, selama waktu itu penyerang memiliki akses tak terpantau ke brankas bank. Tipuan ini ditampilkan dalam film Ocean's 11. Serangan serupa dapat digunakan terhadap otentikasi biometrik (misalnya, serangan sidik jari karet yang dijelaskan dalam Bab 2). Serangan serupa akan melibatkan pelatihan kamera pada Gambar ruangan di bawah pengawasan, kemudian memutar ulang Gambar sementara pencuri bergerak tanpa terdeteksi di seluruh lemari besi.

Seperti yang ditunjukkan contoh-contoh ini, serangan replay dapat menghindari identifikasi biasa, otentikasi, dan pertahanan kerahasiaan, dan dengan demikian memungkinkan penyerang untuk memulai dan
melakukan pertukaran dengan kedok korban. Nomor urut membantu melawan serangan ulangan.

Serangan Modifikasi

Semua serangan ini melibatkan beberapa aspek integritas. Ingat kisaran properti yang dicakup oleh konsep umum integritas; kami mengulanginya dari Bab 1 untuk referensi:

- tepat
- akurat
- tidak dimodifikasi
- dimodifikasi hanya dengan cara yang dapat diterima
- dimodifikasi hanya oleh orang yang berwenang
- dimodifikasi hanya oleh proses resmi
- konsisten
- konsisten secara internal
- bermakna dan berguna

Melindungi properti yang berbeda ini memerlukan tindakan pencegahan yang berbeda, termasuk alat, protokol, dan kriptografi. Dalam bab-bab sebelumnya kami menyajikan beberapa pendekatan ini, dan sekarang kami membangun metode-metode sebelumnya.

3.2.4 Interruption: Hilangnya Layanan

Kelas terakhir dari serangan jaringan yang kami pertimbangkan melibatkan ketersediaan, kaki ketiga dari triad C-I-A. Kita semua tahu betapa frustasinya kehilangan akses ke layanan penting, seperti ketika listrik padam atau sambungan telepon terputus. Tiba-tiba kami melihat semua cara kami bergantung pada layanan itu saat kami menunggu dengan cemas kru perbaikan.

Jaringan, dan terutama Internet, memiliki layanan yang sangat terjamin. Dari desain paling awal untuk Internet, redundansi dan toleransi kesalahan adalah karakteristik penting, dan ketahanannya tetap ada. Sebagian kekuatan ini disebabkan oleh arsitektur mesh dari Internet. Yang disebut last mile, koneksi terakhir antara host dan infrastruktur jaringan yang lebih besar, adalah jalur yang unik, sehingga setiap kegagalan di sana akan mengisolasi host. Tetapi begitu masuk ke jaringan, router memiliki banyak jalur sehingga jika salah satu tidak tersedia, jalur lain dapat digunakan.

Seperti kerentanan lain yang baru saja kita bahas, hilangnya layanan bisa berbahaya atau tidak berbahaya, disengaja atau tidak disengaja. Tidak seperti kegagalan kerahasiaan dan integritas, bagaimanapun, penolakan layanan bukanlah biner: Ya, Anda memiliki layanan atau tidak, tetapi pertanyaan kritis adalah berapa banyak? Kapasitas layanan dapat dikurangi. Apakah penurunan layanan sebesar 0,1 persen atau 1 persen atau 10 persen merupakan bencana besar? Jawabannya tergantung pada jaringan tertentu yang bersangkutan, beban lalu lintasnya, dan kekritisan datanya. Jadi, kita harus mempertimbangkan tidak hanya apakah layanan ada atau tidak, tetapi juga apakah jumlah yang ada memadai.

Routing

Seperti yang baru saja kami jelaskan, protokol perutean Internet rumit. Router harus saling percaya untuk pembaruan status pada aksesibilitas bagian lain dari Internet. Satu informasi buruk dapat meracuni kumpulan data dari banyak router, sehingga mengganggu aliran banyak jalur. Meskipun protokol perutean Internet menyembuhkan diri sendiri, yang berarti mereka memulihkan dari data yang buruk dengan mengkalibrasi ulang ketika mereka menemukan ketidakakuratan, perlu beberapa waktu agar efek kesalahan dihapus dari sistem.

Meskipun jarang dan sangat canggih, serangan terhadap sistem perutean dimungkinkan. Kami menjelaskan beberapa serangan yang relatif sederhana nanti dalam bab ini.

Permintaan yang berlebihan

Meskipun Mae West dilaporkan telah mengatakan "terlalu banyak hal yang baik bisa menjadi luar biasa," sentimen itu hampir tidak berlaku untuk jaringan. Kapasitas jaringan sangat besar tetapi terbatas, dan kapasitas tautan atau komponen tertentu jauh lebih kecil. Jadi, dengan permintaan yang ekstrim, penyerang dapat memFlooding i bagian penting dari jaringan, dari server halaman web ke router atau jalur komunikasi.

Bagaimana komponen yang diFlooding i merespons bervariasi. Beberapa komponen dimatikan sepenuhnya, beberapa menyediakan layanan yang diturunkan (lebih lambat) untuk semua permintaan, dan yang lain menghentikan beberapa aktivitas dalam upaya mempertahankan layanan untuk beberapa pengguna.

Serangan penolakan layanan yang berbahaya biasanya dilakukan melalui permintaan yang berlebihan. Tujuannya adalah untuk membebani kapasitas korban atau mengurangi kemampuan untuk melayani pemohon sah lainnya.

Kegagalan Komponen

Menjadi perangkat keras, komponen gagal; kegagalan ini cenderung sporadis, individual, tak terduga, dan tidak berbahaya. Seperti yang telah kami katakan, Internet cukup kuat sehingga dapat mengatasi kegagalan sebagian besar komponen. Dan penyerang biasanya tidak dapat menyebabkan kegagalan suatu komponen, sehingga masalah ini jarang disebabkan oleh serangan berbahaya. (Lihat Studi Studi Kasus 3-6 untuk penjelasan tentang apa yang tampaknya merupakan kegagalan perangkat keras yang disebabkan.) Namun demikian, teknisi keamanan perlu tetap memperhatikan potensi kerusakan sistem akibat kegagalan peralatan.

Selanjutnya kita beralih ke teknik yang digunakan penyerang untuk menentukan cara memasang serangan. Seperti pencuri yang mencari kerentanan di lingkungan sekitar, penyerang yang berhasil berniat melukai korban tertentu sering menghabiskan waktu untuk menyelidiki kerentanan dan pertahanan korban, dan merencanakan serangan yang sesuai. Investigasi ini bukanlah serangan itu sendiri, tetapi sesuatu yang berkontribusi pada metode dan peluang penyerang.

Pada bulan Juni 2010, fasilitas pengayaan nuklir di Iran terkena virus komputer Stuxnet yang kompleks dan canggih (disebutkan di Bab 3 dan dibahas di Bab 13). Stuxnet menargetkan sistem kontrol industri dengan memodifikasi kode pada pengontrol logika yang dapat diprogram (PLC) untuk membuatnya bekerja dengan cara yang diinginkan penyerang dan untuk menyembunyikan perubahan tersebut dari operator peralatan. Sistem yang digunakan Stuxnet adalah yang menggunakan pengontrol Siemens Simatic, tampaknya di pembangkit nuklir di Bushehr atau Natanz. Stuxnet menargetkan catu daya khusus yang digunakan untuk mengontrol kecepatan perangkat, seperti motor. Perintah yang dimodifikasi malware dikirim ke drive dari perangkat lunak Siemens SCADA, menyebabkan pengontrol mengubah kecepatan perangkat, membuatnya mengubah kecepatan sebentar-sebentar.

Stuxnet menargetkan drive tertentu yang berjalan pada kecepatan tinggi. Kecepatan tinggi tersebut hanya digunakan untuk aplikasi tertentu, salah satunya adalah pengayaan uranium. Menurut Eric Chien dari Symantec "Stuxnet mengubah frekuensi daya keluaran untuk waktu yang singkat menjadi 1410Hz dan kemudian ke 2Hz dan kemudian ke 1064Hz." Frekuensi normal motor adalah 1064 Hz; berlari pada kecepatan 1400 Hz bisa menghancurkan peralatan. Osilasi frekuensi liar seperti itu menyebabkan motor dipercepat, lalu melambat, dan kemudian dipercepat lagi. Pengayaan uranium membutuhkan sentrifugal berputar pada kecepatan yang tepat untuk waktu yang lama; mengubah kecepatan akan secara signifikan mengurangi kualitas produk yang diperkaya.

Memang, beberapa ahli luar berpikir sebanyak 1000 dari sekitar 8000 sentrifugal dalam program pengayaan Iran gagal pada 2009 hingga 2010, selama puncak operasi Stuxnet. Iran memproduksi sentrifugalnya sendiri, yang diketahui gagal secara teratur, meskipun mungkin tidak sebanyak 1000 dari 8000. Virus ini mungkin juga dimaksudkan untuk membuat para insinyur dan perancang pemeliharaan sibuk mengganti perangkat keras yang gagal dan mencari cara untuk menjaga agar seluruh sistem tetap berjalan. Stuxnet dapat berkontribusi pada tingkat kegagalan ini, mungkin contoh pertama dari serangan berbahaya yang menyebabkan kegagalan perangkat keras.

3.2.5 Pemindaian Port

Pemindaian adalah kegiatan inspeksi, dan karena itu tidak menyebabkan kerusakan itu sendiri (jika Anda tidak menganggap belajar tentang lawan Anda sebagai bahaya). Namun, pemindaian sering digunakan sebagai langkah pertama dalam serangan, penyelidikan, untuk menentukan serangan selanjutnya yang mungkin berhasil. Dengan demikian, selanjutnya kami akan memperkenalkan topik probing subnetwork untuk arsitektur dan eksposurnya.

Kerentanan dalam berbagai versi produk perangkat lunak sudah diketahui dengan baik: Vendor memposting daftar kekurangan dan tindakan protektif atau korektif (tambalan dan penyelesaian), dan profesional keamanan memelihara dan mendistribusikan daftar serupa, serta alat untuk menguji kerentanan. Peretas mengedarkan salinan kode dan skrip serangan. Masalah bagi penyerang adalah mengetahui serangan mana yang ditujukan ke mesin mana: Serangan terhadap versi tertentu dari Adobe Reader tidak akan bekerja jika mesin target tidak menjalankan Reader atau menjalankan versi yang tidak mengandung kerentanan tertentu. Mengirim serangan terhadap mesin yang tidak rentan setidaknya memakan waktu tetapi lebih buruk, bahkan mungkin membuat penyerang menonjol atau menjadi terlihat dan dapat diidentifikasi. Penyerang ingin menembakkan panah mereka hanya pada target yang mungkin.

Cara mudah untuk mengumpulkan informasi jaringan adalah dengan menggunakan pemindai port, sebuah program yang, untuk alamat Internet (IP) tertentu, melaporkan port mana yang merespons permintaan dan beberapa kerentanan yang diketahui tampaknya ada. Dan Farmer dan Wietse Venema [FAR90, FAR95] adalah orang-orang pertama yang menjelaskan teknik dalam alat COPS dan SATAN. Sejak itu, alat-alat seperti NESSUS dan Nmap telah memperluas konsep penyelidikan jaringan.

Pemindaian port mirip dengan pemeriksaan fisik rutin dari dokter, terutama pertanyaan awal yang digunakan untuk menentukan riwayat kesehatan. Pertanyaan dan jawaban itu sendiri mungkin tidak tampak signifikan, tetapi mereka menunjuk ke area yang menyarankan penyelidikan lebih lanjut.

Alat Pemindaian Port

Alat pemindaian port sudah tersedia, dan tidak hanya untuk komunitas bawah tanah. Pemindai Nmap, aslinya ditulis oleh Fyodor dan tersedia di www.insecure.org/nmap, adalah alat yang berguna yang dapat diunduh siapa saja. Diberikan sebuah alamat, Nmap akan melaporkan semua port yang terbuka, layanan yang didukung masing-masing, dan pemilik (ID pengguna) daemon yang menyediakan layanan tersebut. (Pemiliknya penting karena menyiratkan hak istimewa apa yang akan diberikan pada seseorang yang menyusup ke layanan itu. Administrator cenderung memberi nama akun istimewa dengan nama seperti admin atau sistem.)

Pemindai lain yang tersedia adalah netcat, yang ditulis oleh Hobbit, di www.l0pht.com/users/l0pht. Produk komersial sedikit lebih mahal, tetapi tidak mahal. Pemindai komersial yang terkenal adalah Nessus (Nessus Corp. [AND03]), Pemindai CyberCop (Rekanan Jaringan), Pemindai Aman (Cisco), dan Pemindai Internet (Sistem Keamanan Internet).

Hasil Pemindaian Port

Seperti dijelaskan sebelumnya dalam bab ini, port hanyalah penunjukan numerik untuk merutekan data ke program tertentu yang menunggunya. Program menunggu, yang disebut daemon atau demon, dikatakan mendengarkan port tertentu; pada kenyataannya, ia mendaftar dengan perangkat lunak manajemen jaringan sehingga menerima data yang ditujukan ke port tersebut. Misalnya, menurut konvensi, port 110 adalah nomor port yang terkait dengan Post Office Protocol untuk email, 80 didedikasikan untuk lalu lintas HTTP (halaman web), dan 123 ditetapkan ke Network Time Protocol untuk sinkronisasi jam. Seiring waktu jumlah layanan telah melebihi kisaran jumlah yang tersedia, sehingga terjadi tabrakan, penggunaan kembali, penggunaan informal, dan realokasi.

Mari kita lanjutkan pembahasan kita sebelumnya tentang permintaan data yang masuk pada port 110, Protokol Kantor Pos. Klien memulai permintaan untuk terhubung dengan server POP dengan protokol yang ditentukan yang diimplementasikan dalam perintah teks ASCII. Server merespons, biasanya mengidentifikasi dirinya sendiri dan terkadang nomor versinya (sehingga klien dan server dapat tersinkronisasi pada kemampuan dan harapan). Kami menunjukkan contoh pertukaran itu pada Gambar 3-11. Baris dari klien diberi label CL dan tanggapan dari server POP diberi label SV. Siapapun dapat memulai pertukaran seperti itu dengan menggunakan Telnet, program emulator terminal.

```
CL:   telnet incoming.server.net 110
SV:   +OK Messaging Multiplexor (Sun Java(tm) System Messaging Server
6.2-6.01 (built Apr  3 2006))
<4d3897ff.11ec04f8@vms108.mailsrvcs.net>
CL:   user v1
SV:   +OK password required for user v1@server.net
CL:   pass p1
SV:   -ERR [AUTH] Authentication failed
CL:   quit
SV:   +OK goodbye
```

Gambar 3-11 Pembuatan Sesi Server POP

Pemindai seperti Nmap menyelidiki berbagai port, menguji untuk melihat layanan apa yang merespons. Contoh output dari Nmap ditunjukkan pada Gambar 3-12. (Nama dan alamat situs telah diubah.) Perhatikan bahwa seluruh pemindaian hanya membutuhkan waktu 34 detik.

```
Nmap scan report
192.168.1.1 / somehost.com (online) ping results
address: 192.168.1.1 (ipv4)
hostnames: somehost.com (user)
The 83 ports scanned but not shown below are in state: closed
Port      State     Service Reason      Product  Version  Extra info
21   tcp  open      ftp     syn-ack     ProFTPD  1.3.1
22   tcp  filtered  ssh     no-response
25   tcp  filtered  smtp    no-response
80   tcp  open      http    syn-ack     Apache   2.2.3    (CentOS)
106  tcp  open      pop3pw  syn-ack     poppassd
110  tcp  open      pop3    syn-ack     Courier pop3d
111  tcp  filtered  rpcbind no-response
113  tcp  filtered  auth    no-response
143  tcp  open      imap    syn-ack     Courier Imapd      rel'd 2004
443  tcp  open      http    syn-ack     Apache  2.2.3      (CentOS)
465  tcp  open      unknown syn-ack
646  tcp  filtered  ldp     no-response
993  tcp  open      imap    syn-ack     Courier Imapd      rel'd 2004
995  tcp  open              syn-ack
2049 tcp  filtered  nfs     no-response
3306 tcp  open      mysql   syn-ack     MySQL   5.0.45
8443 tcp  open      unknown syn-ack
34 sec. scanned
1 host(s) scanned
1 host(s) online
0 host(s) offline
```

Gambar 3-12 Keluaran Pemindai Nmap

Pemindaian port memberi tahu penyerang tiga hal: port atau layanan standar mana yang berjalan dan merespons pada sistem target, sistem operasi apa yang diinstal pada sistem target, dan aplikasi serta versi aplikasi apa yang ada. Informasi ini tersedia untuk diminta dari sistem jaringan; itu dapat diperoleh secara diam-diam, tanpa nama, tanpa identifikasi atau otentikasi, menarik sedikit atau tidak sama sekali perhatian pada pemindaian.

Tampaknya nama sistem operasi atau versi aplikasi sistem tidak akan signifikan, tetapi mengetahui bahwa host tertentu menjalankan versi tertentu—yang mungkin berisi cacat yang diketahui atau bahkan tidak diungkapkan—dari suatu layanan, penyerang dapat merancang serangan untuk mengeksploitasi secara tepat kerentanan itu. Dengan demikian, pemindaian port dapat menjadi langkah pertama dalam serangan yang lebih serius.

Hal lain yang dapat dipelajari penyerang adalah konektivitas. Gambar 3-12 menyangkut satu host. Pada Gambar 3-13 kami telah memperluas pencarian ke seluruh subnetwork (sekali lagi, dengan mengubah nama dan alamat). Seperti yang Anda lihat, jaringan terdiri dari router, tiga komputer, dan satu perangkat tak dikenal.

```
Starting Nmap 5.21 (http://nmap.org) at 2015-00-00
Eastern Daylight Time

Nmap scan report for router (192.168.1.1)
Host is up (0.00s latency).
MAC Address: 00:11:22:33:44:55 (Brand 1}

Nmap scan report for computer (192.168.1.39)
Host is up (0.78s latency).
MAC Address: 00:22:33:44:55:66 (Brand 2)

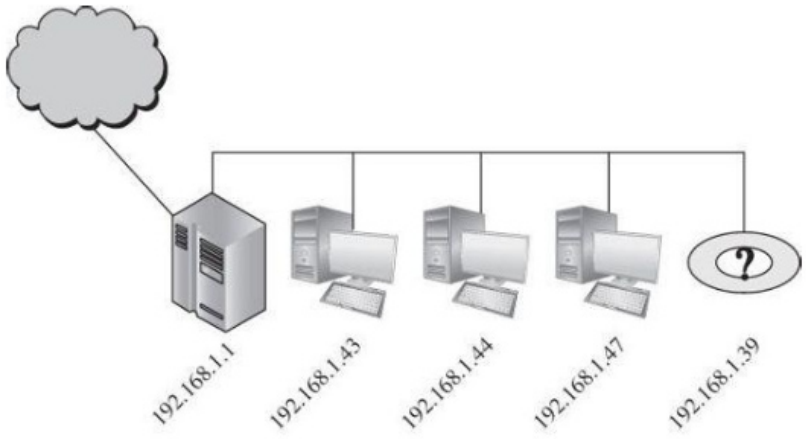
Nmap scan report computer (192.168.1.43)
Host is up (0.010s latency).
MAC Address: 00:11:33:55:77:99 (Brand 3)

Nmap scan report for unknown device 192.168.1.44
Host is up (0.010s latency).
MAC Address: 00:12:34:56:78:9A (Brand 4)

Nmap scan report for computer (192.168.1.47)
Host is up.
```

Gambar 3-13 Pemindaian Nmap dari Jaringan Kecil

Informasi dari Gambar 3-14 memberikan petunjuk penting lainnya: Karena waktu latensi (waktu antara saat sebuah paket dikirim ke perangkat dan perangkat merespons) untuk semua perangkat serupa, mereka mungkin berada di segmen jaringan yang sama. Dengan demikian, Anda dapat membuat sketsa diagram konektivitas jaringan (seperti yang ditunjukkan pada Gambar 3-14).



Gambar 3-14 Diagram Konektivitas Jaringan Kecil

Nmap memiliki banyak pilihan; orang luar dapat membuat sidik jari pemilik dan pengguna, mengidentifikasi layanan umum yang berjalan pada port yang tidak biasa, memetakan konektivitas (rute antar) mesin, atau menyimpulkan jenis perangkat yang tidak diketahui sebenarnya. Perhatikan bahwa dengan hanya beberapa perintah, penyerang dalam dua contoh yang ditampilkan belajar untuk mengetahui :

- Berapa banyak host yang ada
- Apa alamat IP mereka
- Apa alamat fisik (MAC) mereka
- Apa mereknya
- Sistem operasi apa yang dijalankan masing-masing, dan versi apa
- Port apa yang menanggapi permintaan layanan
- Aplikasi layanan apa yang merespons, dan program serta versi apa yang dijalankannya
- Berapa lama respons yang dibutuhkan (yang menunjukkan kecepatan berbagai koneksi jaringan dan dengan demikian dapat menunjukkan desain jaringan)

Untuk penyerang yang malas, Nmap bahkan memiliki opsi yang secara otomatis menghasilkan sejumlah alamat IP acak dan kemudian memindai alamat tersebut. Poin ini sangat penting untuk keamanan komputer. Jika penyerang ingin mengeksploitasi kerentanan yang diketahui dalam versi tertentu dari beberapa perangkat lunak, penyerang tidak perlu menjalankan serangan berulang kali terhadap banyak sistem yang menjalankan versi berbeda—atau bahkan perangkat lunak yang sama sekali berbeda. Sebagai gantinya, penyerang pertama-tama menjalankan pemindaian Nmap baik memilih, katakanlah, 10.000 alamat secara acak, atau memilih semua alamat dalam rentang tertentu, misalnya, 100.200.*.*. Ketika Nmap mengembalikan hasil dari semua pemindaian ini, penyerang dapat menggunakan editor teks sederhana untuk memilih dari output besar hanya baris yang mengidentifikasi versi perangkat lunak yang diinginkan.

Bahaya dari Pemindaian Port

Anda mungkin bertanya apa bahayanya mesin dan layanan yang diketahui seseorang; lagi pula, alasan port terbuka adalah untuk bertukar data. Pemindai hanya mengambil data yang diungkapkan mesin secara sukarela.

Pikirkan alih-alih dua rumah di lingkungan yang dikepung pencuri. Dia tidak tahu apa-apa tentang rumah pertama. Mengenai rumah kedua, dia tahu dua orang tinggal di sana, kamar tidur mereka ada di lantai atas. Pasangan itu memiliki seekor anjing, yang tidur di ruang bawah tanah di balik pintu yang tertutup. Mereka selalu membiarkan jendela belakang terbuka sedikit agar kucing bisa masuk dan keluar. Dan salah satu penghuni baru-baru ini keseleo pergelangan kakinya, jadi dia bergerak perlahan dan dengan sedikit rasa sakit. Jelas rumah kedua lebih menarik bagi pencuri, sebagian karena dia dapat merencanakan serangan yang memanfaatkan kerentanan yang diketahui di rumah itu. Dengan demikian, mengekspos karakteristik sistem komputasi yang tidak perlu dapat berbahaya.

Pemindai jaringan dan kerentanan, di mana Nmap hanyalah salah satu contohnya, memiliki dua tujuan, satu baik dan satu buruk. Penggunaan yang baik adalah oleh administrator jaringan atau pemilik sistem yang akan menjelajahi jaringan mereka dengan alat ini. Alat ini akan melaporkan perangkat mana yang mungkin kedaluwarsa dan versi perangkat lunak yang rentan yang harus

ditingkatkan atau port mana yang tidak perlu diekspos dan harus ditutup. Administrator jaringan besar dapat menggunakan pemindai untuk mendokumentasikan dan meninjau semua perangkat yang terhubung ke jaringan (karena perangkat baru dapat ditambahkan ke jaringan kapan saja). Tapi tentu saja, seperti yang telah kami tunjukkan, penggunaan pemindai jaringan yang buruk adalah memungkinkan penyerang mempelajari suatu sistem. (Hukum tidak ditetapkan apakah memindai komputer tanpa izin adalah ilegal.) Karena pentingnya penggunaan yang baik, perusahaan perangkat lunak komersial yang sehat terus meningkatkan penggunaan dan kegunaan pemindai jaringan yang, sayangnya, juga mendukung penggunaan yang buruk.

Pemindaian port sulit untuk diklasifikasikan. Mereka tentu saja adalah alat yang banyak digunakan oleh penyerang jaringan sebagai langkah pertama dalam upaya yang lebih serius. Apakah mereka kerentanan? Tidak; kerentanannya terletak pada jumlah dan jenis informasi yang diizinkan oleh administrator jaringan untuk diekspor ke program apa pun yang diminta. Apakah mereka ancaman? Tidak juga, karena bukaan yang mereka laporkan tersedia dengan atau tanpa pemindaian port. Haruskah mereka dilarang dengan cara tertentu? Mungkin sudah terlambat untuk tindakan itu, terutama karena setiap programmer yang kompeten dengan pengetahuan dasar tentang protokol jaringan dapat dengan mudah menulis yang dasar. Jadi, paling baik kita dapat mengatakan bahwa teknik pemindaian port ada, dan administrator jaringan harus menggunakan pemindai port itu sendiri untuk menentukan seberapa banyak orang luar dapat mempelajari jaringan mereka. Pemindai port sendiri tidak menyebabkan penolakan layanan atau kegagalan jaringan lainnya, tetapi mereka memfasilitasi dan sering mempercepatnya.

3.2.6 Ringkasan Kerentanan

Seperti yang ditunjukkan oleh contoh yang baru saja disajikan, banyak serangan terhadap infrastruktur jaringan area luas dapat menyebabkan intersepsi, modifikasi, dan penolakan layanan. Karena serangan ini bekerja terhadap jaringan besar, mereka jarang digunakan terhadap satu pengguna tertentu, yang mungkin sulit untuk diisolasi di alam semesta jutaan komunikasi bersamaan. (Seperti yang akan kami jelaskan nanti dalam bab ini, serangan penolakan layanan dapat, dan sering kali, ditujukan terhadap satu korban tertentu.)

Di bagian berikutnya, kami mengeksplorasi bagaimana trik serupa dapat digunakan di jaringan lokal nirkabel, di mana hanya segelintir pengguna yang memungkinkan untuk memfokuskan serangan hanya pada satu. Perhatikan bahwa jaringan ini masih dapat terhubung ke jaringan area yang lebih luas seperti Internet. Jadi aktivitas penuh satu pengguna masih terbuka untuk intersepsi dan modifikasi; titik intrusi hanya berbatasan langsung dengan pengguna.

3.3 Keamanan Jaringan Nirkabel

Pada bagian ini kami menyajikan teknologi jaringan nirkabel. Kami kemudian menjelaskan dua pendekatan untuk mengamankan jaringan ini. Yang pertama secara luas diakui sebagai kegagalan keamanan. Mempelajari upaya yang gagal ini harus menghasilkan wawasan mengapa mengintegrasikan keamanan sulit untuk teknologi yang ada dengan kendala non-keamanan. Diungkapkan secara berbeda, kisah ini adalah contoh utama mengapa insinyur keamanan memohon untuk dimasukkan dalam desain sejak awal: Menambahkan keamanan setelah desain diperbaiki jarang berhasil. Namun, dari cerita ini Anda dapat melihat apa yang seharusnya atau bisa telah diramalkan dan ditangani.

Pendekatan kedua lebih baik, tetapi juga memiliki keterbatasan keamanan. Dalam contoh ini Anda dapat melihat bahwa bahkan dengan contoh jebakan keamanan yang harus dihindari, menyusun pendekatan yang berhasil memerlukan pertimbangan yang cermat terhadap kemungkinan titik kegagalan.

3.3.1 Latar Belakang WiFi

Lalu lintas nirkabel menggunakan bagian dari spektrum radio, sehingga sinyal tersedia bagi siapa saja yang memiliki antena efektif dalam jangkauan. Karena komputasi nirkabel sangat terbuka, diperlukan tindakan untuk melindungi komunikasi antara komputer (disebut klien) dan stasiun pangkalan nirkabel atau titik akses. Mengingat bahwa semua komunikasi ini berada pada frekuensi radio yang telah ditentukan, Anda dapat mengharapkan penyerang menguping untuk mencoba mencegat dan menyamar. Bagian yang harus dilindungi adalah menemukan titik akses, mengautentikasi komputer jarak jauh ke titik akses, dan sebaliknya, dan melindungi aliran komunikasi. Komunikasi nirkabel tidak akan pernah seaman kabel, karena sinyal yang terpapar lebih rentan. Komunikasi nirkabel memiliki kerentanan lain, seperti yang terkait di Studi Kasus 3-7.

Studi Kasus 3-7 : Intersepsi Nirkabel

The New Zealand Herald melaporkan bahwa sebuah perusahaan telekomunikasi besar terpaksa menutup layanan email selulernya karena kelemahan keamanan dalam perangkat lunak jaringan nirkabelnya. Cacat tersebut memengaruhi pengguna di jaringan perusahaan yang mengirim email di ponsel mereka yang mendukung WAP (protokol aplikasi nirkabel).

Kerentanan terjadi ketika pengguna menyelesaikan sesi email. Faktanya, perangkat lunak tidak mengakhiri sesi WAP selama 60 detik lagi. Jika pelanggan jaringan kedua memulai sesi email dalam 60 detik tersebut dan terhubung ke port yang sama dengan pelanggan pertama, pelanggan kedua kemudian dapat melihat pesan pelanggan pertama.

Perusahaan menyalahkan perangkat lunak pihak ketiga yang disediakan oleh portal seluler. Namun demikian, perusahaan telekomunikasi itu merasa sangat malu,

terutama karena "menganggap masalah keamanan dengan jaringan nirkabel" sebagai "faktor utama yang mengancam untuk menghambat perkembangan teknologi [nirkabel]."

Siapa pun yang memiliki kartu jaringan nirkabel dapat mencari jaringan yang tersedia. Konsultan keamanan Chris O'Ferrell telah dapat terhubung ke jaringan nirkabel di Washington D.C. dari luar gedung kantor Senat, Mahkamah Agung, dan Pentagon [NOG02]; yang lain bergabung dengan jaringan di bandara, di pesawat, dan di kedai kopi. Produk Pengamat dari Instrumen Jaringan dan Penganalisis Keamanan Nirkabel IBM dapat menemukan koneksi nirkabel terbuka di jaringan sehingga administrator keamanan dapat mengetahui bahwa jaringan dapat diakses untuk akses nirkabel.

Dan kemudian beberapa pengguna LAN nirkabel menolak untuk mematikan atau melindungi layanan mereka. Pengecer BestBuy dipermalukan oleh pelanggan yang membeli produk nirkabel; saat di tempat parkir, dia memasangnya di komputer laptopnya. Sangat mengejutkan, dia menemukan dia bisa terhubung ke jaringan nirkabel toko. BestBuy kemudian membuat semua mesin kasir nirkabelnya offline. Tetapi rantai apotek CVS mengumumkan rencana untuk melanjutkan penggunaan jaringan nirkabel di semua 4100 tokonya, dengan alasan “Kami menggunakan teknologi nirkabel secara ketat untuk manajemen item internal. Jika kami pernah bergerak ke arah transmisi informasi [pelanggan] melalui LAN nirkabel di dalam toko, kami akan mengenkripsi data” [BRE02a]. Dalam banyak kasus, tidak ada yang mengingat niat awal untuk melindungi data ketika seseorang mengubah aplikasi bertahun-tahun kemudian.

Komunikasi Nirkabel

Untuk menghargai bagaimana keamanan diterapkan pada komunikasi nirkabel dan di mana ia dapat gagal, Anda perlu mengetahui struktur umum komunikasi data nirkabel. Komunikasi data nirkabel (dan juga kabel) diimplementasikan melalui serangkaian pertukaran yang teratur yang disebut protokol. Kami menggunakan protokol dalam kehidupan sehari-hari untuk mencapai pertukaran sederhana. Misalnya, protokol yang sudah dikenal melibatkan membuat dan menerima panggilan telepon.

Jika Anda menelepon teman, Anda melakukan versi langkah-langkah ini:

1. Menekan tombol untuk mengaktifkan telepon Anda.
2. Menekan tombol untuk memilih dan mengirimkan nomor teman (proses yang dulu disebut panggilan telepon).
3. Teman Anda mendengar nada dan menekan tombol untuk menerima panggilan Anda.
4. Teman Anda mengatakan “halo”, atau sapaan lainnya.
5. Anda menyapa.
6. Memulai percakapan.

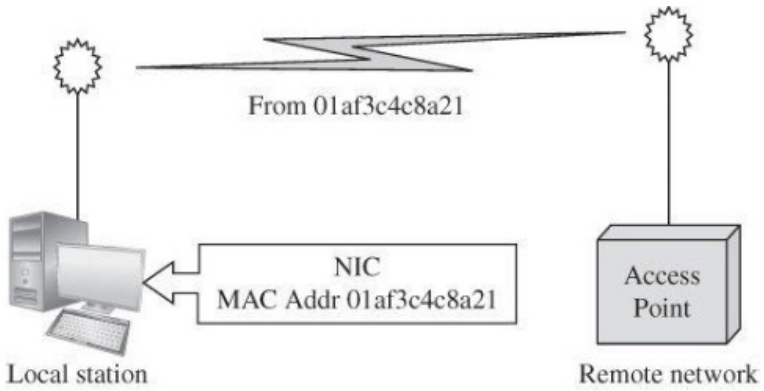
Proses ini tidak berfungsi jika Anda mulai berbicara sebelum teman Anda mendengar dan menjawab telepon, atau jika teman Anda menerima panggilan Anda tetapi tidak pernah mengatakan apa pun. Keenam langkah ini harus diikuti secara berurutan dan dalam bentuk umum ini untuk proses sederhana membuat panggilan telepon berfungsi. Kita semua mempelajari dan menggunakan protokol ini tanpa memikirkan prosesnya, tetapi polanya membantu kita berkomunikasi dengan mudah dan efisien.

Protokol serupa mengatur seluruh proses komunikasi WiFi. Anda dapat menggunakan komputer Anda, yang dibuat di satu negara dengan perangkat lunak yang ditulis di negara lain, untuk terhubung ke titik akses nirkabel di seluruh dunia karena protokol ini merupakan standar yang disepakati secara internasional, yang disebut rangkaian protokol 802.11. Kami sekarang menyajikan poin-poin penting dari
Protokol 802.11 yang penting untuk keamanan.

Suite Protokol 802.11

Semua protokol 802.11 menjelaskan bagaimana perangkat berkomunikasi dalam pita sinyal radio 2,4 GHz (pada dasarnya 2,4 GHz–2,5 GHz) yang dialokasikan untuk WiFi. Pita dibagi menjadi 14 saluran atau subrentang di dalam pita; saluran ini tumpang tindih untuk menghindari gangguan dengan perangkat di sekitar. Perangkat WiFi dirancang untuk hanya menggunakan beberapa saluran, sering kali saluran 1, 6, dan 11. Sinyal nirkabel dapat menempuh jarak hingga 100 meter (300 kaki), meskipun kualitas sinyal berkurang seiring dengan jarak, dan mengganggu objek seperti dinding dan pohon juga mengganggu komunikasi. Protokol 802.11n meningkatkan jangkauan, dan perangkat yang disebut repeater dapat memperluas jangkauan pemancar nirkabel yang ada.

Seperti yang ditunjukkan pada Gambar 3-15, jaringan nirkabel terdiri dari titik akses atau router yang menerima, meneruskan dan mengirimkan data, dan satu atau lebih perangkat, kadang-kadang disebut stasiun, seperti komputer atau printer, yang berkomunikasi dengan titik akses. Titik akses adalah hub dari subjaringan nirkabel. Setiap perangkat harus memiliki kartu antarmuka jaringan, atau NIC, yang mengomunikasikan sinyal radio dengan titik akses. NIC diidentifikasi oleh alamat perangkat keras 48 atau 64-bit unik yang disebut kode akses menengah, atau MAC. (Alamat MAC seharusnya tetap dan unik, tetapi seperti yang akan kami jelaskan nanti dalam bab ini, alamat MAC dapat diubah.) Untuk melihat penyalahgunaan alamat MAC untuk otentikasi, lihat Studi Kasus 3-8.



Gambar 3-15 Stasiun Lokal Berkomunikasi dengan Jaringan Jarak Jauh

Jangkauan Akses WiFi

Jarak merupakan pertimbangan penting dengan WiFi, tetapi sulit untuk menyatakan dengan tepat. Sinyal nirkabel menurun karena gangguan dari objek yang mengganggu, seperti dinding, mesin, dan pohon, serta jarak; penerima tidak akan membuat, atau mungkin memutuskan, koneksi dengan sinyal yang buruk, yang lemah atau kehilangan banyak data. Sinyal luar ruangan, dengan lebih sedikit objek yang mengganggu, umumnya menempuh jarak yang lebih jauh daripada sinyal dalam ruangan.

Studi Kasus 3-8 : Menggunakan Alamat MAC untuk Otentikasi [Ide Buruk]

Dalam apa yang kami harapkan adalah tipuan, sebuah posting yang diduga dari departemen layanan TI Universitas Harvard menunjukkan bahwa Harvard akan mulai menggunakan alamat MAC untuk otentikasi. (http://video2.harvard.edu/wireless/Wireless_Registration_Procedure_072910.pd Pengumuman menyatakan bahwa setelah mendaftar dengan layanan jaringan Harvard, mesin siswa akan dikenali oleh alamat MAC dan siswa tidak perlu lagi memasukkan ID dan PIN Harvard untuk mengakses jaringan nirkabel Harvard. Postingan itu ada di server web Harvard yang tidak jelas, bukan halaman layanan TI utama, dan tampaknya tidak disebutkan di tempat lain di situs web Harvard.

Seperti yang baru saja kami laporkan, programmer jaringan yang cukup terampil dapat mengubah alamat MAC, dan program yang disebut sniffer melaporkan alamat MAC perangkat yang berpartisipasi dalam jaringan nirkabel. Dengan demikian, siapa pun yang ingin menggunakan jaringan WiFi Harvard dapat dengan mudah mendapatkan akses terotentikasi dengan mengendus alamat MAC dari sesi yang sedang berlangsung dan mengatur kartu NIC untuk menampilkan alamat tersebut.

Mungkin situs web ini adalah lelucon dari saingan terdekat Harvard, M.I.T.?

Di sisi lain, antena dapat disetel ke frekuensi komunikasi nirkabel. Memfokuskan langsung pada sumber sinyal juga dapat meningkatkan penerimaan pada jarak yang jauh. Pada Tabel 3-2 kami memperkirakan beberapa rentang yang masuk akal untuk protokol WiFi yang berbeda. Hasil eksperimen dengan 802.11n telah menunjukkan penerimaan pada jarak sekitar 5000 kaki/1600 m dalam kondisi ideal.

Tabel 3-2 Rentang Akses Protokol 802.11 yang Khas

Protocol	Ordinary Signal Range
802.11a	100 ft / 35 m
802.11b	300 ft / 100 m
802.11g	300 ft / 100 m
802.11n	1000 ft / 350 m

Sebagian besar komputer berkemampuan WiFi sekarang berkomunikasi pada protokol 802.11n (dan untuk kompatibilitas pada semua yang sebelumnya juga), sehingga jangkauannya mudah dari satu rumah ke jalan, dan bahkan beberapa rumah jauhnya di lingkungan konvensional. Seperti yang dijelaskan di Studi Kasus 3-9, Google memulai proyek petualangan untuk memetakan konektivitas WiFi di seluruh dunia. Tujuan pemetaan ini tidak jelas, tetapi Uni Eropa menetapkan bahwa Google telah melampaui batas dalam mengumpulkan data ini.

Studi Kasus 3-9 : Proyek Street View Google

Proyek Street View Google, diluncurkan pada 2007, melibatkan mobil dengan kamera yang mengemudikan jalan di berbagai kota untuk mengambil foto pemandangan jalanan. Gambar-Gambar ini digabungkan dengan koordinat GPS untuk memperbaiki setiap Gambar ke lokasi fisiknya.

Menurut Pusat Informasi Privasi Elektronik [EPI10], saat memotret pemandangan di sepanjang jalan ini, mobil Google juga mengoperasikan pemindai dan penerima jaringan nirkabel, menjalankan program untuk memilih lalu lintas jaringan tidak terenkripsi yang ditemui, dan menulis konten itu ke disk komputer, bersama dengan koordinat GPS di mana sinyal itu diterima. Beberapa data itu termasuk kata sandi akun email dan pesan email. Google juga mencegat dan menyimpan pengidentifikasi perangkat jaringan (alamat MAC) dan pengidentifikasi stasiun jaringan nirkabel (SSID) dari jaringan nirkabel yang terdeteksi dari jalanan. Alamat nirkabel yang digabungkan dengan lokasi fisik dapat digunakan untuk menayangkan iklan bertarget. Audit independen program, ditugaskan oleh Google, mendokumentasikan analisis sintaksis dari data yang dikumpulkan untuk dapat menyimpan bidang individual.

Pengumpulan data beroperasi dari tahun 2007 hingga Mei 2010, ketika Google mengumumkan telah keliru mengumpulkan 600 MB data konten nirkabel. Meskipun audit menetapkan bahwa item data yang diambil telah diuraikan untuk memisahkan jenis data yang dienkripsi dan yang tidak dienkripsi, Google berpendapat bahwa penulisan dan penyimpanan data adalah kesalahan.

Pada tahun 2013 Google menyetujui penyelesaian gugatan hukum sebesar $7 juta yang diajukan oleh 37 negara bagian di Amerika Serikat (selain denda $25.000, Google membayar pemerintah AS atas klaim yang dengan sengaja menghalangi penyelidikan terhadap klaim pelanggaran privasi dalam aktivitas itu. Dan pada tahun 2011 ia membayar denda 100.000 € (sekitar $150.000 AS). Pada tahun 2013 Jerman mendenda Google 145.000 € (sekitar $200.000 AS), dan Google membayar Italia 1 juta € (sekitar $1,4 juta AS) atas pelanggaran privasi proyek ini. (Pendapatan kotor Google untuk tahun 2013 adalah $33 miliar AS, jadi jumlah denda ini kurang dari 0,1 persen dari pendapatan Google untuk tahun tersebut.)

Orang dapat berargumen bahwa Google hanya mendengarkan gelombang radio publik, yang diperlihatkan kepada siapa pun dengan penerima yang sesuai.

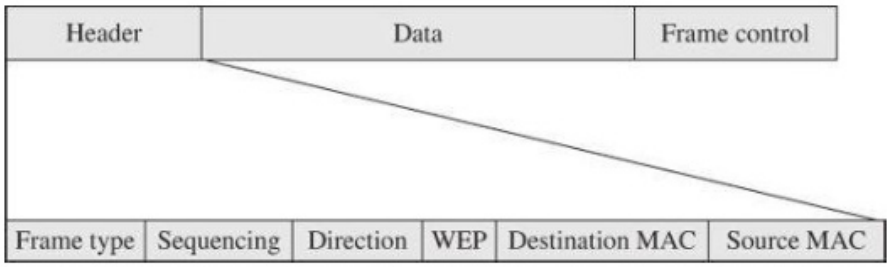
Perpanjangan dari argumen ini adalah bahwa gelombang udara ini tidak lebih terlindungi daripada gelombang suara atau Gambar visual: Saat Anda berbicara di depan umum, Anda tidak memiliki harapan bahwa percakapan Anda bersifat pribadi, dan Anda tahu fotografer amatir dapat menangkap Gambar Anda saat Anda berada di depan tengara mereka menembak. Argumen tandingannya adalah karena berbagai tindakan keamanan yang Anda terapkan, Anda bermaksud agar akses komputer Anda bersifat pribadi. Aspek hukum dari situasi ini kemungkinan akan diperdebatkan untuk beberapa waktu.

Bingkai WiFi

Setiap unit data WiFi disebut bingkai. Setiap frame berisi tiga bidang: header MAC, payload, dan FCS (urutan pemeriksaan bingkai). Header MAC berisi bidang berukuran tetap, termasuk:

- jenis bingkai: kontrol, manajemen, atau data
- ToDS, FromDS: arah frame ini: ke atau dari titik akses
- bit kontrol fragmentasi dan ketertiban
- WEP (privasi setara kabel) atau bit enkripsi: enkripsi, dijelaskan segera
- hingga empat alamat MAC (pengidentifikasi perangkat fisik): alamat pengirim dan penerima, ditambah dua alamat opsional untuk titik pemfilteran lalu lintas

Payload atau badan bingkai adalah data aktual yang ditransmisikan, 0–2304 byte yang strukturnya bergantung pada aplikasi yang menangani data. Urutan pemeriksaan frame adalah pemeriksaan integritas (sebenarnya pemeriksaan redundansi siklik, yang kami jelaskan di Bab 2) untuk memastikan transmisi yang akurat dari seluruh frame. Format bingkai WiFi ditunjukkan pada Gambar 3-16.



Gambar 3-16 Format Bingkai WiFi

Bingkai Manajemen

Dari ketiga jenis bingkai, bingkai manajemen adalah yang paling penting sekarang karena mereka mengontrol pembentukan dan penanganan serangkaian aliran data. Jenis bingkai manajemen yang paling signifikan adalah sebagai berikut:

- Suar. Setiap titik akses secara berkala mengirimkan bingkai suar untuk mengumumkan kehadirannya dan menyampaikan informasi, seperti stempel waktu, pengidentifikasi, dan parameter lain terkait titik akses. Setiap NIC yang berada dalam jangkauan menerima suar ini. Saat Anda terhubung ke layanan WiFi, misalnya, di kedai kopi, komputer Anda menerima sinyal suar dari toko untuk dapat memulai komunikasi. Sinyal suar mengiklankan jaringan yang menerima koneksi.

- Autentikasi. NIC memulai permintaan untuk berinteraksi dengan titik akses dengan mengirimkan identitasnya dalam bingkai otentikasi. Titik akses dapat meminta data otentikasi tambahan dan akhirnya menerima atau menolak permintaan tersebut. Salah satu pihak mengirimkan frame deauthentication untuk mengakhiri interaksi yang sudah ada. Jadi, misalnya, komputer Anda merespons sinyal suar kedai kopi dengan mengembalikan identitasnya (alamat MAC) dalam bingkai otentikasi. NIC meminta koneksi dengan mengirimkan bingkai otentikasi.

SSID

- Permintaan dan tanggapan asosiasi. Setelah otentikasi, NIC meminta titik akses untuk membuat sesi, yang berarti bahwa NIC dan titik akses bertukar informasi tentang kemampuan mereka dan menyetujui parameter interaksi mereka. Bagian penting dari pembentukan asosiasi adalah menyetujui enkripsi. Misalnya, titik akses mungkin dapat menangani tiga algoritma enkripsi yang berbeda, sebut saja A, B, dan C, dan NIC yang meminta hanya dapat menangani dua algoritma, sebut saja B dan D. Dalam asosiasi, keduanya akan menentukan bahwa mereka berbagi algoritma B dan dengan demikian setuju untuk menggunakan bentuk enkripsi tersebut untuk berkomunikasi. Permintaan deasosiasi adalah permintaan untuk mengakhiri sesi.

Salah satu nilai data penting lainnya dalam komunikasi WiFi adalah penunjukan titik akses sehingga perangkat nirkabel dapat membedakan antara titik akses jika menerima lebih dari satu sinyal. Service Set Identifier, atau SSID, adalah identifikasi titik akses; ini adalah string hingga 32 karakter yang dipilih oleh administrator titik akses. SSID adalah pengidentifikasi yang disiarkan titik akses dalam suarnya, dan tautan yang sedang berlangsung mengikat komunikasi NIC terkait ke titik akses yang diberikan. Misalnya, antena nirkabel komputer Anda mungkin mengambil tiga suar: CoffeeShop, Apt203, dan Quicksand.

Jelas SSID harus unik di area tertentu untuk membedakan satu jaringan nirkabel dari yang lain. Untuk titik akses nirkabel versi awal, default yang dipasang dari pabrik, seperti "nirkabel", "tsunami", atau "Linksys" (nama merek), tidak unik; sekarang sebagian besar default pabrik adalah nomor seri yang unik untuk perangkat.

Dengan latar belakang bagaimana komunikasi nirkabel terjadi, kita dapat mulai mengeksplorasi beberapa kerentanan.

3.3.2 Kerentanan di Jaringan Nirkabel

Jaringan nirkabel tunduk pada ancaman terhadap kerahasiaan, integritas, dan ketersediaan seperti halnya aplikasi dan teknologi komputer lainnya. Penyerang dapat bergabung dengan jaringan target dan berpartisipasi dalam pertukaran data, atau hanya mengamati lalu lintas sebagai pengamat.

Kerahasiaan

Tentu saja, jika sinyal data ditransmisikan di tempat terbuka, penerima yang tidak diinginkan mungkin bisa mendapatkan data. Nilai data itu sendiri adalah yang paling sensitif, tetapi komunikasi A dengan titik akses B atau durasi atau volume komunikasi mungkin juga sensitif. Sifat lalu lintas, apakah akses halaman web, jaringan peer-to-peer, email, atau manajemen jaringan, juga dapat dirahasiakan. Terakhir, mode di mana dua unit berkomunikasi—dienkripsi atau tidak dan jika dienkripsi, dengan

algoritma apa—berpotensi sensitif. Dengan demikian, kerahasiaan banyak aspek komunikasi dapat menjadi sensitif.

Integritas

Mengenai integritas, kita harus mempertimbangkan sumber masalah yang berbahaya dan tidak berbahaya. Banyak sumber bahaya yang tidak berbahaya termasuk gangguan dari perangkat lain, kehilangan atau kerusakan sinyal karena jarak atau objek yang mengganggu, masalah penerimaan yang disebabkan oleh cuaca, dan kegagalan komunikasi sporadis dalam perangkat keras dan perangkat lunak yang menerapkan komunikasi protokol.

Kelas pelanggaran integritas yang lebih menarik melibatkan serangan langsung dan berbahaya untuk mengubah konten komunikasi. Untuk komunikasi yang tidak terenkripsi, penyerang mungkin mencoba memalsukan data yang tampaknya berasal dari host atau klien. Karena klien dan server dapat menerima sinyal satu sama lain, penyerang tidak dapat dengan mudah menerima sesuatu dari klien, memodifikasinya, dan mengirimkan versi yang dimodifikasi sebelum sinyal asli klien sampai ke server. Namun, penyerang dapat mencoba mengambil alih aliran komunikasi dengan paksa. Penerima radio WiFi yang menerima dua sinyal lebih memilih yang lebih kuat. Jadi, jika titik akses jahat memotong sinyal dari klien dan mengirimkan sinyal yang kuat kembali, tampak seperti datang dari titik akses server, penjahat mungkin dapat menguasai aliran komunikasi.

Ketersediaan

Ketersediaan melibatkan tiga masalah potensial. Pertama, yang paling jelas, terjadi ketika komponen komunikasi nirkabel berhenti bekerja karena perangkat keras gagal, daya terputus, atau bencana lainnya menyerang. Masalah ketersediaan kedua adalah hilangnya sebagian tetapi tidak semua akses, biasanya dimanifestasikan sebagai layanan yang lambat atau menurun. Layanan dapat menjadi lambat karena gangguan, misalnya, jika daun pohon tertiup angin mengganggu transmisi bingkai, sehingga pihak penerima menyadari hilangnya beberapa data dan harus meminta dan menunggu pengiriman ulang. Layanan juga bisa lambat jika permintaan layanan melebihi kapasitas pihak penerima, sehingga beberapa permintaan layanan dibatalkan atau penerima menangani semua permintaan dengan lambat.

Komunikasi nirkabel juga mengakui masalah ketiga: kemungkinan koneksi jaringan yang salah. Beberapa titik akses WiFi dikenal sebagai hot spot publik dan sengaja tersedia untuk siapa saja yang ingin terhubung. Tetapi pemilik pribadi lainnya tidak ingin berbagi akses mereka dengan siapa pun dalam jangkauan. Meskipun layanan bersama mungkin tidak diperhatikan, itu tetap tidak pantas. Seorang pengguna yang menginginkan akses Internet gratis seringkali bisa mendapatkannya hanya dengan mencari LAN nirkabel yang menawarkan layanan DHCP. Gratis tidak selalu berarti aman, seperti yang dijelaskan di Studi Kasus 3-10. Dalam hal ini, meskipun layanan tersedia, keamanan layanan tersebut mungkin terbatas. Seperti kata pepatah, terkadang Anda mendapatkan apa yang Anda bayar.

Mencari jaringan nirkabel terbuka dalam jangkauan disebut war driving. Untuk menemukan jaringan terbuka, Anda hanya memerlukan komputer yang dilengkapi dengan penerima jaringan nirkabel. Mirip dengan acara pengamatan burung, empat acara World Wide War Driving diadakan (lihat http://www.worldwidewardrive.org/), dua pada tahun 2002, dan masing-masing satu pada tahun 2003 dan 2004. Tujuannya adalah untuk mengidentifikasi sebanyak mungkin nirkabel terbuka yang berbeda. titik akses mungkin dalam satu minggu: Untuk pencarian pertama, 9.374 ditemukan; terakhir, jumlahnya bertambah menjadi 228.537. Jumlahnya tidak sebanding karena seiring penyebaran kata, semakin banyak orang yang terlibat dalam pencarian situs. Untuk masing-masing dari empat peristiwa, sekitar dua pertiga dari situs yang ditemukan tidak mendukung komunikasi terenkripsi. Juga sekitar 30 persen titik akses di setiap peristiwa menggunakan SSID default (pengidentifikasi yang digunakan untuk mengakses titik akses). Biasanya (pada 2002–2004), SSID default adalah sesuatu seperti “nirkabel.” Pemancar nirkabel dengan SSID default dan tanpa enkripsi sama dengan kotak yang mengatakan "inilah saya, silakan gunakan jaringan nirkabel saya."

Saat membantu seorang teman mengatur jaringan rumahnya di Amerika Serikat, seorang konsultan memiliki laptop berkemampuan nirkabel. Ketika dia memindai untuk menemukan titik akses temannya (aman), dia menemukan lima lainnya cukup dekat untuk mendapatkan sinyal yang baik; tiga berjalan tanpa jaminan, dan dua dari ketiganya memiliki SSID yang cukup jelas untuk ditebak dengan mudah dari tetangga mana mereka berasal.

Hanya karena jaringan tersedia tidak berarti aman. Titik akses jahat adalah cara lain untuk mencegat informasi sensitif. Yang harus Anda lakukan adalah menyiarkan titik akses terbuka di kedai kopi atau di dekat gedung perkantoran besar, memungkinkan orang untuk terhubung, dan kemudian menggunakan sniffer jaringan untuk menyalin lalu lintas secara diam-diam. Sebagian besar situs komersial menggunakan enkripsi (seperti algoritme SSL, yang akan kami jelaskan nanti dalam bab ini) saat memperoleh informasi sensitif, sehingga identifikasi keuangan atau pribadi pengguna tidak boleh diekspos. Tetapi banyak jenis data lain, seperti kata sandi atau pesan email, terbuka untuk diambil.

Daya tarik war driving telah berkurang karena beberapa alasan. Pertama, meningkatnya hotspot WiFi publik gratis di kedai kopi, toko buku, hotel, perpustakaan, dan tempat serupa telah mengurangi motivasi untuk mencari sinyal WiFi. Kedua, risiko menghubungkan ke titik akses tidak aman tinggi: Beberapa koneksi WiFi tidak aman adalah magnet yang disengaja untuk memikat klien yang tidak menaruh curiga untuk mencegat data sensitif dari koneksi nirkabel. Akhirnya, karena banyak orang memiliki telepon seluler yang mendukung Internet, mereka menggunakan telepon untuk akses singkat alih-alih komputer dengan WiFi. Dengan demikian, aktivitas penggerak perang untuk menemukan dan memetakan titik akses nirkabel sebagian besar telah berhenti.

Tetapi apakah sah untuk terhubung dengan sinyal nirkabel yang terdeteksi? Dalam kasus terpisah, Benjamin Smith III di Florida pada Juli 2005 dan Dennis Kauchak di Illinois pada Maret 2006 dihukum karena mengakses komputer dari jarak jauh tanpa izin pemiliknya. Kauchak dijatuhi hukuman denda $250. Jadi, meskipun Anda dapat terhubung, mungkin tidak sah untuk melakukannya.

Dengan tiga area keamanan yang mungkin gagal ini, selanjutnya kita akan melihat serangan dan penanggulangan nirkabel tertentu.

Akses WiFi Tidak Resmi

Perangkat yang tidak sah dapat mencoba membuat asosiasi dengan titik akses. Ingat dari protokol WiFi bahwa akses pada dasarnya melibatkan tiga langkah:

1. Jalur akses menyiarkan ketersediaannya dengan mengirimkan suar, undangan bagi perangkat untuk terhubung dengannya.
2. NIC perangkat merespons dengan permintaan untuk mengautentikasi, yang diterima oleh titik akses.
3. NIC perangkat meminta pembentukan asosiasi, yang dinegosiasikan dan diterima oleh titik akses.

Ada ancaman di masing-masing titik ini. Pada langkah 1, siapa pun dapat mengambil dan membalas suar siaran. Pada langkah 2, otentikasi tidak ketat; dalam mode WiFi dasar, titik akses menerima perangkat apa pun, tanpa otentikasi. Pada langkah 3, titik akses mana pun dapat menerima asosiasi dengan perangkat apa pun. Kami dapat melawan serangan akses tidak sah ini di salah satu dari tiga langkah.

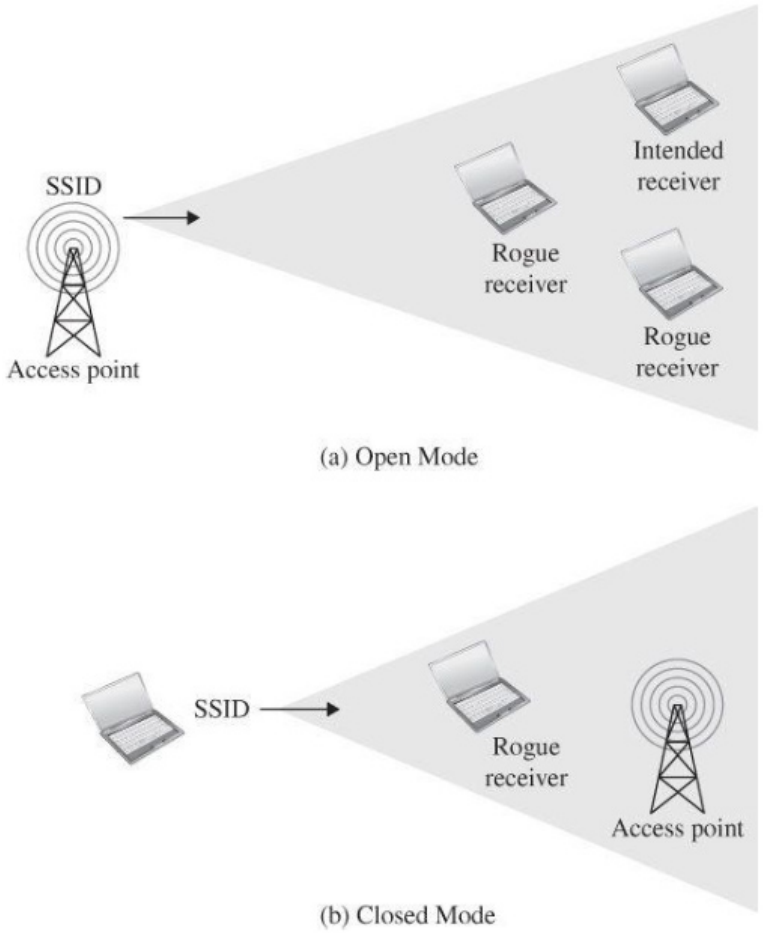
Kelemahan Protokol WiFi

Protokol akses nirkabel memiliki kelemahan bawaan yang dapat membahayakan keamanan. Jelas, komunikasi nirkabel lebih terbuka daripada komunikasi kabel karena kurangnya perlindungan fisik. Untuk alasan apa pun, perancang awal protokol komunikasi nirkabel internasional, suite 802.11, menciptakan situasi yang membuat komunikasi nirkabel rentan, seperti yang kita Gambarkan sekarang.

Mengambil Beacon (Suar)

Klien dan titik akses terlibat dalam jabat tangan otentikasi dan asosiasi untuk menemukan satu sama lain. Pada dasarnya klien mengatakan, “Saya ingin terhubung ke titik akses S” dan titik akses mengatakan, “Saya titik akses S; Saya menerima permintaan Anda untuk terhubung.” Urutan kedua langkah ini penting. Dalam apa yang disebut mode terbuka, titik akses terus-menerus menyiarkan daya tariknya di suarnya, menunjukkan bahwa itu terbuka untuk langkah selanjutnya dalam membangun koneksi. Mode tertutup atau sembunyi-sembunyi, juga dikenal sebagai penyelubungan SSID, membalik urutan langkah-langkahnya: Klien harus terlebih dahulu mengirim sinyal yang mencari titik akses dengan SSID tertentu sebelum titik akses merespons permintaan tersebut dengan undangan untuk terhubung. Kedua mode operasi ini ditunjukkan pada Gambar 3-17.

Dalam mode terbuka, titik akses terus-menerus menyiarkan SSID-nya; dalam mode tertutup, klien terus-menerus menyiarkan permintaan untuk terhubung ke SSID tertentu dari alamat MAC yang diberikan.



Gambar 6-17 Menghubungkan dalam Mode Terbuka dan Tertutup

Beroperasi dalam mode tertutup tampaknya merupakan cara yang berhasil untuk mencegah akses yang tidak sah: Jika Anda tidak mengetahui SSID, Anda tidak dapat meminta sambungan. Namun, mode tertutup membuat klien terbuka. Dalam mode terbuka, klien diam, memantau suar, hingga menemukan yang ingin disambungkan; dengan demikian, klien tidak selalu terlihat. Namun, dalam mode terbuka, klien secara efektif menjadi suar, mengirimkan serangkaian pesan berkelanjutan yang mengatakan, pada intinya, “Saya alamat MAC mmm, mencari SSID sss. Apakah kamu sss?” Dari pesan-pesan itu, tuan rumah nakal dapat mempelajari nilai yang diharapkan yang diperlukan untuk meniru titik akses yang diharapkan klien untuk terhubung.

SSID di Semua Frame

Menyiarkan SSID yang diinginkan dalam mode tertutup mengungkapkan identitas titik akses yang dicari. Lebih buruk lagi, dalam mode tertutup dan terbuka, bahkan setelah jabat tangan awal, semua manajemen dan bingkai data berikutnya berisi SSID yang sama ini, jadi mengendus salah satu dari bingkai ini mengungkapkan SSID. Jadi, siapa pun yang mengendus SSID dapat menyimpan SSID (yang jarang diubah dalam praktiknya) untuk digunakan nanti. Seorang pengintai dapat secara wajar menebak bahwa klien akan mencoba untuk menyambung ke titik akses yang sama ini lagi.

Dengan demikian, rogue memiliki informasi yang dibutuhkan untuk meniru baik klien atau titik akses di masa depan.

Desain protokol yang lebih baik adalah untuk titik akses dan perangkat yang terkait untuk menetapkan nilai data bersama yang akan digunakan selama satu asosiasi ini saja. Dengan cara itu, mencegat permintaan otentikasi awal akan mengungkapkan SSID, tetapi mencegat frame selanjutnya tidak akan.

Otentikasi di Jaringan Nirkabel

Jalur akses dapat mengelola daftar alamat MAC perangkat yang akan menerima koneksi. Dengan demikian, otentikasi pada langkah 2 dapat dicapai dengan hanya menerima perangkat pada daftar penerimaan positif.

Alamat MAC yang Dapat Diubah

Sistem operasi sebenarnya tidak selalu mendapatkan alamat MAC perangkat keras dari kartu NIC, melainkan berkonsultasi dengan data internal, jadi mengubah alamat MAC hanya memerlukan perubahan Tabel alamat kartu jaringan. Petunjuk untuk melakukan itu mudah ditemukan di Internet.

Mengubah alamat MAC NIC tidak hanya merusak otentikasi berbasis MAC pada titik akses, tetapi juga dapat menyebabkan serangan yang lebih besar yang disebut spoofing MAC, di mana satu perangkat meniru yang lain, sehingga mengasumsikan sesi komunikasi perangkat lain.

Pencurian Asosiasi

Sayangnya, jika proses jahat telah mencegat SSID dan memalsukan alamat MAC, dua titik kontrol akses terbaik telah hilang. Melihat dari perspektif lain, bagaimanapun, Anda mungkin berasumsi bahwa perangkat yang telah berhasil lulus uji SSID dan otentikasi sekarang akan diasosiasikan dengan titik akses yang diidentifikasi. Salah lagi!

Meskipun semua frame berisi SSID dari titik akses penerima yang dimaksud, tidak ada yang mencegah titik akses mana pun untuk menerima dan membalas frame apa pun. Faktanya, beberapa perangkat keras dan firmware titik akses diketahui cacat dan akan menerima asosiasi apa pun yang dapat diterimanya. Ini dikenal sebagai jalur akses promiscuous. Untuk contoh asosiasi rentan, lihat Studi Kasus 3-11. Pikirkan hal itu saat Anda mempertimbangkan untuk menyambung ke hot spot layanan WiFi gratis di bar atau bandara.

Asosiasi Pilihan

Situasi WiFi umum melibatkan penduduk yang terhubung ke jaringan rumah mereka, siswa yang terhubung ke jaringan sekolah, pekerja bisnis yang terhubung ke jaringan perusahaan, dan pelanggan yang terhubung ke jaringan kedai kopi atau toko buku. Pengguna biasa mungkin sering terhubung ke beberapa jaringan, dengan koneksi satu kali sesekali ke hotel atau jaringan bandara saat bepergian, atau museum selama kunjungan sehari. Untuk menyederhanakan koneksi, perangkat lunak antarmuka nirkabel membuat daftar titik koneksi favorit (rumah, sekolah, kantor) yang akan coba disambungkan secara otomatis. Biasanya tidak ada kebingungan, karena jaringan ini akan memiliki nama yang berbeda (sebenarnya SSID): Komputer Anda akan terhubung secara otomatis ke jaringan WestHallDorm.

Firefox adalah browser yang populer, sebagian karena pengguna telah menulis dan berkontribusi add-on, dua miliar yang menakjubkan pada hitungan terakhir, untuk melakukan berbagai hal mulai dari mengelola janji temu dan menampilkan ramalan cuaca hingga mengunduh video dan menerjemahkan teks dari satu bahasa kepada yang lain.

Kontribusi baru-baru ini, Firesheep, memungkinkan pengguna bergabung ke koneksi WiFi yang dibuat pengguna lain dengan mengklik mouse. Untuk menggunakan Firesheep, yang perlu Anda lakukan adalah bergabung dengan jaringan publik terbuka, seperti jaringan WiFi gratis di kedai kopi lokal Anda. Saat Anda terhubung, Firesheep diam-diam memindai semua lalu lintas lainnya di jaringan, memilih pesan yang menunjukkan hubungan yang mapan dengan beberapa layanan, seperti email berbasis web atau situs jejaring sosial. Seperti yang kami jelaskan di Bab 4, banyak dari situs ini mengelola asosiasi aktif mereka dengan menggunakan cookie yang dikirim dari situs ke browser pengguna dan kembali ke situs lagi. Masalahnya adalah sering kali cookie tersebut dikomunikasikan tanpa enkripsi, sepenuhnya jelas, yang berarti bahwa siapa pun yang mencegat dan mengirimkan cookie akan bergabung dengan sesi pengguna asli.

Firesheep membuat prosesnya ramah pengguna. Saat memindai jaringan, ia memilih situs sosial populer, misalnya, Facebook, mengambil nama pengguna dan bahkan Gambar dan menampilkannya di layar. Klik pada foto dan Anda masuk sebagai pengguna tersebut. Penanggulangannya, enkripsi, jarang diterapkan oleh situs-situs ini (walaupun sebagian besar lembaga keuangan mengenkripsi seluruh sesi, jadi transaksi perbankan Anda mungkin tidak diekspos oleh Firesheep). Kata penulis ekstensi, Eric Butler:

> *Situs web memiliki tanggung jawab untuk melindungi orang-orang yang bergantung pada layanan mereka. [Sites] telah terlalu lama mengabaikan tanggung jawab itu, dan inilah saatnya bagi semua orang untuk menuntut web yang lebih aman. Harapan saya adalah bahwa Firesheep akan membantu pengguna menang.*

Memang, tiga minggu setelah Butler merilis Firesheep pada Oktober 2010 dengan demonstrasi di San Diego pada konferensi keamanan ToorCon, Microsoft mengumumkan bahwa mereka menambahkan enkripsi sesi penuh (SSL, yang akan kami jelaskan nanti dalam bab ini) ke email berbasis web Hotmail. program. Setelah bertahun-tahun didorong, situs email dan jejaring sosial berbasis web yang populer sekarang menggunakan enkripsi sesi penuh. Namun, situs baru yang tidak dilindungi dibawa online setiap hari.

Pertimbangkan, bagaimanapun, akses WiFi gratis. Kedai kopi, toko buku, bandara, dan kota menawarkan layanan ini, dan beberapa administrator jaringan ingin memudahkan pelanggan dengan memberi nama layanan FreeWiFi. Jika Anda menginstruksikan perangkat lunak Anda (atau dalam beberapa kasus jika Anda tidak menginstruksikan perangkat lunak Anda untuk tidak melakukannya), FreeWiFi akan disimpan sebagai titik akses yang akan terhubung secara otomatis setiap kali melihat SSID tersebut di masa mendatang. Sayangnya, nama SSID tidak terikat pada karakteristik fisik apa pun: Komputer Anda tidak membedakan antara FreeWiFi sebagai titik akses di kedai kopi Anda atau titik jahat di kota asing yang dimaksudkan untuk memikat pengunjung yang tidak curiga ke wilayah itu. Komputer Anda akan terus mencari titik akses dengan SSID FreeWiFi dan menyambung ke titik mana pun yang ditemukannya. Meskipun kelemahan utama di sini adalah perangkat lunak yang menyimpan daftar titik akses pilihan untuk koneksi otomatis, protokol juga salah karena tidak memastikan identifikasi titik koneksi yang unik.

Daftar kerentanan dalam jaringan nirkabel ini panjang, yang seharusnya memberi tahu Anda bahwa komunikasi nirkabel sulit untuk diamankan. Sayangnya, keamanan WiFi telah bermasalah hampir sejak awal, seperti yang Anda lihat saat kami beralih dari kerentanan ke penanggulangan. Dalam bab ini kami mempertimbangkan dua contoh dari jenis penanggulangan yang sama: protokol. Paket protokol pertama diperkenalkan bersama dengan protokol lain yang mendefinisikan komunikasi nirkabel; suite protokol kedua adalah pengganti dari apa yang hampir sejak awal ditemukan paling efektif secara marginal. Jadi, kami menyatakan ini sebagai satu penanggulangan yang gagal dan satu penanggulangan yang ditingkatkan tetapi bukan penanggulangan yang sempurna. Kami mengGambarkan kegagalan pertama.

3.3.3 Penanggulangan yang Gagal: WEP (Wired Equivalent Privacy)

Pada saat yang sama komite standar IEEE sedang merancang protokol untuk mengaktifkan komunikasi digital nirkabel, mereka menyadari bahwa mereka juga memerlukan beberapa mekanisme untuk melindungi keamanan komunikasi tersebut, dan karenanya mereka memasukkan tindakan pencegahan dalam desain protokol awal. Privasi setara kabel, atau WEP, dimaksudkan sebagai cara komunikasi nirkabel untuk memberikan privasi yang setara dengan komunikasi kabel konvensional. Kabel fisik mudah dilindungi karena dapat diamankan secara fisik, dan lebih sulit untuk dicegat atau disadap tanpa deteksi. Untuk membuat komunikasi nirkabel dapat dipasarkan, perancang protokol berpikir bahwa mereka perlu menawarkan kerahasiaan yang sebanding dengan komunikasi kabel. Hasilnya adalah WEP, yang merupakan bagian dari standar 802.11 asli ketika diterbitkan pada tahun 1997.

Kelemahan dalam WEP telah diidentifikasi pada awal tahun 2001, dan kelemahannya sekarang sangat parah sehingga koneksi WEP dapat dipecahkan dengan perangkat lunak yang tersedia dalam beberapa menit.

Standar nirkabel 802.11 yang asli adalah upaya untuk menstandarisasi bidang komunikasi nirkabel yang muncul, dan karenanya berisi detail yang signifikan tentang transmisi, frekuensi, interaksi perangkat-ke-perangkat, dan operasi.

Kelemahan Keamanan WEP

Protokol WEP dimaksudkan untuk memberikan kekebalan kepada pengguna terhadap penyadapan dan serangan peniruan identitas, yang pada saat itu bukanlah ancaman serius. (Alasan itu mirip dengan mengatakan perlindungan terhadap kecelakaan kendaraan tidak menjadi perhatian yang signifikan ketika mobil ditemukan, karena hanya sedikit orang yang memiliki mobil. Seiring dengan meningkatnya penggunaan mobil, pabrikan telah menambahkan sejumlah fitur keamanan, seperti kantung udara, kursi sabuk pengaman, dan pintu yang diperkuat. Protokol WiFi lebih lambat beradaptasi.)

Keamanan WEP melibatkan beberapa teknik yang relatif sederhana untuk menanggapi apa yang diharapkan sebagai serangan yang tidak canggih dan jarang terjadi. WEP menggunakan kunci enkripsi yang dibagikan antara klien dan titik akses. Untuk mengotentikasi pengguna, titik akses mengirimkan nomor acak ke klien, yang dienkripsi klien menggunakan kunci bersama dan kembali ke titik akses. Sejak saat itu, klien dan titik akses diautentikasi dan dapat berkomunikasi menggunakan kunci enkripsi bersama mereka. Ada beberapa masalah dengan pendekatan yang tampaknya sederhana ini, yang sekarang kami jelaskan.

Kunci Enkripsi Lemah

Pertama, standar WEP memungkinkan kunci enkripsi 64- atau 128-bit, tetapi setiap kunci dimulai dengan vektor inisialisasi 24-bit (IV), yang memiliki efek mengurangi panjang kunci menjadi 40 atau 104 bit. (Alasan untuk panjang kunci ini adalah karena pemerintah AS mengamanatkan bahwa perangkat lunak kriptografi untuk ekspor dibatasi pada algoritme yang menggunakan kunci tidak lebih dari 40 bit. Pembatasan tersebut telah dicabut.)

Pengguna memasukkan kunci dalam bentuk apa pun yang nyaman, biasanya dalam heksadesimal atau sebagai string alfanumerik yang diubah menjadi angka. Memasukkan 64 atau 128 bit dalam hex memerlukan pemilihan dan kemudian mengetikkan 16 atau 32 simbol dengan benar untuk klien dan titik akses. Tidak mengherankan, string hex seperti C0DEC0DE … (yaitu nol antara C dan D) adalah umum. Frasa sandi rentan terhadap serangan kamus.

Dengan demikian, pengguna cenderung menggunakan kunci yang tidak benar-benar acak. Situasinya seperti meminta anak-anak yang sangat kecil untuk memilih angka antara 1 dan 100 dan kemudian mencoba menebaknya. Jika Anda menentukan anak-anak hanya mengetahui angka 1 sampai 10, peluang Anda untuk menebak dengan benar meningkat dari 1 dalam 100 menjadi 1 dalam 10, meskipun ruang target masih 1 hingga 100. Distribusi non-acak memperkecil peluang menebak dengan benar.

Kunci Statis

Kunci enkripsi WEP dibagi antara pengirim dan penerima. Ini berarti bahwa nilai yang sama harus dimasukkan pada titik akses dan perangkat jarak jauh. Meskipun pengguna kadang-kadang bersedia untuk memilih dan memasukkan kunci kedua tempat, mereka tidak ingin sering melakukannya. Dengan demikian, kunci enkripsi yang sama cenderung digunakan untuk waktu yang lama.

Penyerang khusus yang dapat memantau sejumlah besar lalu lintas jaringan nirkabel akan mengumpulkan banyak titik data untuk menyimpulkan kunci. Jika kunci sering berubah, titik data

dari kunci lama tidak memberikan bantuan dalam menyimpulkan kunci baru, tetapi kunci yang tidak sering diubah memungkinkan untuk menyimpulkan dari sejumlah besar titik data. Jadi, kunci yang jarang diubah meningkatkan kemungkinan penyerang dapat menyimpulkannya.

Proses Enkripsi Lemah

Bahkan jika kuncinya kuat, itu benar-benar memiliki panjang efektif hanya 40 atau 104 bit karena cara itu digunakan dalam algoritma. Serangan brute force terhadap kunci 40-bit berhasil dengan cepat. Bahkan untuk versi 104-bit, kelemahan dalam algoritma RC4 dan penggunaannya mengalahkan keamanan WEP. Alat seperti WEPCrack dan AirCrack-ng memungkinkan penyerang untuk memecahkan enkripsi WEP, biasanya dalam beberapa menit. Pada konferensi tahun 2005, FBI mendemonstrasikan betapa mudahnya sesi nirkabel yang diamankan dengan WEP dapat diputus.

Algoritma Enkripsi Lemah

Masalah ketiga dengan WEP adalah cara melakukan enkripsi. WEP tidak menggunakan RC4 sebagai algoritma enkripsi secara langsung; sebagai gantinya, RC4 menghasilkan urutan panjang angka acak, yang disebut urutan kunci, yang diturunkan dari vektor inisialisasi 24-bit dan kunci 40-bit. WEP menggabungkan urutan kunci dengan data menggunakan fungsi eksklusif-OR. Sayangnya, jika penyerang dapat menebak nilai yang didekripsi dari setiap bingkai terenkripsi tunggal, memasukkan nilai itu ke dalam fungsi eksklusif-OR mengungkapkan segmen dari urutan kunci tersebut. Urutan kunci yang sama digunakan kembali untuk semua pesan, sehingga segmen akan berulang pada titik yang sama. IV dikomunikasikan sebagai plaintext, sehingga penyerang dapat mencegatnya untuk serangan pencarian kunci yang lengkap. Masalah lain yang diketahui melibatkan penggunaan OR eksklusif.

Inisialisasi Tabrakan Vektor

Masalah enkripsi terakhir dengan WEP menyangkut vektor inisialisasi, yang menjadi 24 bit pertama dari kunci enkripsi. Siklus 24 bit ini dalam pola yang dapat diprediksi sampai semua
Pola 24-bit telah digunakan (sekitar 16 juta iterasi), di mana titik vektor inisialisasi kembali ke nilai pertama dan siklus dimulai lagi. Setidaknya begitulah teorinya. Dalam praktiknya, nilai vektor inisialisasi tertentu terjebak dalam loop dan tidak pernah berubah, sementara yang lain tidak menggilir semua 16 juta pola 24-bit. Dan beberapa perubahan pertama tidak sepenuhnya acak tetapi memiliki tingkat prediktabilitas tertentu.

Penyerang yang tertarik dapat menguji semua 16 juta vektor inisialisasi yang mungkin dalam beberapa jam, dan kelemahan seperti vektor inisialisasi yang tidak berubah (disebut lemah) mengurangi jumlah pengujian lebih jauh, sehingga mempercepat pencarian.

Pemeriksaan Integritas Rusak

Seolah-olah masalah enkripsi saja tidak cukup, WEP tidak dirancang untuk integritas yang kuat. Seperti yang sudah Anda ketahui, komunikasi nirkabel dapat mengalami kehilangan dan gangguan data. Dengan demikian, perancang protokol memasukkan nilai pemeriksaan untuk menunjukkan apakah bingkai tiba utuh atau beberapa bit telah hilang atau secara tidak sengaja berubah dalam transmisi. Penerima menghitung ulang nilai cek dan jika tidak cocok, menandakan kesalahan transmisi dan meminta salinan lain untuk dikirim.

Pemeriksaan integritas menggunakan algoritma yang terkenal. Jika penyerang jahat ingin mengubah bagian dari komunikasi, penyerang hanya mengubah data, menghitung nilai pemeriksaan integritas baru, dan mengganti pemeriksaan asli dengan yang baru. Jadi, ketika bingkai tiba, nilai pemeriksaan baru akan cocok dengan data yang dimodifikasi, dan penerima tidak akan menyadari bahwa nilai data telah dimodifikasi secara jahat.

Tidak Ada Otentikasi

Kelemahan terakhir dalam protokol WEP adalah tidak memiliki otentikasi. Perangkat apa pun yang dapat memberi nama SSID yang benar dan menunjukkan alamat MAC yang benar dianggap sah. Seperti yang kita lihat, bahkan jika SSID tidak disiarkan, itu tersedia dalam bingkai lain, seperti alamat MAC, sehingga penyerang dapat dengan mudah menampilkan SSID dan mengkonfigurasi ulang NIC untuk menunjukkan alamat MAC yang diperlukan. Dengan demikian, penyerang tidak benar-benar terhalang untuk memalsukan node jaringan yang valid.

Intinya: Keamanan WEP Tidak Dapat Diterima

Semua kelemahan WEP ini adalah keterbatasan dari protokol WEP dasar. Dalam beberapa tahun pengenalan standar WEP, peneliti (lihat [FLU01] dan [ARB02]) menghasilkan program demonstrasi aktual yang menunjukkan kemampuan untuk menyimpulkan kunci RC4 dalam hitungan menit. Seperti yang dijelaskan Studi Kasus 3-12, kelemahan ini tidak hanya teoretis; penyerang benar-benar mengeksploitasi kerentanan ini dan mengganggu sistem nirkabel, menyebabkan hilangnya sejumlah besar data sensitif. Desain protokol WEP tidak menggunakan kriptografi secara efektif, gagal melakukan otentikasi, tidak memiliki kontrol yang efektif atas modifikasi yang disengaja, dan tidak dapat menjamin ketersediaan untuk pengguna yang berwenang. Dengan kekurangan dalam protokol itu sendiri, tidak ada implementasi yang ditingkatkan atau mode penggunaan yang aman yang dapat mengimbanginya.

Studi Kasus 3-12 : TJ Maxx Pencurian Data

Pada tahun 2005, sekelompok 11 peretas mencuri lebih dari 45 juta nomor kartu kredit dan debit dari toko pakaian TJ Maxx and Marshalls dan mitra bisnis mereka; para penjahat melakukannya tanpa pernah menginjakkan kaki di dalam toko.

Pencuri memasang antena di luar satu toko TJ Maxx dan menyadap komunikasi nirkabel antara perangkat pemindai genggam, mesin kasir, dan komputer di jaringan nirkabel toko. Dengan antena berbentuk seperti teleskop, seseorang dengan komputer laptop sederhana dapat mencegat sinyal WiFi bermil-mil jauhnya dari titik akses. Pencuri ini bekerja dari tempat parkir.

Jaringan diamankan dengan protokol WEP yang mudah dikompromikan, meskipun kerentanan keamanan yang serius di WEP telah ditunjukkan dan didokumentasikan bertahun-tahun sebelumnya. Data yang diperoleh dari intersepsi nirkabel tidak hanya mencakup detail transaksi tetapi juga ID login dan kata sandi akun yang lebih penting yang memungkinkan kru memasang sniffer di jaringan dan meretas ke database pusat perusahaan induk yang memiliki TJ Maxx dan Marshalls.

Albert Gonzales dari Miami divonis pada Maret 2010 karena menjadi biang keladi dari kelompok yang mencakup dua warga negara AS lainnya, tiga orang Rusia, dua orang Cina, dan masing-masing satu dari Estonia dan Belarusia. Gonzales divonis 20 tahun penjara.

TJ Maxx bukan satu-satunya pengecer yang rentan. Pada tahun 2009, unit Pertahanan Udara Motorola mensurvei pengecer di kota-kota besar di seluruh dunia, termasuk Atlanta, Boston, New York City, Paris, Seoul, dan Sydney. Menurut siaran pers mereka pada 28 Januari 2009, mereka menemukan bahwa 44 persen jaringan dan perangkat nirkabel pengecer dapat disusupi. Jaringan nirkabel di toko tidak menggunakan enkripsi 32 persen setiap saat, dan 25 persen jaringan menggunakan teknologi keamanan WEP yang lemah.

Perhatikan dari kota-kota ini dan kebangsaan kelompok Gonzales bahwa keamanan komputer merupakan masalah internasional. Targetnya memang tersebar luas, dan banyaknya jaringan yang rentan menarik penyerang yang cakap dari berbagai latar belakang. Sayangnya, beberapa pengecer mulai menggunakan teknologi nirkabel hanya untuk mengomunikasikan informasi dengan sensitivitas rendah, seperti data inventaris. Namun, ketika mereka memperluas aplikasi jaringan mereka dan mulai mengomunikasikan data yang lebih sensitif, mereka melupakan atau mengabaikan paparan penggunaan keamanan yang lemah. Untuk itu, keamanan harus ditinjau setiap kali ada perubahan penggunaan, arsitektur, atau implementasi sistem.

3.3.4 Suite Protokol yang Lebih Kuat: WPA (WiFi Protected Access)

Paket protokol WEP diterbitkan pada tahun 1997 dan diratifikasi pada tahun 1999, yang berarti bahwa produk yang menerapkan WEP mulai muncul sekitar tahun 1999. Pada posting sci.crypt 1995, Wagner dan Roos secara independen menemukan masalah dalam struktur kunci RC4 tetapi, karena RC4 tidak banyak digunakan sebelum dimasukkan ke dalam standar WEP, masalah ini belum dipelajari secara luas.

Indikasi pertama kelemahan WEP yang serius diterbitkan pada tahun 2001, hanya dua tahun setelah penerimaan formal protokol WEP. Periode yang begitu singkat dapat menjadi hasil dari banyak penyebab. Tentu saja kendala pada kekuatan kriptografi pada tahun 1997 membatasi pilihan keamanan pengembang protokol. Selanjutnya, meremehkan keseriusan ancaman terhadap teknologi baru dan karenanya tidak digunakan kemungkinan membuat perancang protokol mengambil pendekatan yang mudah. Ketika kekurangan WEP diterbitkan pada tahun 2001, menjadi jelas bahwa WEP bukanlah protokol yang memadai untuk WiFi. (Sayangnya, seperti yang dijelaskan di Studi Kasus 3-13, bahkan para ahli pun gagal mempraktikkan keamanan yang kuat.)

Studi Kasus 3-13 : Lakukan Seperti yang Saya Katakan, Bukan Seperti yang Saya Lakukan

Anda akan mengharapkan konferensi profesional keamanan komputer untuk mengikuti praktik keamanan terbaik. Tidak, hanya itu yang mereka nasihati, bukan apa yang mereka lakukan.

Pada Konferensi Keamanan RSA 2010, yang menarik banyak praktisi keamanan komputer dan pemimpin industri, divisi Pertahanan Udara Motorola memindai gelombang nirkabel untuk melihat siapa yang terhubung dengan apa. Mereka mengamati lebih dari 2.000 koneksi. Mereka menemukan [WIL10] bahwa 116 klien telah terhubung point-to-point ke SSID berisiko seperti "WiFi Publik Gratis" dan "Akses Internet Gratis". Koneksi point-to-point (disebut ad hoc) adalah ke komputer lain, tidak harus ke titik akses dan server.

Lebih buruk lagi, 62 persen dari jaringan yang ditemukan menjalankan protokol WEP atau protokol TKIP yang lebih kuat tetapi masih cacat, hampir sepuluh tahun setelah kurangnya keamanan WEP telah ditunjukkan secara meyakinkan, dan hampir lima tahun setelah kerentanan (dijelaskan nanti dalam bab ini) di TKIP dipublikasikan.

Motorola tidak melacak perangkat yang menggunakan keamanan yang lemah, jadi tidak ada yang tahu berapa banyak mesin pengguna dibandingkan dengan mesin demonstrasi di ruang pameran. Namun, orang bertanya-tanya apa yang dikatakan statistik ini tentang penggunaan praktik keamanan terbaik oleh masyarakat umum.

Untuk alasan ini, pada tahun 2001 IEEE mulai merancang skema otentikasi dan enkripsi baru untuk nirkabel, seperti yang kami jelaskan di bagian selanjutnya.

Alternatif untuk WEP adalah WiFi Protected Access atau WPA, dirancang pada tahun 2003. Standar IEEE 802.11i, yang dikenal sebagai WPA2, telah disetujui pada tahun 2004, dan merupakan perpanjangan dari WPA. Meskipun nama WPA2 benar, standarnya secara informal dikenal sebagai WPA. Sebenarnya, ada perbedaan antara ini: WPA adalah pengganti asli untuk WEP; WPA2 melampaui WPA dengan membutuhkan dukungan untuk algoritma enkripsi AES yang kuat. Selanjutnya, untuk menggunakan penunjukan "WiFi Certified" bermerek dagang, perangkat harus disertifikasi oleh aliansi WiFi. Dalam praktiknya, semua perangkat WiFi yang dijual kini memenuhi standar WPA2. Dalam buku ini kami mengikuti penggunaan umum dan menggunakan WPA untuk merujuk pada protokol WPA dan WPA2. Bagaimana WPA menjadi lebih baik dari WEP?

Kelebihan WPA dibanding WEP

WPA berangkat untuk mengatasi kekurangan yang diketahui pada WEP, dan dengan demikian banyak fitur WPA secara langsung mengatasi kelemahan WEP. Berikut adalah beberapa cara di mana WPA lebih unggul dari WEP.

Kunci Enkripsi Non-Statis

Pertama, WEP menggunakan kunci enkripsi yang tidak berubah sampai pengguna memasukkan kunci baru di klien dan titik akses. Kriptologis menyesalkan kunci enkripsi statis karena kunci tetap memberi penyerang sejumlah besar ciphertext untuk mencoba menganalisis dan banyak waktu untuk menganalisisnya. WPA memiliki pendekatan perubahan kunci, yang disebut Program Integritas Kunci Temporal (TKIP), dimana kunci enkripsi diubah secara otomatis pada setiap paket. WPA juga menggunakan hierarki kunci untuk membuat kunci baru untuk setiap sesi. Kunci ini mengizinkan titik akses (disebut autentikator) dan perangkat penghubung (disebut pemohon) untuk membuat dan menukar kunci untuk kerahasiaan dan integritas yang unik untuk sesi asosiasi.

Autentikasi

Kedua, WEP menggunakan kunci enkripsi sebagai autentikator, meskipun tidak aman. WPA menggunakan protokol otentikasi yang dapat diperluas (EAP) yang dengannya otentikasi dapat dilakukan dengan kata sandi, token, sertifikat, atau mekanisme lainnya. Untuk pengguna jaringan kecil (rumah), ini mungkin masih berarti rahasia bersama, yang tidak ideal. Pengguna cenderung memilih kunci yang lemah, seperti angka pendek atau frasa sandi yang terkena serangan kamus.

Enkripsi yang Kuat

Algoritma enkripsi untuk WEP adalah RC4, yang memiliki kelemahan kriptografi baik dalam panjang kunci dan desain. Dalam WEP, vektor inisialisasi untuk RC4 hanya 24 bit, ukuran yang sangat kecil sehingga sering terjadi tabrakan; lebih lanjut, WEP tidak memeriksa penggunaan ulang vektor inisialisasi. WPA2 menambahkan AES sebagai kemungkinan algoritma enkripsi (walaupun RC4 juga masih didukung untuk kompatibilitas). AES (dijelaskan dalam Bab 2) adalah algoritma enkripsi yang jauh lebih kuat, sebagian karena menggunakan kunci enkripsi yang lebih panjang (yang meningkatkan waktu untuk pencarian lengkap dari hari ke ribuan tahun).

Perlindungan Integritas

WEP menyertakan pemeriksaan integritas 32-bit yang terpisah dari bagian data. Tetapi karena enkripsi WEP tunduk pada serangan cryptanalytic, pemeriksaan integritas juga tunduk, sehingga penyerang dapat memodifikasi konten dan pemeriksaan terkait tanpa harus mengetahui kunci enkripsi terkait. WPA menyertakan pemeriksaan integritas 64-bit yang dienkripsi.

Inisiasi Sesi

Protokol pengaturan untuk WPA dan WPA2 jauh lebih kuat daripada untuk WEP. Setup untuk WPA melibatkan tiga langkah protokol: otentikasi, jabat tangan empat arah (untuk memastikan bahwa klien dapat menghasilkan kunci kriptografi dan untuk menghasilkan dan menginstal kunci untuk enkripsi dan integritas di kedua ujungnya), dan jabat tangan kunci grup opsional (untuk multicast komunikasi). Lehembre memberikan Gambaran yang baik tentang protokol WPA.

WPA dan WPA2 mengatasi kekurangan keamanan yang dikenal di WEP. Arazi dkk. membuat kasus yang kuat untuk kriptografi kunci publik dalam jaringan sensor nirkabel, dan argumen serupa dapat dibuat untuk aplikasi nirkabel lainnya (walaupun tuntutan komputasi yang lebih berat dari enkripsi kunci publik merupakan faktor pembatas pada perangkat nirkabel dengan kemampuan prosesor yang terbatas). Penggunaan WEP menurun mendukung WPA, seperti yang dijelaskan di Studi Kasus 3-14.

Sejak diperkenalkan pada tahun 2004, penggunaan WPA terus meningkat. Pada tahun 2008, Asosiasi Keamanan Informasi Profesional Hong Kong melakukan survei dengan mengemudikan perang, memantau, dan membuat katalog titik akses yang dapat ditemukan. Mereka menentukan bahwa, dari lebih dari 10.000 titik akses yang diidentifikasi di Hong Kong dan Makau, 43 persen menggunakan WEP dan 40 persen WPA.

RSA Security melakukan survei, juga pada tahun 2008, dari pusat bisnis utama. Mereka menemukan penggunaan WPA atau WPA2 sebesar 49 persen (dari titik akses yang disurvei) di New York City, 71 persen di Paris, dan 48 persen di London.

Meskipun persentase penggunaan WPA terus meningkat di seluruh dunia, tingkat adopsi sangat kecil, mengingat keunggulan keamanan utama WPA dibandingkan WEP (atau, lebih buruk, tanpa keamanan sama sekali).

Serangan Pada WPA

Tak lama setelah munculnya suite protokol WPA, para peneliti menemukan dan menjelaskan kekurangannya.

Man-in-the-Middle

Mishra dan Arbaugh [MIS02] mengidentifikasi dua kelemahan potensial dalam desain protokol WPA. Yang pertama, yang disebut serangan man-in-the-middle (kami menunjukkan contoh lain dari serangan di tengah-tengah di Bab 4), dieksploitasi ketika penyerang yang cerdik dapat mengganggu percakapan yang sah, mencegat dan mungkin mengubah keduanya. pihak, untuk diam-diam mendapatkan atau memodifikasi data yang dilindungi. Serangan Mishra dan Arbaugh menggunakan orang jahat di tengah untuk membajak sesi, yaitu, orang luar menggantikan pengguna yang sah dan melanjutkan sesi itu dalam otoritas pengguna.

Serangan berhasil melalui serangan spoofing alamat MAC. Selama urutan asosiasi antara perangkat dan titik akses, perangkat menyajikan kredensial untuk mengautentikasi dan titik akses mengirim pesan yang mengonfirmasi autentikasi. Pada saat itu, orang jahat di tengah mengubah alamat MAC-nya ke titik akses dan mengirimkan permintaan ke perangkat untuk disassociate. Disasosiasi adalah sarana bagi salah satu pihak untuk mengakhiri asosiasi dan dapat terjadi karena penyelesaian aktivitas, kelebihan beban, atau alasan lain. Perangkat yang meminta menghentikan pengaitan dan memulai lagi proses pengaitan; sementara itu, orang luar yang jahat telah mengubah alamat MAC ke perangkat yang tidak terkait dan melanjutkan hubungan dengan titik akses seolah-olah itu adalah pengguna asli.

Masalah yang memungkinkan serangan ini adalah bahwa frame tidak memiliki perlindungan integritas; oleh karena itu, pesan disasosiasi dari host jahat tidak diidentifikasi sebagai tidak autentik.

Otentikasi Tidak Lengkap (Incomplete Authentication)

Serangan kedua terhadap WPA menunjukkan kelemahan terkait dalam urutan otentikasi.

Pada satu titik pemohon (klien) diharuskan untuk mengotentikasi ke titik akses, tetapi pemohon tidak memiliki dasar untuk jaminan bahwa titik akses itu sah, yaitu bahwa pihak jahat tidak mengirimkan sinyal yang berpura-pura menjadi titik akses. Dengan demikian, pemohon dapat dipaksa untuk mengungkapkan data otentikasi kepada pihak ketiga yang tidak berwenang.

Ingat tentang pentingnya saling curiga dalam program: Setiap rutinitas perlu mencurigai bahwa semua rutinitas lain yang berinteraksi dengannya mungkin salah atau tidak bersahabat. Serangan yang diajukan menunjukkan contoh kegagalan untuk saling curiga.

Pencarian Key Search

Keterbatasan kriptografi yang diketahui adalah bahwa seluruh ruang kemungkinan kunci kriptografi dapat dicari untuk menemukan yang benar. Penanggulangan terhadap serangan ini adalah dengan menggunakan kunci yang panjang, sehingga jumlah kunci yang harus dicari menjadi penghalang. Kunci DES 56-bit telah terbukti rentan terhadap musuh dengan daya komputasi yang signifikan, dan panel kriptografer pada tahun 1996 [BLA96] menyarankan agar kunci berukuran 100 bit atau lebih untuk keamanan tinggi. Saran ini tergantung pada kunci yang benar-benar acak; seperti menggunakan aaaaaa sebagai kata sandi, menggunakan nomor pola yang dapat diprediksi melemahkan kunci. Karena pemilihan kunci sangat penting, manajemen kunci WPA menjadi sorotan. WPA menggunakan kunci dasar 256-bit, yang tampaknya cukup lama untuk aman.

Untuk membuat kunci bersama antara titik akses dan perangkat, administrator harus memasukkan angka yang sangat besar dengan benar dua kali, sekali untuk titik akses dan sekali untuk perangkat. Untuk menyederhanakan entri sejumlah besar, banyak implementasi WPA mengizinkan frasa sandi, serangkaian karakter, yang kemudian diubah menjadi angka. Moskowitz [MOS03] mengamati bahwa orang cenderung tidak memilih string karakter sepenuhnya secara acak, dan dengan demikian menebak serangan dengan string populer berhasil lebih cepat daripada pencarian lengkap yang lengkap. Moskowitz mencatat, bagaimanapun, bahwa algoritma yang digunakan WPA untuk mengubah frasa sandi menjadi kunci enkripsi (komputer) memakan waktu, yang mengurangi kemampuan penyerang untuk menguji sejumlah besar frasa sandi potensial sebagai kunci.

Serangan serupa bergantung pada orang yang memilih frasa sandi pendek, karena serangan yang melelahkan akan berkembang secara berurutan melalui semua frasa sandi potensial satu karakter, lalu dua karakter, dan seterusnya.

Akhirnya, pada tahun 2008, peneliti Martin Beck dan Erik Tews mempresentasikan serangan terhadap fitur WEP yang dibawa ke WPA. Serangan itu merusak integritas konten terenkripsi. Kami telah menjelaskan ketidakamanan algoritma RC4 yang digunakan oleh WEP, menerapkan kunci 40- atau 104-bit, dan serangan Tews-Beck menemukan kelemahan lain di sana. Para peneliti juga menyerang WPA dengan teknik mereka, yang mereka sebut chopchop karena mereka memotong dan mengganti satu byte blok dan melihat perubahan integritas blok. Dengan berulang kali memotong dan mengganti, mereka menyimpulkan kunci integritas. Serangan itu merusak WPA asli karena menggunakan mekanisme integritas yang disebut TKIP (Protokol Integritas Kunci Sementara)

yang dirancang agar kompatibel dengan WEP. Serangan canggih paling efektif terhadap blok data pendek yang penyerang mengetahui beberapa data plaintext yang mendasarinya; hasil serangan memungkinkan penyerang untuk mengganti beberapa paket dengan data lain tanpa terdeteksi. Ohigashi dan Morii meningkatkan teknik dengan membuatnya lebih cepat.

Serangan ini penting karena menunjukkan kerentanan yang sebelumnya tidak diketahui. Namun, itu hanya menghasilkan paket-paket tertentu yang berhasil dalam aliran WPA. Itu tidak merusak WPA atau TKIP secara umum dan, yang lebih penting, tidak efektif melawan WPA2 menggunakan algoritma AES.

Kesimpulan: WPA Cukup Aman

Kerentanan yang diidentifikasi terjadi dalam kasus terbatas dan tidak memengaruhi sebagian besar pengguna atau WPA. Kehati-hatian dalam memilih kunci enkripsi dapat memastikan bahwa kunci tersebut cukup panjang dan acak untuk kebal dari serangan tebak-tebakan.

Lebih serius daripada kelemahan apa pun dalam rangkaian algoritma WPA adalah jumlah data yang dikomunikasikan orang tanpa perlindungan. Perlindungan data pengguna adalah masalah aplikasi, bukan masalah jaringan, dan dengan demikian ini adalah sesuatu yang harus dipecahkan oleh pengguna dan program.

Sejauh ini dalam buku ini, kami telah memfokuskan hampir secara eksklusif pada serangan kerahasiaan dan integritas, baik dalam komputasi konvensional maupun dalam jaringan. Alat penanggulangan kami sangat bergantung pada trio otentikasi, kontrol akses, dan enkripsi, serta alat tujuan khusus seperti pemrograman defensif, pemisahan, dan hak istimewa paling rendah. Sekarang kita beralih ke kerentanan keamanan yang sangat kuat di jaringan: penolakan layanan, atau hilangnya ketersediaan. Untuk melawan ancaman seperti itu, kami menemukan bahwa kami membutuhkan serangkaian tindakan pencegahan yang sangat berbeda.

3.4 Denial of Service (DoS)

Serangan Denial of Service (DoS) merupakan jenis serangan terhadap sistem dalam jaringan internet dengan cara menghabiskan resource yang dimiliki oleh suatu sistem sehingga tidak dapat menjalankan fungsinya dengan benar dan secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan sistem yang diserang tersebut. Serangan DoS memanfaatkan kelemahan sistem pada keterbatasan sumber daya, baik bandwidth, kemampuan menyimpan memori, server dan kelemahan lainnya. Kebanyakan DoS menyerang bisnis kecil hingga menengah yang tidak memiliki sumber daya yang besar. Pada dasarnya tujuan penyerang hanya untuk membuat sistem lumpuh, tapi tak jarang juga ada yang kemudian meminta biaya tebusan untuk menghentikan serangan. Dalam serangan DoS penyerang menggunakan satu komputer dan satu koneksi internet saja ketika meluncurkan serangan. Untuk melancarkan serangan yang berskala lebih besar, penyerang bisa menggunakan banyak komputer dan banyak koneksi internet yang dikontrol secara bersamaan dengan menggunakan botnet. Botnet merupakan sejumlah komputer yang terinfeksi malware tanpa disadari oleh penggunanya.

Contoh: Kegagalan Besar-besaran Web Estonia

Kami memulai bagian ini dengan contoh serangan layanan besar. Dan meskipun pelaku serangan ini masih belum diketahui, cukup jelas bahwa serangan ini bermotif politik.

Pejabat di Republik Estonia memutuskan pada tahun 2007 untuk memindahkan sebuah monumen yang disebut "Prajurit Perunggu", yang memperingati keterlibatan Rusia dalam Perang Dunia II. Mengambil langkah itu sebagai penghinaan terhadap Rusia, orang-orang memblokade kedutaan Estonia di Moskow, dan protes meletus di Estonia, yang memiliki populasi etnis minoritas Rusia yang besar.

Hampir segera setelah demonstrasi dimulai, situs web Estonia dibombardir dengan lalu lintas, dengan kecepatan 100–200 megabit per detik. Meskipun serangan baru-baru ini telah mencapai 1.000 kali volume, pada tahun 2007, lalu lintas 100 megabit per detik tidak pernah terdengar.

Di antara situs yang diserang adalah dari

- Presiden
- Parlemen
- Banyak departemen pemerintah
- Partai-partai politik
- Organisasi berita utama
- Bank-bank besar
- Perusahaan telekomunikasi

Serangan dimulai pada 27 April setelah patung itu dipindahkan, dan terus berlanjut selama beberapa hari. Pada tanggal 8–9 Mei, periode ketika Rusia merayakan kemenangannya atas Nazi dalam Perang Dunia II, serangan kembali meningkat, dan meningkat lagi pada pertengahan Mei sebelum akhirnya mereda.

Estonia adalah salah satu negara yang paling terkomputerisasi di dunia dan telah memelopori e-government; perlambatan di situs pemerintah dan komersial utama selama hampir sebulan berdampak serius pada kemampuan warganya untuk melakukan bisnis.

Tim tanggap darurat komputer Estonia menetapkan bahwa serangan itu sebagian besar datang dari luar Estonia. Para ahli bertindak cepat untuk menutup situs yang diserang dan menerapkan kontrol lain untuk membatasi lalu lintas masuk. Tim tanggap darurat dari Uni Eropa dan Amerika Serikat dipanggil untuk membantu mengelola serangan.

Menentukan sumber serangan itu tidak mungkin, Sumber serangan seperti itu seringkali tidak jelas, karena menentukan dari mana lalu lintas dirutekan dari yang terbaru tidak sama dengan mengidentifikasi sumber asli serangan. Meskipun Menteri Luar Negeri Estonia menuduh Kremlin terlibat, Menteri Pertahanan mengakui tidak ada bukti pasti tentang hal itu. Seorang Rusia dihukum di Estonia karena peran kecil dalam serangan itu. Tanggung jawab untuk perencanaan, koordinasi, dan pemasangan serangan belum dan mungkin tidak akan pernah ditetapkan.

Tindakan terisolasi? Tidak. Pada Januari 2013, situs web New York Times dibombardir oleh serangan penolakan layanan besar-besaran, seperti halnya situs Washington Post dan Wall Street Journal. Diduga, situs web ini diserang oleh peretas yang memiliki hubungan dengan China. Pada Agustus 2013, sebuah kelompok yang diidentifikasi sebagai Tentara Elektronik Suriah diduga menutup akses ke situs web New York Times selama 20 jam. Pada Juni 2014, kelompok yang sama diduga mengalihkan pembaca Reuters dari cerita yang mengGambarkan serangan Suriah ke pesan yang melaporkan bahwa situs tersebut telah diretas. Penolakan layanan untuk tujuan politik adalah alat yang ampuh. Lembaga keuangan juga menjadi sasaran serangan dari sumber yang tidak diketahui.

Serangan penolakan layanan, atau DoS, adalah upaya untuk mengalahkan ketersediaan, yang ketiga dari tiga properti dasar yang harus dipertahankan dalam keamanan komputer. Denial of service berarti persis seperti namanya: pengguna ditolak aksesnya ke layanan atau data resmi. Kerahasiaan dan integritas berkaitan dengan pencegahan akses yang tidak sah; ketersediaan berkaitan dengan menjaga akses resmi.

Konsep sederhana DDoS attack adalah memFlooding i lalu lintas jaringan dengan banyak data. Konsep Denial of Service bisa dibagi menjadi 3 tipe penggunaan, yakni sebagai berikut :

- Request flooding merupakan teknik yang digunakan dengan memFlooding i jaringan menggunakan banyak request. Akibatnya, pengguna lain yang terdaftar tidak dapat dilayani.
- Traffic flooding merupaka teknik yang digunakan dengan memFlooding i lalu lintas jaringan dengan banyak data. Akibatnya, pengguna lain tidak bisa dilayani.
- Mengubah sistem konfigurasi atau bahkan merusak komponen dan server juga termasuk tipe denial of service, tetapi cara ini tidak banyak digunakan karena cukup sulit untuk dilakukan

Kerahasiaan dan integritas adalah dua hal yang saling berkaitan: Data atau objek disimpan atau tidak disimpan secara pribadi dan tidak dimodifikasi; ketersediaan bisa lebih bernuansa, karena mungkin ada layanan tetapi dalam jumlah yang tidak mencukupi atau pada respons yang tidak dapat diterima. Anda tahu bahwa halaman web membutuhkan beberapa detik untuk dimuat, tetapi seiring berjalannya waktu Anda menjadi lebih frustrasi atau curiga bahwa itu tidak akan pernah ditampilkan; kemudian, tiba-tiba muncul dan Anda bertanya-tanya mengapa begitu lama. Dengan demikian, penolakan layanan berkisar dari hilangnya akses sepenuhnya hingga perlambatan yang nyata dan tidak dapat diterima hingga ketidaknyamanan.

3.4.1 Bagaimana Layanan Ditolak

Pada bagian ini kami menjelaskan apa yang menyebabkan penolakan layanan. Banyak penyebab yang tidak berbahaya dan seringkali sporadis dan spontan, sehingga hanya sedikit yang dapat dilakukan untuk mengatasinya. Kami fokus pada penyebab jahat karena itulah yang dapat ditangani. Untungnya, beberapa kelas penanggulangan efektif terhadap serangan penolakan layanan yang berbahaya. Pertama, kami mempertimbangkan beberapa penyebabnya.

Pikirkan sejenak tentang bagaimana Anda dapat menolak akses di jaringan komputer.

- Salah satu kelemahan potensial adalah kapasitas sistem. Jika permintaan lebih tinggi dari yang dapat ditangani sistem, beberapa data tidak akan bergerak dengan baik melalui jaringan. Serangan ini juga dikenal sebagai serangan berbasis volume atau volumetrik.
- Sama halnya dengan kapasitas jaringan dasar yang berlebihan, serangan dapat menguras aplikasi yang melayani jaringan tertentu, dalam apa yang disebut serangan berbasis aplikasi.
- Cara lain untuk menolak layanan adalah dengan memotong atau menonaktifkan hubungan komunikasi antara dua titik. Banyak pengguna tidak akan dapat menerima layanan, terutama jika tautan itu adalah satu titik yang harus dilalui banyak lalu lintas.
- Penyebab terakhir dari penolakan akses adalah kegagalan perangkat keras atau perangkat lunak. Meskipun mirip dengan kegagalan tautan komunikasi, dalam hal ini masalahnya berkaitan dengan mesin atau program, yang perlindungannya dapat melibatkan konsep seperti toleransi kesalahan.

DOS dapat terjadi dari volume yang berlebihan, aplikasi yang gagal, tautan yang terputus, atau kegagalan perangkat keras atau perangkat lunak.

Pertama kami memeriksa masalah kapasitas yang tidak mencukupi.

Flooding

Bayangkan seorang guru di kelas yang penuh dengan anak-anak berusia enam tahun. Setiap anak menuntut perhatian guru. Mula-mula, guru mendengar seorang anak dan memberikan perhatian kepada anak itu. Kemudian anak kedua menelepon, dan guru memusatkan perhatian pada anak itu sambil mencoba mengingat apa yang dibutuhkan anak pertama. Melihat teriakan itu berhasil, anak-anak tiga, empat, dan lima berteriak memanggil guru, tetapi ini membuat frustrasi anak-anak lain yang juga menuntut perhatian. Tentu saja, setiap anak yang berteriak lebih keras dari yang sebelumnya, dan segera kelas menjadi hiruk-pikuk teriakan anak-anak, sehingga tidak mungkin bagi guru untuk melakukan apa pun kecuali menyuruh mereka semua untuk diam, menunggu giliran, dan bersabarlah (tidak ada yang datang secara alami untuk anak berusia enam tahun). Guru menjadi begitu terbebani dengan tuntutan sehingga satu-satunya solusi adalah mengabaikan semua tuntutan saat ini dan memulai dari awal.

Penyerang dapat mencoba efek kelebihan beban yang sama dengan menyajikan perintah lebih cepat daripada yang dapat ditangani oleh server; server sering mengantri perintah yang tidak terpenuhi pada saat kelebihan beban untuk layanan saat puncak mereda, tetapi jika perintah terus datang terlalu cepat, server akhirnya kehabisan ruang untuk menyimpan permintaan. Serangan seperti itu disebut overload atau Flooding .

Target serangan Flooding dapat berupa aplikasi, seperti sistem manajemen basis data; sistem operasi atau salah satu komponennya, misalnya, file atau server cetak; atau perangkat jaringan seperti router. Alternatifnya, serangan flooding dapat diarahkan terhadap sumber daya, seperti Tabel alokasi memori atau halaman web. Pada hari Michael Jackson meninggal, Google menerima begitu banyak pertanyaan tentang dia sehingga para insinyur Google mengira mereka sedang diserang dan mengambil tindakan mengelak yang, ironisnya, membatasi akses ke layanan berita Google.

Serangan denial-of-service flooding dapat disebut volumetrik, artinya hanya berusaha menjenuhkan atau menguras kapasitas tautan telekomunikasi penting.

Serangan Flooding terjadi dari permintaan yang melebihi kapasitas, dari penyebab yang berbahaya atau alami.

Akses Terblokir

Sebagai analogi fisik lainnya, pertimbangkan kecelakaan lalu lintas yang menghentikan lalu lintas di kedua arah jalan dua lajur yang sibuk. Ketika pengendara mulai berbaris di belakang kecelakaan itu, pada titik tertentu seorang pengemudi menyimpulkan bahwa pendekatan yang tepat adalah menyelinap ke jalur lalu lintas yang akan datang untuk menghindari semua mobil yang berhenti dan, tentu saja, yang lain segera mengikuti. Mereka sampai sejauh kecelakaan dan harus berhenti. Apa yang kemudian terjadi adalah bahwa dua jalur lalu lintas menumpuk di titik kecelakaan di kedua sisi kecelakaan, yang berarti bahwa polisi dan kendaraan darurat lainnya tidak dapat melewati dua garis padat mobil di kedua arah untuk mencapai kecelakaan. Bahkan ketika mobil-mobil cacat didorong keluar dari jalan untuk menghindari kecelakaan, semua jalur dipenuhi mobil yang tidak bisa bergerak karena tidak ada ruang baik di depan maupun di belakang.

Dalam keamanan komputer, penyerang mungkin hanya mencegah layanan berfungsi. Penyerang dapat mengeksploitasi kerentanan perangkat lunak dalam aplikasi dan menyebabkan aplikasi mogok. Atau penyerang dapat mengganggu mekanisme perutean jaringan, mencegah permintaan akses masuk ke server. Namun pendekatan lain bagi penyerang untuk memanipulasi data kontrol akses, menghapus izin akses untuk sumber daya, atau menonaktifkan mekanisme kontrol akses sehingga tidak ada yang bisa disetujui untuk akses.

Kegagalan Akses

Entah jahat atau tidak, perangkat keras dan perangkat lunak gagal dari waktu ke waktu; tentu saja, sepertinya selalu kegagalan yang tidak berbahaya seperti itu hanya terjadi pada saat-saat kritis. Perangkat lunak berhenti bekerja karena cacat, atau perangkat keras aus atau berhenti secara misterius. Kegagalan bisa bersifat sporadis, artinya hilang atau terkoreksi sendiri secara spontan, atau kegagalan bisa permanen, seperti dari komponen yang rusak.

Ini, kemudian, adalah tiga akar ancaman terhadap ketersediaan:

- kapasitas yang tidak mencukupi; kelebihan beban
- akses yang diblokir
- komponen tidak responsif

Penyerang akan mencoba mengaktualisasikan salah satu dari jenis ancaman ini dengan mengeksploitasi kerentanan terhadap mereka. Pada bagian berikutnya kita akan memeriksa beberapa kerentanan potensial ini. Di Studi Kasus 3-16 kami menjelaskan insiden yang dihasilkan dari kombinasi faktor—tidak ada yang berbahaya—termasuk usia komponen, desain jaringan kuno, dan protokol komunikasi yang salah.

Studi Kasus 3-16 : Sistem Rumah Sakit Beth Israel-Deaconess Down

Pada tahun 2002, Pusat Medis Beth Israel-Deaconess Boston diakui oleh Information Week sebagai Peringkat ke-16 dari 500 grup IT inovatif teratas di Amerika Serikat. Pada tahun yang sama rumah sakit mengalami insiden penolakan layanan yang membuat seluruh rumah sakit kembali menggunakan formulir kertas yang telah mereka tinggalkan bertahun-tahun sebelumnya.

Pada hari Rabu, 13 November 2002, gejala pertama yang terlihat adalah bahwa email yang biasanya instan membutuhkan waktu sepuluh detik untuk dikirim. Insinyur jaringan mengamati bahwa satu sakelar jaringan inti dipenuhi oleh lonjakan lalu lintas yang tiba-tiba dari sumber yang tidak dikenal. Untuk mengatasi volume ini dari penyebab yang tidak diketahui, para insinyur mulai menghancurkan jaringan, menutup koneksi untuk menyederhanakan arus lalu lintas dalam jaringan dan juga untuk membantu mengidentifikasi sumber masalah. Kemudian para insinyur akan mengetahui bahwa menutup bagian jaringan sebenarnya memperburuk masalah. Ternyata jaringan meronta-ronta karena sesuatu yang disebut loop protokol pohon rentang. Arsitektur jaringan rumah sakit mencakup banyak sakelar, yang masing-masing menggunakan algoritme pohon rentang, yang pada dasarnya adalah peta rute terpendek ke setiap tujuan yang diketahui dalam jaringan. Setiap sakelar bertanggung jawab untuk menguji koneksinya dan berkomunikasi dengan sakelar tetangga untuk membangun pohon rentangnya sendiri. Tetapi untuk menghindari pengulangan tanpa akhir (simpul A menentukan bahwa cara ke simpul C adalah pergi dulu ke simpul B, tetapi simpul B menganggap jalur yang lebih baik adalah melalui simpul A, sehingga komunikasi berulang tanpa henti antara simpul A dan B), algoritma membatasi perhitungan panjang jalur pada tujuh. Di Beth Israel, satu transfer data yang sangat besar terjebak dalam loop yang lebih panjang yang sangat memperlambat lalu lintas. Tetapi ketika para insinyur mulai memotong sirkuit, tindakan itu menyebabkan semua sakelar mencoba menghitung ulang jalur pohon rentang mereka, yang pada gilirannya memperlambat lalu lintas dan menyebabkan para insinyur memutuskan lebih banyak tautan, yang pada gilirannya menyebabkan lebih banyak penghitungan ulang sakelar.

Bagian penting dari masalah adalah bahwa desain jaringan sesuai untuk tahun 1996, ketika pertama kali dipasang, tetapi arsitektur jaringan belum ditingkatkan untuk memperhitungkan ekspansi besar, karena Beth Israel membawa beberapa rumah sakit regional untuk bergabung dengan jaringan TI-nya. , atau untuk kemajuan teknologi, karena router menggantikan sakelar di segmen jaringan besar dengan konektivitas yang kompleks. Jaringan tahun 1996 berfungsi secara memadai pada tahun 2002 pada saat tekanan rendah, tetapi ledakan besar lalu lintas jaringan memFlooding i jaringan, menolak akses cepat ke semua pengguna.

Permintaan tes laboratorium, grafik catatan pasien, pesanan resep, hasil x-ray digital, catatan tagihan, semua data yang biasanya dapat ditangani dengan mudah secara elektronik tiba-tiba berhenti bekerja. Pada hari Kamis 14 November administrator

memutuskan untuk menyerahkan seluruh sistem elektronik untuk memungkinkan insinyur jaringan akses penuh ke jaringan. Rumah sakit kembali menggunakan formulir kertas, jelas lebih lambat dan lebih rumit. Namun meskipun demikian, jaringan tersebut sangat padat sehingga sulit untuk memetakan konektivitas dari 25.000 node-nya. Rumah sakit memanggil pemasok peralatan jaringannya, Cisco, untuk membantu mendesain ulang dan mengimplementasikan kembali jaringannya. Selama akhir pekan, insinyur rumah sakit dan Cisco menguji komponen dan segmen dan mengganti sakelar dengan router yang tidak terkena masalah pohon rentang. Pada Senin 18 November, jaringan baru berkinerja andal dan pengguna kembali menggunakan jaringan TI alih-alih kertas.

Insiden ini terjadi pada tahun 2002, tetapi kerentanannya tetap relevan. Organisasi enggan untuk mendesain ulang dan mengimplementasikan kembali jaringan kompleks yang telah berkembang dari waktu ke waktu; biaya dan ketidaknyamanan adalah dua motivator kuat untuk mempertahankan status quo. Tetapi ketika anggota staf pindah, orang-orang melupakan arsitektur jaringan dan alasan desain, sehingga pemeliharaan sering kali terdiri dari meninggalkan segala sesuatunya sendiri sebanyak mungkin. Seperti yang ditunjukkan oleh contoh ini, strategi itu memiliki rentang hidup yang terbatas dan sering kali menimbulkan konsekuensi bencana.

Seperti yang ditunjukkan Studi Kasus 3-16, penolakan layanan dapat muncul dari penyebab yang berbahaya atau tidak berbahaya. Pada awal insiden, mungkin sulit untuk membedakan antara serangan yang disengaja dan kegagalan perangkat keras atau perangkat lunak acak. Selanjutnya, seperti dalam situasi ini, beberapa penyebab, tidak ada satu pun yang cukup menyebabkan masalah, dapat berinteraksi secara serius. Namun mencari penyebab individu dapat menjadi tantangan bagi administrator, terutama ketika dihadapkan dengan masalah langsung mencoba untuk mendapatkan sistem yang gagal beroperasi kembali.

Dari tiga penyebab dasar layanan yang gagal—kurangnya kapasitas atau kelebihan beban, akses yang diblokir, dan komponen yang tidak responsif—kita sekarang bergerak untuk mengidentifikasi kerentanan yang dapat menyebabkan kegagalan ini.

Lebih Lengkap tentang Serangan Flooding

Jenis serangan denial-of-service berbahaya yang paling umum adalah Flooding.Sepertinya memFlooding i korban akan membutuhkan sumber daya yang luar biasa. Namun, mengeksploitasi kelemahan dalam protokol dan utilitas jaringan dapat menghasilkan penolakan layanan; pada kenyataannya, beberapa baris kode dari satu komputer dapat menjatuhkan entitas jaringan yang tampaknya lebih kuat. Pada bagian ini kita memeriksa bagaimana serangan Flooding dirakit.

Sumber Daya Tidak Memadai

Dalam contoh kita tentang guru dan anak-anak berusia enam tahun, guru tidak dapat menangani tuntutan dari semua siswa: satu per satu, mungkin, tetapi tidak sekaligus. Satu guru dengan dua atau tiga siswa mungkin bisa mengatasinya, atau sepuluh guru dengan tiga puluh siswa, tetapi tidak satu lawan tiga puluh. Demikian pula dengan sistem komputasi, penyerang dapat mencoba mengkonsumsi sejumlah besar sumber daya yang langka.

Membanjiri korban pada dasarnya adalah serangan yang tidak canggih, meskipun cara melakukan flooding bisa menjadi canggih. Cara lain untuk menolak layanan adalah dengan memblokir akses ke sumber daya, yang akan kami pertimbangkan selanjutnya.

Kapasitas Tidak Memadai

Jika bandwidth penyerang lebih besar dari korban, penyerang dapat memFlooding i korban dengan asimetri. Seorang korban selalu berpotensi rentan terhadap penyerang dengan lebih banyak sumber daya. Contoh sumber daya yang tidak mencukupi mungkin slot dalam Tabel koneksi jaringan, ruang dalam buffer, atau siklus prosesor. Flooding terjadi karena bandwidth yang masuk tidak mencukupi atau sumber daya—perangkat keras, daya komputasi, perangkat lunak, atau kapasitas Tabel —tidak memadai.

Denial of service terutama terlihat dalam serangan jaringan, di mana penyerang dapat menghabiskan terlalu banyak bandwidth jaringan yang tersedia. Kami mempertimbangkan kelelahan kapasitas jaringan berikutnya.

Serangan Jaringan Flooding Disebabkan oleh Kode Berbahaya

Serangan penolakan layanan yang paling primitif adalah memFlooding i koneksi. Jika penyerang mengirimi Anda data sebanyak yang dapat ditangani oleh sistem komunikasi Anda, Anda tidak dapat menerima data lainnya. Bahkan jika paket sesekali mencapai Anda dari orang lain, komunikasi kepada Anda akan sangat terganggu. Ironisnya, masalah ini diperparah oleh kekokohan protokol TCP: Jika, karena kemacetan, paket 1 dan 2 tertunda tetapi paket 3 berhasil lolos terlebih dahulu, pengendali protokol akan melihat bahwa 1 dan 2 hilang. Penerima menerima dan menyimpan paket 3, tetapi pengirim dapat mengirim ulang paket 1 dan 2, yang menambah kemacetan.

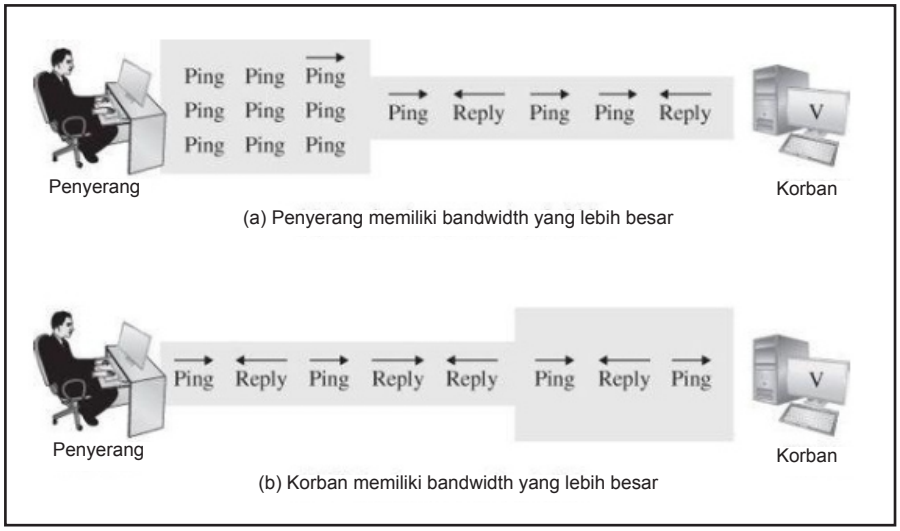
Serangan yang lebih canggih menggunakan atau menyalahgunakan elemen protokol Internet. Selain TCP dan UDP, ada protokol kelas ketiga, yang disebut ICMP atau Internet Control Message Protocols. Biasanya digunakan untuk diagnostik sistem, protokol ini tidak terkait aplikasi pengguna. Protokol ICMP termasuk:

- ping, yang meminta tujuan untuk mengembalikan balasan, dimaksudkan untuk menunjukkan bahwa sistem tujuan dapat dijangkau dan berfungsi
- echo, yang meminta tujuan untuk mengembalikan data yang dikirim ke sana, dimaksudkan untuk menunjukkan bahwa tautan koneksi dapat diandalkan (ping sebenarnya adalah versi echo)
- destination unreachable, yang menunjukkan bahwa alamat tujuan tidak dapat diakses
- sumber quench, yang berarti tujuan menjadi jenuh dan sumber harus menunda pengiriman paket untuk sementara waktu

Protokol ini memiliki kegunaan penting untuk manajemen jaringan. Tetapi mereka juga dapat digunakan untuk menyerang suatu sistem. Protokol ditangani dalam tumpukan jaringan, sehingga serangan mungkin sulit dideteksi atau diblokir pada host penerima. Tetapi keanehan atau kelalaian dalam protokol atau implementasinya dapat membuka jalan bagi penyerang untuk mengeksploitasi kelemahan untuk memFlooding i kode yang mendukung fungsi protokol. Kami memeriksa bagaimana protokol ini dapat digunakan untuk menyerang korban. Dan kami menekankan bahwa paket tidak diautentikasi: Penyerang dapat menggunakan paket ping atau gema untuk menjenuhkan jaringan sama mudahnya seperti administrator menggunakannya untuk mengelola kinerja jaringan.

Ping Kematian

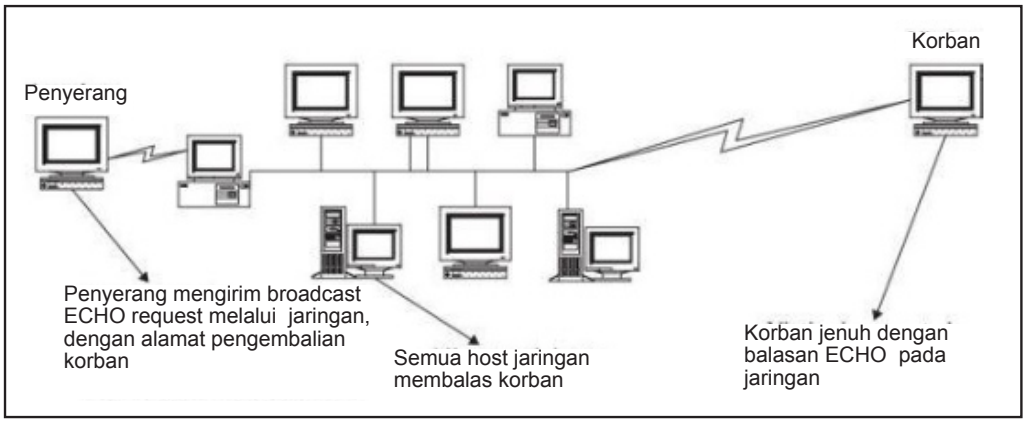
Ping Kematian (Ping of death) adalah serangan sederhana, menggunakan perintah ping yang biasanya digunakan untuk menguji waktu respons dari sebuah host. Karena ping membutuhkan penerima untuk menanggapi paket, yang perlu dilakukan penyerang hanyalah mengirim banyak ping ke korban yang dituju. Serangan dibatasi oleh bandwidth terkecil pada rute serangan, seperti yang ditunjukkan pada Gambar 3-18. Jika penyerang berada pada koneksi 10 megabyte (MB) dan jalur ke korban 100 MB atau lebih, secara matematis penyerang saja tidak dapat memFlooding i korban. Tetapi serangan berhasil jika angkanya dibalik: Penyerang pada koneksi 100 MB pasti dapat memFlooding i korban 10 MB. Paket ping akan memenuhi bandwidth korban.



Gambar 3-18 Serangan Ping

Smurf

Serangan smurf adalah variasi dari serangan ping. Ia menggunakan kendaraan yang sama, paket ping, dengan dua putaran ekstra. Pertama, penyerang memilih jaringan korban tanpa disadari yang menjadi kaki tangannya. Penyerang memalsukan alamat sumber dalam paket ping sehingga seolah-olah berasal dari korban, yang berarti penerima akan merespons korban.



Gambar 3-19 Serangan Smurf

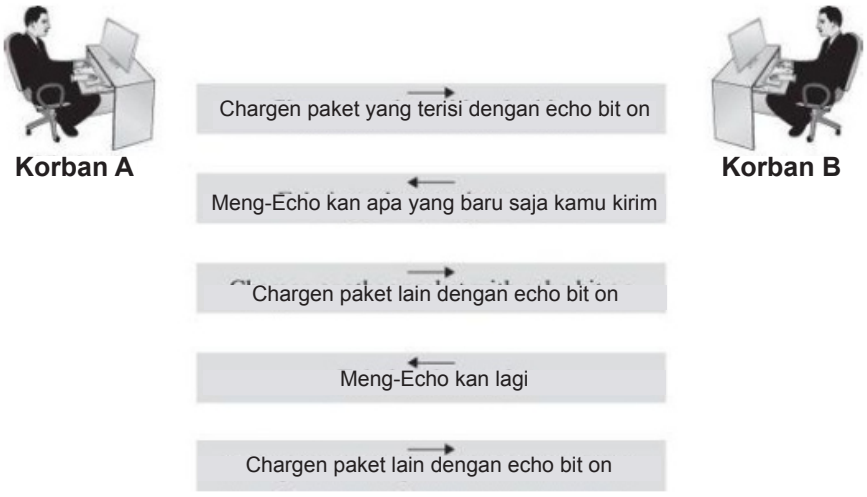
Kemudian, penyerang mengirimkan permintaan ini ke jaringan dalam mode siaran dengan menyetel byte terakhir alamat ke semua 1; paket mode siaran didistribusikan ke semua host di subnetwork.

Serangan tersebut digambarkan pada Gambar 3-19, menunjukkan serangan siaran tunggal yang dipantulkan kembali pada korban. Dengan cara ini penyerang menggunakan seluruh subnetwork untuk melipatgandakan efek serangan.

Echo–Chargen

Serangan echo-charge bekerja antara dua host. Chargen adalah protokol ICMP yang menghasilkan aliran paket untuk menguji kapasitas jaringan. Echo adalah protokol ICMP lain yang digunakan untuk pengujian; host yang menerima gema mengembalikan semua yang diterimanya ke pengirim.

Penyerang memilih dua korban, A dan B, dan kemudian menyiapkan proses charge pada host A yang menghasilkan paketnya sebagai paket gema dengan tujuan host B. Jadi, A memFlooding i B dengan paket gema. Tetapi karena paket-paket ini meminta penerima untuk mengirimkannya kembali ke pengirim, host B membalas dengan mengembalikannya ke host A. Seperti yang ditunjukkan pada Gambar 3-20, rangkaian ini menempatkan infrastruktur jaringan A dan B ke dalam loop tanpa akhir, karena A menghasilkan string gema yang B patuh kembali ke A, seperti dalam permainan tenis. Atau, penyerang dapat menjadikan B sebagai alamat sumber dan tujuan dari paket pertama, sehingga B menggantung dalam satu lingkaran, terus-menerus membuat dan membalas pesannya sendiri.



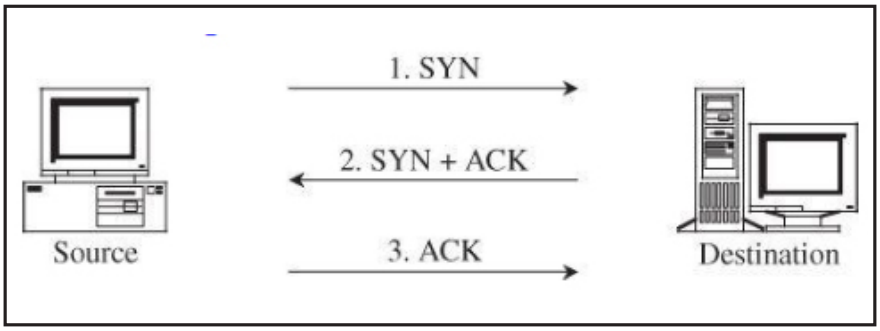
Gambar 3-20 Serangan Echo–Chargen

SYN Flood

Serangan penolakan layanan populer lainnya adalah Flooding SYN. Serangan ini menggunakan paket protokol TCP, membuat sifat berorientasi sesi dari protokol ini bekerja melawan korban. Untuk protokol seperti Telnet atau SMTP, rekan protokol membuat koneksi virtual, yang disebut sesi, untuk menyinkronkan sifat interaksi perintah-respons bolak-balik. Sesi dibuat dengan jabat tangan TCP tiga arah. Setiap paket TCP memiliki bit flag, salah satunya dilambangkan SYN (sinkronisasi) dan satu dilambangkan ACK (acknowledge). Pertama, untuk memulai koneksi TCP, originator mengirimkan paket dengan bit SYN aktif. Kedua, jika penerima siap untuk membuat koneksi, ia membalas dengan paket dengan bit SYN dan ACK aktif. Akhirnya, pihak pertama menyelesaikan pertukaran untuk mendemonstrasikan saluran komunikasi yang jelas dan lengkap dengan mengirimkan paket dengan bit ACK aktif, seperti yang ditunjukkan pada Gambar 3-21.

Kadang-kadang paket hilang atau rusak dalam transmisi. Tujuan (yang kita sebut penerima) mempertahankan antrian yang disebut koneksi SYN_RECV, melacak item-item yang SYN–ACK telah

dikirim tetapi belum ada ACK yang sesuai yang diterima. Biasanya, koneksi ini selesai dalam waktu singkat. Jika paket SYN–ACK (2) atau ACK (3) hilang, pada akhirnya host tujuan akan mengatur waktu koneksi yang tidak lengkap dan membuangnya dari antrian tunggu.



Gambar 3-21 Jabat Tangan TCP Tiga Arah

Penyerang dapat menolak layanan ke target dengan mengirimkan banyak permintaan SYN, yang mana target merespons dengan benar dengan SYN-ACK; namun, penyerang tidak pernah membalas dengan ACK untuk menyelesaikan koneksi, sehingga mengisi antrian SYN_RECV korban. Biasanya, antrian SYN_RECV cukup kecil, menampung 10 atau 20 entri. Karena potensi penundaan perutean di Internet, waktu penahanan tipikal untuk antrian SYN_RECV bisa beberapa menit. Jadi penyerang hanya perlu mengirim permintaan SYN baru setiap beberapa detik, dan antrian akan terisi.

Penyerang yang menggunakan pendekatan ini biasanya melakukan satu hal lagi: Mereka memalsukan alamat pengirim yang tidak ada dalam paket SYN awal. Mengapa? Karena dua alasan. Pertama, penyerang tidak ingin mengungkapkan alamat sumber sebenarnya jika seseorang harus memeriksa paket dalam antrian SYN_RECV untuk mencoba mengidentifikasi penyerang. Kedua, penyerang ingin membuat paket SYN berbahaya tidak dapat dibedakan dari paket SYN yang sah untuk membuat koneksi nyata. Memilih alamat sumber yang berbeda (palsu) untuk masing-masing membuatnya unik, seperti lalu lintas biasa. Paket SYN-ACK ke alamat yang tidak ada menghasilkan respons ICMP Destination Unreachable, tetapi ini bukan ACK yang menunggu koneksi TCP. (TCP dan ICMP adalah suite protokol yang berbeda, jadi balasan ICMP tidak selalu kembali ke pengendali TCP pengirim.)

Serangan ini menyalahgunakan fitur sah dari protokol jaringan untuk membanjiri korban, tetapi fitur tersebut tidak dapat dinonaktifkan karena memiliki tujuan yang diperlukan dalam rangkaian protokol. Namun, kapasitas jaringan yang luar biasa bukan satu-satunya cara untuk menolak layanan. Di bagian selanjutnya kita akan memeriksa serangan yang menghabiskan sumber daya lain yang tersedia.

3.4.2 Flooding Jaringan dengan Resource Exhaustion

Komputer mendukung banyak aplikasi dengan membagi waktu antar aplikasi; penelitian sistem operasi telah membantu orang merancang algoritme yang efektif untuk memutuskan berapa banyak (berapa proporsi) waktu pemrosesan untuk dialokasikan ke aplikasi mana. Berpindah dari satu aplikasi ke aplikasi lain, yang disebut pengalihan konteks, membutuhkan waktu dan memori karena status aplikasi saat ini disimpan dan status aplikasi berikutnya dimuat ulang. Nilai register harus

ditulis ke memori, aktivitas asinkron yang luar biasa harus diselesaikan, dijatuhkan atau direkam, dan memori harus dipertahankan atau dibebaskan. Jika ada sedikit proses aktif dan sedikit sakelar konteks, overhead untuk setiap sakelar dapat diabaikan, tetapi dengan meningkatnya jumlah proses aktif, proporsi waktu yang dihabiskan dalam pengalihan konteks juga bertambah, yang berarti proporsi waktu untuk komputasi aktual berkurang. Dengan terlalu banyak proses, sistem dapat memasuki keadaan yang disebut thrashing, di mana kinerjanya gagal karena pengalihan konteks yang hampir terus menerus.

Waktu bukan satu-satunya sumber daya yang bisa habis. Buffer untuk email masuk dapat kewalahan oleh Flooding pesan masuk yang tiba-tiba. Logging dan file log dapat diFlooding i oleh sejumlah besar kesalahan atau kondisi kesalahan yang harus ditangani. Buffer untuk merakit kembali komunikasi yang terfragmentasi juga bisa habis.

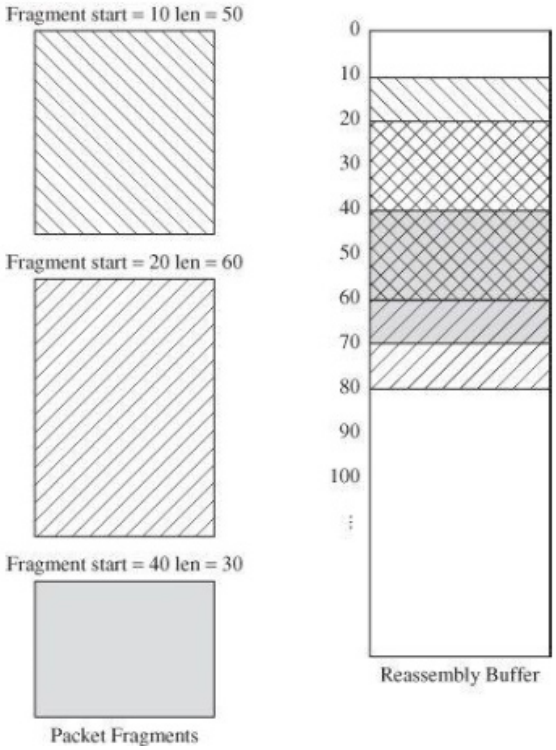
Bahkan identifikasi dan otentikasi dapat menjadi rentan dalam serangan kelelahan. Untuk melindungi dari serangan tebak-tebakan otomatis, beberapa layanan otentikasi menonaktifkan akses akun untuk sementara atau permanen setelah beberapa nomor, seperti tiga atau lima, upaya login yang gagal. Dengan demikian, pengguna jahat dapat memblokir akses dengan berulang kali gagal login sebagai korban.

Fragmentasi IP: Teardrops

Serangan tetesan air mata menyalahgunakan fitur yang ironisnya dimaksudkan untuk meningkatkan komunikasi jaringan. Datagram IP jaringan adalah objek dengan panjang variabel. Untuk mendukung aplikasi dan kondisi yang berbeda, protokol datagram mengizinkan satu unit data untuk difragmentasi, yaitu, dipecah menjadi beberapa bagian dan ditransmisikan secara terpisah. Setiap fragmen menunjukkan panjang dan posisi relatifnya dalam unit data. Ujung penerima menyusun kembali fragmen-fragmen itu menjadi satu unit data tunggal.

Seperti yang ditunjukkan pada Gambar 3-22, dalam serangan tetesan air mata, penyerang mengirimkan serangkaian datagram yang tidak dapat disatukan dengan benar. Satu datagram mungkin mengatakan itu adalah posisi 0 untuk panjang 60 byte, posisi lain 30 untuk 90 byte, dan posisi lain 41 untuk 173 byte. Ketiga bagian ini tumpang tindih, sehingga tidak dapat dipasang kembali dengan benar. Dalam kasus ekstrim, sistem operasi terkunci dengan unit data parsial yang tidak dapat dipasang kembali, sehingga menyebabkan penolakan layanan.

Penyebab lain penolakan layanan didasarkan pada perutean jaringan: Jika Tabel perutean tidak lagi mengarah ke suatu situs, situs tersebut secara efektif tidak dapat dijangkau. Kami menjelaskan serangan perutean berikutnya.



Gambar 3-22 Serangan Teardrops

3.4.3 Denial of Service dengan Kegagalan Addressing

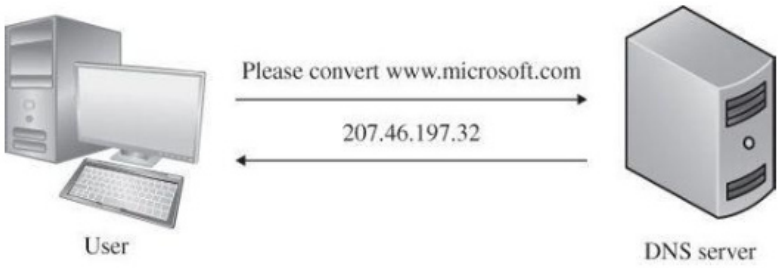
Seperti yang kami jelaskan sebelumnya, cara lain penyerang dapat menolak layanan adalah dengan mencegah akses, secara fisik atau logis. Di bagian ini kami mempertimbangkan cara untuk mencegah data sampai ke korban. Anda dapat melihat bahwa siapa pun yang dapat memutuskan, mengganggu, atau membebani kapasitas sistem dapat menolak layanan. Ancaman fisik agak jelas dan dijelaskan kemudian dalam bab ini. Kami mempertimbangkan sebagai gantinya beberapa serangan elektronik yang dapat menyebabkan penolakan layanan. Di bagian ini kita melihat cara layanan dapat ditolak dengan sengaja atau tidak sengaja.

Misrouting adalah serangan yang mencapai dua tujuan. Misalkan alamat rumah tetangga Anda adalah 217 Main Street, tetapi Anda mencatat nomor di rumahnya dan menempatkan 217 di atas rumah Anda sendiri. Kemudian, semua surat tetangga Anda akan dikirim ke rumah Anda dan tetangga Anda tidak akan mendapatkannya. Anda akan diposisikan secara ideal untuk memeriksa (dan mungkin membuka) semua yang seharusnya diterima tetangga Anda, dan Anda akan memblokir semua pengiriman ke tetangga Anda. Perubahan pengalamatan ini akan memfasilitasi intersepsi dan penolakan layanan. Situasi serupa terjadi dengan alamat jaringan, seperti yang kami jelaskan sekarang.

Pemalsuan DNS

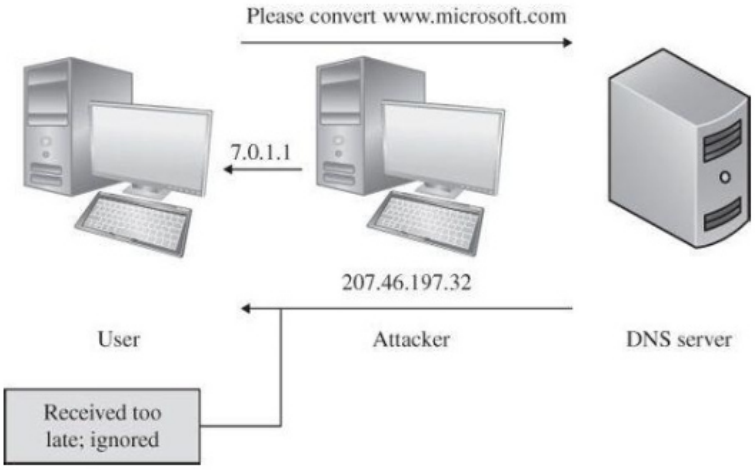
Inti dari pengalamatan Internet adalah protokol yang disebut protokol DNS atau Domain Name System. DNS adalah database terjemahan nama Internet ke alamat, dan protokol DNS menyelesaikan nama ke alamat. Untuk efisiensi, server DNS membuat cache dari nama domain yang baru saja

digunakan; dengan serangan yang disebut keracunan DNS, penyerang mencoba memasukkan entri yang tidak akurat ke dalam cache itu sehingga permintaan di masa mendatang diarahkan ke alamat yang dipilih penyerang. Permintaan dan respons DNS standar ditunjukkan pada Gambar 3-23, di mana pengguna meminta terjemahan URL microsoft.com, dan server nama merespons dengan alamat 207.46.197.32.



Gambar 3-23 Menyelesaikan Nama Domain ke Alamat

Layanan DNS diimplementasikan pada server jarak jauh, sehingga serangan man-in-the-middle melibatkan penyerang yang mencegat dan membalas kueri sebelum server DNS sebenarnya dapat merespons. Situasi seperti itu, yang disebut DNS spoofing, ditunjukkan pada Gambar 3-24. Dalam contoh itu, penyerang dengan cepat merespons dengan alamat 7.0.1.1 (mungkin alamat yang dikendalikan oleh penyerang). Dengan perubahan itu, penyerang dapat masuk ke tengah komunikasi pengguna dengan www.microsoft.com, meneruskan apa pun yang diinginkan penyerang ke situs web Microsoft yang sebenarnya. Browser pengguna mengabaikan respons yang benar dari server DNS yang datang setelah browser menerima alamat palsu dari penyerang.

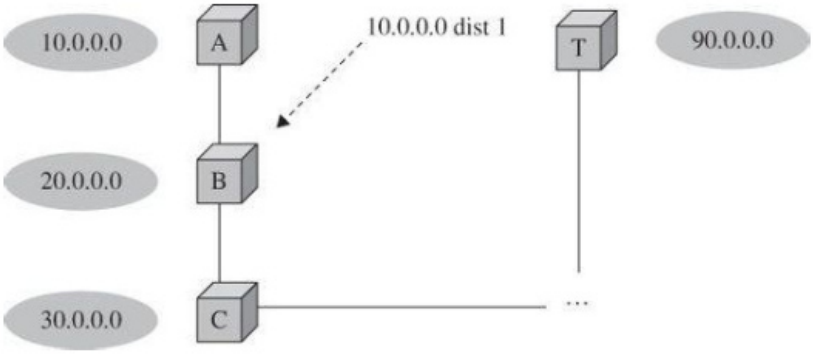


Gambar 3-24 Resolusi Alamat yang Melibatkan DNS Spoofing

Perutean Ulang Routing

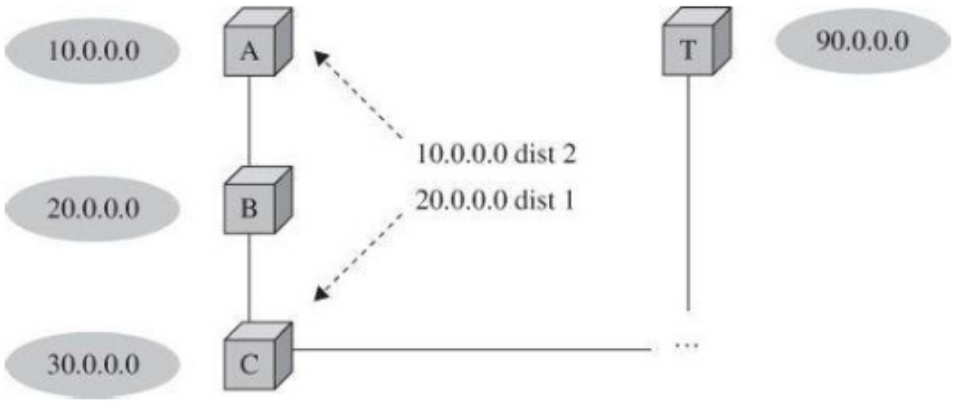
Salah satu contoh serangan man-in-the-middle melibatkan satu node yang mengarahkan ulang jaringan sehingga semua lalu lintas mengalir melalui node yang menyerang, yang mengarah ke potensi intersepsi. Router jaringan adalah konfederasi longgar dari komponen yang saling percaya yang mengatur pengiriman semua data melalui jaringan, termasuk Internet. Penjelasan man-in-the-middle untuk router agak rumit, jadi kami menyajikan versi sederhana yang menyoroti peran tengah; untuk deskripsi yang lebih lengkap tentang fenomena ini.

Setiap router mengirim pesan ke router lain, daftar alamat yang memiliki jalur; router lain kemudian menambahkan jalur mereka dan meneruskan daftar yang diperluas ke router lain juga. Dengan cara ini, semua router mempelajari koneksi router lain. Pada Gambar 3-25, empat router mengontrol empat subnet: A mengontrol subnet 10.0.0.0; B, 20.0.0.0, dan seterusnya. A berdekatan dengan B, B berdekatan dengan C, dan T adalah router lain yang tidak berdekatan dengan salah satu dari tiga lainnya. A mengiklankan ke tetangganya bahwa jaraknya 1 dari mesin mana pun di subnet 10.0.0.0.



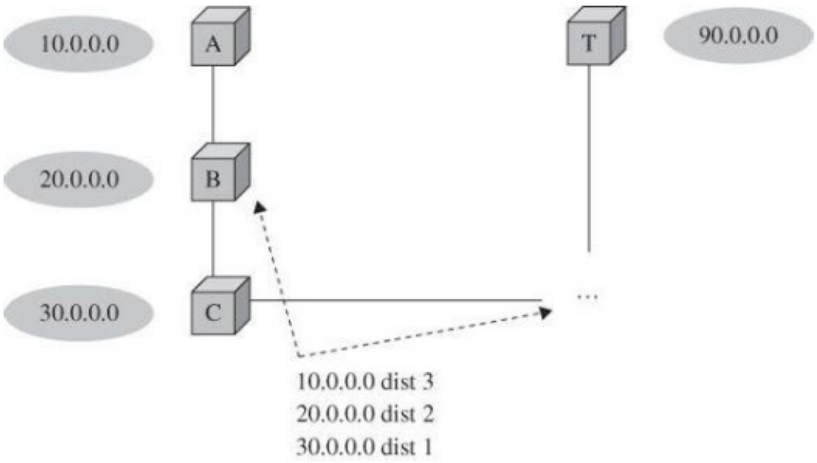
Gambar 3-25 Router Mengiklankan Subnetnya

Karena B baru mengetahui bahwa router A hanya berjarak 1 dari subnet 10.0.0.0, B mengiklankan ke tetangganya A dan C bahwa jarak 1 dari subnetnya sendiri dan jarak 2 dari subnet 10.0.0.0, seperti yang ditunjukkan pada Gambar 3-26. Tentu saja, A tidak peduli bahwa ia dapat mencapai alamat 10.0.0.0 dengan melewati B; itu akan menjadi loop yang tidak masuk akal, tetapi itu mencatat bahwa B adalah jalur terdekat ke alamat 20.0.0.0.



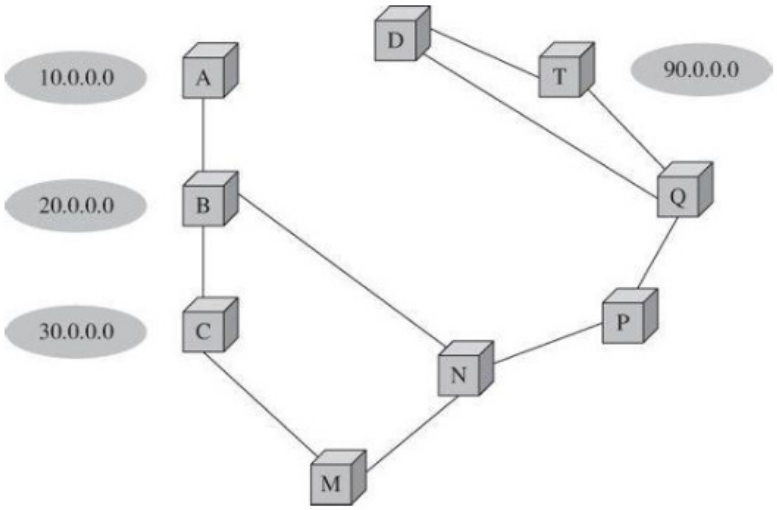
Gambar 3-26 Router Mengiklankan Subnetnya Sendiri dan Subnet Tetangganya

Gambar 3-27 menunjukkan bagaimana C mengambil apa yang baru saja dipelajari dari B dan menyiarkannya ke router lain yang berdekatan dengannya. Dengan cara ini, router mengumumkan kemampuannya di seluruh jaringan. Seiring waktu, router berbagi informasi yang merinci topologi jaringan lengkap. Setiap router memelihara Tabel tujuan dan langkah selanjutnya, jadi jika C memiliki sesuatu untuk subnetwork 10.0.0.0, Tabel nya akan menunjukkan bahwa ia harus meneruskan aliran data tersebut ke B.



Gambar 3-27 Router Menyebarkan Informasi Routing

Pada Gambar 3-28 kami sedikit memperumit adegan dengan menambahkan lebih banyak router; untuk kesederhanaan kami tidak menunjukkan subnetwork mereka. Semua router ini akan mengiklankan konektivitas mereka, dari mana mereka dapat menentukan jalur terpendek antara setiap pasangan titik. Perhatikan bahwa A agak terisolasi dari T; jalur terpendeknya adalah B-N-P-Q-T.



Gambar 3-28 Diagram Konektivitas Router yang Lebih Kompleks

Router beroperasi pada kepercayaan implisit; apa yang dilaporkan router diyakini benar. Namun, router terkadang mengalami malfungsi atau administratornya memasukkan data yang tidak akurat, sehingga Tabel perutean dapat rusak karena penyebab yang tidak berbahaya (dan berbahaya). Dalam contoh kita, jika router A mengiklankan jarak 1 dari subnetwork 90.0.0.0, sebagian besar lalu lintas ke subnetwork tersebut akan dirutekan ke A, karena jarak tersebut akan mengalahkan jalur apa pun kecuali T itu sendiri. Jika A menerima lalu lintas itu, ia dapat dengan mudah mencegat dan memodifikasi lalu lintas apa pun ke jaringan itu, sehingga router jahat dalam jaringan dapat memicu serangan man-in-the-middle dengan cara ini.

Router Mengambil alih Jaringan

Pada konferensi Defcon 2008, sebagian besar peserta tidak menyadari bahwa dua peneliti telah mengubah rute jaringan nirkabel konferensi melalui peralatan mereka. Para peneliti, Pilosov dan Kapela [PIL08] menjelaskan dan mendemonstrasikan serangan mereka. Meskipun serangannya lebih mendetail daripada yang ingin kami sajikan di sini, serangan ini memperluas pendekatan yang baru saja dijelaskan. Makalah lain (seperti [HEP09, KUH07, dan BEL89]) telah membahas kerentanan serupa.

Router mengomunikasikan jalur yang tersedia dengan protokol BGP (Border Gateway), yang kompleks, sehingga serangan terhadapnya canggih tetapi tentu saja layak. Detail seperti waktu dan nomor urut harus ditangkap dan digunakan dengan benar agar pembaruan BGP dapat dikenali dan diterima oleh seluruh jaringan. Selain itu, serangan pada protokol bergantung pada perangkat yang berada di "ujung" subjaringan, yaitu, terhubung langsung ke dua subjaringan yang berbeda. Meskipun penyerang dapat mewakili berada di tepi subnetwork lokal, misalnya, jaringan nirkabel di hotel atau laboratorium, lebih sulit untuk mewakili berada di tepi subnetwork yang lebih besar, misalnya, meniru ISP dalam koneksi langsung ke Internet. Namun, penyerang yang berhasil dapat mengalihkan, membaca, menyalin, memodifikasi, atau menghapus semua lalu lintas jaringan yang diserang.

Source Routing dan Spoofing Addressing

Lalu lintas internet biasanya menempuh rute terbaik yang tersedia; yaitu, setiap router menentukan jalur terbaik berikutnya (disebut hop berikutnya) untuk mengarahkan unit data. Namun, pengirim, menggunakan proses yang disebut perutean sumber, dapat menentukan beberapa atau semua titik perantara di mana unit data ditransfer. Dengan perutean sumber yang ketat, jalur lengkap dari sumber ke tujuan ditentukan; dengan perutean sumber yang longgar, titik perantara tertentu (beberapa atau semua) yang diperlukan ditentukan.

Salah satu penggunaan perutean sumber adalah untuk menguji atau memecahkan masalah router dengan memaksa lalu lintas untuk mengikuti jalur tertentu yang kemudian dapat dilacak oleh seorang insinyur. Penggunaan perutean sumber yang lebih kejam adalah memaksa data mengalir melalui perute jahat atau tautan jaringan. Jelas, menambahkan perutean sumber ke aliran data memungkinkan orang di tengah untuk memaksa lalu lintas mengalir melalui routernya. Karena potensi penyalahgunaannya, perutean sumber longgar diblokir oleh banyak perute Internet.

3.4.4 PengalihanTraffic

Seperti yang kita lihat sebelumnya, pada lapisan jaringan, router adalah perangkat yang meneruskan lalu lintas dalam perjalanannya melalui jaringan perantara antara jaringan host sumber dan jaringan tujuan. Jadi jika penyerang dapat merusak perutean, lalu lintas dapat hilang.

Router menggunakan algoritma yang kompleks untuk memutuskan bagaimana merutekan lalu lintas. Apa pun algoritmanya, mereka pada dasarnya mencari jalur terbaik (di mana "terbaik" diukur dalam beberapa kombinasi jarak, waktu, biaya, kualitas, dan sejenisnya). Router hanya mengetahui router yang mereka gunakan untuk berbagi koneksi jaringan langsung, dan mereka menggunakan protokol gateway untuk berbagi informasi tentang kemampuan mereka. Setiap router memberi

tahu tetangganya tentang seberapa baik ia dapat mencapai alamat jaringan lain. Karakteristik ini memungkinkan penyerang untuk mengganggu jaringan.

Untuk melihat caranya, perlu diingat bahwa terlepas dari kecanggihannya, router hanyalah sebuah komputer dengan dua atau lebih antarmuka jaringan. Misalkan router mengiklankan ke tetangganya bahwa ia memiliki jalur terbaik ke setiap alamat lain di seluruh jaringan. Segera semua router akan mengarahkan semua lalu lintas ke satu router itu. Satu router mungkin keFlooding an, atau mungkin hanya menurunkan banyak lalu lintasnya. Dalam kedua kasus, banyak lalu lintas tidak pernah sampai ke tujuan yang diinginkan.

Seperti yang kami sebutkan sebelumnya, router saling percaya untuk menyediakan data yang akurat. Kadang-kadang, karena korupsi yang tidak berbahaya, router akan mengirim data yang salah, tetapi kegagalan sporadis ini memiliki efek lokal dan sembuh sendiri dari waktu ke waktu berkat keandalan jaringan. Namun, router yang sengaja menyesatkan (atau perangkat yang menyamar sebagai router) dapat bertahan karena kepercayaan implisit. Seperti yang Anda ketahui, tindakan pencegahan standar untuk mengecualikan penipu adalah identifikasi dan otentikasi. Tetapi untuk efisiensi, protokol komunikasi router dirancang tanpa otentikasi. Hanya sekarang langkah-langkah otentikasi ditambahkan ke protokol router.

3.4.5 Serangan DNS

Serangan penolakan layanan terakhir kami sebenarnya adalah kelas serangan berdasarkan konsep server nama domain. Server nama domain meminta server nama lain untuk menyelesaikan nama domain yang tidak diketahuinya. Untuk efisiensi, ia menyimpan jawaban yang diterimanya sehingga dapat mengubah nama itu lebih cepat di masa mendatang. Alamat yang dipetakan oleh server DNS dapat disimpan selama berminggu-minggu atau berbulan-bulan.

Flaw pada Server Nama Aplikasi Perangkat Lunak

Dalam implementasi Unix yang paling umum, server nama menjalankan perangkat lunak yang disebut Berkeley Internet Name Domain, atau BIND, atau bernama (singkatan untuk "nama daemon"). BIND memiliki banyak kekurangan, termasuk buffer overflow yang sekarang dikenal. Dengan menyalip server nama atau menyebabkannya menyimpan entri palsu, penyerang dapat mengalihkan perutean lalu lintas apa pun, dengan implikasi yang jelas untuk penolakan layanan.

Serangan Domain Top Level

Cara lain untuk menolak layanan melalui kegagalan resolusi alamat melibatkan melumpuhkan sistem DNS Internet itu sendiri. Pada bulan Oktober 2002, Flooding besar lalu lintas memFlooding i server DNS domain tingkat atas Internet, server yang membentuk dasar dari struktur pengalamatan Internet. Ada 13 server domain tingkat atas yang tersebar di seluruh dunia; server ini menerjemahkan tingkat teratas, atau bagian terakhir dari alamat jaringan: .com, .edu, .fr, .uk, .org, atau .biz bagian dari URL. Pada serangan tahun 2002, kira-kira setengah dari Flooding lalu lintas datang dari hanya 200 alamat. Meskipun beberapa orang berpikir masalahnya adalah seperangkat firewall yang salah konfigurasi, tidak ada yang tahu pasti apa yang menyebabkan serangan itu, dan bahkan apakah itu serangan atau insiden anomali.

Sekali lagi pada tahun 2007, hal serupa terjadi. Pada tanggal 6 Februari 2007, server nama root DNS diserang dengan dua serangan penolakan layanan besar-besaran selama total enam jam. Kali ini jelas merupakan serangan, setidaknya sebagian berasal dari kawasan Asia-Pasifik [ICA07]. Dalam situasi ini juga, dampak serangan berkurang secara signifikan karena, antara tahun 2002 dan 2007, Internet mulai menggunakan desain baru untuk server nama root.

Disebut anycast, teknologi ini memungkinkan fungsi pencarian tersebar di banyak komputer, bahkan ratusan. Dengan demikian, serangan pada satu server DNS, atau bahkan sejumlah kecil server, memiliki dampak yang kecil.

Sebuah serangan pada bulan Maret 2005 menggunakan cacat pada firewall Symantec untuk memungkinkan perubahan dalam catatan DNS yang digunakan pada mesin Windows. Namun, tujuan serangan ini bukanlah penolakan layanan. Dalam serangan ini, cache DNS yang diracuni mengarahkan pengguna ke situs iklan yang menerima uang dari klien setiap kali pengguna mengunjungi situs tersebut. Namun demikian, serangan itu juga mencegah pengguna mengakses situs yang sah.

Serangan ini mencoba untuk menolak layanan dengan membatasi kemampuan sistem untuk menyelesaikan alamat. Karena resolusi alamat didistribusikan di Internet, serangan ini cenderung lebih efektif menyebabkan penolakan layanan lokal dan kurang efektif terhadap segmen besar.

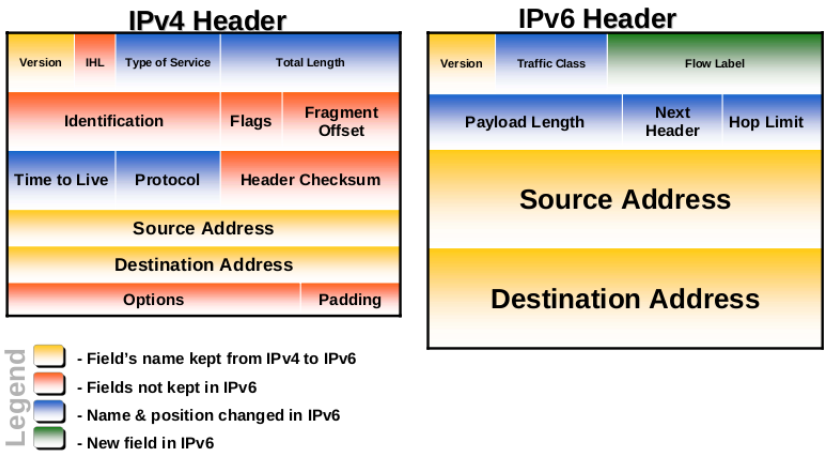
Serangan penolakan layanan sering kali merupakan serangan tingkat kedua. Pertama, penyerang memasukkan kode serangan ke sistem target dan kemudian, setelah kode ditempatkan, penyerang memicu kode tersebut untuk menerapkan serangan penolakan layanan. Selanjutnya kami mempertimbangkan bagaimana penyerang dapat menyusup ke sistem target untuk memulai serangan penolakan layanan.

Session Hijack

Dalam serangan pembajakan sesi, penyerang memungkinkan pertukaran dimulai antara dua pihak tetapi kemudian mengalihkan komunikasi, seperti halnya seorang pria di tengah. Pikirkan, misalnya, masuk ke situs keuangan, menyelesaikan otentikasi, dan kemudian kehilangan sesi karena penyerang. Situs keuangan biasanya terlindungi dengan baik dengan enkripsi, tetapi situs lain mungkin rentan, misalnya, situs yang mengomunikasikan catatan medis atau mendukung interaksi antara siswa dan guru.

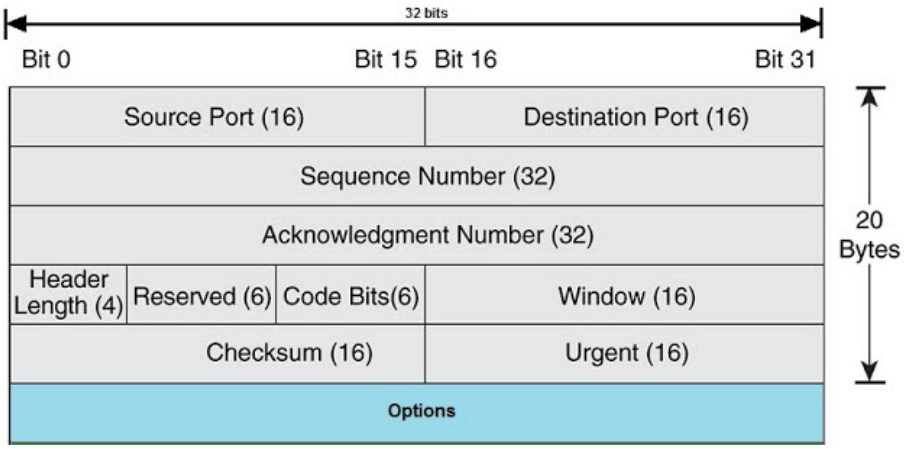
Pembajakan sesi difasilitasi oleh elemen desain protokol TCP/IP. Pertama, perhatikan header protokol IP, seperti yang ditunjukkan pada Gambar 3-29. Bagian penting adalah byte 12-19, yang berisi alamat IP sumber dan tujuan. Tujuan dari tujuan itu jelas; sumber diperlukan agar penerima dapat menghasilkan pesan tanggapan kepada pengirim. Pada titik mana pun di sepanjang perjalanan dari sumber ke tujuan, penyerang dapat mengubah alamat sumber itu, sehingga mengarahkan respons ke penyerang, bukan pengirim asli.

Dalam pembajakan sesi, penyerang secara harfiah mencuri koneksi TCP yang sudah ada dengan menulis ulang alamat sumber dan tujuan.



Gambar 3-29 IP Header

Sekarang perhatikan header protokol TCP, seperti yang ditunjukkan pada Gambar 3-30. Seluruh paket TCP terkandung dalam datagram IP dari Gambar 3-29; dengan demikian semua Gambar 3-29 terdapat dalam bidang Data (byte 20 dan seterusnya) dari Gambar 3-30.

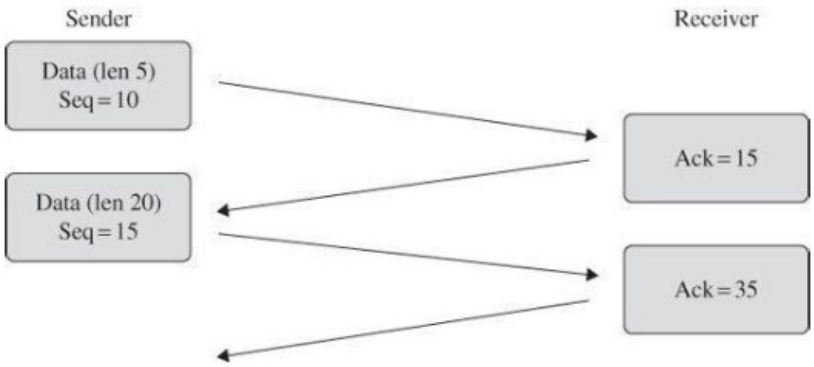


Gambar 3-30 TCP Header

Jika paket tiba tidak sesuai pesanan, penangan protokol menggunakan urutan TCP dan nomor pengakuan, byte 4-11 pada Gambar 3-30, untuk merekonstruksi item data. Protokol TCP dirancang dengan mempertimbangkan jaringan yang tidak stabil, sehingga berisi fitur untuk mengenali dan memperbaiki kesalahan, tidak hanya kerusakan pada data pesan tetapi juga kerusakan pada data kontrol yang ditampilkan di header ini.

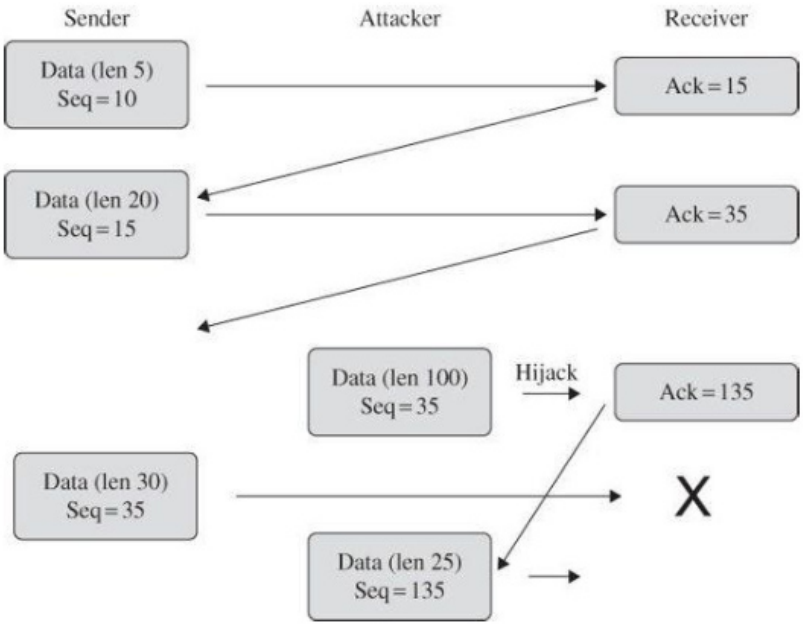
Pengirim membuat dan mengirim paket 1, lalu 2, lalu 3, dan seterusnya, dan penerima mengembalikan paket dengan nomor pengakuan saat paket diterima, seperti yang ditunjukkan pada Gambar 3-31. Kami menyederhanakan penjelasan sedikit dengan hanya menampilkan urutan dari sudut pandang klien. Klien mengirimkan pointer buffer saat ini, dan server mengakui pointer yang sama. (Untuk protokol lengkap, masing-masing mengakui penunjuk terakhir yang lain dan mengirimkan penunjuk saat ini yang memperhitungkan penerimaan data terbaru.) Jika klien mengirim paket dengan urutan yang salah dan pasangan nomor pengakuan, ini mengganggu sinkronisasi dan penerima membuang paket sampai menerima satu yang cocok dengan nomor pengakuan sebelumnya. Jika mereka

tidak menyinkronkan ulang, mereka mengakhiri dan membangun kembali sesi. Protokol dengan demikian menyembuhkan diri sendiri karena begitu kedua ujungnya disinkronkan ulang, mereka dapat menentukan pertukaran terakhir yang berhasil dan mentransmisikan ulang dari titik itu ke depan.



Gambar 3-31 Pertukaran TCP Normal

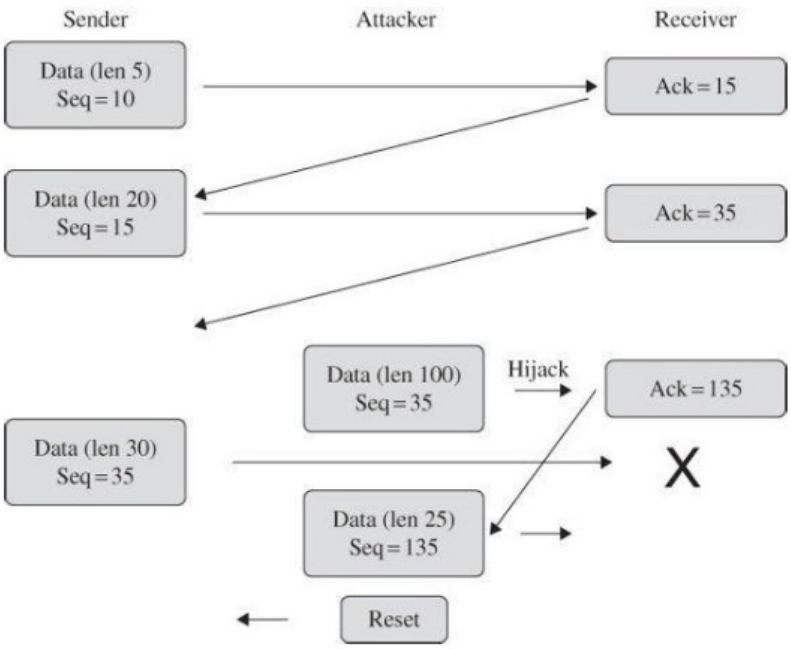
Penyerang dapat memanfaatkan koreksi ini dengan memasukkan paket yang mempertahankan sinkronisasi dengan penerima tetapi menghancurkan sinkronisasi dengan pengirim sebenarnya. Penyerang dan penerima sekarang disinkronkan ulang dan melanjutkan pertukaran yang dimulai oleh pengirim asli. Dengan cara ini, seperti yang ditunjukkan pada Gambar 3-32, penyerang diam-diam menyelinap ke dalam sesi, menggantikan pengirim asli. Paket yang dimasukkan ini adalah replay yang dibuat dengan hati-hati oleh seorang pria di tengah. Serangan ini ditemukan oleh Robert Morris, Sr. dan diperluas oleh Steven Bellovin.



Gambar 3-32 Pembajakan TCP

Sementara itu, seperti yang ditunjukkan pada Gambar 3-33, penyerang mengirimkan perintah RST (reset) ke pengirim asli, meyakinkan pengirim bahwa penerima telah menutup koneksi asli. Pengirim

dapat mencoba membuka koneksi baru dengan penerima, tanpa menyadari bahwa penyerang melanjutkan sesi sebelumnya. Bergantung pada aplikasi yang sedang berjalan, penyerang dapat menerima pengirim sebagai pengguna baru (mungkin mengharuskan pengguna untuk mengautentikasi ulang) atau menolak pengguna karena menduplikasi koneksi yang sudah berlangsung.



Gambar 3-33 Mengatur Ulang Pengirim Asli

Jadi, dengan serangan pembajakan sesi, penyerang dapat meluncur ke aliran komunikasi yang sedang berlangsung tanpa terlihat jelas oleh salah satu dari dua pihak asli; komunikasi berlanjut dengan penyerang menggantikan pengirim asli, sementara pengirim itu dihentikan. Karena kehilangan koneksi sesaat terjadi karena banyak alasan yang tidak berbahaya, pengguna cenderung tidak mencurigai adanya serangan dalam situasi ini; sesi sering dibuat kembali oleh penangan protokol jaringan tanpa sepengetahuan pengguna.

Penyerang hanya menyatu dengan aliran komunikasi, mengambil alih interaksi dari pengirim asli. Serangan berhasil karena penyerang dapat melihat dan memanipulasi header TCP dan IP, tetapi tentu saja ini perlu terlihat di seluruh jaringan karena itulah yang memungkinkan lalu lintas dikirimkan. Namun, kami menunjukkan di bagian berikutnya, cara untuk melindungi dari pembajakan, baik dengan menyembunyikan data penghubung di dalam aplikasi maupun dengan menyembunyikan data header.

Keracunan Cache DNS

Serangan keracunan cache DNS adalah cara untuk menumbangkan pengalamatan untuk menyebabkan server DNS mengarahkan klien ke alamat yang ditentukan. Serangan keracunan DNS secara konseptual sederhana adalah memalsukan pesan ke pencatat DNS, meminta nama domain tertentu diubah dari satu alamat ke alamat lainnya. Permintaan ini biasanya terjadi ketika situs web dipindahkan dari satu penyedia hosting ke penyedia hosting lainnya atau ketika organisasi mengubah struktur alamatnya. Namun, penyerang jahat dapat menggunakan permintaan perubahan

DNS untuk mengarahkan lalu lintas yang ditujukan untuk nama domain tertentu. Karena persyaratan otentikasi yang kuat, pendaftar jarang menyerah pada pemalsuan seperti itu.

Serangan yang lebih mungkin adalah menggunakan pesan protokol DNS yang dengannya semua server nama Internet mengoordinasikan terjemahan alamat mereka. Dan Kaminsky [KAM08] memperluas beberapa metode yang diketahui sebelumnya untuk meracuni cache DNS. Protokol DNS rumit, tetapi Anda tidak perlu memahami detailnya untuk menghargai serangan ini.

Klien yang memerlukan alamat yang sesuai dengan nama domain mengirimkan kueri ke server nama DNS lokalnya. Jika server tersebut tidak memiliki jawabannya, ia akan meneruskan kueri ke server nama root; kueri diteruskan ke server nama yang lebih spesifik sampai seseorang membalas secara otoritatif dengan alamatnya. Alamat itu disebarkan melalui rantai server yang terlibat dalam menyelesaikan kueri dan akhirnya kembali ke klien. Server di sepanjang jalan men-cache respons sehingga mereka dapat merespons langsung ke kueri di masa mendatang untuk alamat yang sama.

Kaminsky melihat kekurangan dalam perkembangan ini: yaitu, kueri ini tetap terbuka hingga dijawab dan bahwa respons yang cocok dengan nomor ID untuk kueri akan di-cache. Jika penyerang dapat menebak urutan nomor ID kueri, penyerang dapat memalsukan respons yang memenuhi ID kueri terbuka; balasan palsu itu dapat memberikan alamat apa pun sebagai tanggapan. Sampai tanggapan dihapus dari cache, semua lalu lintas untuk alamat yang diminta akan diarahkan ke alamat yang diberikan dalam balasan palsu. Jadi, dengan memprediksi nomor urut dengan benar dan menghasilkan lalu lintas jaringan ke server nama tertentu, penyerang dapat mengarahkan lalu lintas secara diam-diam ke alamat yang dipilih. Dalam keracunan cache, konversi DNS nama-ke-alamat yang salah ditempatkan dan tetap dalam cache terjemahan.
Contoh ini menunjukkan kerentanan nomor urut yang dapat diprediksi. Penanggulangan untuk jenis serangan ini adalah serangkaian nomor urut yang tidak dapat diprediksi, sebaiknya diambil dari berbagai kemungkinan.

Selama bertahun-tahun, badan pengatur Internet telah bekerja untuk menerapkan perlindungan terhadap serangan replay dan pembajakan tersebut. Tujuan ini ditangani dengan DNSSEC, ekstensi keamanan DNS. Pada bulan Juni 2010, server DNS root pertama diberi kunci pribadi untuk menandatangani catatan DNS; server root lainnya akan diberikan kunci. Setiap catatan DNS di tingkat root akan ditandatangani dan diterbitkan, bersama dengan kunci publik administrator root, di DNS itu sendiri. Saat catatan server nama root ditandatangani, server nama lain secara bertahap akan memperoleh kunci publik dan menandatangani catatan mereka. Pada akhirnya, permintaan alamat klien juga akan memerlukan perolehan dan pemeriksaan tanda tangan dari semua catatan yang merupakan bagian dari jalur resolusi nama.

3.4.6 Memanfaatkan Kerentanan yang Diketahui

Penyerang tidak memiliki kekurangan alat untuk memulai serangan. Alat peretas sering kali dimulai dengan kerentanan yang diketahui, kadang-kadang yang terkenal yang patchnya telah lama tersedia; orang memiliki kebiasaan gagal menerapkan tambalan ke sistem yang lebih lama atau yang berada di lokasi terpencil. Kegagalan untuk menambal sistem menjadi masalah serius karena waktu antara publisitas mengenai kerentanan dan eksploitasi pertama. Symantec [SYM10] melaporkan bahwa

pada tahun 2009, jarak antara pengungkapan dan eksploitasi rata-rata kurang dari satu hari untuk 28 kerentanan yang ditambal Microsoft di Internet Explorer; eksploitasi muncul rata-rata dua hari setelah kerentanan diketahui. Jendela antara hari patch tersedia dan hari kerentanan pertama kali dieksploitasi memang sangat singkat. Selanjutnya, pada tahun 2009, Symantec mengidentifikasi 12 eksploitasi zero-day. Eksploitasi zero-day adalah eksploitasi yang terjadi sebelum kerentanan diketahui publik dan karenanya sebelum tambalan tersedia.

Beberapa alat, seperti R-U-Dead-Yet dan EvilGrade, memeriksa banyak kerentanan. Trojan horse, virus, dan jenis malware lainnya dapat menjadi dasar serangan penolakan layanan. Salah satu toolkit serangan yang populer tetapi sangat efektif adalah Zeus, yang harganya kurang dari $700 tetapi juga beredar gratis di bawah tanah hacker. Perusahaan keamanan Symantec telah mendokumentasikan lebih dari 90.000 varian Zeus [SYM10]. Dalam alat seperti ini, penolakan layanan terkadang merupakan produk sampingan; alat ini mengeksploitasi kerentanan yang pada akhirnya menyebabkan sistem crash, sehingga menolak layanan, atau setidaknya mengganggunya. Seperti yang kami jelaskan nanti dalam bab ini, mengeksploitasi kerentanan sering kali merupakan langkah pertama dalam mengendalikan komputer penyerang yang kemudian direkrut menjadi tentara penyerang.

3.4.7 Pemutusan Secara Fisik

Akhirnya, kami mempertimbangkan penyebab penolakan layanan terakhir kami: kegagalan fisik. Jaringan terdiri dari peralatan, konektor, dan media transmisi, yang mana saja bisa gagal. Kabel yang rusak, papan sirkuit yang rusak, atau sakelar atau router yang tidak berfungsi dapat menyebabkan penolakan layanan yang sama berbahayanya dengan serangan peretas. Dan sama seperti penyerang menyerang tanpa peringatan dan seringkali tanpa penyebab yang jelas, kegagalan perangkat keras tidak terduga.

Kegagalan Transmisi

Komunikasi gagal karena berbagai alasan. Misalnya, garis dipotong. Atau gangguan jaringan membuat paket tidak dapat dikenali atau tidak terkirim. Mesin di sepanjang jalur transmisi gagal karena alasan perangkat keras atau perangkat lunak. Perangkat dihapus dari layanan untuk diperbaiki atau diuji. Perangkat jenuh dan menolak data yang masuk hingga dapat menghapus kelebihannya. Banyak dari masalah ini bersifat sementara atau diperbaiki secara otomatis (dielakkan) di jaringan utama, termasuk Internet.

Namun, beberapa kegagalan tidak dapat dengan mudah diperbaiki. Putusnya saluran komunikasi tunggal ke komputer Anda (misalnya, dari jaringan ke kartu antarmuka jaringan Anda atau saluran telepon ke modem Anda) hanya dapat diperbaiki dengan membuat tautan alternatif atau memperbaiki yang rusak. Administrator jaringan akan mengatakan "layanan ke seluruh jaringan tidak terpengaruh," tapi itu sedikit menghibur Anda.

Kegagalan Komponen

Komponen, misalnya, router, papan sirkuit, firewall, perangkat pemantauan, perangkat penyimpanan, dan sakelar, gagal karena alasan yang tidak diketahui. Usia, cacat pabrik, lonjakan daya, panas, dan gangguan dapat memengaruhi perangkat keras. Jaringan seringkali merupakan rantai komponen

yang rapuh, yang semuanya diperlukan untuk menjaga jaringan tetap beroperasi. Dalam kasus terburuk, kegagalan komponen apa pun menyebabkan seluruh jaringan gagal. Di Studi Kasus 3-17 kami menjelaskan bagaimana kegagalan satu atau dua papan sirkuit mempengaruhi Negara Bagian Virginia.

Kegagalan perangkat keras hampir selalu merupakan kejadian alami. Meskipun kerusakan perangkat keras yang disebabkan jarang terjadi, itu bukan tidak mungkin. Sebagai contoh, worm Stuxnet yang sebelumnya dijelaskan di Studi Kasus 3-7 dapat menjalankan peralatan mekanis sampai pada titik kegagalan.

Kami telah mempertimbangkan apa yang mungkin disebut serangan penolakan layanan individu, tindakan yang menonaktifkan satu host atau menolak layanan ke satu alamat atau segmen jaringan. Misalnya situasi ini disesalkan untuk host atau alamat yang terpengaruh. Meskipun kerusakan semacam itu dapat melumpuhkan pengguna biasa, instalasi besar seperti perusahaan atau fasilitas utama pemerintah tidak terpengaruh karena memiliki kapasitas dan ketahanan yang besar. Namun, alasan yang lebih serius untuk mempelajari serangan ini adalah karena mereka dapat digunakan sebagai komponen berulang dalam serangan yang jauh lebih besar yang dapat dan sangat memengaruhi pengguna utama. Di bagian selanjutnya kita mempelajari serangan penolakan layanan terdistribusi ini.

Studi Kasus 3-17 : Negara Bagian Virginia Dihentikan Karena Kegagalan TI

Pada tanggal 25 Agustus 2010, layanan komputer untuk 26 dari 89 lembaga Negara Bagian Virginia gagal, mempengaruhi 13 persen dari server file negara bagian. Lembaga negara tidak dapat mengakses data yang diperlukan untuk melayani pelanggan. Mungkin yang paling terpengaruh adalah Departemen Kendaraan Bermotor negara bagian, yang tidak dapat mengeluarkan SIM atau kartu identitas. Departemen Pajak Negara Bagian dan Dewan Pemilihan Negara Bagian juga terkena dampak yang parah, karena tidak memiliki akses ke database selama hampir seminggu; lembaga negara lainnya terpengaruh hingga tiga hari. Selama pemadaman, Departemen Perpajakan tidak dapat mengakses rekening pembayar pajak, negara bagian tidak dapat mengeluarkan cek pengangguran, dan tunjangan kesejahteraan dibayarkan hanya karena upaya besar oleh karyawan yang bekerja selama akhir pekan.

Penyebab hilangnya layanan akhirnya ditemukan karena komponen perangkat keras yang gagal, khususnya perangkat jaringan area penyimpanan (SAN) EMC. Ironisnya, perangkat keras tersebut dimaksudkan untuk meningkatkan keandalan penyimpanan data dengan mendukung redundansi dan pencadangan umum dan memungkinkan data dikumpulkan dari berbagai jenis perangkat penyimpanan yang berbeda. Dalam SAN dua papan sirkuit gagal, menyebabkan hilangnya akses secara luas; satu papan ditemukan rusak dan, ketika diganti, jaringan penyimpanan gagal begitu parah sehingga seluruh sistem harus dimatikan selama lebih dari dua hari. Pabrikan mengatakan kegagalan besar seperti itu belum pernah terjadi sebelumnya dan teknologinya memiliki tingkat keandalan 99,999 persen.

Ketika perangkat keras bekerja kembali, pejabat negara dan teknisi dari Northrop Grumman, kontraktor negara bagian yang menjalankan seluruh sistem, menemukan bahwa basis data utama telah rusak dan satu-satunya tindakan adalah membangun kembali basis data dari salinan cadangan pada pita. Data yang baru saja dimasukkan—mewakili 3 persen dari database—hilang secara permanen.

Tidak semua masalah penolakan layanan adalah akibat dari serangan jahat, tetapi konsekuensi penolakan layanan bisa sama parahnya dari penyebab jahat atau tidak berbahaya.

3.5 Distributed Denial of Service (DDoS)

Distributed Denial of Service atau DoS - Penolakan Layanan Terdistribusi - merupakan serangan yang terbilang cukup kuat untuk melukai sebuah infrastruktur dari suatu organisasi. Serangan ini bertujuan untuk mencegah pengguna menikmati layanan yang diberikan suatu server dan pada akhirnya server tersebut akan down. Serangan DDoS menghancurkan perusahaan komersial yang bergantung pada komputasi untuk interaksi pelanggan, serta fungsi back-end seperti manajemen inventaris dan penjadwalan. Pemerintah terus memindahkan layanan ke web, sehingga akses yang gagal berarti warga tidak dapat menangani interaksi pemerintah biasa. Kemajuan terbaru dalam rekam medis elektronik telah membawa keuntungan, tetapi karena ketergantungan pada mode manajemen data itu tumbuh, merawat pasien akan menjadi berbahaya tanpa akses data. Dan kontrol perangkat komputerisasi dari lampu lalu lintas ke satelit berarti bahwa kegagalan layanan dapat menyebabkan komplikasi serius di dunia fisik juga. Untuk alasan ini, kami mengeksplorasi penyebab dan tindakan pencegahan untuk penolakan layanan.

Serangan penolakan layanan yang baru saja kami jelaskan sangat kuat, dan Studi Kasus 3-19 menunjukkan kepada kita bahwa banyak yang diluncurkan. Tetapi seorang penyerang dapat membuat serangan dua tahap yang melipatgandakan efeknya berkali-kali. Efek multiplikasi ini memberikan kekuatan untuk penolakan layanan terdistribusi.

Studi Kasus 3-18 : Denial of Service: Apa Perbedaan yang Dibuat Satu Dekade

Berapa banyak aktivitas penolakan layanan yang ada? Seperti kebanyakan insiden keamanan komputer, statistik representatif yang andal sulit diperoleh karena tidak ada pengumpulan data pusat, pendekatan pengambilan sampel bervariasi sehingga hanya ada sedikit cara untuk membandingkan nilai, dan tidak ada yang tahu populasi yang dijelaskan oleh hasil tersebut. Beberapa hasil penolakan layanan dari awal 2000-an dan 2010-an memang menunjukkan perubahan yang tak terbantahkan.

Para peneliti di University of California, San Diego (UCSD) mempelajari jumlah aktivitas penolakan layanan di Internet. Karena banyak serangan DoS menggunakan alamat pengirim fiktif, para peneliti menegaskan bahwa lalu lintas ke alamat yang

tidak ada menunjukkan jumlah serangan penolakan layanan. Mereka memantau ruang alamat yang besar dan tidak terpakai di Internet selama tiga minggu pada tahun 2001.
Penemuan mereka:

- Lebih dari 12.000 serangan ditujukan pada lebih dari 5.000 target selama periode tiga minggu.
- Flooding SYN tampaknya menyumbang lebih dari setengah serangan.
- Separuh serangan berlangsung kurang dari sepuluh menit, dan 90 persen serangan berlangsung kurang dari satu jam.

Steve Gibson dari Gibson Research Corporation (GRC) mengalami beberapa serangan denial-of-service pada pertengahan 2001. Dia mengumpulkan data untuk keperluan forensiknya sendiri [GIB01]. Serangan pertama berlangsung selama 17 jam, dan pada saat itu dia berhasil mengkonfigurasi ulang router yang menghubungkannya ke Internet untuk memblokir serangan tersebut. Selama 17 jam itu dia menemukan situsnya diserang oleh 474 PC berbasis Windows. Serangan selanjutnya berlangsung 6,5 jam sebelum berhenti dengan sendirinya. Serangan-serangan ini kemudian diketahui telah diluncurkan oleh seorang anak berusia 13 tahun dari Kenosha, Wisconsin.

Pada akhir dekade, banyak hal telah berubah.

Perusahaan jaringan Arbor Networks mengkhususkan diri dalam menyediakan produk keamanan jaringan untuk membantu ISP dalam menjaga keamanan tulang punggung jaringan mereka. Karena aktivitas mereka dengan ISP, mereka diposisikan untuk mengukur sejumlah besar lalu lintas penolakan layanan. Dalam sebuah analisis yang mencakup tahun 2009, mereka menghitung lebih dari 350.000 serangan penolakan layanan, yang diterjemahkan menjadi satu serangan setiap 90 detik, di mana lebih dari 20.000 melebihi 1 Gbps (gigabit per detik), ukuran volume lalu lintas yang diarahkan. pada sasaran yang diserang. Banyak tautan koneksi Internet organisasi menangani paling banyak 1 Gbps, jadi serangan lebih dari 1 Gbps tidak hanya memFlooding i situs web tetapi seluruh organisasi target dan mulai mencadangkan, memFlooding i infrastruktur jaringan ISP. Sebagai perbandingan, layanan DSL perumahan saat ini mencapai puncak sekitar 3 megabit (1/1000 gigabit) per detik, dan modem kabel untuk pelanggan perumahan biasanya tidak lebih cepat dari 30 Mbps. Pada tahun 2010 [ARB10], Arbor Networks menemukan setidaknya satu serangan yang mencapai 100 Gbps.

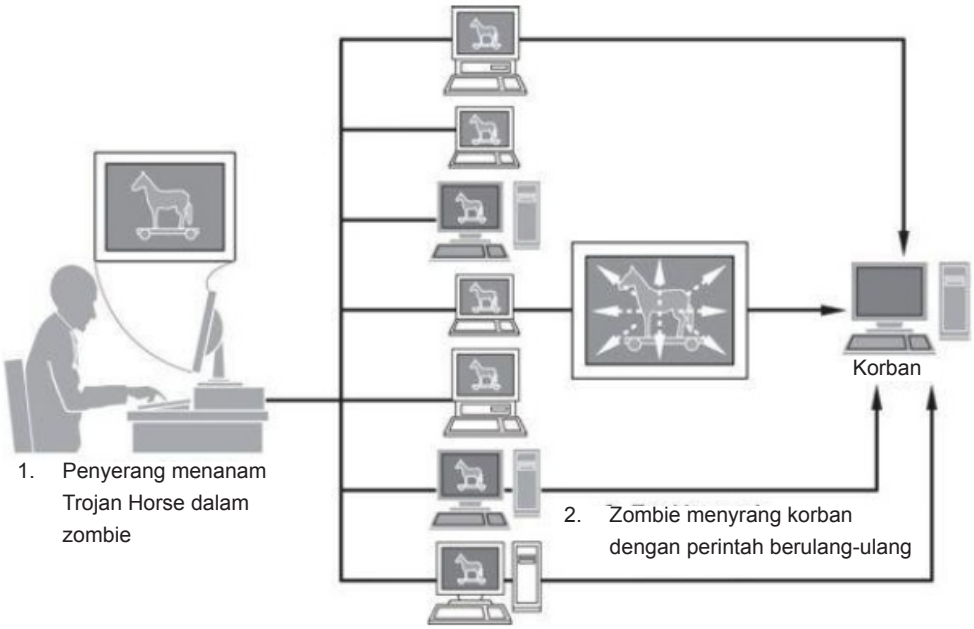
Arbor Networks mengamati bahwa serangan yang lebih besar dari 1 Gbps juga cenderung berlangsung lama. Mereka menemukan bahwa hampir 4.000 serangan lebih dari 1 Gbps berlangsung selama lebih dari 8 jam, dan sekitar 3.500 di antaranya lebih dari 4 Gbps dan 2.000 di antaranya lebih dari 10 Gbps berlangsung selama itu.

Hebatnya, volumenya terus meningkat. Menurut sebuah laporan oleh Incapsula [INC14] pada tahun 2014 serangan 10 Gbps, di ujung atas pada tahun 2009, lemah; 33 persen DDoS melebihi 20 Gbps, dan pada Februari 2014 mereka mencatat satu serangan yang mencengangkan 180 Gbps. Volume di tingkat jaringan bukan satu-satunya ukuran pertumbuhan keparahan serangan DDoS. Laporan yang sama mengGambarkan situs web yang diserang oleh 6 hingga 8 juta permintaan per menit, yang beberapa situs siap untuk tangani.

Pada akhir 2012, bank AS J.P Morgan Chase, SunTrust, Wells Fargo, dan PNC terkena serangan DDoS selama beberapa hari, karena lembaga keuangan dan situs media berita menjadi sasaran.

Serangan Denial-of-service juga mulai menargetkan aktivitas jaringan tertentu. Serangan penolakan layanan klasik mencoba menghabiskan seluruh bandwidth tautan, tetapi serangan baru-baru ini menargetkan firewall, server DNS, infrastruktur untuk layanan VoIP, penyeimbang beban, dan sejenisnya. Karena layanan ini memerlukan komputasi, mereka lebih lambat dan kewalahan oleh volume lalu lintas yang lebih kecil daripada serangan kelelahan bandwidth sederhana.

Untuk memasang serangan penolakan layanan (atau DDoS) terdistribusi, penyerang melakukan dua hal, seperti yang diilustrasikan pada Gambar 3-34.



Gambar 3-34 Serangan Denial-of-Service Terdistribusi

Pada tahap pertama, penyerang ingin mengerahkan pasukan mesin yang dikompromikan untuk menyerang korban. Menggunakan serangan yang mudah (seperti mengeksploitasi buffer overflow atau menipu pengguna untuk membuka dan menginstal kode yang tidak dikenal dari lampiran email), dalang menanam Trojan horse pada mesin jarak jauh. Trojan ihorse tu tidak selalu menyebabkan

kerusakan yang nyata pada mesin yang terinfeksi; pada kenyataannya, mesin harus tetap sehat (dan terinfeksi) sehingga dapat berpartisipasi dalam serangan terhadap korban yang sebenarnya. File kode asing dapat diberi nama untuk editor atau utilitas populer, terikat ke layanan sistem operasi standar, atau dimasukkan ke dalam daftar proses (daemon) yang diaktifkan saat startup. Tidak peduli bagaimana itu terletak di dalam sistem, itu mungkin tidak akan menarik perhatian.

Penyerang mengulangi proses ini dengan banyak komputer target. Masing-masing sistem yang dikompromikan ini kemudian menjadi apa yang dikenal sebagai zombie. Pengguna sistem target melakukan pekerjaan normal mereka, tidak menyadari zombie penduduk. Banyak serangan kerentanan saat ini mengunduh kode ke mesin yang disusupi untuk mengubahnya menjadi zombie.

Pada titik tertentu penyerang memilih korban dan mengirimkan sinyal ke semua zombie untuk melancarkan serangan. Kemudian, alih-alih korban mencoba bertahan dari serangan denial-of-service dari satu host jahat, korban harus mencoba melawan serangan dari banyak zombie yang semuanya bertindak sekaligus. Tidak semua zombie perlu menggunakan serangan yang sama; misalnya, beberapa bisa menggunakan serangan smurf, dan yang lain bisa menggunakan SYN floods untuk mengatasi potensi kelemahan yang berbeda.

3.5.1 Serangan Scripted Denial-of-Service

Selain efek penggandaannya yang luar biasa, serangan denial-of-service terdistribusi merupakan masalah serius karena mudah diluncurkan dari skrip. Mengingat kumpulan serangan penolakan layanan dan metode propagasi, seseorang dapat dengan mudah menulis prosedur untuk menanam kuda Trojan yang dapat meluncurkan salah satu atau semua serangan penolakan layanan. Alat serangan DDoS pertama kali muncul pada pertengahan 1999. Beberapa alat DDoS asli termasuk Tribal Flood Network (TFN), Trin00, dan TFN2K (Tribal Flood Network, edisi tahun 2000). Ketika kerentanan baru yang memungkinkan Trojan horse untuk ditanam ditemukan dan ketika serangan penolakan layanan baru ditemukan, alat kombinasi baru muncul. Untuk detail lebih lanjut tentang topik ini, lihat.

Menurut Tim Tanggap Darurat Komputer AS (CERT), pemindaian untuk menemukan inang yang rentan (potensial zombie) kini disertakan dalam alat kombinasi; satu alat sekarang mengidentifikasi zombienya, menginstal kuda Troya, dan mengaktifkan zombie untuk menunggu sinyal serangan. Symantec mengonfirmasi bahwa paket exploit sekarang menyertakan kode untuk mengubah sistem yang disusupi menjadi zombie. Pemilihan target (zombie) baru-baru ini sebagian besar dilakukan secara acak, artinya penyerang tampaknya tidak peduli dengan zombie mana yang mereka infeksi. Pengungkapan ini sebenarnya adalah berita buruk karena artinya tidak ada organisasi atau host yang dapat diakses yang aman dari serangan. Mungkin karena jumlahnya sangat banyak dan karena penggunanya dianggap kurang berpengetahuan tentang manajemen dan perlindungan komputer, mesin berbasis Windows menjadi target serangan yang lebih populer daripada sistem lain. Yang paling menakutkan adalah temuan yang telah kami sajikan bahwa waktu semakin singkat antara penemuan kerentanan dan eksploitasi yang meluas.

Studi Kasus 3-19 menjelaskan contoh penyerang dengan daya tembak yang lebih besar. Pertempuran itu bukan satu lawan satu tetapi banyak lawan satu: Penyerang meminta pasukan agen untuk menyerang sekaligus dari segala arah. Serangan yang ditemui di bilah sisi terjadi tepat saat komunitas

penyerang maju ke mode serangan baru. Penyelidik memahami serangan penolakan layanan biasa; apa yang dia tidak mengerti pada awalnya adalah serangan penolakan layanan terdistribusi, di mana dampaknya dikalikan dengan kekuatan banyak penyerang.

Studi Kasus 3-19 : Diserang oleh Tentara

Barrett Lyon adalah seorang hacker putus sekolah yang berubah menjadi konsultan komputer yang memiliki fokus fenomenal dan paham teknis. Untuk membantu satu klien memperluas dan menstabilkan jaringan aplikasi web, ia mendapatkan rujukan yang menghasilkan lebih banyak rujukan.

Perusahaan taruhan online BetCRIS telah diganggu dengan serangan sesekali yang memFlooding i situs web mereka hingga satu hari, di mana tidak ada petaruh yang dapat memasang taruhan dan karenanya BetCRIS tidak menghasilkan uang, kehilangan bisnis sebanyak $5 juta dalam sehari. Selama musim semi 2003, kepala BetCRIS mendapat pesan email dari peretas anonim yang memperingatkan bahwa dia akan membuat BetCRIS terkena serangan penolakan layanan kecuali jika dia dibayar $500. Setelah membayar, manajer BetCRIS meminta referensi rekan kerja dan menghubungi Lyon untuk meminta saran. Lyon merekomendasikan membeli beberapa perangkat keras yang dirancang untuk menangkis serangan semacam itu; manajer BetCRIS menginstalnya dan merasa aman untuk masa depan.

Pada akhir November, BetCRIS mendapat permintaan lain: Sebuah pesan email mengumumkan “Situs Anda sedang diserang” dan meminta $40.000 untuk meninggalkan BetCRIS sendirian selama setahun. Berpikir bahwa solusi yang direkomendasikan Lyon sudah memadai, manajer BetCRIS mengabaikan permintaan tersebut.

Serangan denial-of-service besar-besaran membanjiri mesin tujuan khusus dalam sepuluh menit, menyebabkan situs BetCRIS mogok; serangan itu juga membanjiri ISP BetCRIS, yang menjatuhkan BetCRIS sebagai klien untuk menyelamatkan pelanggannya yang lain. Saat serangan berlangsung, tuntutan berkembang menjadi $60.000 dan akhirnya $1 juta dolar. Selama waktu ini Lyon menyadari bahwa ini bukanlah serangan denial-of-service biasa yang diluncurkan dari beberapa mesin, tetapi melibatkan ratusan, mungkin ribuan, lebih.

Lyon tahu serangan itu harus memiliki beberapa kesamaan. Dia mencari alamat IP yang dekat sehingga dia dapat memblokir seluruh rentang, tetapi hanya menemukan sedikit. Beberapa serangan mengejar router sementara yang lain tampak seperti pelanggan biasa. Lyon dengan cepat menulis kode untuk memblokir hal-hal yang dia bisa dan membeli peralatan untuk menjadi ISP sendiri untuk melayani BetCRIS. Sementara itu, penyerang mengejar tetangga bisnis BetCRIS di komunitas perjudian online, serta mantan ISP BetCRIS. Setelah beberapa hari pertempuran bolak-balik, Lyon menang: Situs web BetCRIS telah dicadangkan, stabil, dan kinerjanya normal.

Semua mengatakan, biaya pertempuran sekitar $ 1 juta, hanya apa yang diinginkan penyerang sebagai pemerasan. Dalam pertempuran, Lyon belajar banyak tentang bentuk serangan baru yang baru muncul pada tahun 2003, serangan penolakan layanan terdistribusi.

Bot

Ketika kekuatan diperlukan, panggil tentara. Dalam situasi ini, tentara yang kami maksud adalah jaringan mesin yang siap, bersedia, dan mampu membantu serangan itu. Namun, tidak seperti tentara sungguhan, baik mesin maupun pemiliknya tidak menyadari bahwa mereka adalah bagian dari serangan.

Zombie (atau bot, hacker untuk robot) adalah mesin yang menjalankan potongan kode berbahaya di bawah kendali jarak jauh. Objek kode ini adalah kuda Troya yang didistribusikan ke sejumlah besar mesin korban. Karena mereka mungkin tidak mengganggu atau membahayakan komputer pengguna (selain memakan sumber daya komputasi dan jaringan), mereka sering tidak terdeteksi.

Botnet

Botnet merupakan gabungan dari dua kata, yaitu robot dan network. Dilansir dari Wikipedia, botnet adalah sekumpulan program yang saling terhubung melalui internet yang berkomunikasi dengan program-program sejenis untuk melakukan tugas tertentu. Umumnya botnet digunakan untuk mengirimkan surat elektronik spam, berpartisipasi dalam serangan DDos, atau sebagai fitur keamanan kanal IRC.

Botnet juga bisa dikatakan sebagai kumpulan aplikasi bot (robot) yang dikonfigurasi agar dapat berjalan otomatis dalam sebuah jaringan. Tiap komputer yang tergabung dalam jaringan botnet akan mengoperasikan perintah dari penggerak botnet (bot master) yang dilakukan secara remote. Bisa disimpulkan bahwa jika komputer telah terinfeksi oleh botnet, maka saat tersambung ke jaringan komputer tersebut akan menjalankan perintah yang diberikan oleh bot master.

Botnet, jaringan bot, digunakan untuk serangan penolakan layanan besar-besaran, diimplementasikan dari banyak situs yang bekerja secara paralel terhadap korban. Mereka juga digunakan untuk spam dan serangan email massal lainnya, di mana volume email yang sangat besar dari satu titik mungkin diblokir oleh penyedia layanan pengirim. Contoh operasi botnet dijelaskan di Studi Kasus 3-20.

Studi Kasus 3-20 : Awas! KashmirBlack, Botnet yang Dijalankan dari Indonesia, Serang Situs Web

Peneliti keamanan dunia maya telah menemukan jaringan botnet besar-besaran bernama KashmirBlack, dijalankan dari Indonesia, yang telah menyerang situs web yang menjalankan sistem manajemen konten populer seperti WordPress, Drupal, dan Joomla!.

Menurut lembaga keamanan siber yang berbasis di AS, Imperva, seperti dikutip dari www.newkerala.com, Senin (26/10/2020), botnet yang sangat canggih diyakini

telah menginfeksi ratusan ribu situs web dengan menyerang platform CMS (content magement system) yang mendasarinya.

Tujuan utama botnet tampaknya menginfeksi situs web dan kemudian menggunakan server mereka untuk penambangan cryptocurrency. Berbasis di Indonesia, para peretas memiliki infrastruktur command-and-control (Ciamp;C) untuk mengoperasikan KashmirBlack.

Sebuah botnet adalah sekumpulan program yang saling terhubung melalui internet yang berkomunikasi dengan program-program sejenis untuk melakukan tugas tertentu. Botnet bisa dipakai untuk menjaga keamanan kanal IRC, mengirimkan pos-el spam atau berpartisipasi dalam serangan DDos.

Menurut peneliti, botnet tersebut merupakan hasil karya seorang hacker bernama "Exect1337", salah satu anggota kru peretas Indonesia PhantomGhost. Peladen (server) Ciamp; C adalah mesin terpusat yang dapat mengirim perintah dan menerima telemetri dari mesin yang merupakan bagian dari botnet.

"The KashmirBlack Ciamp; C memiliki tiga peran utama menyediakan skrip Perl yang menginfeksi server korban dengan skrip berbahaya botnet, menerima laporan temuan dan hasil serangan dari bot dan menyediakan bot dengan instruksi serangan," kata Imperva. "Dengan menganalisis instruksi serangan yang diterima oleh 'bot penyebar' tiruan kami, kami menemukan bahwa sebagian besar situs target korban serangan berada di AS," kata para peneliti.

KashmirBlack Ciamp;C memiliki pemindai yang mencari situs yang menjalankan platform CMS, membuat instruksi serangan dengan situs yang baru ditemukan, dan mendorongnya ke dalam antrean menunggu bot untuk menerimanya dan menyerang. Menurut ZDNet, beberapa eksploitasi menyerang CMS itu sendiri, sedangkan yang lain menyerang beberapa komponen dan pustaka dalamnya.

"Selama penelitian kami, kami menyaksikan evolusinya dari botnet volume menengah dengan kemampuan dasar menjadi infrastruktur besar yang akan tetap ada," kata peneliti Imperva.

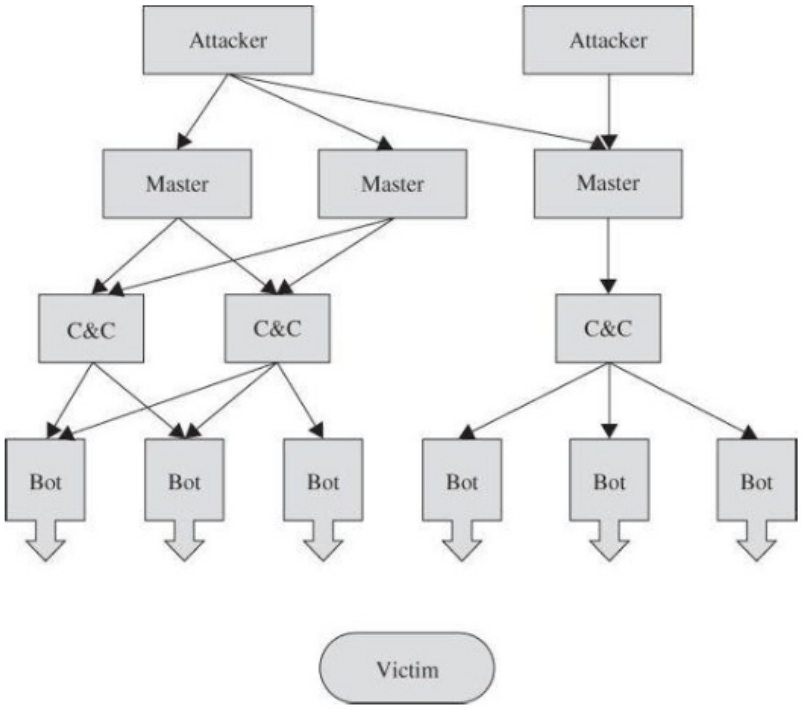
Tim menyarankan beberapa tindakan yang harus dilakukan jika peladen Anda terinfeksi oleh botnet KashmirBlack. "Matikan proses berbahaya, hapus file berbahaya, hapus pekerjaan yang mencurigakan dan asing dan hapus plugin dan tema yang tidak digunakan." Administrator situs harus memastikan file inti CMS dan modul pihak ketiga selalu dimutakhir dan dikonfigurasi dengan benar.

"Kata sandi yang kuat dan unik direkomendasikan karena ini adalah pertahanan pertama terhadap serangan brute force," kata Imperva. *Sumber : https://teknologi.bisnis.com/read/20201027/84/1310428/awas-kashmirblack-botnet-yang-dijalankan-dari-indonesia-serang-situs-web*

Botnet Command dan Control Update

Sama seperti tentara konvensional, jaringan bot membutuhkan hierarki komando; bot mengharuskan petugas untuk memberi tahu mereka kapan harus menyerang, melawan siapa, dan dengan senjata apa. Markas bot disebut pusat komando dan kendali. Struktur dasar tentara seperti itu ditunjukkan pada Gambar 3-35. Dalang ingin diisolasi dari konfigurasi sebenarnya, untuk mengurangi kemungkinan deteksi. Juga, jika bagian dari tentara diisolasi dan diturunkan, penyerang menginginkan redundansi untuk dapat berkumpul kembali, sehingga penyerang membangun redundansi. Penyerang mengendalikan satu atau lebih pengendali utama yang membangun pusat komando dan kendali.

Pusat komando dan kontrol mengontrol bot individu, memberi tahu mereka kapan harus memulai dan menghentikan serangan terhadap korban mana. Komunikasi dari pusat komando dan kendali ke bot dapat didorong, dengan pusat mengirimkan instruksi ke bot, atau ditarik, dengan masing-masing bot bertanggung jawab untuk secara berkala menelepon ke rumah ke pengontrol untuk menentukan apakah ada pekerjaan yang harus dilakukan. Untuk menghindari deteksi, master sering mengubah pusat komando dan kontrol, yang model pushnya lebih efektif, karena bot individu tidak harus diberi tahu tentang alamat komputer perintah dan kontrol yang baru.



Gambar 3-35 Struktur Perintah-dan-Kontrol Botnet

Bot berkoordinasi satu sama lain dan dengan tuannya melalui saluran jaringan biasa, seperti saluran Internet Relay Chat (IRC), jaringan peer-to-peer (yang telah digunakan untuk berbagi musik melalui Internet) atau protokol jaringan lainnya (termasuk HTTP) . Terstruktur sebagai web yang terkoordinasi secara longgar, botnet tidak mengalami kegagalan dari satu bot atau grup bot mana pun, dan dengan banyak saluran untuk komunikasi dan koordinasi, botnet sangat tangguh. Semua aktivitas perintah dan kontrol ini harus dilakukan secara diam-diam agar tidak menarik perhatian administrator jaringan atau dinonaktifkan.

Rent-A-Bot

Orang yang menginfeksi mesin untuk mengubahnya menjadi bot disebut botmaster. Seorang botmaster dapat memiliki (dalam arti mengontrol) ratusan atau ribuan bot. Karena mesin yang terinfeksi milik pengguna yang tidak curiga yang menggunakannya untuk komputasi nyata, bot ini tidak selalu tersedia. Terkadang pemilik sebenarnya mematikan mesin mereka, memutuskan sambungan dari Internet, atau menggunakannya secara intensif sehingga hanya sedikit kapasitas yang tersisa untuk berfungsi sebagai bot. Namun, sebagian besar waktu, mesin ini diam, tersedia untuk pekerjaan jahat.

Botmaster sering memiliki dua kegunaan untuk botnet: Pertama, botnet harus tersedia untuk serangan ketika botmaster ingin mengejar korban. Seperti disebutkan di bilah sisi sebelumnya dalam bab ini, serangan bisa berlangsung berjam-jam. Namun, aktivitas penolakan layanan cenderung ditargetkan, tidak acak, sehingga satu botmaster tidak mungkin memiliki jumlah korban yang tidak terbatas untuk mengarahkan bot. Jadi, untuk mendatangkan sedikit pemasukan, botmaster terkadang juga menyewakan botnetnya kepada orang lain. Peneliti Dancho Danchev melaporkan bahwa pada tahun 2013, botnet dari 1.000 host dapat disewa seharga $25–$120 AS, atau $200–$500 AS untuk 10.000 host selama 24 jam.

Bergabung dengan Botnets

Ketika bisnis dan pemerintah telah memindahkan kehadiran mereka secara online, protes dan bentuk ketidakpuasan publik lainnya terhadap mereka telah mengikuti. Semakin banyak orang yang telah termotivasi untuk menggunakan poster protes, aksi duduk di jalan raya, dan bom molotov yang setara dengan dunia maya.

Beberapa tahun terakhir telah menunjukkan peningkatan yang stabil dalam kecanggihan alat dan taktik penggunaan online yang tidak terpengaruh. Aplikasi jejaring sosial, teknologi Web 2.0, dan ketersediaan umum dari apa yang paling tepat digambarkan sebagai alat serangan siber “kelas militer” menjadikannya tugas sepele bagi para pemrotes untuk meluncurkan serangan yang melumpuhkan dari mana saja di seluruh dunia.

Adopsi besar-besaran portal jejaring sosial dan layanan micro-blogging pada gilirannya menciptakan generasi baru kemampuan Command-and-Control (CnC) terpusat yang dengan cepat dan mudah mengatur protes untuk peserta internasional dari semua lapisan masyarakat. Kesederhanaan teknologi ini dapat dimanfaatkan untuk koordinasi serangan terhadap pemerintah dan organisasi komersial tidak dapat diremehkan.

Botnet selalu dianggap sebagai ancaman parah yang menghilangkan PC dan server dari kendali TI. Namun, kompromi botnet selalu datang dari pemasangan malware bot yang tidak disengaja dan tidak diketahui. Penerimaan malware bot yang disengaja dan disengaja, betapapun terpuji penyebabnya, menghadirkan tantangan berbahaya bagi organisasi mana pun yang peduli tentang mempertahankan kontrol atas aset jaringan dan menunjukkan tanggung jawab fidusia yang tepat.

Malicious Autonomous Mobile Agents

Bot termasuk dalam kelas kode yang lebih umum dikenal sebagai *malicious autonomous mobile agents.* Bekerja sebagian besar sendiri, program-program ini dapat menginfeksi komputer di mana

saja mereka dapat mengakses, menyebabkan penolakan layanan serta jenis bahaya lainnya. Tentu saja, kode tidak berkembang, muncul, atau bermutasi dengan sendirinya; harus ada pengembang yang terlibat pada awalnya untuk mengatur proses dan, biasanya, untuk membuat skema pembaruan. Agen semacam itu kadang-kadang disebut agen inokulasi.

Saat bot atau agen menjalankan dan memperoleh pembaruan, tidak setiap agen akan diperbarui sekaligus. Satu agen mungkin berada di sistem yang dimatikan, yang lain di sistem yang saat ini tidak memiliki konektivitas jaringan eksternal, dan agen lain mungkin berjalan di domain sumber daya yang dibatasi. Dengan demikian, saat agen keluar-masuk kontak dengan layanan pembaruan mereka, beberapa akan diperbarui dan yang lain akan menjalankan versi yang lebih lama. Masalah koordinasi pasukan agen yang berbeda adalah topik penelitian aktif, berdasarkan masalah jenderal Bizantium.

Autonomous Mobile Protective Agents

Misalkan seorang insinyur keamanan memecahkan kode logika agen; insinyur kemudian mungkin meminta agen untuk memperjuangkan orang baik dengan memodifikasinya agar terlihat normal bagi saudara kandungnya tetapi sebenarnya untuk menyebarkan infeksi balik. Jadi, misalnya, agen yang dimodifikasi mungkin mencari agen musuh lainnya dan memberikan mereka "pembaruan" yang sebenarnya menonaktifkannya.

Konsep ini tidak terlalu mengada-ada seperti kedengarannya. Dengan cara yang sama seperti penyerang telah mengembangkan jaringan untuk membahayakan, peneliti keamanan telah mendalilkan bagaimana agen yang baik dapat membantu menyembuhkan setelah infeksi kode berbahaya.

Seorang remaja Jerman, Sven Jaschen, menulis dan merilis worm yang disebut NetSky pada Februari 2004. Dia mengklaim niatnya adalah untuk menghapus infeksi worm MyDoom dan Bagle yang tersebar luas dari komputer yang terinfeksi dengan menutup kerentanan yang dieksploitasi oleh worm tersebut. NetSky disebarkan melalui email. Namun, Jaschen segera terlibat dalam pertempuran dengan pencipta Bagle dan MyDoom, yang menghasilkan versi kode yang lebih baik, yang menghasilkan versi baru NetSky, dan seterusnya, dengan total 30 jenis NetSky yang terpisah. Menurut salah satu pakar keamanan, Mikko Hypponen dari f-Secure, NetSky lebih efektif dalam mengurangi aliran spam daripada apa pun yang terjadi di Kongres atau pengadilan AS. Sayangnya, itu juga menghabiskan banyak sumber daya sistem dan membombardir banyak klien komersial dengan email. Versi worm yang lebih baru meluncurkan serangan penolakan layanan terhadap tempat-tempat yang tidak disukai Jaschen. Dua tahun setelah rilis virus, itu masih infeksi virus paling umum di seluruh dunia, menurut perusahaan keamanan Sophos.

Dua bulan setelah merilis NetSky, pada ulang tahunnya yang kedelapan belas, Jaschen menulis dan merilis virus berbasis Internet yang sangat merusak bernama Sasser yang memaksa komputer untuk reboot terus-menerus. Dia ditangkap oleh otoritas Jerman, dan dihukum serta dijatuhi hukuman percobaan 31 bulan dan masa percobaan tiga tahun.

3.5.2 Mengatasi Serangan DDoS

Serangan DDoS tidak sulit untuk dicegah, setidaknya secara teori. Kebanyakan bot wajib militer menggunakan kerentanan terkenal, yang patch telah didistribusikan untuk beberapa waktu. Jadi, jika seluruh dunia hanya menginstal patch pada waktu yang tepat, ancaman DDoS akan berkurang. Beberapa pengguna komputer, bagaimanapun, tidak memiliki salinan resmi dari sistem operasi dan perangkat lunak lainnya, sehingga mereka tidak dapat berlangganan dan mendapatkan patch melalui rantai pabrikan. Perangkat lunak komputer adalah salah satu dari sejumlah kecil komoditas, termasuk senjata api ilegal dan obat-obatan terlarang, di mana pasar gelap juga mempengaruhi konsumen yang sah. Serangan DDoS melibatkan beberapa programmer dan analis berbakat dalam permainan yang menguntungkan untuk membuat perisai rumit di sekitar kelemahan duniawi lama yang berderit. Sampai kita menghapus kekurangannya, tidak ada yang bisa diperbaiki di sekitarnya. Itu adalah titik di mana teori bertemu dengan praktik.

Administrator dapat mengatasi serangan DoS biasa melalui teknik seperti penyetelan (menyesuaikan jumlah server aktif), penyeimbangan beban (menyeimbangkan beban komputasi di seluruh server yang tersedia), shunning (mengurangi layanan yang diberikan ke lalu lintas dari rentang alamat tertentu), dan daftar hitam (menolak koneksi dari alamat tertentu). Teknik yang sama ini digunakan untuk melawan serangan DDoS, diterapkan pada skala yang lebih besar dan pada perimeter jaringan. Sejauh ini sebagian besar serangan DDoS tampaknya membuat pernyataan atau memusatkan perhatian, jadi setelah berlangsung beberapa saat, penyerang menyimpulkan bahwa poin telah dibuat dan berhenti. Beberapa serangan, seperti yang dijelaskan sebelumnya di Studi Kasus 3-20, bertujuan untuk memeras uang dari para korban; seperti jenis serangan pemerasan lainnya, membayar suap mungkin tidak menghentikan serangan.

Diskusi penolakan layanan ini menyimpulkan pemeriksaan kami terhadap ancaman yang rentan terhadap komputasi jaringan. Denial of service adalah masalah yang khas dan membutuhkan penanggulangannya sendiri. Serangan jaringan lain yang melibatkan intersepsi dan modifikasi menggunakan kontrol yang lebih terkenal.

Serangan ini juga merupakan bagian terakhir dalam analisis kami tentang ancaman keamanan dan kerentanan terhadap jaringan komputer. Bagian ini telah menyentuh ketiga elemen triad C-I-A, dengan penyadapan dan penyamaran (serangan terhadap kerahasiaan), korupsi dan pemutaran ulang data (integritas), dan penolakan layanan (ketersediaan). Bagian tentang jaringan WiFi menunjukkan kerentanan yang dapat menyebabkan kegagalan masing-masing dari ketiganya. Bagian itu juga menunjukkan bahwa standar yang dikembangkan dengan hati-hati pun dapat menunjukkan kelemahan keamanan yang serius. Bagian II bab ini akan membahawa tentang perlindungan untuk jaringan komputer.

Bagian II—Pertahanan Strategis: Penanggulangan Keamanan

Dalam sisa bab ini, kami mempertimbangkan tiga kategori kontrol: Pertama, seperti yang dapat Anda bayangkan, kontrol enkripsi yang sudah dikenal adalah alat yang kuat untuk menjaga kerahasiaan dan integritas dalam jaringan. Kami menjelaskan secara arsitektur bagaimana enkripsi dapat digunakan dan kemudian memperkenalkan dua aplikasi spesifik kriptografi ke jaringan: komunikasi terenkripsi

antara browser dan situs webnya, yang disebut enkripsi SSL, dan tautan terenkripsi dalam jaringan, yang disebut jaringan pribadi virtual atau VPN. Kemudian kami memperkenalkan alat perlindungan jaringan yang disebut firewall, yang sebenarnya hanyalah contoh dari monitor referensi yang sudah dikenal. Kami mengakhiri studi kontrol dengan perangkat lain, yang disebut deteksi intrusi atau sistem perlindungan, yang memantau lalu lintas jaringan untuk mengidentifikasi dan melawan ancaman jaringan berbahaya tertentu.

3.6 Kriptografi dalam Keamanan Jaringan

Ingat dari Bab 2 bahwa ada dua kelas enkripsi yang luas: sistem simetris (kunci rahasia) dan asimetris (kunci publik). Yang pertama adalah pekerja keras kriptografi, yang digunakan untuk enkripsi massal data dalam jumlah besar. Deskripsi itu sangat cocok dengan lalu lintas jaringan, dan begitulah cara penggunaannya. Kelas kedua dari algoritma kriptografi unggul dalam membangun hubungan yang dapat dipercaya antara dua pihak yang sebelumnya mungkin tidak memilikinya, yang juga berlaku secara alami dalam situasi jaringan. Pada bagian ini kami menjelaskan bagaimana kedua pendekatan tersebut dapat memberikan kekuatan keamanan dalam jaringan.

3.6.1 Enkripsi Jaringan

Enkripsi mungkin adalah alat yang paling penting dan serbaguna untuk pakar keamanan jaringan. Kita telah melihat di bab-bab sebelumnya bahwa enkripsi sangat kuat untuk menyediakan privasi, keaslian, integritas, dan pemisahan. Karena jaringan melibatkan risiko yang lebih besar, mereka sering mengamankan data dengan enkripsi, mungkin dalam kombinasi dengan kontrol lain. Sebelum kita mulai mempelajari penggunaan enkripsi untuk melawan ancaman keamanan jaringan, mari kita tekankan empat poin.

- Enkripsi hanya melindungi apa yang dienkripsi (yang seharusnya jelas tetapi tidak). Kenali bahwa data terekspos di antara ujung jari pengguna dan proses enkripsi sebelum dikirim, dan data terekspos lagi setelah didekripsi di ujung jarak jauh. Enkripsi terbaik tidak dapat melindungi dari kuda Trojan berbahaya yang mencegat data sebelum titik enkripsi.
- Merancang algoritma enkripsi sebaiknya diserahkan kepada para profesional. Kriptografi dipenuhi dengan kehalusan, dan perubahan yang tampaknya kecil dapat berdampak besar pada keamanan.
- Enkripsi tidak lebih aman daripada manajemen kuncinya. Jika penyerang dapat menebak atau menyimpulkan kunci enkripsi yang lemah, permainan berakhir.
- Enkripsi bukanlah obat mujarab atau peluru perak. Desain sistem yang cacat dengan enkripsi masih merupakan desain sistem yang cacat. Orang yang tidak mengerti enkripsi terkadang salah mengira bahwa debu peri yang ditaburkan pada sistem untuk perlindungan magis. Buku ini tidak akan diperlukan jika debu peri seperti itu ada.

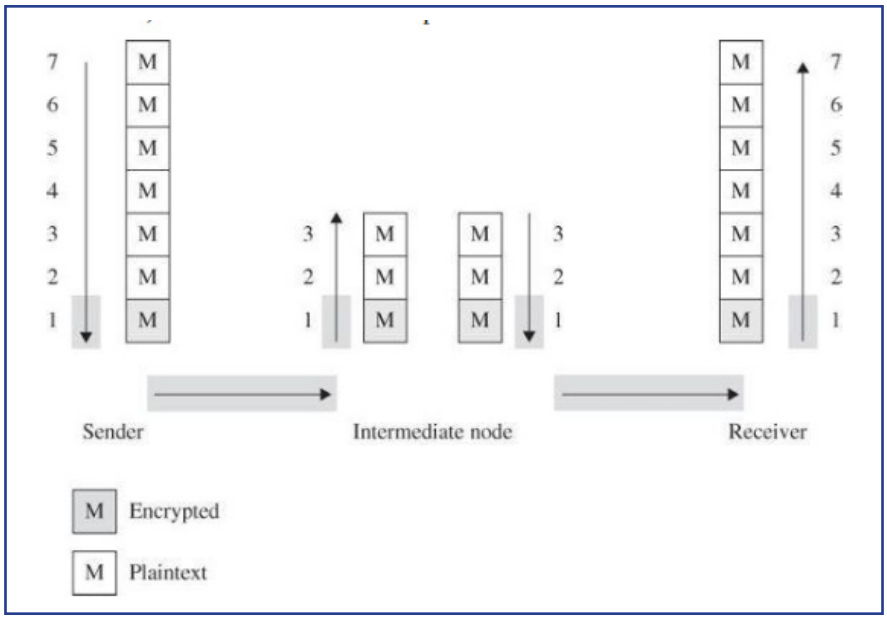
Dalam aplikasi jaringan, enkripsi dapat diterapkan antara dua host (disebut enkripsi tautan) atau antara dua aplikasi (disebut enkripsi ujung ke ujung). Kami mempertimbangkan keduanya di bawah ini. Dengan kedua bentuk enkripsi, distribusi kunci selalu menjadi masalah. Kunci enkripsi harus dikirimkan ke pengirim dan penerima dengan cara yang aman. Di bagian selanjutnya dari bab ini, kami juga menyelidiki teknik untuk distribusi kunci yang aman dalam jaringan. Akhirnya, kami mempelajari fasilitas kriptografi untuk lingkungan komputasi jaringan.

Mode Enkripsi Jaringan

Enkripsi dapat digunakan dalam jaringan melalui dua mode umum: link dan end-to-end. Mereka melakukan fungsi yang berbeda dan memiliki kekuatan dan kelemahan yang berbeda. Dan mereka bahkan dapat digunakan bersama, meskipun agak berlebihan.

Link Encryption

Dalam link encryption (enkripsi tautan), data dienkripsi tepat sebelum sistem menempatkannya pada tautan komunikasi fisik. Dalam hal ini, enkripsi terjadi pada lapisan 1 atau 2 dalam model OSI. (Situasi serupa terjadi dengan protokol TCP/IP, yang memiliki model berlapis serupa tetapi lebih pendek.) Demikian pula, dekripsi terjadi tepat saat komunikasi tiba dan memasuki komputer penerima. Sebuah model enkripsi link ditunjukkan pada Gambar 3-36. Seperti yang Anda lihat, data berjalan dalam plaintext melalui lapisan atas model sampai mereka dienkripsi sesaat sebelum transmisi, pada level 1. Pengalamatan terjadi pada level 3. Oleh karena itu, pada node perantara, enkripsi harus dihapus agar untuk menentukan di mana selanjutnya untuk meneruskan data, dan konten diekspos.

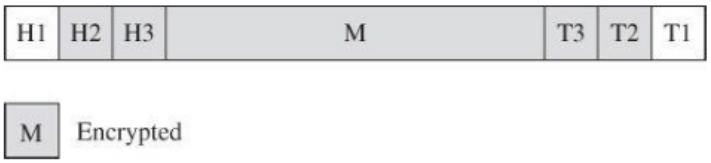


Gambar 3-36 Model Enkripsi Tautan

Enkripsi melindungi pesan dalam perjalanan antara dua komputer, tetapi pesan dalam plaintext di dalam host. (Sebuah pesan dalam plaintext dikatakan “dalam kejelasan.”) Perhatikan bahwa karena enkripsi ditambahkan pada lapisan protokol bawah, pesan tersebut diekspos di semua lapisan pengirim dan penerima lainnya. Jika kami memiliki keamanan fisik yang baik dan kami memercayai perangkat lunak yang mengimplementasikan fungsi lapisan atas, kami mungkin tidak terlalu mengkhawatirkan potensi kerentanan ini. Pesan terbuka untuk diakses dalam dua lapisan dari semua host perantara yang dapat dilalui oleh pesan. Pesannya jelas di host perantara, dan salah satu host ini mungkin tidak terlalu dapat dipercaya.

Enkripsi tautan tidak terlihat oleh pengguna. Enkripsi menjadi layanan transmisi yang dilakukan oleh lapisan protokol jaringan tingkat rendah, seperti perutean pesan atau deteksi kesalahan transmisi. Gambar 3-37 menunjukkan pesan terenkripsi tautan yang khas, dengan bidang yang

diarsir dienkripsi. Karena beberapa header dan trailer data link diterapkan sebelum blok dienkripsi, bagian dari masing-masing blok tersebut diarsir. Saat pesan M ditangani di setiap lapisan, informasi header dan kontrol ditambahkan di sisi pengirim dan dihapus di sisi penerima. Perangkat enkripsi perangkat keras beroperasi dengan cepat dan andal; dalam hal ini, enkripsi tautan tidak terlihat oleh sistem operasi dan juga operator.

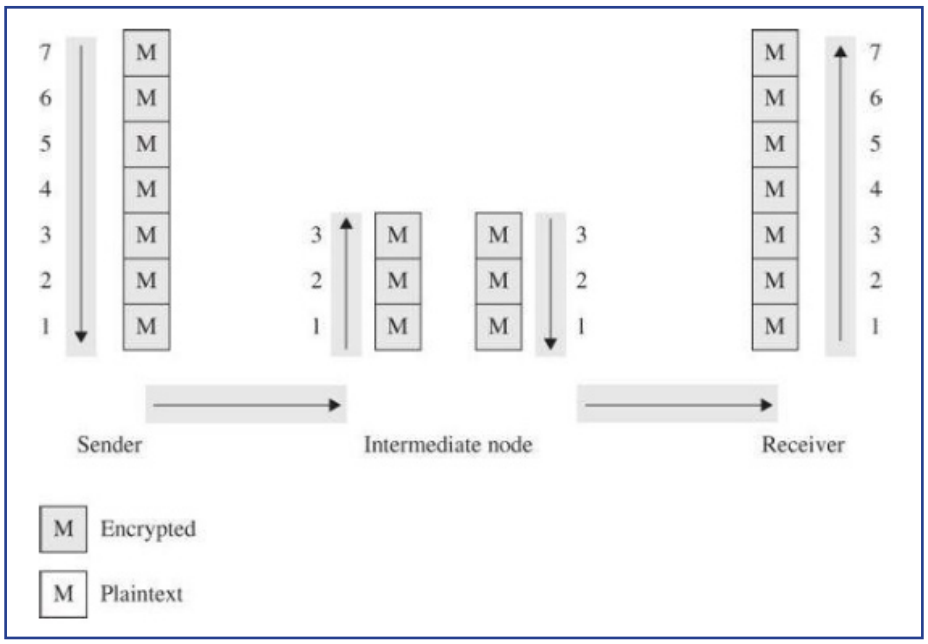


Gambar 3-37 Enkripsi Tautan

Enkripsi tautan sangat tepat ketika saluran transmisi adalah titik kerentanan terbesar. Jika semua host di jaringan cukup aman tetapi media komunikasi digunakan bersama dengan pengguna lain atau tidak aman, enkripsi tautan adalah kontrol yang mudah digunakan. Enkripsi tautan juga diinginkan ketika semua komunikasi pada satu jalur harus dilindungi, misalnya, jika tautan antara dua kantor dari satu perusahaan, di mana semua komunikasi internal akan dilindungi.

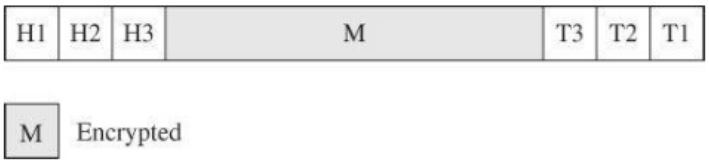
End-to-End Encryption

Seperti namanya, End-to-End Encryption memberikan keamanan dari satu ujung transmisi ke ujung lainnya. Enkripsi dapat diterapkan antara pengguna dan host oleh perangkat keras. Atau, enkripsi dapat dilakukan dengan perangkat lunak yang berjalan pada komputer host. Dalam kedua kasus tersebut, enkripsi dilakukan pada level tertinggi, biasanya oleh aplikasi pada OSI level 7, tetapi terkadang 5 atau 6. Model enkripsi ujung-ke-ujung ditunjukkan pada Gambar 3-38.



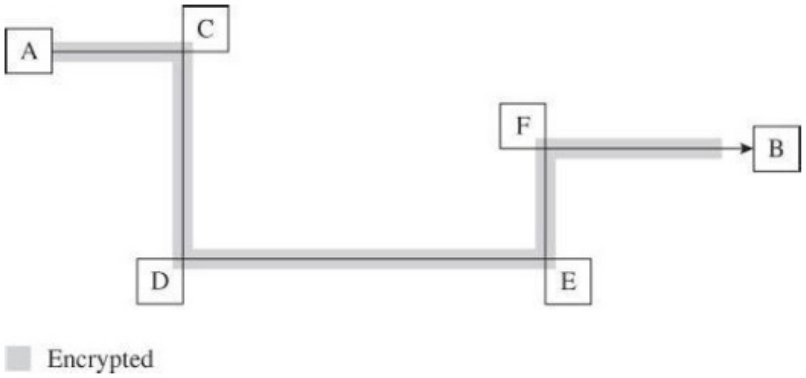
Gambar 3-38 Enkripsi Tingkat Aplikasi (End-to-End)

Karena enkripsi mendahului semua perutean dan pemrosesan transmisi lapisan, pesan ditransmisikan dalam bentuk terenkripsi ke seluruh jaringan. Tentu saja, hanya bagian data dari pesan yang dilindungi, tetapi seringkali header tidak sesensitif data. Enkripsi mengatasi potensi kelemahan di lapisan bawah dalam model transfer. Jika lapisan bawah gagal menjaga keamanan dan mengungkapkan data yang diterimanya, kerahasiaan data tidak terancam. Gambar 3-39 menunjukkan pesan tipikal dengan enkripsi ujung ke ujung, sekali lagi dengan bidang terenkripsi yang diarsir.



Gambar 3-39 Enkripsi End-to-End

Ketika enkripsi ujung ke ujung digunakan, pesan yang dikirim melalui beberapa host dilindungi. Isi data pesan masih dienkripsi, seperti yang ditunjukkan pada Gambar 3-40, dan pesan dienkripsi (dilindungi dari pengungkapan) saat dalam perjalanan. Oleh karena itu, meskipun pesan harus melewati node yang berpotensi tidak aman (seperti C hingga F) di jalur antara A dan B, pesan dilindungi dari pengungkapan saat dalam perjalanan.



Gambar 3-40 Pesan Dilindungi dalam Transit

Perbandingan Metode Enkripsi

Cukup mengenkripsi pesan bukanlah jaminan mutlak bahwa itu tidak akan terungkap selama atau setelah transmisi. Namun, dalam banyak kasus, kekuatan enkripsi adalah perlindungan yang memadai, mengingat kemungkinan pencegat melanggar enkripsi dan ketepatan waktu pesan. Seperti banyak aspek keamanan, kita harus menyeimbangkan kekuatan perlindungan dengan kemungkinan serangan.

Dengan mode tautan, semua transmisi dilindungi sepanjang tautan tertentu. Biasanya, host yang diberikan hanya memiliki satu tautan ke jaringan, yang berarti bahwa semua lalu lintas jaringan yang dimulai pada host tersebut akan dienkripsi untuk host tersebut. Tetapi skema enkripsi ini menyiratkan bahwa setiap host lain yang menerima komunikasi ini juga harus memiliki fasilitas kriptografi untuk mendekripsi pesan. Selanjutnya, semua host harus berbagi kunci. Sebuah pesan dapat melewati satu atau lebih host perantara dalam perjalanan ke tujuan akhirnya. Jika pesan dienkripsi di sepanjang beberapa tautan jaringan tetapi tidak yang lain, maka sebagian dari keuntungan enkripsi hilang.

Oleh karena itu, enkripsi tautan biasanya dilakukan pada semua tautan jaringan jika dilakukan sama sekali.

Sebaliknya, enkripsi ujung ke ujung diterapkan pada "tautan logis", yang merupakan saluran virtual antara dua proses, pada tingkat yang jauh di atas jalur fisik. Karena host perantara sepanjang jalur transmisi tidak perlu mengenkripsi atau mendekripsi pesan, mereka tidak memerlukan fasilitas kriptografi. Jadi, enkripsi hanya digunakan untuk pesan dan aplikasi yang membutuhkannya. Selain itu, enkripsi dapat dilakukan dengan perangkat lunak, sehingga kita dapat menerapkannya secara selektif, satu aplikasi pada satu waktu atau bahkan ke satu pesan dalam aplikasi tertentu.

Keuntungan selektif dari enkripsi ujung ke ujung juga merupakan kerugian terkait kunci enkripsi. Di bawah enkripsi ujung ke ujung, saluran kriptografi virtual ada di antara setiap pasangan pengguna. Untuk memberikan keamanan yang tepat, setiap pasangan pengguna harus berbagi kunci kriptografi yang unik. Jumlah kunci yang diperlukan sama dengan jumlah pasangan pengguna, yaitu n * (n – 1)/2 untuk n pengguna. Jumlah ini meningkat pesat seiring dengan bertambahnya jumlah pengguna.

Enkripsi tautan lebih cepat, lebih mudah bagi pengguna, dan menggunakan lebih sedikit kunci. Enkripsi end-to-end lebih fleksibel, dapat digunakan secara selektif, dilakukan pada level pengguna, dan dapat diintegrasikan dengan aplikasi. Tidak ada bentuk yang tepat untuk semua situasi.

Dalam beberapa kasus, kedua bentuk enkripsi dapat diterapkan. Pengguna yang tidak mempercayai kualitas enkripsi tautan yang disediakan oleh suatu sistem juga dapat menerapkan enkripsi ujung ke ujung. Administrator sistem yang memperhatikan keamanan skema enkripsi ujung ke ujung yang diterapkan oleh program aplikasi juga dapat memasang perangkat enkripsi tautan. Jika kedua enkripsi relatif cepat, duplikasi keamanan ini memiliki sedikit efek negatif.

Enkripsi tingkat tautan sangat cocok untuk menerapkan jaringan pribadi dengan menggunakan sumber daya publik. Jaringan pribadi virtual, yang dijelaskan di bagian berikutnya, adalah teknik yang menyediakan privasi di jaringan publik.

3.6.2 Enkripsi Peramban (Browser Encryption)

Browser dapat mengenkripsi data untuk perlindungan selama transmisi. Browser dan server menegosiasikan kunci enkripsi yang umum, jadi meskipun penyerang membajak sesi di tingkat protokol TCP atau IP, penyerang, yang tidak memiliki kunci yang tepat, tidak dapat bergabung dengan pertukaran data aplikasi.

Enkripsi SSH

SSH (secure shell) adalah sepasang protokol (versi 1 dan 2) yang awalnya ditetapkan untuk Unix tetapi sekarang tersedia di sebagian besar sistem operasi. SSH menyediakan jalur yang diautentikasi dan dienkripsi ke shell atau penerjemah perintah sistem operasi. Kedua versi SSH menggantikan utilitas Unix seperti Telnet, rlogin, dan rsh untuk akses jarak jauh. SSH melindungi dari serangan spoofing dan modifikasi data dalam komunikasi.

Protokol SSH melibatkan negosiasi antara situs lokal dan remote untuk algoritma enkripsi (misalnya, DES atau AES) dan otentikasi (termasuk kunci publik dan Kerberos).

SSH server adalah suatu protokol atau jalur masuk aman dengan menggunakan file yang dikirim dari jaringan unix system. Semua alur jalannya jaringan dirahasiakan hanya dapat membaca satu jaringan dan juga tidak dapat menangkap password.

SSH adalah jaringan koneksi yang aman, jika Anda menggunakan jaringan ini maka akan memiliki pilihan untuk dalam melakukan autentikasi dari pengguna remote sebelum terkoneksi jaringan. Sistem ini juga mampu mengirimkan input dari SSH client kepada host atau server pusat, lalu mengirim kembali hasil ke server client. Ketika melakukan koneksi SSH ke salah satu alamat IP server, Anda dapat mengubah pengaturan yang terdapat di dalam server sesuai dengan pengguna yang tengah dipakai. Selain terkoneksi dengan jaringan OSX, Anda juga dapat melakukan koneksi kepada sistem operasi lain, seperti Windows dan Linux.

Perangkat atau sistem ini membuat Anda dapat bergerak bebas di jaringan dengan melalui struktur file akun hosting, selain melakukan monitoring pada log file, Anda juga dapat memulai dan menghentikkan service atau layanan.

SSH bisa dipakai untuk melakukan instal software di dalam akun hosting Anda atau terkenal dengan sebutan manage database MySQL, sistem ini memungkinkan Anda untuk lebih banyak berinteraksi dari penggunaan standar website.

Enkripsi SSL dan TLS

Protokol Secure Sockets Layer (SSL) awalnya dirancang oleh Netscape pada pertengahan 1990-an untuk melindungi komunikasi antara browser web dan server. Itu melewati tiga versi: SSL 1.0 (pribadi), SSL 2.0 (1995), dan SSL 3.0 (1996). Pada tahun 1999, Internet Engineering Task Force mengupgrade SSL 3.0 dan menamai upgrade TLS, untuk keamanan lapisan transport. TLS 1.0, yang terkadang juga dikenal sebagai SSL 3.1, didokumentasikan dalam Internet RFC 2246; dua versi yang lebih baru diberi nama TLS 1.1 (RFC 4346, 2006) dan TLS 1.2 (RFC 5246, 2008). Akronim SSL sering digunakan untuk mewakili suite protokol SSL dan TLS.
Dalam model jaringan OSI, aplikasi dijalankan pada level tertinggi (terjauh dari sinyal listrik), yang disebut level 7, dan SSL diimplementasikan pada level 4, di atas pengalamatan jaringan (level 3) dan media fisik (level 1). SSL beroperasi antara aplikasi (seperti browser) dan protokol TCP/IP untuk menyediakan otentikasi server, otentikasi klien opsional, dan saluran komunikasi terenkripsi antara klien dan server.

Suite Cipher

Klien dan server menegosiasikan algoritma enkripsi, yang disebut cipher suite, untuk otentikasi, enkripsi sesi, dan hashing. Untuk memungkinkan perluasan dan penghentian algoritme dari waktu ke waktu, pihak pertama yang membuka interaksi, seringkali klien, menyatakan algoritme pilihannya, dan pihak kedua merespons dengan yang tertinggi dalam daftar yang dapat ditanganinya.

Dari Wikipedia kita mendapatkan informasi bahwa cipher suite adalah suatu set algoritma yang mengamankan jaringan yang menggunakan TLS atau Transport Layer Security.

Set algoritma ini meliputi:

- Pertukaran kunci - Algoritme ini asimetris (algoritme kunci publik) dan bekerja dengan baik dengan sejumlah kecil data. Mereka digunakan untuk melindungi informasi yang diperlukan untuk membuat kunci bersama untuk transaksi yang aman.
- Enkripsi massal - Tugas ini akan mengenkripsi pesan yang dipertukarkan antara klien dan server. Algoritme ini simetris dan cenderung berkinerja sangat baik, bahkan dengan sejumlah besar data yang ditransfer.
- Otentikasi pesan - Algoritma ini menghasilkan pesan hash dan tanda tangan yang memastikan integritas sebuah pesan.

Semua hal di atas menggunakan ALG_ID - tipe data yang menentukan pengenal algoritme - agar sistem operasi mengetahui Cipher Suite mana yang akan digunakan. Anda dapat melihat daftar semua Cipher Suites yang tersedia untuk Schannel.dll di situs web Microsoft .

Mengubah Suites Cipher di Schannel.dll
Sekarang setelah Anda mengetahui lebih banyak tentang cipher suite dan Schannel.dll, sekarang saatnya untuk membahas cara mengubah Algoritma dan Protokol Kriptografi mana yang sebenarnya digunakan. Penting untuk dicatat bahwa meskipun Anda mengubah apa yang digunakan Schannel.dll, perangkat lunak yang akan Anda gunakan juga harus mendukung protokol. Berikut adalah daftar berbagai sistem operasi Windows yang mungkin Anda gunakan sebagai server.

- Windows Server Standard 2012
- Windows Server Datacenter 2012
- Windows Server Perusahaan 2008 R2
- Windows Server Standard 2008 R2
- Windows Server Datacenter 2008 R2
- 7 Enterprise Windows
- Jendela 7 Profesional
- Windows Server Perusahaan 2008
- Windows Server Standard 2008
- Windows Server Datacenter 2008
- 2003 Microsoft Windows Server, Enterprise Edition (32-bit x86)
- 2003 Microsoft Windows Server, Standard Edition (32-bit x86)
- Microsoft Windows Server 2003, Edisi Web
- Microsoft Windows XP Professional
- Microsoft Windows XP Home Edition
- Microsoft Windows Server 2000
- Server Lanjut Microsoft Windows 2000
- Microsoft Windows 2000 Professional Edition
- Microsoft Windows NT Server 4.0 Edisi Standar
- Microsoft Windows NT Server 4.0 Edisi Perusahaan
- Edisi Pengembang Microsoft Windows NT Workstation 4.0
- Windows NT 4.0 Paket Layanan 6, Windows 2000, Windows XP, Windows 2003

Pertama, kita akan melihat sistem operasi Windows 2003 dan sebelumnya. Untuk mengaktifkan dan menonaktifkan protokol yang berbeda, Anda harus menggunakan Regedt32.exe untuk menemukan kunci registri berikut ini:

```
HKEY_LOCAL_MACHINESYSTEMCurrentControlSetControlSecurityProvidersSCHANNEL
```

Selanjutnya, kita akan membahas berbagai subkunci yang tersedia - dan di mana Anda ingin membuat perubahan. Pada dasarnya, untuk mengaktifkan salah satu dari yang di bawah ini, setel data nilai DWORD ke 0xffffffff atau setel ke 0x0 untuk menonaktifkan subkunci tertentu.

- SCHANNELProtokol - Untuk mengaktifkan sistem untuk menggunakan protokol yang tidak akan dinegosiasikan secara default (seperti TLS 1.1 dan TLS 1.2), ubah data nilai DWORD dari nilai DisabledByDefault menjadi 0x0 dalam kunci registri berikut di bawah kunci Protokol:
- Subkunci SCHANNELCiphers - Kunci registri Cipher di bawah kunci SCHANNEL digunakan untuk mengontrol penggunaan algoritme simetris seperti DES dan RC4. Berikut ini adalah kunci registri yang valid di bawah kunci sandi.
- Subkey SCHANNEL / Hash - Kunci registri Hash di bawah kunci SCHANNEL digunakan untuk mengontrol penggunaan algoritme hashing seperti SHA-1 dan MD5. Berikut ini adalah kunci registri yang valid di bawah kunci Hash.
- Subkunci SCHANNEL / KeyExchangeAlgorithms - Kunci registri KeyExchangeAlgorithms di bawah kunci SCHANNEL digunakan untuk mengontrol penggunaan algoritme pertukaran kunci seperti RSA. Berikut ini adalah kunci registri yang valid di bawah kunci KeyExchangeAlgorithms

Protokol SSL sederhana namun efektif, dan merupakan protokol komunikasi aman yang paling banyak digunakan di Internet. (Perhatikan, bagaimanapun, ada kelemahan dalam algoritme MD5 di mana peneliti dapat memalsukan sertifikat yang tampaknya valid untuk digunakan dengan SSL. Ada juga serangan injeksi teks biasa terhadap TLS 1.2, yang diGambarkan sebagai CVE-2009-3555. Cacatnya melibatkan perbaikan di sisi server, sehingga banyak layanan aplikasi web perlu diperbaiki.)

SSL Session

Karena SSL umumnya digunakan dengan halaman web, sering disebut sebagai HTTPS (HTTP Secure), dan Anda akan melihat awalan https: di bilah alamat browser, serta gembok tertutup di sudut setiap kali SSL masuk. operasi. Untuk menggunakan SSL, klien meminta sesi SSL.

Setiap kali seorang pengunjung web mengunjungi situs yang menggunakan teknologi SSL, website akan menciptakan sebuah link yang terenkripsi antara sesi browser mereka dan web server. SSL adalah standar industry/protokol untuk komunikasi web yang aman dan digunakan untuk melindungi jutaan transaksi online setiap hari.

SSL memungkinkan informasi sensitif seperti data kartu kredit,username,password dan informasi penting ditransmisikan cari server ke client atau sebaliknya dengan aman karena data yang dikirim akan diaca (dienkripsi).

Web server harus memiliki sertifikat SSL sebelum dapat membuat koneksi SSL. Ketika seseorang mengaktifkan protokol SSL di server web mereka, mereka diminta untuk menjawab pertanyaan

yang akan membangun identitas mereka. Pertanyaan meminta informasi tentang kedua situs dan perusahaan. Setelah sertifikat SSL yang diminta, server web menciptakan dua kunci kriptografi, yaitu Public Key dan Private Key.

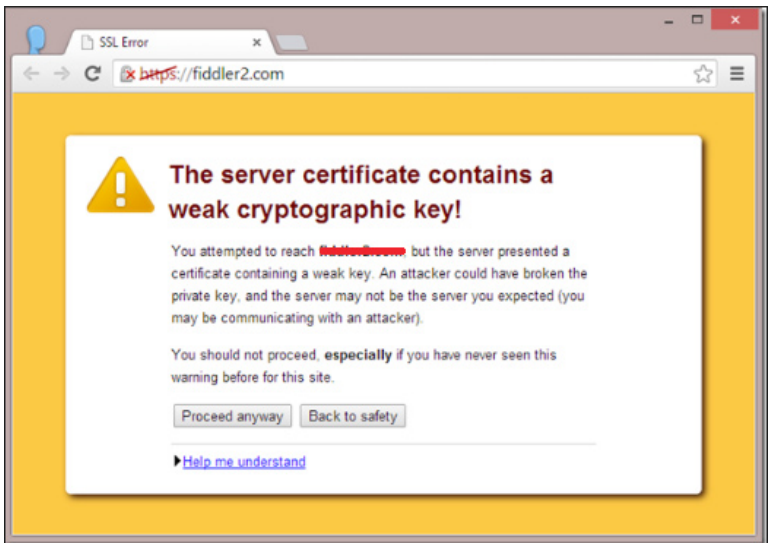
- Public key akan diberikan ke browser bersama dengan certificate ketika koneksi terenkripsi (secure connection)antara browser dan server terbentuk ,Publilc key ini akan digunakan oleh browser untuk mengenkripsi data yang yang akan dikirim ke server.
- Private Key akan digunakan oleh server untuk mendecrypt informasi terenkripsi dari browser, Private key ini sifatnya sangat rahasia dan tidak ada yang boleh tau (bocor) karena kunci ini yang digunakan untuk membongkar enkripsi data dari dan ke server.

Sertifikat SSL dapat dibedakan menjadi 2 berdasarkan pemberi otorisasi yaitu :

Self Signed Certificate SSL

Self Signed Certificate dapat diperoleh dengan gratis karena kita yang membuat dan kita pula yang menandatanginya. Jika suatu website /aplikasi menggunakan protocol SSL dengan Self signed certificate biasanya pada aplikasi client (browser) akan muncul peringatan jika certificate tidak dipercaya. Ini hal yang wajar karena didalam browser terdapat daftar root CA yang dijadikan acuan oleh browser untuk menentukan kepercayaan terdapat suatu certificate.

Hal ini dilakukan untuk memastikan bahwa certificate yang digunakan valid dan tidak sedang diacak-acak oleh hacker (Man In The Midle Attack). Seperti yang diperlihatkan pada Gambar 3.41



Ganbar 6.41 peringatan yang diterima ketika Sertifikat SSL tidak terpercaya.

Trusted Certificate Authority (Very-Signed SSL)

Jenis certificate ini harus didapatkan dengan cara membayar ke pihak CA ataupun menggunakan layanan opensource seperti LetsEncrypt yang memberikan certificate secara gratis dan terpercaya (Sudah terdaftar dan diakui resmi di dunia internet)

Untuk mendapatkan Trusted Certificate Authority kita harus membuat Certificate Signing Request(CSR) di webserver yang kita gunakan, dalam proses pembuatan certificate ini akan dihasilkan Private

Key dan Public Key. agar certificate yang kita buat dipercaya oleh browser maka ceritificate yang berisi public key harus dikirim dan di tandatangi oleh organisasi terpercaya pemberi certificate (Certificate Authority atau CA) seperti Comodo,symantec,GoDaddy,GlobalSign,Verizone dll . dalam proses ini CA tidak tau private key yang kita gunakan karena yang kita kirim adalah certificate yang berisi public key.Setelah Certificate yang kitabuat telah ditandatangani oleh CA maka kita harus memasang sertifikat tersebut di web server yang kita gunakan (CA's Root Certificate).Oleh karena itulah SSL Certificate mempunyai masa berlaku dan untuk mendapatkan nya kita harus membeli ke pihak Certificate Authority(CA) ketika kita akan membuat website dengan protocol HTTPS

Proses enkripsi menggunakan SSL

- Client /Browser meminta koneksi SSL (SSL Hello)
- Server membalas permintaan dengan mengirimkan SSL Certificate yang berisi Public key
- Client menerima dan memfalidasi keabsahan certificate tersebut (ngecek pihak CA yang menandatangi,masa berlaku,owner dll)
- Client membuat Symmetric Encryption key (sering disebut session key) dan meng enkripsi session key dengan public key yang ada di dalam certificate lalu mengirimkan nya ke server.
- Server medekripsi dengan private key data dari client yang berisi symmetric session key, dan menggukanan key tersebut untuk mengirim data dari server ke client . koneksi SSL pun terbentuk

Alasan menggunakan SSL

Alasan utama mengapa menggunakan SSL adalah untuk menjaga informasi sensitif selama dalam proses pengiriman melalui Internet dengan cara dienkripsi, sehingga hanya penerima pesan yang dapat memahami dari hasil enkripsi tersebut. Hal ini sangat penting, karena informasi yang kita kirimkan di Internet membutuhkan proses perjalanan dari komputer ke komputer sampai mencapai server tujuan. Komputer lain yang ada diantara Anda dan server dapat melihat nomor Kartu Kredit Anda, username dan passwords, dan informasi sensitive lainnya bila hal ini tidak dienkripsi dengan Sertifikat SSL.

Ketika sertifikat SSL digunakan, informasi menjadi tidak dapat terbaca oleh siapapun kecuali ke server yang memang dituju saat mengirim informasi tersebut. Hal ini melindungi informasi tersebut dari hackers dan pencuri identitas. Dengan memasang SSL pada website bisnis kita maka konsumen akan merasa aman karena informasi yang dikirim dari dan ke server sudah diacak dan hanya server yang bisa membaca informasi tersebut.Sehingga akan terjalin kepercayaan konsumen terhadap website kita.

Rute Onion

Istilah TOR diambil dari singkatan kata "The Onion Router" yang tidak lain merupakan sebuah jaringan virtual untuk meningkatkan keamanan dan kerahasiaan data di dunia maya. Lantas, kenapa alat ini disebut Onion atau bawang merah dalam bahasa Indonesia? Menurut laman wikipedia, filosofi penamaan ini berdasarkan pada skema TOR yang terdiri dari banyak lapisan dengan tujuan menyembunyikan traffic atau identitas asli.
Karena mengutamakan sistem anonim, TOR Browser beroperasi di jaringan khusus yang terdiri dari banyak server. Menariknya, setiap kita menggunakannya maka sistem akan memilih secara rancom

tiga buah relay untuk menyambungkan koneksi ke Internet. Setiap koneksi yang digunakan juga harus melewati proses enkripsi sehingga masng-masing relay tersebut hanya bisa melihat IP dari tiap ujung-ujungnya saja, karena lintasan jaringan sepenuhnya membentuk pola simpul.

Berkat pola keamanan berlapis tersebut, hasilnya lalu lintas internet pengguna tidak akan mudah dilacak oleh orang lain melalui alamat IP. Sementara disisi lain, pengguna bisa menembus filter atau pemblokian terhadap suatu konten yang dimuat di dalam website.

Meski bersifat anonim dan bisa menembus semua konten di internet, Tor browser sangat lemah dalam hal kecepatan akses internet. Hal ini tentu sebagai dampak dari adanya relay menuju beberapa server sekaligus yang tersebar di seluruh belahan dunia sebelum akhirnya sampai ke situs tujuan.

Contohnya, ketika Anda yang berada di Indonesia ingin mengakses website dengan server di Singapura maka jaringan harus di relay dulu melewati beberapa negara di dunia secara acak, bahkan bisa melewati server di Amerika Serikat lebih dulu sebelum sampai ke tujuan. Maka dari itulah setip konten yang diakses melalui Tor browser bisa dipastikan datang lebih lambat dari kecepatan normal.

Proyek Tor (https://www.torproject.org/) mendistribusikan perangkat lunak gratis dan mendaftarkan jaringan terbuka yang menggunakan perutean bawang untuk bertahan dari analisis lalu lintas. Tor (yang merupakan singkatan dari The Onion Router) melindungi dengan mentransfer komunikasi di sekitar jaringan terdistribusi lebih dari 5.000 relai yang dijalankan oleh sukarelawan di seluruh dunia: Tor mencegah orang luar yang menonton koneksi Internet mempelajari situs apa yang dikunjungi pengguna, dan mencegah situs mempelajari lokasi fisik pengguna. Menurut Bloomberg BusinessWeek 23 Jan 2014, pengguna Tor berkisar dari aktivis Iran yang menghindari sensor pemerintah mereka untuk mengirimkan Gambar dan berita selama protes setelah pemilihan presiden 2009, hingga warga negara China yang secara teratur menggunakannya untuk mengatasi pembatasan ketat negara di Internet konten dan akses. Tor juga memfasilitasi apa yang disebut sisi gelap Internet, atau Darknet, yang digunakan untuk mengimplementasikan lalu lintas ilegal dalam pornografi anak, obat-obatan, dan kartu kredit curian serta detail identitas.

3.6.3 IP Security Protocol Suite (IPsec)

IPSec (singkatan dari IP Security) adalah suatu protokol yang dipakai untuk mengamankan transmisi datagram dalam suatu internetwork berbasis TCP/IP. IPSec mendefiniskan beberapa standar untuk melakukan enkripsi data dan juga integritas data pada lapisan kedua dalam DARPA Reference Model (internetwork layer). IPSec melakukan enkripsi terhadap data pada lapisan yang sama dengan protokol IP dan menggunakan teknik tunneling untuk mengirimkan informasi menempuh jaringan Internet atau dalam jaringan Intranet secara terlindung. IPSec dirumuskan oleh badan Internet Engineering Task Force (IETF) dan diimplementasikan di dalam banyak sistem operasi. Windows 2000 adalah sistem operasi pertama dari Microsoft yang mendukung IPSec.

IPSec diimplementasikan pada lapisan transport dalam OSI Reference Model untuk melindungi protokol IP dan protokol-protokol yang lebih tinggi dengan menggunakan beberapa kebijakan keamanan yang dapat dikonfigurasikan untuk memenuhi kebutuhan keamanan pengguna, atau jaringan. IPSec umumnya diletakkan sebagai suatu lapsian tambahan di dalam stack protokol TCP/

IP dan diatur oleh setiap kebijakan keamanan yang diinstalasikan dalam setiap mesin komputer dan dengan suatu skema enkripsi yang dapat dinegosiasikan antara pengirim dan penerima. Kebijakan-kebijakan keamanan tersebut mengandung himpunan filter yang diasosiasikan dengan kelakuan tertentu. Ketika suatu alamat IP, nomor port TCP dan UDP atau protokol dari suatu paket datagram IP cocok dengan filter tertentu, karenanya kelakukan yang dikaitkan dengannya hendak diaplikasikan terhadap paket IP tersebut.

Dalam sistem operasi Windows 2000, Windows XP, dan Windows Server 2003, kebijakan keamanan tersebut dibuat dan diputuskan pada level domain Active Directory atau pada host individual dengan menggunakan snap-in IPSec Management dalam Microsoft Management Console (MMC). Kebijakan IPSec tersebut, mengandung beberapa peraturan yang menentukan kebutuhan keamanan untuk beberapa wujud komunikasi. Peraturan-peraturan tersebut dipakai ntuk memulai dan mengontrol komunikasi yang terlindung berlandaskan sifat lalu lintas IP, sumber lalu lintas tersebut dan tujuannya. Peraturan-peraturan tersebut dapat menentukan metode-metode autentikasi dan negosiasi, atribut ronde tunneling, dan jenis koneksi.

Untuk membuat suatu sesi komunikasi yang terlindung antara dua komputer dengan menggunakan IPSec, karenanya dibutuhkan suatu framework protokol yang disebut dengan ISAKMP/Oakley. Framework tersebut mencakup beberapa algoritma kriptografi yang telah ditentukan sebelumnya, dan juga dapat diperluas dengan menambahkan beberapa sistem kriptografi tambahan yang dibuat oleh pihak ketiga. Selama ronde negosiasi dilakukan, persetujuan hendak tercapai dengan cara autentikasi dan kemanan yang hendak dipakai, dan protokol pun hendak membuat suatu kunci yang dapat dipakai bersama (shared key) yang nantinya dipakai sebagi kunci enkripsi data. IPSec mendukung dua buah sesi komunikasi keamanan, yakni sebagai berikut:

- protokol Authentication Header (AH): menawarkan autentikasi pengguna dan perlindungan dari beberapa serangan (umumnya serangan man in the middle), dan juga menyediakan fungsi autentikasi terhadap data serta integritas terhadap data. Protokol ini mengizinkan penerima untuk merasa yakin bahwa identitas si pengirim adalah mempunyai mempunyainya, dan data pun tidak dimodifikasi selama transmisi. Namun, protokol AH tidak menawarkan fungsi enkripsi terhadap data yang ditransmisikannya. Informasi AH diisi ke dalam header paket IP yang dikirimkan dan dapat dipakai secara sendirian atau bersamaan dengan protokol Encapsulating Security Payload.
- protokol Encapsulating Security Payload (ESP): Protokol ini melakukan enkapsulasi serta enkripsi terhadap data pengguna untuk meningkatkan kerahasiaan data. ESP juga dapat memiliki skema autentikasi dan perlindungan dari beberapa serangan dan dapat dipakai secara sendirian atau bersamaan dengan Authentication Header. Sama seperti halnya AH, informasi mengenai ESP juga diisi ke dalam header paket IP yang dikirimkan.

Beberapa perangkat keras serta perangkat lunak dapat dikonfigurasikan untuk mendukung IPSec, yang dapat dilakukan dengan menggunakan enkripsi kunci publik yang dipersiapkan oleh Certificate Authority (dalam suatu public key infrastructure) atau kunci yang dipakai bersama yang telah ditentukan sebelumnya (skema Pre-Shared Key/PSK) untuk melakukan enkripsi secara privat.

Sebagai bagian dari paket IPv6, paket protokol keamanan IP, atau IPsec, diadopsi oleh IETF. Dirancang untuk mengatasi kekurangan mendasar seperti menjadi subjek spoofing, penyadapan, dan

pembajakan sesi, protokol IPsec mendefinisikan sarana standar untuk menangani data terenkripsi. IPsec diimplementasikan pada lapisan IP (3), sehingga melindungi data yang dihasilkan di semua lapisan di atasnya, khususnya, informasi kontrol TCP dan UDP, serta data aplikasi. Oleh karena itu, IPsec tidak memerlukan perubahan pada sejumlah besar protokol atau aplikasi TCP dan UDP yang ada.

IPsec agak mirip dengan SSL, karena mendukung otentikasi dan kerahasiaan dengan cara yang tidak memerlukan perubahan signifikan baik di atasnya (dalam aplikasi) atau di bawahnya (dalam protokol TCP). Seperti SSL, SSL dirancang untuk tidak bergantung pada algoritme kriptografi tertentu dan memungkinkan kedua pihak yang berkomunikasi untuk menyepakati seperangkat protokol yang saling mendukung.

3.6.4 Security Association

Security Association adalah suatu hubungan simplex yang menghasilkan layanan keamanan lalulintas yang dibawanya. Layanan keamanan ini dihasilkan oleh SA dengan penggunaan AH atau ESP tapi bukan penggunaan keduanya. Jika proteksi AH dan ESP diterapkan dalam aliran lalulintas, kemudian dua tau lebih SA di-create untuk menghasilkan proteksi dalam aliran lalulintas. Untuk mengamankan komunikasi dua arah antara dua Host, atau antara dua security gateway maka dibutuhkan dua SA (satu di masing-masing arah).

Security Association secara unik dikenali dari tiga komponen yaitu Security Parameter Index (SPI), alamat tujuan IP dan protokol keamanan (AH atau ESH). Nilai SPI mencakup nilai 1 sampai 255 yang ditetapkan oleh IANA (Internet Assigned Number Authority) untuk penggunaan dimasa yang akan datang. Nilai SPI nol (0) ditetapkan untuk penggunaan implementrasi khusus lokal dan tidak dikirim lewat kabel. Sebagai contoh implementasi manajemen kunci mempunyai nilai SPI nol yang berarti tidak ada Security Association selama periode ketika implementasi IPSec telah meminta bahwa entitas manajemen kunci tersebut menetapkan SA baru, tetapi SA belum masih belum ditetapkan. Pada prinsipnya, alamat tujuan merupakan unicast address, IP broadcast address atau multicat group address. Bagaimanapun mekanisme manajemen SA IPSec saat ini hanya didefinisikan untuk SA unicast. Oleh karena itu, untuk diskusi pemaparan selanjutnya SA dideskripsikan utamanya untuk komunikasi point-to-point, meskipun konsepnya dapat diaplikasikan untuk kasus komunikasi point-to-multipoint.

Seperti telah dituliskan dibagian sebelumnya didefinisikan dua tipe SA yaitu mode transport dan mode tunnel. SA mode transport adalah SA antara dua Host. Dalam kasus dimana link security ingin digunakan antara dua sistem intermediate sepanjang path mode transport juga dapat digunakan antara dua security gateway. Dalam kasus terbaru mode transport juga digunakan untuk mensuport IP-in-IP atau GRE tunneling melalui SA mode transport. Catatan bahwa fungsi kontrol akses merupakan bagian yang penting dari IPSec secara significan dibatasi dalam konteks ini. Sehingga penggunaan mode transport harus dievaluasi secara hati-hati sebelum digunakan. Dalam Ipv4, header protokol keamanan mode transport terlihat setelah header IP dan beberapa pilihan lain dan sebelum protokol layer yang lebih tinggi (seperti TCP atau UDP) dalam IPv6, header protokol keamanan terlihat setelah header base IP dan ekstensionnya, tetapi mungkin juga terlihat sebelum atau sesudah pilihan tujuan dan sebelum protokol layer yang lebih tinggi. Dalam kasus ESP SA

mode tranport menyediakan layanan keamanan hanya untuk protokol layer yang lebih tinggi darinya, tidak untuk IP header atau ekstension header yang mendahului ESP header. Dalam kasus untuk AH, proteksi juga ditambahkan ke bagian yang dipilih dari IP header, bagian yang dipilih dari ekstension header dan option pilihan (yang terdapat pada header IPv4, Hop-by-Hop ekstension header Ipv6, atau ekstension header tujuan pada Ipv6).

SA mode tunnel sebenarnya adalah SA yang diaplikasikan di IP tunnel. Dengan hanya sepasang pengecualian, kapan saja ujung manapun dari SA adalah Security gateway, SA harus mode tunnel. SA diantara dua security gateway pada dasarnya adalah SA mode tunnel seperti SA antara Host dan Security Gateway. Catatan bahwa dalam kasus dimana lalulintas ditujukan untuk security gateway seperti SNMP commands, security gateway berlaku sebagai Host dan mode transport diperbolehkan. Tapi dalam kasus tersebut security gateway tidak berlaku sebagai gateway. Seperti tertulis diatas security gateway mungkin mendukung mode transport untu menyediakan link (Hubungan) keamanan. Dua Host dapat menyusun sebuah mode tunnel antara mereka. Kebutuhan untuk transit traffic SA meliputi security gateway untuk menjadi SA tunnel mengacu ke kebutuhan untuk menghindari problem potensial dengan memperhatikan fragmentasi dan penyusunan ulang dari paket IPSec dan dalam keadaan dimana path yang banyak (melalui security gateway yang berbeda) menuju ke tujuan yang sama dibelakang security gateway.

Untuk SA mode tunnel ada header IP luar yang menspesifikasikan pemrosesan tujuan IPSec ditambah dengan header IP dalam yang menunjukkan tujuan terakhir dari peket tersebut. Header protokol keamana terlihat setelah header IP luar dan sebelum header IP dalam. Jika AH diterapkan dalam mode tunnel bagian header IP luar diperoleh proteksi seperti halnya paket IP yang disalurkan (semua header IP dalam diproteksi seperti halnya layer protokol yang lebih tinggi). Jika ESP diterapkan proteksi dikerjakan hanya pada paket yang disalurkan tidak pada header luar.

Fungsi Security Association

Layanan keamanan yang diberikan oleh SA bergantung pada protokol security yang dipilih, mode SA, endpoint SA, dan pada pemilihan layanan optional yang disertakan dalam protokol. Sebagai contoh, AH menyediakan layanan otentifikasi data asal dan integritas hubungan untuk datagram IP. AH juga memberikan layanan anti-replay, untuk membantu counter menolak service attack. AH merupakan protokol yang cocok utnuk diterapkan jika kerahasiaan kurang dipedulikan. AH juga menyediakan otentiofikasi untuk bagian yang dipilih dari header IP, yang mana mungkin dibutuhkan di beberapa keadaan. Sebagai contoh, jika integritas dari sebuah header option IPv4 atau header ekstension IPv6 harus diproteksi dalam rute antara pengirim dan penerima, AH menyediakan layanan tersebut (kecuali untuk bagian header IP yang tidak dapat ditebak dan dapat berubah.)

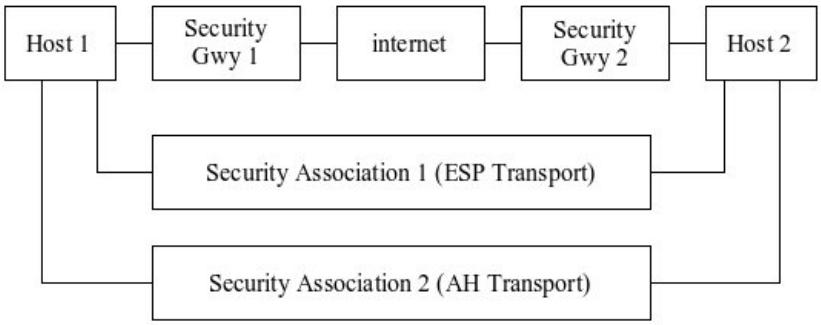
ESP memyediakan kerahasiaan untuk lalulintas. (kekuatan layanan kerahasiaan ini bergantung pada algoritma enkripsi yang diterapkan). ESP juga menyediakan layanan otentifikasi. Jika otentifiaksi dirundingkan untuk SA ESP, penerima juga memilih untuk menguatkan layanan anti-reply dengan fitur fyang sama seperti layanan anti-reply pada AH. Jangkauan otentifikasi yang diberikan oleh ESP lebih dangkal daripada AH, seperti header IP yang berada diluar header ESP tidak terlindungi. Hanya jika protokol layer berikutnya perlu untuk diotentifikasi, maka otentifikasi ESP merupakan pilihan yang tepat dan akan lebih efisien ruang daripada AH. Perlu diingat meskipun kerahasiaan dan otentifikasi merupakan layanan optional tetapi kedua layanan tersebut tidak dapat dihilangkan. Setidaknya salah satu dari layanan tersebut harus dipilih.

Apabila layanan kerahasiaan (confidentiality) dipilih, maka SA ESP (mode tunnel) antara dua security gateway dapat memberikan kerahasiaan aliran lalulintas parsial. Penggunaan mode tunnel menperbolehkan header IP dalam (inner IP header) untuk dienkripsikan, merahasiakan identitas dari sumber dan tujuan dari lalulintas tersebut. Lebih dari, lapisan ESP payload juga dapat dilibatkan untuk menyembunyikan ukuran dari peket. Layanan confidentiality aliran lalulintas yang sama juga dapat diberikan ketika mobile user menggunakan alamat IP dynamic dalam konteks dialup, dan menetapkan SA ESP (mode tunnel) menjadi firewall (berlaku sebagai security gateway).

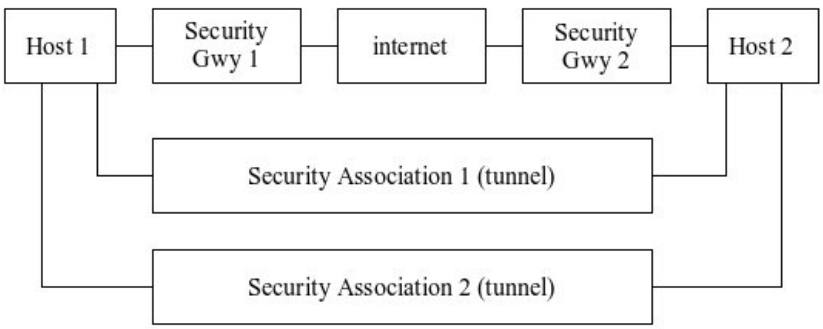
Menggabungkan Security Association

Security Association dapat digabungkan menjadi satu kemasan dengan dua cara, yaitu transport adjacency dan iterated tunneling.

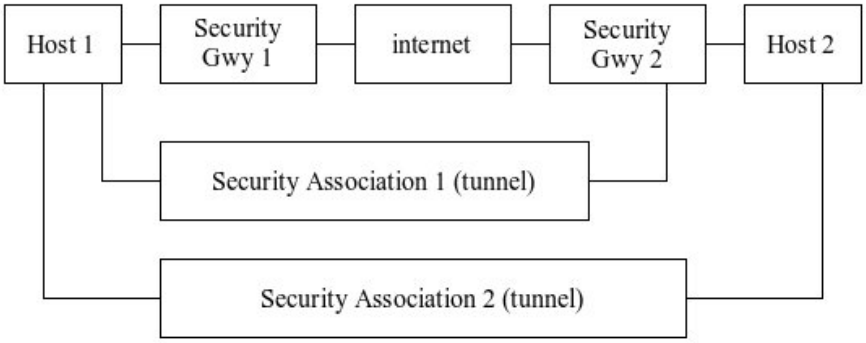
1. Transport Adjacency. Cara ini mengacu pada penggunaan lebih dari satu protokol security dalam datagram IP yang sama, tanpa melibatkan tunneling.



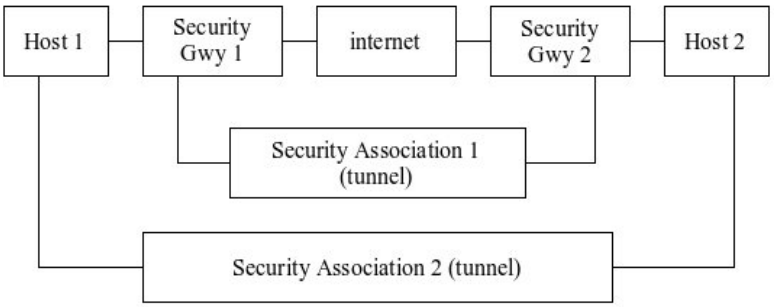
2. Iterated Tunneling. Cara ini mengacu pada pengaplikasian multiple layer dari protokol security yang dipengaruhi melalui IP tunneling. Pendekatan ini membolehkan nesting yang multiple level, karena masing-masing tunnel dimulai dan diakhiri pada tempat IPSec yang berbeda sepanjang path (lintasan). Ada tiga basic kasus pada iterated tunneling, yaitu:
   - Kedua ujung (endpoint) untuk SA adalah sama. Tunnel dalam atau tunnel luar keduanya dapat berupa AH atau ESP.



- Salah satu dari ujung SA adalah sama.



- Tidak ada ujung yang sama.



3.6.5 Virtual Private Network

Enkripsi tautan dapat memberi pengguna jaringan perasaan bahwa mereka berada di jaringan pribadi, bahkan ketika itu adalah bagian dari jaringan publik. Selanjutnya, diterapkan pada tingkat tautan, enkripsi dan dekripsi tidak terlihat oleh pengguna. Untuk alasan ini, pendekatan ini disebut jaringan pribadi virtual (atau VPN).

Virtual Private Network atau biasa disebut VPN adalah Sebuah cara aman untuk mengakses local area network yang berada pada jangkauan tertentu, dengan menggunakan internet atau jaringan umum lainnya untuk melakukan transmisi data paket secara pribadi. salah satu fungsi adanya vpn adalah keamanan dalam berkomunikasi atau dalam pertukaran data, juga tidak memungkinkan pihak lain untuk menyusup ke traffic (lalu lintas jaringan) yang tidak semestinya.

Menurut IETF, *Internet Engineering Task Force, VPN is an emulation of [a] private Wide Area Network (WAN) using shared or public IP facilities, such as the Internet or private IP backbones*.VPN merupakan suatu bentuk private internet yang melalui public network (internet), dengan menekankan pada keamanan data dan akses global melalui internet. Hubungan ini dibangun melalui suatu tunnel (terowongan) virtual antara 2 node. Seperti yang dijelaskan pada Gambar 3.48. VPN Menghubungkan PC dengan jaringan publik atau internet namun sifatnya privat, karena bersifat privat maka tidak semua orang bisa terkoneksi ke jaringan ini dan mengaksesnya. Hal ini di lakukan untuk menjaga kemanan dalam pertukaran data atau informasi.

Konsep kerja VPN pada dasarnya VPN Membutuhkan sebuah server yang berfungsi sebagai penghubung antar PC. Jika digambarkan kira-kira seperti ini

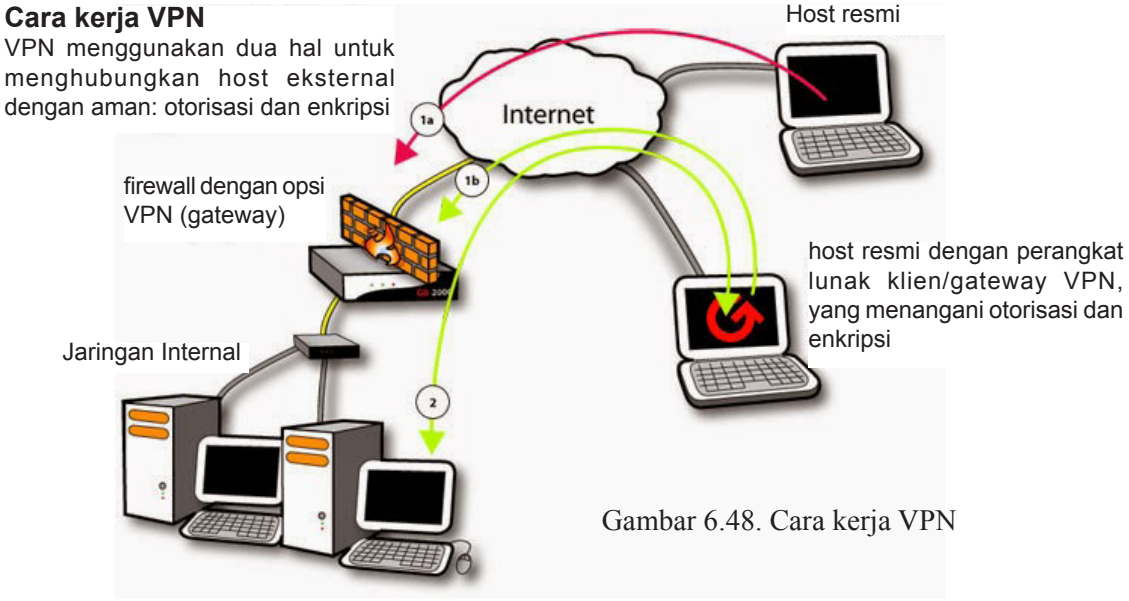
internet <—> VPN Server <—-> VPN Client <—-> Client

bila digunakan untuk menghubungkan 2 komputer secara private dengan jaringan internet maka seperti ini:

Komputer A <—> VPN Clinet <—> Internet <—> VPN Server <—> VPN Client <—> Komputer B

Jadi semua koneksi diatur oleh VPN Server sehingga dibutuhkan kemampuan VPN Server yang memadai agar koneksinya bisa lancar.

Biasanya, keamanan fisik dan keamanan administratif cukup kuat untuk melindungi transmisi di dalam perimeter situs (misalnya kantor, gedung, pabrik, atau kampus). Dengan demikian, paparan terbesar bagi pengguna terjadi ketika komunikasi meninggalkan lingkungan yang dilindungi. Enkripsi tautan antara dua titik akhir yang diamankan dapat mencapai hasil ini.



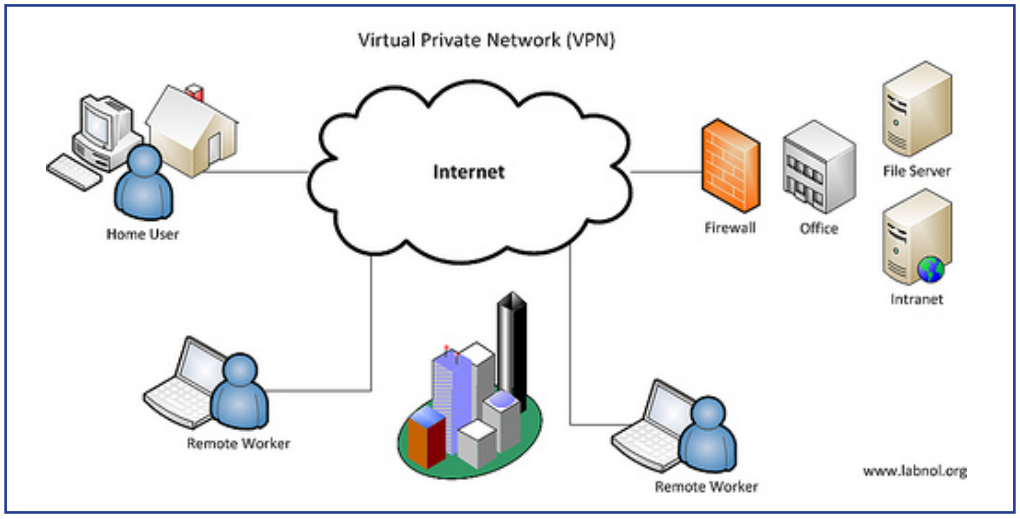
Gambar 6.48. Cara kerja VPN

Untuk jaringan pribadi virtual, kami mempertimbangkan dua kasus. Yang pertama, sebuah perusahaan memiliki dua kantor yang terpisah secara fisik, dan para karyawan ingin bekerja sebagai satu unit, bertukar data sensitif seolah-olah mereka berada di satu kantor yang dilindungi. Setiap kantor memelihara jaringannya sendiri. Kedua kantor dapat menerapkan jaringan pribadi dengan memperoleh, mengelola, dan memelihara peralatan jaringan mereka sendiri untuk menyediakan tautan pribadi antara dua situs. Solusi ini seringkali mahal, dan perusahaan bertanggung jawab penuh untuk memelihara koneksi. Seringkali perusahaan seperti itu tidak berada dalam bisnis jaringan, tetapi mempertahankan satu tautan mengharuskan mereka untuk menjadi atau menyewa administrator jaringan. Namun, perusahaan mungkin tidak menyukai risiko mengomunikasikan informasi sensitif perusahaan melalui jaringan publik dan bersama.

Alternatifnya adalah jaringan pribadi virtual antar kantor. Dengan enkripsi tautan, semua komunikasi antar situs dienkripsi. Sebagian besar biaya solusi ini adalah dalam memperoleh dan menyiapkan jaringan. Beberapa komunikasi karyawan akan melibatkan rencana sensitif dan data rahasia; komunikasi lainnya akan menjadi obrolan kantor biasa tentang tim olahraga atau rencana makan siang. Hampir tidak ada salahnya mengenkripsi obrolan serta lalu lintas penting karena waktu tambahan untuk mengenkripsi dan mendekripsi semua lalu lintas biasanya tidak signifikan dibandingkan dengan waktu transmisi jaringan.

Firewall (dijelaskan di bagian selanjutnya) dapat mengimplementasikan VPN. Saat pengguna pertama kali menjalin komunikasi dengan firewall, pengguna dapat meminta sesi VPN dengan firewall. Klien pengguna dan firewall menegosiasikan kunci enkripsi sesi, dan firewall dan klien kemudian menggunakan kunci itu untuk mengenkripsi semua lalu lintas di antara keduanya. Dengan cara ini, jaringan yang lebih besar dibatasi hanya untuk mereka yang diberi akses khusus oleh VPN. Dengan kata lain, pengguna merasa seolah-olah jaringan yang lebih besar bersifat pribadi, meskipun sebenarnya tidak. Dengan VPN, kami mengatakan bahwa komunikasi melewati terowongan terenkripsi.

Virtual Private Network atau VPN adalah suatu jaringan pribadi yang dibuat dengan menggunakan jaringan publik, atau dengan kata lain menciptakan suatu WAN yang sebenarnya terpisah baik secara fisikal maupun geografis sehingga secara logikal membentuk satu netwok tunggal, paket data yang mengalir antar site maupun dari user yang melakukan remote akses akan mengalami enkripsi dan authentikasi sehingga menjamin keamanan, integritas dan validitas data. Lebih jelasnya, VPN adalah sebuah koneksi Virtual yang bersifat private mengapa disebut demikian karena pada dasarnya jaringan ini tidak ada secara fisik hanya berupa jaringan virtual dan mengapa disebut private karena jaringan ini merupakan jaringan yang sifatnya private yang tidak semua orang bisa mengaksesnya. Perhatikan Gambar 3.49



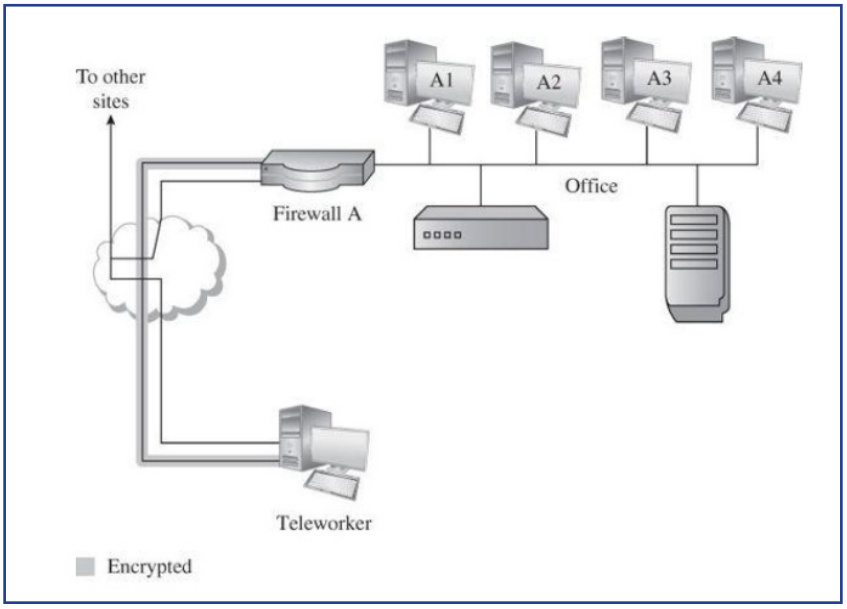
Gambar 3.49 Jaringan VPN

Sekarang perhatikan kasus kedua dari seorang telecommuter: Jeannie, seorang karyawan yang bekerja dari rumah. Agar produktif dari rumah, dia perlu menggunakan file pusat dan sumber daya yang dapat dia akses dengan mudah dari kantor. Namun yang jelas, perusahaan tidak ingin sumber daya tersebut diekspos ke publik. Dari rumahnya, Jeannie menggunakan teknologi seperti DSL

atau kabel untuk menyambung ke penyedia Internet yang mengarahkan sebagian lalu lintasnya ke kantornya dan sisanya ke situs web lain. Dengan demikian, dia muncul di kantornya seperti pengguna web lainnya. Dia juga dapat menggunakan VPN untuk komunikasi kantor yang aman.

Jaringan pribadi virtual dibuat ketika firewall berinteraksi dengan layanan otentikasi di dalam perimeter. Firewall dapat meneruskan data autentikasi pengguna ke server autentikasi dan, setelah konfirmasi identitas terotentikasi, firewall memberi pengguna hak istimewa keamanan yang sesuai. Misalnya, Jeannie mungkin diizinkan untuk mengakses sumber daya yang tidak tersedia untuk pengguna umum. Firewall mengimplementasikan kontrol akses ini berdasarkan VPN. VPN dengan akses istimewa ditunjukkan pada Gambar 3-50. Pada Gambar tersebut, firewall melewati identitas (hak istimewa) server internal Jeannie.

Enkripsi adalah alat yang ampuh, tetapi penggunaannya penuh dengan masalah. Algoritme berjalan dengan sendirinya dengan rapi, dan banyak implementasi dalam perangkat keras dan perangkat lunak mudah digunakan dan dapat diandalkan. Mengelola kunci untuk mendukung banyak pengguna virtual itu rumit tetapi didorong oleh Tabel, menjadikannya tugas yang baik untuk komputer. Kunci juga harus dilindungi dalam penyimpanan di kedua ujungnya.



Gambar 3-50 VPN dengan Akses Istimewa

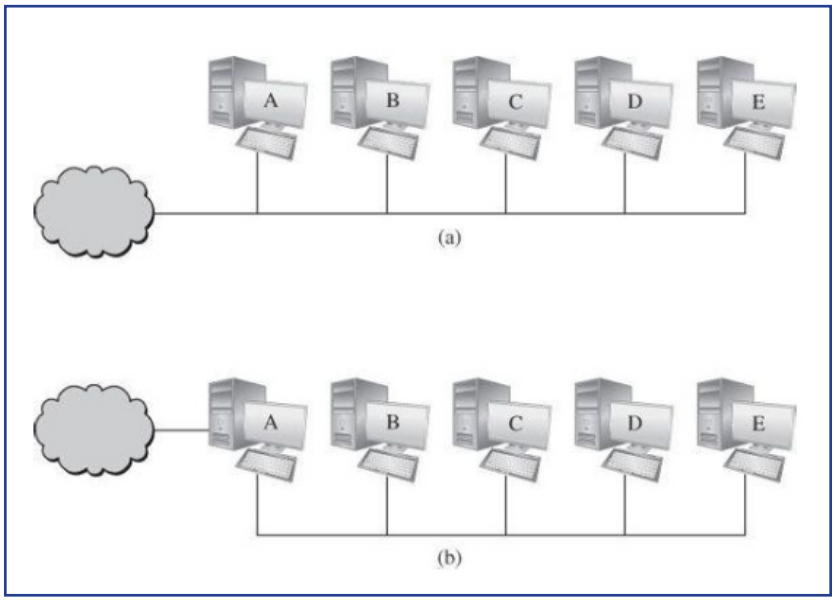
3.6.6 Sistem Arsitektur

Jika Anda mencoba membatasi informasi yang diungkapkan oleh pemindaian port tentang jaringan dan host serta layanannya, pendekatan alami adalah dengan mengelompokkan jaringan, dengan banyak host pada segmen yang tidak langsung terlihat ke luar.

Sebagai contoh, pikirkan tentang sistem telepon rumah sakit yang khas. Beberapa fungsi, seperti sumber daya manusia atau layanan pasien, perlu menerima panggilan langsung dari pihak luar, dan nomor telepon tersebut dapat dipublikasikan dalam direktori. Tetapi Anda tidak ingin nomor telepon ruang operasi atau laboratorium diagnostik atau bahkan rumah tangga atau pemeliharaan tersedia untuk orang luar. Rumah sakit akan menerbitkan nomor operator umum; jika orang luar

memiliki alasan yang meyakinkan untuk perlu terhubung dengan ruang operasi, operator dapat menentukan itu dan meneruskan panggilan atau mungkin mengarahkannya ke orang lain yang dapat membantu dengan lebih baik. Eksekutif tertentu mungkin memiliki asisten administrasi yang menyaring panggilan mereka, mengizinkan beberapa panggilan segera, menerima pesan untuk orang lain, dan mengarahkan lagi orang lain. Arsitektur yang tersirat dalam deskripsi layanan telepon rumah sakit ini adalah sejumlah kecil telepon yang dapat diakses secara eksternal (relatif terhadap jumlah telepon internal yang lebih besar), dan beberapa titik tersedak lainnya yang menyaring dan mengalihkan semua panggilan lainnya.

Situasi serupa terjadi dengan jaringan. Bandingkan jaringan Gambar 3-51 (a) dengan jaringan Gambar 3-51 (b). Pada Gambar 3-51 (a), kelima komputer A–E terlihat dari luar jaringan, sedangkan pada Gambar 3-51 (b) hanya komputer A yang terlihat. Jaringan perangkat B– E di bagian (b) dikenal sebagai subnet yang dilindungi, dan perangkat A disebut gateway dual-homed.

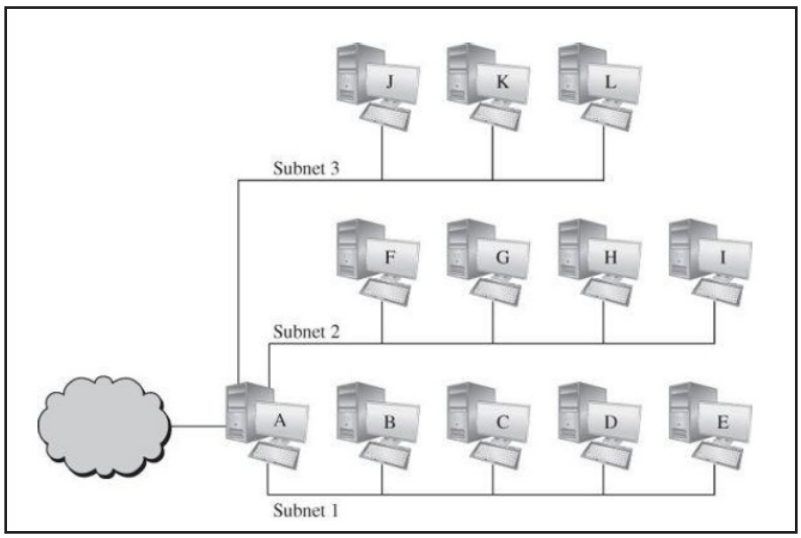


Gambar 3-51 (a) Perangkat Terlihat. (b) Perangkat yang Kurang Terlihat

Arsitektur (a) memberikan beberapa keunggulan dibandingkan arsitektur (b). Pertama, host A menjadi satu titik kegagalan: Jika gateway A tidak tersedia karena alasan apa pun, gateway A tidak dapat meneruskan lalu lintas ke atau dari B–E, yang berarti mereka secara efektif terputus dari jaringan. Selanjutnya, perangkat gateway A menjadi potensi kemacetan, sehingga perangkat B melalui E berbagi akses melalui A; jika A lambat atau jika salah satu B–E menggunakan bandwidth jaringan dalam jumlah besar, kinerja mesin lain akan terganggu.

Kita bahkan dapat memperluas gagasan tentang subnet yang dilindungi menjadi dua atau lebih subnet, seperti yang ditunjukkan pada Gambar 3-52. Ketiga subnet tersebut dapat digunakan untuk departemen atau grup pengguna yang terpisah, atau dapat dialokasikan secara geografis. Tentu saja, semakin banyak subnet yang didukung gateway A, semakin besar risiko jika perangkat A gagal.

Konfigurasi ulang arsitektur jaringan membatasi atau mempersulit pergerakan, tetapi tidak membahas tujuan keamanan pusat dari akses terkontrol. Untuk mencapai tujuan itu, kami bergantung pada perangkat yang disebut firewall, yang akan kami jelaskan selanjutnya.



Gambar 3-52 Beberapa Subnet Terproteksi

3.7 Firewall

Firewall pada bangunan, sesuai dengan namanya, adalah dinding yang dimaksudkan untuk menghambat penyebaran api dari satu bagian bangunan ke bagian lain, misalnya antara satu apartemen dengan apartemen lainnya. Firewall dibangun dari bahan yang tahan terhadap api dengan intensitas atau durasi tertentu; mereka mencegah penyebaran api tetapi tidak dijamin atau dimaksudkan untuk menghentikan kebakaran yang sangat hebat.

Sebagai perangkat keamanan komputer, firewall jaringan serupa, melindungi satu subnet dari bahaya dari subnet lain. Penggunaan utama firewall adalah untuk melindungi subnetwork internal dari banyak ancaman yang telah kami jelaskan di Internet liar. Firewall juga dapat digunakan untuk memisahkan segmen jaringan internal, misalnya, untuk menjaga kerahasiaan tinggi dari jaringan penelitian sensitif dalam organisasi yang lebih besar.

3.7.1 Definisi Firewall

Firewall adalah salah satu perangkat keamanan yang paling penting untuk jaringan. Firewall secara resmi ditemukan pada awal 1990-an, tetapi konsepnya benar-benar mencerminkan monitor referensi (diperkenalkan pada Bab 2) dari dua dekade sebelumnya.

Firewall adalah perangkat yang menyaring semua lalu lintas antara jaringan yang dilindungi atau "di dalam" dan jaringan yang kurang dapat dipercaya atau "di luar". Biasanya firewall berjalan pada perangkat khusus; karena ini adalah satu titik di mana lalu lintas disalurkan, kinerja itu penting, yang berarti bahwa hanya fungsi firewall yang harus dijalankan pada mesin firewall.

Dalam praktiknya, firewall adalah komputer dengan memori, perangkat penyimpanan, kartu antarmuka untuk akses jaringan, dan perangkat lainnya. Ini menjalankan sistem operasi dan menjalankan program aplikasi. Seringkali perangkat keras, sistem operasi, dan aplikasi dijual sebagai satu paket, sehingga aplikasi firewall (program) terkadang juga disebut firewall.

Sistem firewall biasanya tidak memiliki compiler, linker, loader, editor teks umum, debugger, pustaka pemrograman, atau alat lain yang mungkin digunakan penyerang untuk memperluas serangan dari komputer firewall. Karena firewall adalah kode yang dapat dieksekusi, penyerang dapat mengkompromikan kode itu dan mengeksekusi dari perangkat firewall. Misalnya, Cisco menjalankan sistem operasi berpemilik IOS pada sakelar, router, dan firewallnya. Pada tahun dari Juli 2013 hingga Juni 2014 Cisco merilis 27 patch untuk IOS untuk menangani berbagai masalah dari kerentanan yang dapat menyebabkan penolakan layanan, untuk mengatasi masalah terjemahan dan perutean. Untuk waktu yang lama perangkat lunak berpemilik pada perangkat jaringan hampir tidak menjadi target penyerang, tetapi bahkan perangkat ini sekarang menarik perhatian peretas.

Tujuan dari firewall adalah untuk menjaga hal-hal "buruk" di luar lingkungan yang dilindungi. Untuk mencapai itu, firewall menerapkan kebijakan keamanan yang dirancang khusus untuk mengatasi hal-hal buruk yang mungkin terjadi. Misalnya, kebijakannya mungkin untuk mencegah akses apa pun dari luar (sambil tetap mengizinkan lalu lintas lewat dari dalam ke luar). Atau, kebijakan mungkin mengizinkan akses hanya dari tempat tertentu, dari pengguna tertentu, atau untuk aktivitas tertentu. Bagian dari tantangan melindungi jaringan dengan firewall adalah menentukan kebijakan keamanan mana yang memenuhi kebutuhan instalasi. Firewall menegakkan aturan yang telah ditentukan yang mengatur lalu lintas apa yang dapat mengalir.

Orang-orang di komunitas firewall (pengguna, pengembang, dan pakar keamanan) tidak setuju tentang cara kerja firewall. Secara khusus, komunitas terbagi tentang perilaku default firewall. Kita dapat mengGambarkan dua aliran pemikiran sebagai "apa yang tidak secara tegas dilarang diizinkan" (izin default) dan "apa yang tidak secara tegas diizinkan dilarang" (default menyangkal). Pengguna, yang selalu tertarik dengan fitur baru, lebih memilih yang pertama. Pakar keamanan, dengan mengandalkan pengalaman beberapa dekade, sangat menyarankan yang terakhir. Administrator yang menerapkan atau mengkonfigurasi firewall harus memilih salah satu dari dua pendekatan, meskipun administrator sering dapat memperluas kebijakan dengan mengatur parameter firewall.

3.7.2 Desain Firewall

Seperti yang telah kami jelaskan, firewall adalah perangkat sederhana yang secara ketat dan efektif mengontrol aliran data ke dan dari jaringan. Dua kualitas mengarah pada efektivitas itu: kebijakan arus lalu lintas yang dipahami dengan baik dan desain serta implementasi yang dapat dipercaya.

Kebijakan (Policy)

Firewall menerapkan kebijakan keamanan, yaitu seperangkat aturan yang menentukan lalu lintas apa yang dapat atau tidak dapat melewati firewall. Seperti banyak masalah dalam keamanan komputer, idealnya kami menginginkan kebijakan sederhana, seperti lalu lintas "baik" dapat lewat tetapi lalu lintas "buruk" diblokir. Sayangnya, mendefinisikan "baik" dan "buruk" tidak sederhana atau algoritmik. Firewall datang dengan contoh kebijakan, tetapi setiap administrator jaringan perlu menentukan lalu lintas apa yang diizinkan masuk ke jaringan tertentu.

Contoh konfigurasi firewall sederhana ditunjukkan pada Tabel 3-5. Tabel diproses dari atas ke bawah, dan aturan pencocokan pertama menentukan tindakan firewall. Karakter * cocok dengan nilai apa pun di bidang itu. Kebijakan ini menyatakan lalu lintas masuk apa pun ke port 25 (transfer email)

atau port 69 (yang disebut transfer file sepele) diizinkan ke atau dari host mana pun di subjaringan 192.168.1. Dengan aturan 3, semua host di dalam diizinkan lalu lintas keluar di mana saja pada port 80 (pengambilan halaman web). Selanjutnya, dengan aturan 4 lalu lintas luar ke host internal di alamat tujuan 192.168.1.18 (mungkin server web) diperbolehkan. Semua lalu lintas lain ke jaringan 192.168.1 ditolak.

Tabel 3-5 Contoh Konfigurasi Firewall

Rule	Type	Source Address	Destination Address	Destination Port	Action
1	TCP	*	192.168.1.*	25	Permit
2	UDP	*	192.168.1.*	69	Permit
3	TCP	192.168.1.*	*	80	Permit
4	TCP	*	192.168.1.18	80	Permit
5	TCP	*	192.168.1.*	*	Deny
6	UDP	*	192.168.1.*	*	Deny

Kepercayaan

Firewall adalah contoh monitor referensi, konsep keamanan komputer yang mendasar. Ingat dari bahwa monitor referensi memiliki tiga karakteristik:

- selalu dipanggil
- bukti kerusakan
- kecil dan cukup sederhana untuk analisis yang ketat

Firewall adalah bentuk khusus dari monitor referensi. Dengan memposisikan firewall secara hati-hati dalam arsitektur jaringan, kita dapat memastikan bahwa semua akses jaringan yang ingin kita kendalikan harus melewati firewall. Firewall diposisikan sebagai koneksi fisik tunggal antara jaringan yang dilindungi (internal) dan yang tidak terkendali (eksternal). Penempatan ini memastikan kondisi "selalu dipanggil".

Firewall biasanya terisolasi dengan baik, membuatnya sangat kebal terhadap modifikasi. Biasanya firewall diimplementasikan pada komputer yang terpisah, dengan koneksi langsung hanya ke jaringan luar dan dalam. Isolasi ini diharapkan memenuhi persyaratan "tamperproof". Lebih jauh lagi, platform firewall menjalankan sistem operasi sederhana yang menjalankan layanan minimal yang memungkinkan kompromi sistem operasi atau aplikasi firewall. Misalnya, firewall mungkin menghasilkan log lalu lintas yang ditolak, tetapi mungkin tidak menginstal alat untuk melihat dan mengedit log itu; modifikasi, jika perlu, dapat dilakukan pada mesin yang berbeda di lingkungan yang dilindungi. Dengan cara ini, bahkan jika penyerang harus berkompromi dengan sistem firewall, tidak ada alat yang dapat digunakan untuk menyamarkan atau menghapus entri log yang mungkin menunjukkan insiden tersebut.

Terakhir, perancang firewall sangat menyarankan agar fungsionalitas firewall tetap sederhana. Sayangnya, seiring waktu, tuntutan pada fungsionalitas firewall telah meningkat (seperti audit lalu lintas, antarmuka pengguna grafis, bahasa untuk mengekspresikan dan menerapkan aturan kebijakan yang kompleks, dan kemampuan untuk menganalisis lalu lintas yang sangat terstruktur), sehingga sebagian besar firewall saat ini tidak dapat dianggap kecil. atau sederhana. Namun demikian, produsen firewall telah bertahan dari sebagian besar upaya pemasaran untuk menambahkan fungsionalitas yang tidak relevan yang efek bersihnya hanya untuk mengurangi dasar keyakinan bahwa firewall beroperasi seperti yang diharapkan.

3.7.2 Jenis Firewall

Firewall memiliki berbagai kemampuan, tetapi secara umum, firewall termasuk dalam salah satu dari sejumlah kecil jenis. Setiap jenis melakukan hal yang berbeda; tidak ada satu jenis yang selalu benar atau lebih baik dan yang lainnya salah. Di bagian ini, pertama-tama kita memotivasi perlunya berbagai jenis firewall dan kemudian memeriksa setiap jenis untuk melihat apa itu, cara kerjanya, dan apa kekuatan dan kelemahannya. Berbagai jenis firewall menerapkan berbagai jenis kebijakan; misalnya, firewall sederhana yang disebut router penyaringan menilai hanya berdasarkan data header: alamat. Firewall yang lebih kompleks melihat konten yang dikomunikasikan untuk membuat keputusan akses. Kesederhanaan dalam kebijakan keamanan bukanlah hal yang buruk; pertanyaan penting untuk ditanyakan ketika memilih jenis firewall adalah ancaman apa yang perlu dilawan oleh instalasi.

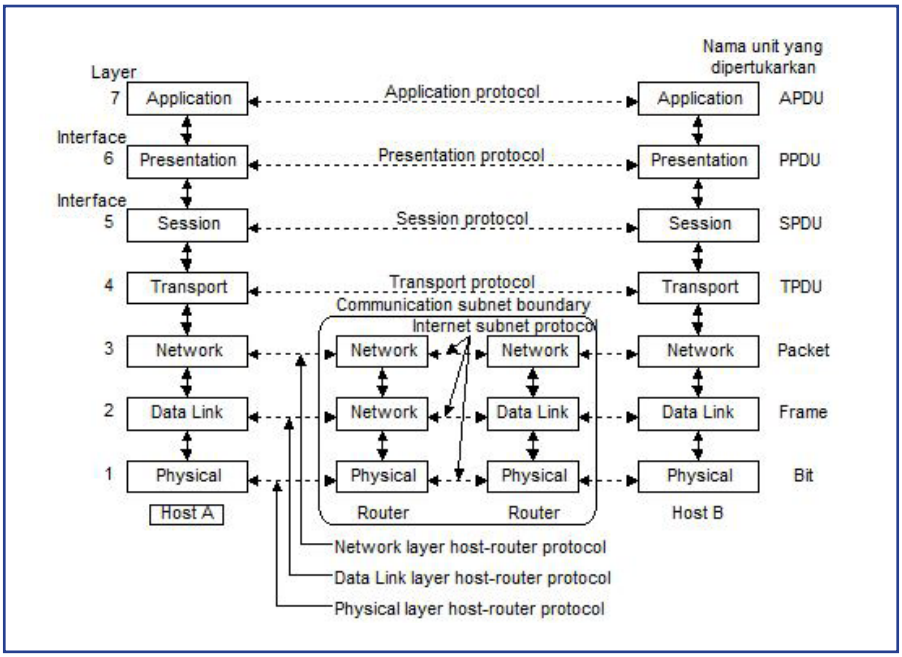
Karena firewall adalah jenis host, sering kali dapat diprogram sebagai workstation berkualitas baik. Sementara router penyaringan bisa sangat primitif, kecenderungannya adalah untuk menerapkan bahkan router pada komputer lengkap dengan sistem operasi karena editor dan alat pemrograman lainnya membantu dalam mengonfigurasi dan memelihara router. Namun, pengembang firewall adalah minimalis: Mereka mencoba menghilangkan dari firewall semua yang tidak sepenuhnya diperlukan untuk fungsionalitas firewall. Ada alasan bagus untuk batasan minimal ini: untuk memberikan bantuan sesedikit mungkin kepada penyerang yang berhasil. Jadi, firewall cenderung tidak memiliki akun pengguna sehingga, misalnya, mereka tidak memiliki file kata sandi untuk disembunyikan. Memang, firewall yang paling diinginkan adalah firewall yang berjalan dengan nyaman di ruang belakang; kecuali untuk pemindaian berkala atas log auditnya, jarang ada alasan untuk menyentuhnya.

Latar Belakang Teknologi Jaringan

Sebelum kami menjelaskan firewall, kami perlu mengulangi dan memperluas sedikit teknologi jaringan yang kami perkenalkan di awal bab ini. Gambar 3-53 menggambarkan apa yang dikenal sebagai model jaringan ISO Open Systems Interconnect (OSI).

Dalam model ini, data dibangkitkan pada lapisan atas (7—Aplikasi) oleh beberapa program aplikasi. Kemudian data melewati enam lapisan lainnya; pada setiap lapisan data diformat ulang, dikemas, dan ditangani. Misalnya, lapisan transport melakukan pengecekan dan koreksi kesalahan untuk memastikan aliran data yang andal, lapisan jaringan menangani pengalamatan untuk menentukan cara merutekan data, dan lapisan tautan data membagi data menjadi blok-blok yang dapat dikelola untuk transfer yang efisien. Lapisan terakhir, lapisan fisik, berkaitan dengan teknologi listrik atau lainnya dimana sinyal ditransmisikan melalui beberapa media fisik. Di tempat tujuan, data masuk di

bagian bawah tumpukan yang sama dan berjalan ke atas melalui lapisan, di mana detail pengalamatan dihapus dan item dikemas ulang dan diformat ulang. Akhirnya, mereka dikirim ke aplikasi di sisi tujuan. Setiap lapisan memainkan peran yang jelas dalam komunikasi. Arsitektur ini lebih konseptual daripada yang sebenarnya, tetapi memfasilitasi diskusi tentang fungsi jaringan.



Gambar 3-53 Model Referensi OSI

Jenis firewall yang berbeda sesuai dengan ancaman yang berbeda. Pertimbangkan contoh pemindaian port yang digunakan untuk memulai bab ini. Misalkan Anda mengidentifikasi penyerang yang memeriksa sistem Anda beberapa kali. Bahkan jika Anda memutuskan pertahanan Anda kuat, Anda mungkin ingin memblokir semua lalu lintas luar—bukan hanya pemindaian port—dari alamat penyerang. Dengan begitu, bahkan jika penyerang mengetahui kerentanan di sistem Anda, Anda akan mencegah serangan berikutnya dari alamat yang sama. Tapi itu hanya menangani satu penyerang pada satu waktu.

Sekarang pertimbangkan bagaimana pemindaian port beroperasi. Pemindai mengirimkan probe terlebih dahulu ke port 1, kemudian ke port 2, 3, 4, dan seterusnya. Port ini mewakili layanan, beberapa di antaranya Anda harus tetap hidup agar klien eksternal dapat mengaksesnya. Tetapi tidak ada klien eksternal normal yang perlu mencoba menyambung ke semua port Anda. Jadi, Anda dapat mendeteksi dan memblokir probe dari sumber mana pun yang tampaknya mencoba menyelidiki jaringan Anda. Bahkan jika urutan probe bukan 1-2-3-4 (pemindai mungkin mengacak urutan probe untuk membuat deteksi lebih sulit), menerima beberapa upaya koneksi ke port yang tidak biasa dari sumber yang sama mungkin merupakan sesuatu yang harus dihentikan setelah Anda melihat cukup banyak penyelidikan untuk mengidentifikasi serangan itu. Untuk itu, firewall Anda perlu merekam dan menghubungkan probe koneksi individual.

Serangan jaringan yang berbeda mungkin menargetkan aplikasi tertentu. Misalnya, cacat mungkin diketahui tentang versi xy dari server web merek z, yang melibatkan aliran data dari string karakter tertentu. Firewall Anda dapat mencari string karakter yang diarahkan ke port server web. Jenis

serangan yang berbeda ini dan cara yang berbeda untuk mendeteksinya menyebabkan beberapa jenis firewall.

Jenis firewall termasuk:

- gateway penyaringan paket atau router penyaringan
- firewall inspeksi stateful
- gateway tingkat aplikasi, juga dikenal sebagai proxy
- circuit-level gateways
- penjaga
- firewall pribadi

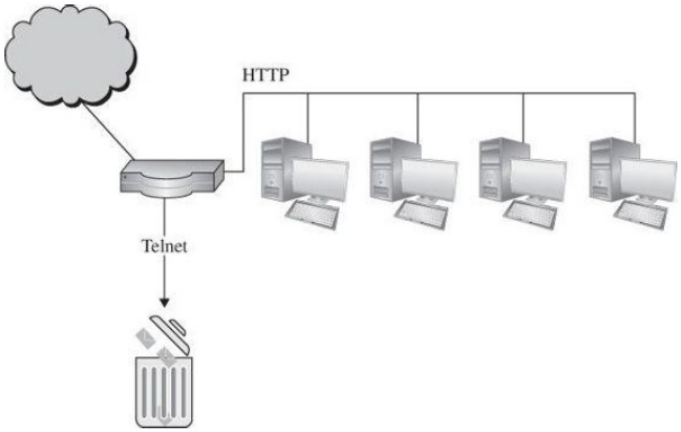
Kami menjelaskan jenis ini di bagian berikut.

Gerbang Penyaringan Paket (Packet Filtering Gateway)

Gateway penyaringan paket atau router penyaringan adalah yang paling sederhana, dan dalam beberapa situasi, jenis firewall yang paling efektif. Sebuah gateway penyaringan paket mengontrol akses berdasarkan alamat paket (sumber atau tujuan) atau jenis protokol transport tertentu (seperti lalu lintas web HTTP), yaitu dengan memeriksa informasi kontrol dari setiap paket tunggal. Firewall dapat menyaring lalu lintas sebelum sampai ke jaringan yang dilindungi. Jadi, jika pemindaian port berasal dari alamat 100.200.3.4, Anda dapat mengonfigurasi firewall gateway penyaringan paket untuk membuang semua paket dari alamat tersebut. Gambar 3-53 menunjukkan filter paket yang memblokir akses dari (atau ke) alamat dalam satu jaringan; filter memungkinkan lalu lintas HTTP tetapi memblokir lalu lintas dengan menggunakan protokol Telnet. Filter paket beroperasi pada OSI level 3.

Filter paket tidak “melihat ke dalam” sebuah paket; mereka memblokir atau menerima paket hanya berdasarkan alamat IP dan port. Dengan demikian, setiap detail di bidang data paket (misalnya, mengizinkan perintah Telnet tertentu saat memblokir layanan lain) berada di luar kemampuan filter paket.

Filter paket dapat melakukan layanan penting untuk memastikan validitas alamat di dalam. Inang dalam biasanya memercayai inang dalam lainnya justru karena mereka bukan orang luar: Di luar tidak terkendali dan penuh dengan makhluk berbahaya. Tetapi satu-satunya cara host dalam dapat mengenali host dalam lainnya adalah dengan alamat yang ditampilkan di bidang sumber pesan. Alamat sumber dalam paket dapat dipalsukan, sehingga aplikasi di dalam mungkin mengira sedang berkomunikasi dengan host lain di dalam alih-alih pemalsu luar. Filter paket berada di antara jaringan dalam dan jaringan luar, sehingga dapat menentukan apakah sebuah paket dari luar memalsukan alamat dalam, seperti yang ditunjukkan pada Gambar 3-54.



Gambar 3-54 Packet Filter

Ketika kami mengatakan filter "duduk di antara" dua jaringan, kami benar-benar bermaksud menghubungkannya ke jaringan di dalam dan luar, dengan dua kartu antarmuka yang terpisah. Filter paket dapat dengan mudah membedakan lalu lintas di dalam dari luar berdasarkan antarmuka mana paket tiba.

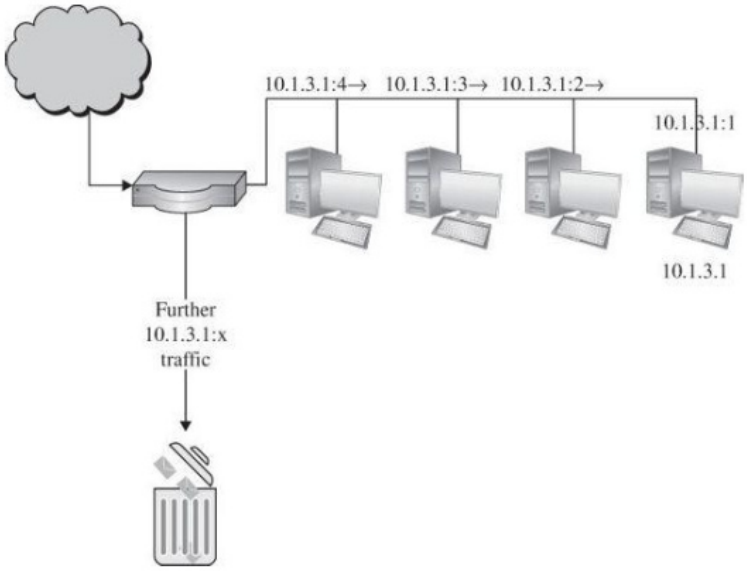
Filter paket penyaringan mungkin dikonfigurasi untuk memblokir semua paket dari luar yang mengklaim alamat sumbernya adalah alamat dalam. Dalam contoh ini, filter paket memblokir semua paket yang mengklaim berasal dari alamat mana pun dalam bentuk 100.50.25.x (tetapi, tentu saja, mengizinkan paket apa pun dengan tujuan 100.50.25.x). Filter paket menerima atau menolak hanya berdasarkan informasi header—alamat, ukuran, jenis protokol—dari setiap paket dengan sendirinya. Pemrosesan seperti itu sederhana, efisien, dan cepat, sehingga firewall pemfilteran paket sering kali berfungsi sebagai penjaga pintu yang kokoh untuk dengan cepat menghilangkan lalu lintas yang jelas tidak diinginkan.

Kerugian utama dari router packet filtering adalah kombinasi dari kesederhanaan dan kompleksitas. Inspeksi router sederhana; untuk melakukan pemfilteran yang canggih, aturan yang ditetapkan harus sangat detail. Sebuah set aturan rinci akan menjadi kompleks dan karena itu rentan terhadap kesalahan. Misalnya, memblokir semua lalu lintas port 23 (Telnet) sederhana dan mudah. Tetapi jika beberapa lalu lintas Telnet diizinkan, setiap alamat IP yang diizinkan harus ditentukan dalam aturan; dengan cara ini, set aturan bisa menjadi sangat panjang.
Firewall Inspeksi Stateful

Filtering firewall bekerja pada paket satu per satu, menerima atau menolak setiap paket dan pindah ke paket berikutnya. Mereka tidak memiliki konsep "negara" atau "konteks" dari satu paket ke paket berikutnya. Firewall inspeksi stateful mempertahankan informasi status dari satu paket ke paket lainnya di aliran input.

Ingat deskripsi mengamati probe terhadap port 1, 2, 3, 4, dan seterusnya; aktivitas itu adalah contoh penggunaan firewall inspeksi stateful. Dengan sendirinya, pemeriksaan terhadap port 1 tidak ada artinya: Ini kemungkinan besar merupakan upaya sah untuk terhubung ke layanan port 1 atau kesalahan tunggal, tetapi dapat juga menandakan dimulainya serangan pemindaian port. Firewall mencatat bahwa alamat 100.200.3.4 mengirim paket koneksi ke port 1 pada 01:37.26.

Ketika pemeriksaan terhadap port 2 tiba, firewall dapat merekam koneksi kedua dari 100.200.3.4, pada 01:37.29. Setelah dua koneksi lagi pada 01:37.34 dan 01:37.36, koneksi berikutnya pada 01:37.39 memenuhi aturan firewall untuk jumlah port yang berbeda dalam waktu singkat, sehingga mengaktifkan aturan untuk memblokir koneksi dari 100.200.3.4, seperti yang ditunjukkan pada Gambar 3-55. Firewall berkembang melalui beberapa status (jumlah permintaan koneksi dari alamat 100.200.3.4) dari paket yang berbeda hingga jumlah tersebut melebihi ambang batas untuk perilaku yang dapat diterima. Nama inspeksi stateful mengacu pada akumulasi bukti ancaman di beberapa paket.



Gambar 3-55 Inspeksi Stateful Memblokir Beberapa Probe

Salah satu pendekatan klasik yang digunakan oleh penyerang adalah memecah serangan menjadi beberapa paket dengan memaksa beberapa paket memiliki panjang yang sangat pendek sehingga firewall tidak dapat mendeteksi karakteristik serangan yang terbagi menjadi dua atau lebih paket. Firewall inspeksi stateful akan melacak urutan paket dan kondisi dari satu paket ke paket lainnya untuk menggagalkan serangan semacam itu.

Aplikasi Proxy

Filter paket hanya melihat header paket, bukan data di dalam paket. Oleh karena itu, filter paket akan meneruskan apa pun ke port 25, dengan asumsi aturan penyaringannya mengizinkan koneksi masuk ke port itu. Tetapi aplikasi itu kompleks dan terkadang mengandung kesalahan. Lebih buruk lagi, aplikasi (seperti agen pengiriman email) sering bertindak atas nama semua pengguna, sehingga mereka memerlukan hak semua pengguna (misalnya, untuk menyimpan pesan email masuk sehingga pengguna di dalam dapat membacanya). Aplikasi yang cacat, berjalan dengan hak semua pengguna, dapat menyebabkan banyak kerusakan.

Gateway proxy aplikasi, juga disebut bastion host, adalah firewall yang mensimulasikan efek (tepat) aplikasi pada level 7 sehingga aplikasi hanya menerima permintaan untuk bertindak dengan benar. Gateway proxy adalah perangkat berkepala dua: Dari dalam, gateway tampak seperti koneksi luar (tujuan), sedangkan untuk pihak luar, host proxy merespons seperti yang dilakukan oleh orang dalam.

Aplikasi Proxy menjalankan aplikasi semu. Misalnya, ketika surat elektronik ditransfer ke suatu lokasi, proses pengiriman di satu situs dan proses penerimaan di tujuan berkomunikasi dengan protokol yang menetapkan legitimasi transfer surat dan kemudian benar-benar melewati pesan surat. Protokol antara pengirim dan tujuan didefinisikan dengan hati-hati. Sebuah gateway proxy pada dasarnya mengganggu di tengah pertukaran protokol ini, tampak seperti tujuan dalam komunikasi dengan pengirim yang berada di luar firewall, dan tampak seperti pengirim dalam komunikasi dengan penerima sebenarnya di dalam. Proxy di tengah memiliki kesempatan untuk menyaring transfer email, memastikan bahwa hanya perintah dan konten protokol email yang dapat diterima yang dikirim ke kedua arah. (Biasanya firewall fokus untuk melindungi penerima orang dalam dari konten berbahaya yang dikirim dari luar.)

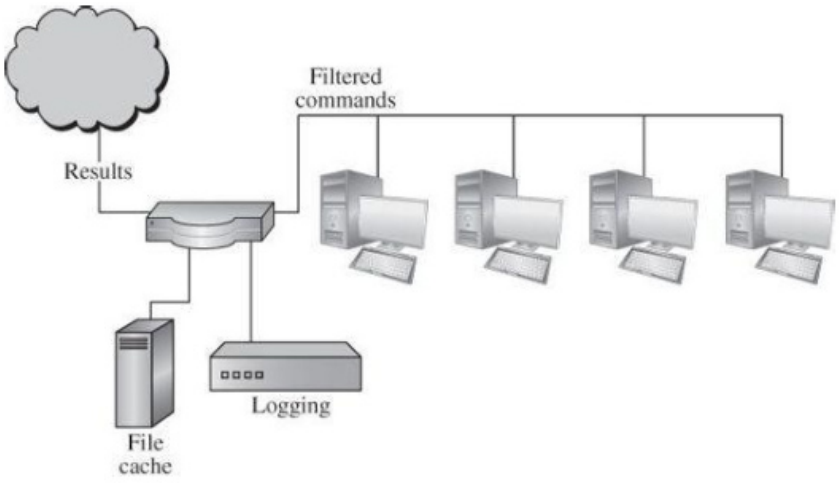
Sebagai contoh aplikasi proxy, pertimbangkan protokol FTP (transfer file). Perintah protokol khusus mengambil (mendapatkan) file dari lokasi yang jauh, menyimpan (menempatkan) file ke host jarak jauh, membuat daftar file (ls) dalam direktori pada host jarak jauh, dan memposisikan proses (cd) pada titik tertentu dalam direktori pohon pada host jarak jauh. Perintah protokol FTP sebenarnya adalah bagian dari perintah yang dapat dijalankan pengguna dari workstation untuk memanipulasi file. Beberapa administrator mungkin ingin mengizinkan mendapat tetapi memblokir penempatan, dan hanya mencantumkan file tertentu atau melarang perubahan dari direktori tertentu (sehingga orang luar hanya dapat mengambil file dari direktori yang telah ditentukan sebelumnya). Proxy akan mensimulasikan kedua sisi pertukaran protokol ini. Misalnya, dalam satu contoh, proxy mungkin menerima perintah get tetapi menolak perintah put. Dalam situasi lain, proxy dapat memfilter respons lokal terhadap permintaan untuk membuat daftar file sehingga hanya mengungkapkan sebagian file yang ingin diekspos oleh administrator dalam kepada orang luar.

Untuk memahami tujuan sebenarnya dari gateway proxy, mari kita pertimbangkan beberapa contoh antara lain :

- Sebuah perusahaan ingin membuat daftar harga online sehingga orang luar dapat melihat produk dan harga yang ditawarkan. Ia ingin memastikan bahwa (a) tidak ada orang luar yang dapat mengubah harga atau daftar produk dan (b) orang luar hanya dapat mengakses daftar harga, bukan file yang lebih sensitif yang disimpan di dalamnya.
- Sebuah sekolah ingin mengizinkan siswanya untuk mengambil informasi apapun dari sumber daya World Wide Web di Internet. Untuk membantu memberikan layanan yang efisien, sekolah ingin mengetahui situs apa yang telah dikunjungi dan file apa dari situs tersebut yang telah diambil; file yang sangat populer akan di-cache secara lokal.
- Instansi pemerintah ingin menjawab pertanyaan melalui sistem manajemen basis data. Namun, agensi ingin menyaring hasil sehingga tidak ada nama atau identifikasi yang dikembalikan dalam hasil—hanya dihitung dalam kategori.
- Sebuah perusahaan dengan beberapa kantor ingin mengenkripsi bagian data dari semua email ke alamat di kantor lainnya. (Proxy yang sesuai di ujung jarak jauh akan menghapus enkripsi.)

Masing-masing persyaratan ini dapat dipenuhi dengan proxy. Dalam kasus pertama, proxy akan memantau data protokol transfer file untuk memastikan bahwa hanya file daftar harga yang diakses dan file tersebut hanya dapat dibaca, tidak dimodifikasi. Persyaratan sekolah dapat dipenuhi dengan prosedur logging sebagai bagian dari web browser. Kebutuhan agensi dapat dipenuhi oleh proxy tujuan khusus yang berinteraksi dengan sistem manajemen basis data, melakukan kueri tetapi

menyaring output. Aplikasi firewall dapat mengenkripsi dan mendekripsi pesan email tertentu untuk situasi terakhir. Fungsi-fungsi ini ditunjukkan pada Gambar
3-56.



Gambar 3-56 Fungsi Firewall Proxy

Proxy pada firewall dapat disesuaikan dengan kebutuhan spesifik, seperti detail logging tentang akses. Mereka bahkan dapat menghadirkan antarmuka pengguna yang umum untuk fungsi internal yang mungkin berbeda. Misalkan jaringan internal memiliki campuran jenis sistem operasi, tidak ada yang mendukung otentikasi kuat melalui token tantangan-tanggapan. Proxy dapat meminta autentikasi yang kuat (nama, sandi, dan respons tantangan), memvalidasi respons tantangan itu sendiri, dan kemudian hanya meneruskan detail autentikasi nama dan sandi sederhana dalam bentuk yang diperlukan oleh sistem operasi host internal tertentu.

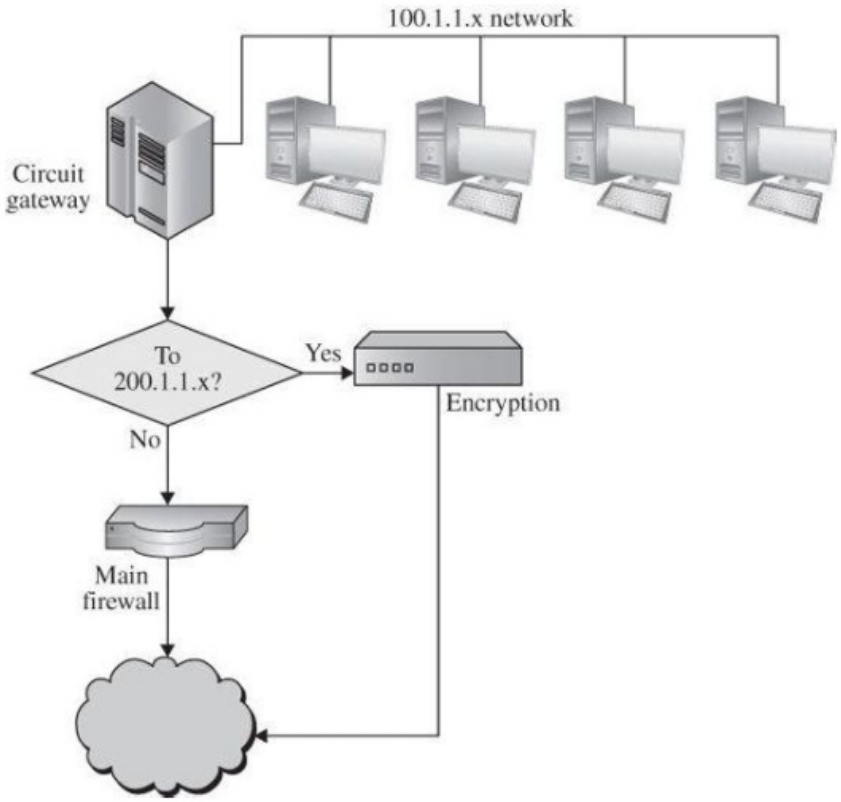
Perbedaan antara proxy dan router penyaringan adalah bahwa proxy menafsirkan aliran protokol sebagai aplikasi, untuk mengontrol tindakan melalui firewall berdasarkan hal-hal yang terlihat dalam protokol, bukan hanya pada data header eksternal.

Circuit-Level Gateway

Circuit-Level Gateway (Gateway tingkat sirkuit) adalah firewall yang pada dasarnya memungkinkan satu jaringan menjadi perpanjangan dari yang lain. Ini beroperasi pada OSI level 5, level sesi, dan berfungsi sebagai gateway virtual antara dua jaringan. Sirkuit adalah koneksi logis yang dipertahankan untuk jangka waktu tertentu, kemudian dirobohkan atau diputuskan. Firewall memverifikasi sirkuit saat pertama kali dibuat. Setelah rangkaian diverifikasi, data selanjutnya yang ditransfer melalui rangkaian tidak diperiksa. Gateway tingkat sirkuit dapat membatasi koneksi mana yang dapat dibuat melalui gateway.

Salah satu penggunaan gateway tingkat sirkuit adalah untuk mengimplementasikan jaringan pribadi virtual, yang dijelaskan sebelumnya dalam bab ini. Misalkan sebuah perusahaan memiliki dua kantor, masing-masing dengan jaringannya sendiri, di alamat 100.1.1.x dan 200.1.1.x. Selanjutnya, perusahaan ingin memastikan bahwa komunikasi antara dua ruang alamat ini bersifat pribadi, sehingga administrator jaringan memasang sepasang perangkat enkripsi. Gateway tingkat sirkuit memisahkan semua lalu lintas ke dan dari jaringan 100 dan 200, seperti yang ditunjukkan pada

Gambar 3-57. (Angka ini hanya menunjukkan jaringan 100; struktur paralel ada pada jaringan 200.) Gerbang sirkuit pada jaringan 100 mengarahkan semua lalu lintas ke jaringan 200 melalui perangkat enkripsi. Saat lalu lintas kembali, firewall pada subjaringan 100 merutekan semua lalu lintas dari jaringan 200 melalui unit enkripsi (untuk dekripsi) dan kembali ke gateway 100. Dengan cara ini, arus lalu lintas antara 100 dan 200 jaringan secara otomatis disaring (sehingga tidak ada lalu lintas lain yang dapat menyamar sebagai bagian dari pasangan jaringan yang dilindungi ini), dan dienkripsi untuk kerahasiaan. Pengguna tidak mengetahui kriptografi dan manajemen terjamin akan perlindungan kerahasiaan.



Gambar 3-57 Circuit-Level Gateway

Guard

Guard (Penjaga) adalah firewall yang canggih. Seperti firewall proxy, ia menerima unit data protokol, menafsirkannya, dan memancarkan unit data protokol yang sama atau berbeda yang mencapai hasil yang sama atau hasil yang dimodifikasi. Penjaga menentukan layanan apa yang akan dilakukan atas nama pengguna sesuai dengan informasi yang tersedia, seperti apa pun yang dapat dipastikan dengan andal tentang identitas pengguna (di luar), interaksi sebelumnya, dan sebagainya. Tingkat kontrol yang dapat diberikan penjaga hanya dibatasi oleh apa yang dapat dihitung. Tetapi penjaga dan firewall proxy cukup mirip sehingga perbedaan di antara mereka terkadang tidak jelas. Artinya, kita dapat menambahkan fungsionalitas ke firewall proxy hingga mulai terlihat seperti penjaga.

Kegiatan jaga bisa sangat rinci, seperti yang diilustrasikan dalam contoh berikut:

- Sebuah universitas ingin mengizinkan mahasiswanya menggunakan email hingga batas pesan atau karakter email yang begitu banyak dalam beberapa hari terakhir. Meskipun hasil ini dapat dicapai dengan memodifikasi penangan email, hasil ini lebih mudah dilakukan dengan memantau titik umum yang dilalui semua aliran email, protokol transfer email.
- Sebuah sekolah ingin siswanya dapat mengakses World Wide Web tetapi, karena kapasitas koneksinya ke web, sekolah hanya mengizinkan begitu banyak byte per detik (yaitu, memungkinkan mode teks dan grafik sederhana tetapi tidak mengizinkan kompleks grafis, video, musik, atau sejenisnya).
- Sebuah perpustakaan ingin menyediakan dokumen-dokumen tertentu, tetapi untuk mendukung penggunaan yang adil dari materi berhak cipta, itu akan memungkinkan pengguna untuk mengambil hanya begitu banyak karakter pertama dari sebuah dokumen. Setelah jumlah tersebut, perpustakaan akan meminta pengguna untuk membayar biaya yang akan diteruskan kepada penulis.
- Sebuah perusahaan sedang mengembangkan produk baru berdasarkan minyak bumi dan gas helium, dengan kode nama "minyak ringan". Dalam aliran data keluar apa pun, seperti transfer file, email, halaman web, atau aliran data lainnya, kata "petroleum", "helium", atau "light oil" akan diganti dengan "magic". Firewall dianggap terutama sebagai filter masuk: hanya mengizinkan lalu lintas yang sesuai (yang sesuai dengan kebijakan keamanan firewall). Contoh ini menunjukkan bahwa firewall atau penjaga dapat dengan mudah menyaring lalu lintas keluar.
- Perusahaan ingin mengizinkan karyawannya mengambil file melalui FTP. Namun, untuk mencegah masuknya virus, pertama-tama akan melewati semua file yang masuk melalui pemindai virus. Meskipun banyak dari file-file ini akan menjadi teks atau grafik yang tidak dapat dieksekusi, administrator perusahaan berpikir bahwa biaya pemindaian mereka (file mana yang akan lulus) akan diabaikan.

Masing-masing skenario ini dapat diimplementasikan sebagai proxy yang dimodifikasi. Karena keputusan proxy didasarkan pada beberapa kualitas data komunikasi, kami menyebut proxy sebagai penjaga. Karena kebijakan keamanan yang diterapkan oleh penjaga agak lebih kompleks daripada tindakan proxy, kode penjaga juga lebih kompleks dan karenanya lebih rentan terhadap kesalahan. Firewall yang lebih sederhana memiliki lebih sedikit kemungkinan cara untuk gagal atau ditumbangkan. Contoh proses penjagaan adalah yang disebut Tembok Api Besar China, yang dijelaskan di Studi Kasus 3-23.

Kami sengaja mengatur jenis firewall ini dari yang sederhana hingga yang kompleks. Penyaringan sederhana sangat mekanistik dan algoritmik, menyiratkan bahwa kode untuk mengimplementasikannya teratur dan mudah. Penentuan konten yang kompleks semakin mendekati kecerdasan mesin, aktivitas yang lebih heuristik dan bervariasi. Analisis yang lebih kompleks membutuhkan lebih banyak waktu, yang dapat memengaruhi kinerja dan kegunaan firewall. Tidak ada pendekatan firewall tunggal yang benar atau lebih baik; masing-masing memiliki konteks penggunaannya yang sesuai.

Penguasa di Republik Rakyat Tiongkok ingin mengontrol data yang dapat diakses oleh penduduknya. Perusahaan konten seperti Google dan Yahoo/Microsoft telah diberitahu bahwa jika mereka ingin berbisnis di China, mereka perlu menggunakan versi khusus dari aplikasi web mereka yang menyaring "kata-kata yang menyinggung". Ketika Skype ingin memasuki pasar Cina, mereka juga diberitahu bahwa mereka harus menghapus pesan teks; hasilnya: Teks Skype sekarang menghilangkan kata-kata seperti “Falun Gong” dan “Dalai Lama.”

Bloomberg Business News melaporkan bahwa China mempekerjakan 30.000 orang untuk memantau konten di situs web dan melaporkan situs yang melanggar standar [ELG06]. Semua lalu lintas Internet melewati bank firewall yang dikendalikan pemerintah. Setiap email atau pesan teks yang berisi kata-kata terlarang dijatuhkan di firewall.

Sebagai syarat menjalankan bisnis di China, Google diminta untuk menyediakan kemampuan pencarian khusus yang tidak mengizinkan akses ke situs terlarang tertentu atau membuat konten yang tidak dapat diterima. Tiananmen adalah salah satu istilah yang sensitif, seperti 4 Juni (tanggal pemberontakan); masukkan itu ke mesin pencari dan, menurut CNN, Anda mendapatkan hasil "Menurut hukum dan peraturan yang relevan, hasilnya tidak ditampilkan." Tapi masukkan 8x8, yang tentu saja bernilai 64 atau 6/4, singkatan dari 4 Juni, dan Anda mungkin menemukan beberapa entri blog yang belum disensor [SHA11]. Facebook dan Twitter, tentu saja, disensor, tetapi orang-orang menemukan cara licik untuk menghindari sensor itu.
Setelah mematuhi pembatasan Cina selama beberapa tahun, Google secara resmi meninggalkan Cina daratan pada musim panas 2010. Awalnya, lalu lintas Google dialihkan ke server di Hong Kong, secara teknis Cina tetapi beroperasi dengan sangat bebas. Firewall Cina dan server pengalamatan mengalihkan upaya untuk menjangkau situs eksternal.

Meskipun secara teknis bukan firewall, Great Firewall of China, yang secara resmi dikenal dengan nama yang lebih menarik, Proyek Perisai Emas, tentu saja menjalankan fungsi firewall. Namun, seperti yang ditunjukkan oleh contoh yang dikutip, memfilter konten lebih sulit daripada sekadar menyaring alamat.

3.7.3 Personal Firewall

Firewall biasanya melindungi (sub) jaringan dari beberapa host. Mahasiswa dan karyawan di kantor berada di belakang firewall yang sebenarnya. Semakin banyak pengguna rumahan, pekerja individu, dan usaha kecil menggunakan modem kabel atau koneksi DSL dengan akses tak terbatas dan selalu aktif. Orang-orang ini membutuhkan firewall, tetapi komputer firewall terpisah untuk melindungi satu stasiun kerja dapat terlihat terlalu rumit dan mahal. Orang-orang ini membutuhkan kemampuan firewall dengan harga yang lebih rendah.

Firewall personal adalah program aplikasi yang berjalan pada workstation yang dilindunginya. Firewall pribadi dapat melengkapi pekerjaan firewall konvensional dengan menyaring jenis data yang akan diterima oleh satu host, atau dapat mengimbangi kurangnya firewall biasa, seperti pada DSL pribadi atau koneksi modem kabel.

Firewall personal adalah program yang berjalan pada satu host untuk memantau dan mengontrol lalu lintas ke host tersebut. Itu hanya dapat bekerja bersama dengan dukungan dari sistem operasi.

Sama seperti firewall jaringan menyaring lalu lintas masuk dan keluar untuk jaringan itu, firewall pribadi menyaring lalu lintas pada satu workstation. Workstation dapat rentan terhadap kode berbahaya atau agen aktif berbahaya (kontrol ActiveX atau applet Java), kebocoran data pribadi yang disimpan di workstation, dan pemindaian kerentanan untuk mengidentifikasi potensi kelemahan. Implementasi komersial firewall pribadi termasuk SaaS Endpoint Protection dari McAfee, F-Secure Internet Security, Microsoft Windows Firewall, dan Zone Alarm dari CheckPoint.

Firewall personal dikonfigurasi untuk menerapkan beberapa kebijakan. Misalnya, pengguna dapat memutuskan bahwa situs tertentu, seperti komputer di jaringan perusahaan, sangat dapat dipercaya, tetapi sebagian besar situs lain tidak. Vendor terkadang menyediakan dan memelihara daftar situs tidak aman yang aksesnya diblokir oleh produk mereka secara default. Pengguna menentukan kebijakan yang mengizinkan pengunduhan kode, berbagi data tanpa batas, dan akses manajemen dari segmen korporat tetapi tidak dari situs lain. Firewall pribadi juga dapat menghasilkan log akses, yang dapat berguna untuk memeriksa jika sesuatu yang berbahaya tidak lolos dari firewall.

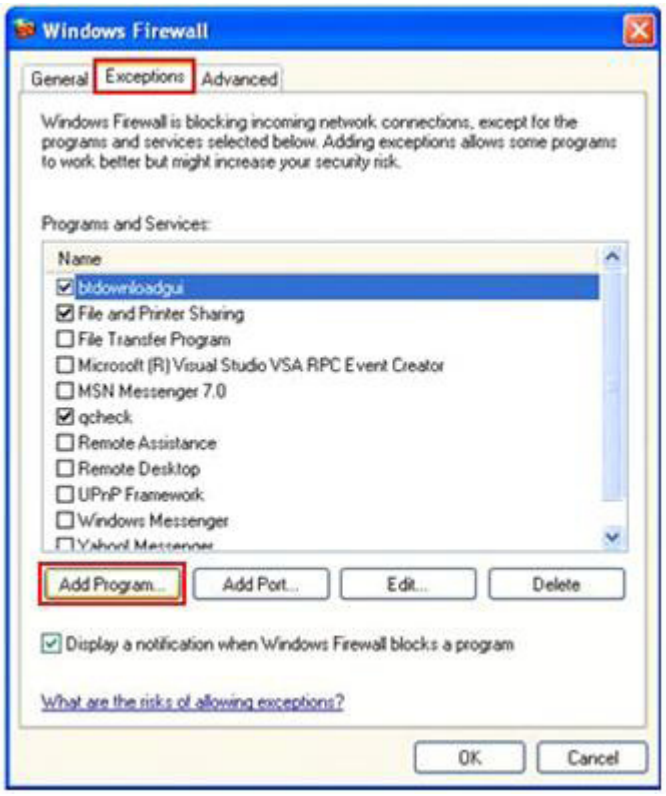
Menggabungkan pemindai malware dengan firewall personal adalah efektif dan efisien. Biasanya, pengguna lupa menjalankan pemindai secara teratur, tetapi mereka ingat untuk menjalankannya sesekali, misalnya selama seminggu. Namun, meninggalkan eksekusi pemindai ke memori pengguna berarti pemindai mendeteksi masalah hanya setelah fakta—seperti ketika virus telah diunduh dalam lampiran email. Dengan kombinasi pemindai virus dan firewall personal, firewall mengarahkan semua email masuk ke pemindai virus, yang memeriksa setiap lampiran saat mencapai host target dan sebelum dibuka.

Firewall personal berjalan di komputer yang coba dilindunginya. Dengan demikian, penyerang yang cerdik kemungkinan akan mencoba serangan yang tidak terdeteksi yang akan menonaktifkan atau mengkonfigurasi ulang firewall di masa mendatang. Seperti yang dijelaskan di Studi Kasus 3-24, pengguna dapat mengalahkan kebijakan keamanan firewall mereka sendiri. Anda telah mempelajari di Bab 4 bahwa kode yang terhubung ke sistem operasi dapat menjadi rootkit itu sendiri, sebuah potensi ancaman, sementara di sisi lain, kode tersebut dapat rentan terhadap serangan licik melalui sistem operasi oleh rootkit. Namun, terutama untuk modem kabel, DSL, dan koneksi "selalu aktif" lainnya, stasiun kerja statis adalah target yang terlihat dan rentan untuk komunitas serangan yang selalu ada. Firewall personal dapat memberikan perlindungan yang wajar kepada klien yang tidak berada di belakang firewall jaringan.

Studi Kasus 3-24 : Membuat Lubang di Firewall

Firewall memiliki manfaat keamanan yang jelas, tetapi terkadang mereka mencegah pengguna yang bermaksud baik untuk mengakses data dan fungsi yang diperlukan. Misalnya, firewall biasanya mencegah pengguna di satu sistem menggunakan File Transfer Protocol (ftp) untuk mengunggah atau mengunduh file di sistem lain. Untuk alasan ini, seseorang di dalam firewall kadang-kadang “melubangi” firewall agar orang luar yang tepercaya bisa masuk untuk sementara waktu. Lubang seperti itu sebenarnya merupakan pengecualian yang dimasukkan ke dalam aturan kebijakan firewall. Lubang-lubang ini memungkinkan file untuk dibagikan, aplikasi dapat diakses, dan banyak lagi. Secara teknis disebut backdoor SSH, lubang firewall dapat diatur dengan berbagai cara. Setelah pekerjaan orang luar selesai, orang dalam menutup lubang dan perlindungan dipulihkan.

Beberapa sistem operasi mengizinkan aturan yang sengaja melanggar firewall. Misalnya, Windows XP secara resmi mengizinkan pengguna untuk membuat lubang dengan menetapkan "pengecualian" pada layar administratif untuk firewall Windows, yang ditunjukkan pada Gambar 3-58. Pengecualian dapat membuka port atau, lebih disukai, mengaktifkan program atau layanan tertentu untuk memiliki akses di dalam firewall.



Gambar 3-58 Firewall Exception

Apa kerugian dari pelanggaran firewall semacam itu? Mereka melemahkan firewall, mungkin sampai menonaktifkannya. Pelanggaran semacam itu berisiko secara tidak sengaja membiarkan orang lain (selain lalu lintas yang pengecualiannya dibuat)

untuk masuk melalui lubang pada saat yang bersamaan. Jadi, apakah etis untuk melubangi firewall? Hanya jika benar-benar diperlukan, bersifat sementara, dan dilakukan dengan izin dari administrator sistem. Situasi seperti itu dapat muncul dalam keadaan darurat, ketika informasi atau layanan yang dilindungi diperlukan untuk mengatasi masalah yang tidak biasa. Tantangannya adalah untuk memastikan bahwa keadaan darurat tidak menjadi praktik standar dan pengecualian dihapus setelah digunakan.

Firewall hanya dapat melindungi apa yang dapat dikontrolnya, jadi jika subjaringan memiliki koneksi eksternal yang tidak disaring oleh firewall, firewall tidak dapat mengontrol lalu lintas pada koneksi yang tidak disaring tersebut. Contohnya adalah perangkat dengan koneksi Internet langsungnya sendiri (mungkin koneksi nirkabel jahat). Seperti yang kita lihat sebelumnya dalam bab ini, visibilitas ke satu perangkat, mungkin melalui koneksi nirkabel yang disebutkan di sini, dapat memberikan visibilitas dan akses penyerang ke perangkat lain. Untuk alasan ini, satu-satunya jalur ke setiap perangkat jaringan yang dilindungi harus melewati firewall jaringan.
Meskipun contoh-contoh ini adalah penyederhanaan, mereka menunjukkan jenis konfigurasi yang dilindungi firewall. Selanjutnya, kami meninjau jenis serangan yang dapat dan tidak dapat dilindungi oleh firewall.

Apa yang Dapat Diblokir—dan Tidak Dapat—Diblokir oleh Firewall

Seperti yang telah kita lihat, firewall bukanlah solusi lengkap untuk semua masalah keamanan komputer. Firewall hanya melindungi perimeter lingkungannya terhadap serangan dari pihak luar yang ingin mengeksekusi kode atau mengakses data pada mesin di lingkungan yang dilindungi. Ingatlah poin-poin ini tentang firewall.

- Firewall dapat melindungi lingkungan hanya jika firewall mengontrol seluruh perimeter. Artinya, firewall hanya efektif jika tidak ada koneksi tanpa perantara yang menembus perimeter. Jika bahkan satu host di dalam terhubung ke alamat luar, dengan koneksi nirkabel misalnya, seluruh jaringan dalam rentan melalui host yang tidak terlindungi.
- Firewall tidak melindungi data di luar perimeter; data yang telah melewati (keluar) dengan benar melalui firewall sama terbukanya dengan seolah-olah tidak ada firewall.
- Firewall adalah bagian instalasi yang paling terlihat dari luar, sehingga menjadi target serangan yang paling menarik. Untuk alasan ini, beberapa lapisan perlindungan yang berbeda, yang disebut pertahanan mendalam, lebih baik daripada mengandalkan kekuatan hanya satu firewall.
- Firewall harus dikonfigurasi dengan benar, konfigurasi tersebut harus diperbarui saat lingkungan internal dan eksternal berubah, dan laporan aktivitas firewall harus ditinjau secara berkala untuk bukti upaya penyusupan atau berhasil.
- Firewall adalah target penetrator. Sementara firewall dirancang untuk menahan serangan, itu tidak bisa ditembus. Desainer sengaja menjaga firewall kecil dan sederhana sehingga bahkan jika penetrator merusaknya, firewall tidak memiliki alat lebih lanjut, seperti compiler, linker, loader, dan sejenisnya, untuk melanjutkan serangan.
- Firewall hanya menjalankan kontrol kecil atas konten yang masuk ke dalam, yang berarti bahwa data yang tidak akurat atau kode berbahaya harus dikontrol dengan cara lain di dalam perimeter.

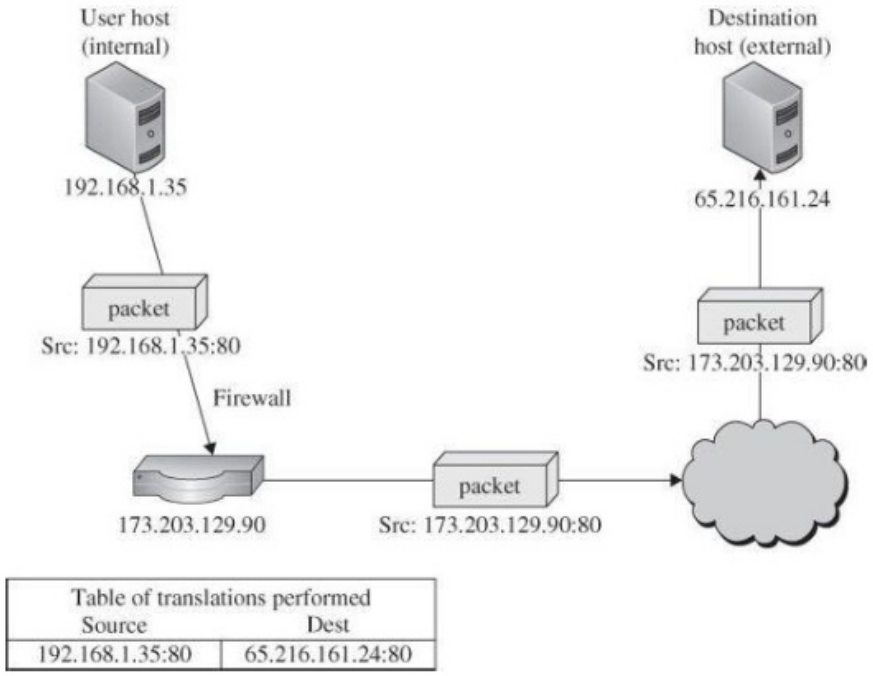
Firewall adalah alat penting dalam melindungi lingkungan yang terhubung ke jaringan. Namun, lingkungan harus dilihat secara keseluruhan, semua kemungkinan eksposur harus dipertimbangkan, dan firewall harus sesuai dengan strategi keamanan yang lebih besar dan komprehensif. Firewall saja tidak dapat mengamankan lingkungan.

3.7.4 Network Address Translation (NAT)

Firewall melindungi host internal dari aliran data masuk atau keluar yang tidak dapat diterima. Namun, seperti yang ditunjukkan sebelumnya dalam bab ini, terkadang orang luar dapat memperoleh informasi berharga hanya dengan mempelajari arsitektur, konektivitas, atau bahkan ukuran jaringan internal. Ketika host internal menyajikan alamat IP-nya kepada pihak luar (diperlukan jika pihak luar diharapkan untuk membalas), pihak luar dapat menyimpulkan beberapa struktur jaringan dari pola alamat. Selanjutnya, setelah dirilis, alamat ini akan selamanya diketahui dan dapat dieksploitasi oleh orang luar. Dengan mudah, firewall juga dapat mencegah informasi ini keluar.

Hingga saat ini, sistem pengalamatan pada komputer dalam sebuah jaringan masih mengandalkan IPv4. Yang secara matematis, jumlah dari IPv4 inilah berkisar 4,2 Miliyar IP address. Meski terlihat banyak, nyatanya ini masih diragukan untuk memenuhi kebutuhan dunia, karena total terdapat lebih dari 7 miliyar jiwa. Jelas hal ini akan sangat kurang pasokan IP Address bila kita asumsikan semua orang ingin bisa terhubung ke internet.

Demi mengatasi keterbatasan IP address inilah orang mulai mencari bagaimana caranya agar setiap orang bisa terhubung dengan jumlah IP address yang disebutkan tadi. Kemudian munculah sebuah metode pengalamatan pada jaringan yang disebut dengan Network Address Translation. Dengan hadirnya metode NAT ini, semua orang sangat mungkin terhubung ke internet.



Gambar 3-63 Network Address Translation

Satu-satunya komplikasi pada skema ini terjadi jika dua host internal menghubungi alamat tujuan yang sama melalui port yang sama, yang mungkin diharapkan jika dua host internal secara independen ingin mengakses halaman web di www.google.com, misalnya. Dalam hal ini, firewall akan menulis ulang nomor port sumber dari satu host yang meminta ke nomor acak yang berbeda sehingga firewall dapat menerjemahkan kembali dengan benar. Host internal 192.168.1.35 mungkin menjadi 173.203.129.90 port 4236, dan 192.168.1.57 mungkin menjadi 173.203.129.90 port 4966.

Dunia luar hanya melihat satu alamat eksternal, 173.203.129.90 untuk seluruh jaringan internal yang diamankan, sehingga orang luar tidak dapat menyimpulkan struktur jaringan internal. Faktanya, orang luar tidak tahu apakah satu komunikasi pada satu waktu berasal dari host internal yang sama dengan komunikasi selanjutnya, sehingga agak melindungi pengguna internal individu. Akhirnya, mengetahui alamat yang seharusnya dari orang dalam tidak akan membantu orang luar nanti: Jika orang luar membuat lalu lintas ke alamat yang sama di lain waktu, firewall akan menolak lalu lintas karena alamat pengirim tidak lagi ada di Tabel terjemahan. Meskipun terutama digunakan karena masalah lain (nomor alamat publik terbatas), terjemahan alamat jaringan melakukan peran keamanan yang signifikan.

3.7.5 Data Loss Prevention

Atau bisa disebut, apakah itu data leakage prevention? DLP adalah sekumpulan kebijakan, produk, teknologi, dan teknik yang dirancang untuk mencegah hilangnya informasi sensitif bisnis atau organisasi.

Semua strategi DLP harus menggabungkan kumpulan solusi yang memantau, mendeteksi, dan menghentikan aliran data yang tidak sah. Solusi ini termasuk mencegah pengguna menghancurkan data sensitif secara tidak sengaja serta melindungi dari pelanggaran data dari luar organisasi.

Jaringan DLP difokuskan pada pemantauan pengguna akhir dalam organisasi seperti halnya melindungi pelanggaran dari agen eksternal melalui kerentanan jaringan. Karyawan yang tidak berwenang, misalnya, tanpa sengaja dapat mengubah, menghapus, atau mengirimkan informasi yang dapat merusak reputasi perusahaan.

Cara Kerja Data Loss Prevention

Teknologi DLP menggunakan sekumpulan aturan bisnis untuk mendeteksi transfer data yang tidak normal dan mencari data sensitif yang mungkin menjadi bagian dari komunikasi elektronik pengguna akhir. Tujuan sistem DLP adalah untuk menghentikan pengiriman informasi seperti informasi intelektual, data keuangan, detail pelanggan, atau informasi karyawan, baik secara tidak sengaja atau sengaja, keluar dari jaringan perusahaan.

Misalnya, jika seorang karyawan mencoba mengirim email dengan lampiran spreadsheet yang menyertakan informasi perusahaan yang sensitif, izin mereka akan ditolak oleh sistem DLP terkait, dalam hal ini, sistem pencegahan ekstrusi. Penolakan ini dapat terjadi meskipun karyawan biasanya diizinkan mengakses spreadsheet, tetapi kebijakan melarangnya dibagikan di luar jaringan organisasi.

Solusi DLP mengalami peningkatan popularitas setelah skandal WikiLeaks Bradley/Chelsea Manning, di mana seorang anggota militer AS membocorkan sejumlah informasi rahasia ke situs web WikiLeaks, dan skandal Edward Snowden, di mana kontraktor NSA membocorkannya. sejumlah besar dokumen rahasia ke berbagai organisasi berita besar.

DLP dapat diimplementasikan dalam beberapa cara: Sistem berbasis agen mungkin diinstal sebagai rootkit OS yang memantau perilaku pengguna, termasuk koneksi jaringan, akses file, dan aplikasi yang dijalankan. Solusi berbasis jaringan memantau koneksi jaringan, terutama transfer file. Solusi lain mungkin khusus untuk aplikasi, seperti agen perangkat lunak untuk memantau email. Solusi DLP umumnya akan mencari berbagai indikator:

- Kata kunci. Kata atau frasa tertentu, seperti “rahasia”, “rahasia”, atau “hak milik”, adalah indikator kuat dari data sensitif. Solusi DLP juga memungkinkan pelanggan untuk mencari kata kunci yang memiliki arti khusus untuk bisnis tertentu, seperti nama kode untuk produk baru.
- Pola lalu lintas. Beberapa pola lalu lintas yang mungkin menunjukkan perilaku mencurigakan adalah transfer file massal, koneksi ke email luar atau layanan berbagi file, email ke penerima yang tidak dikenal, dan koneksi ke layanan jaringan yang tidak dikenal.
- Pengkodean/enkripsi. DLP dapat dengan mudah dikalahkan dengan enkripsi yang kuat, karena tidak ada solusi DLP yang dapat menentukan sensitivitas file yang tidak dapat dibaca. Untuk mengatasi masalah ini, solusi DLP biasanya memblokir file keluar yang tidak dapat mereka dekode atau dekripsi. Banyak pemindai malware memperlakukan file masuk, seperti lampiran email terenkripsi, dengan cara yang sama.

Meskipun solusi DLP berguna untuk mencegah kebocoran data yang tidak disengaja, solusi tersebut lebih rapuh daripada solusi keamanan lain yang kita bahas dalam buku ini. Penyerang yang gigih sering kali dapat menemukan cara untuk mentransfer data ke dalam sistem, meskipun solusi DLP yang efektif dapat memperlambat proses atau memperingatkan personel keamanan tepat waktu untuk mencegahnya.

Sistem DLP memberikan transisi yang baik ke topik berikutnya. Firewall kadang-kadang disebut perangkat tepi, artinya mereka diposisikan pada batas subnetwork. Pendekatan DLP dapat diintegrasikan ke dalam firewall, dipasang di sistem operasi, atau digabungkan ke program aplikasi lain yang memanipulasi tanggal sensitif. Dengan demikian, teknologi DLP tidak terbatas pada tepi subjaringan yang dilindungi. Selanjutnya kami mempelajari sistem deteksi dan perlindungan intrusi, memantau produk yang juga ditempatkan di dalam subnetwork.

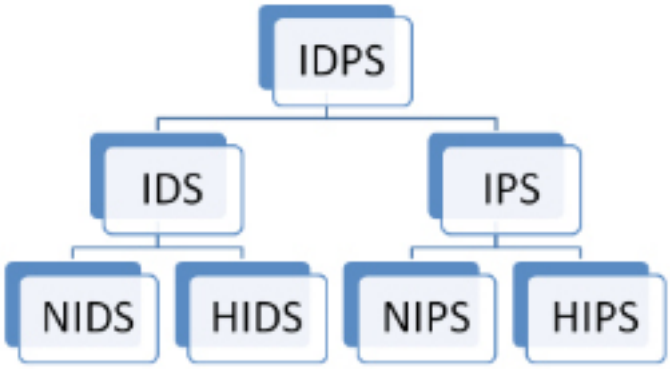
3.8 Intrusion Detection System

3.8.1 Pemahaman

Intrution Detection System atau IDS adalah perangkat (atau aplikasi) yang memonitor jaringan dan / atau sistem untuk kegiatan berbahaya atau pelanggaran kebijakan dan memberikan laporan ke administrator atau station manajemen jaringan. Pencegahan intrusi / pemyusupkan adalah proses melakukan deteksi intrusi dan mencoba untuk menghentikan insiden yang mungkin terdeteksi.

Intrusion Detection and Prevention System (IDPS) atau Sistem pendeteksi intrusi dan pencegahan terutama difokuskan pada identifikasi kemungkinan insiden, mencatat informasi tentang insiden tersebut, mencoba untuk menghentikan mereka, dan melaporkan mereka ke administrator keamanan. Selain itu, organisasi dapat menggunakan IDPS untuk keperluan lain, seperti mengidentifikasi masalah dengan kebijakan keamanan, mendokumentasikan ancaman yang ada, dan menghalangi orang dari melanggar kebijakan keamanan. IDPS telah menjadi tambahan yang diperlukan untuk infrastruktur keamanan hampir setiap organisasi.

IDS (Intrution Detection System) adalah sebuah sistem yang melakukan pengawasan terhadap lalulintas (traffic) jaringan dan pengawasan terhadap kegiatan-kegiatan yang mencurigakan didalam sebuah sistem jaringan. Jika ditemukan kegiatan-kegiatan yang mencurigakan berhubungan dengan lalulintas jaringan, maka IDS akan memberikan peringatan kepada sistem atau administrator jaringan. Dalam banyak kasus IDS juga merespon terhadap lalulintas yang tidak normal / anomali melalui aksi pemblokiran user atau alamat IP (Internet Protocol) yang melakukan usaha pengaksesan jaringan tersebut.



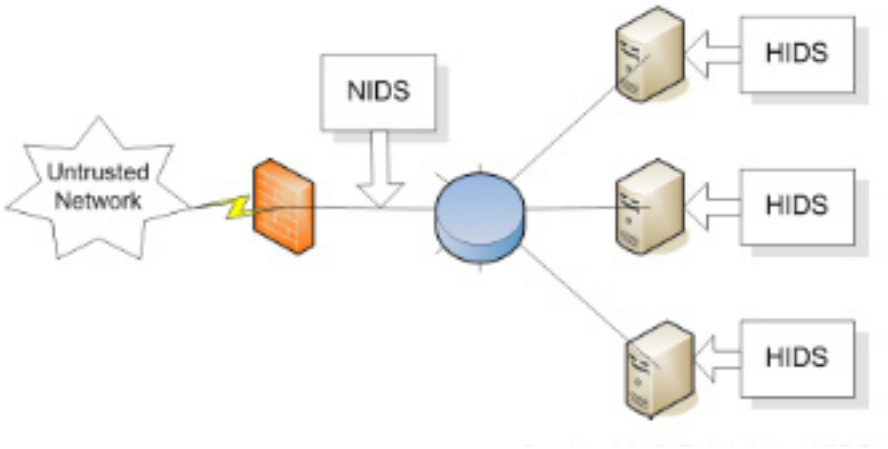
Gambar 3.48 Hirarki IDPS

IPS (Intrusion Prevention System) adalah sebuah sistem yang menggabungkan fungsi firewall dan fungsi IDS dengan proporsional. Teknologi ini dapat digunakan untuk mencegah serangan yang akan masuk ke jaringan lokal dengan memeriksa dan mencatat semua paket data serta mengenali paket dengan sensor, disaat serangan telah teridentifikasi, IPS akan menolak akses (block) dan mencatat (log) semua paket data yang teridentifikasi tersebut. Jadi IPS bertindak sepeti layaknya firewall yang akan melakukan allow dan block yang dikombinasikan dengan IDS yang dapat mendeteksi paket secara detail. IPS menggunakan signatures dari paket untuk mendeteksi aktivitas lalulintas di jaringan dan terminal, dimana pendeteksian paket yang masuk dan keluar (inbound-outbound)

dapat di cegah sedini mungkin sebelum merusak atau mendapatkan akses ke dalam jaringan lokal. Jadi early detection dan prevention menjadi penekanan pada IPS ini.
IDS dan IPS secara umum dikenal sebagai IDPS (Intrusion Detection and Prevention Systems). Biasanya dalam suatu perangkat keras memiliki fungsi IDS maupun IPS. Gambar 1 menunjukkan pembagian fungsi IDPS. Terdapat dua tipe IDS, yaitu NIDS dan HIDS. IPS juga memiliki dua tipe, yaitu NIPS dan HIPS. Seperti yang dilukiskan dalam Gambar 3.48

NIDS (Network Based IDS)

Semua lalu lintas yang mengalir ke sebuah jaringan akan dianalisis untuk mencari apakah ada percobaan serangan atau penyusupan ke dalam sistem jaringan. NIDS umumnya terletak di dalam segmen jaringan penting di mana server berada atau terdapat pada "pintu masuk" jaringan. Idealnya semua traffic yang berasal dari luar dan dalam jaringan di lakukan proses scan, namun cara ini dapat menyebabkan bottleneck yang mengganggu kecepatan akses di seluruh jaringan. Posisi dari NIDS dalam suatu jaringan dapat dilihat pada Gambar 3.49.



Gambar 3.49 Peletakkan NIDS dan HIDS dalam Sebuah Jaringan

HIDS (Host based IDS)

IDS jenis ini diletakkan pada host yang berdiri sendiri atau perlengkapan dalam sebuah jaringan. Sebuah HIDS melakukan pengawasan terhadap paket-paket yang berasal dari dalam maupun dari luar hanya pada satu alat saja dan kemudian memberi peringatan kepada user atau administrator sistem jaringan akan adanya kegiatankegiatan yang mencurigakan yang terdeteksi oleh HIDS. Posisi dari HIDS dalam suatu jaringan dapat dilihat pada Gambar 3.49

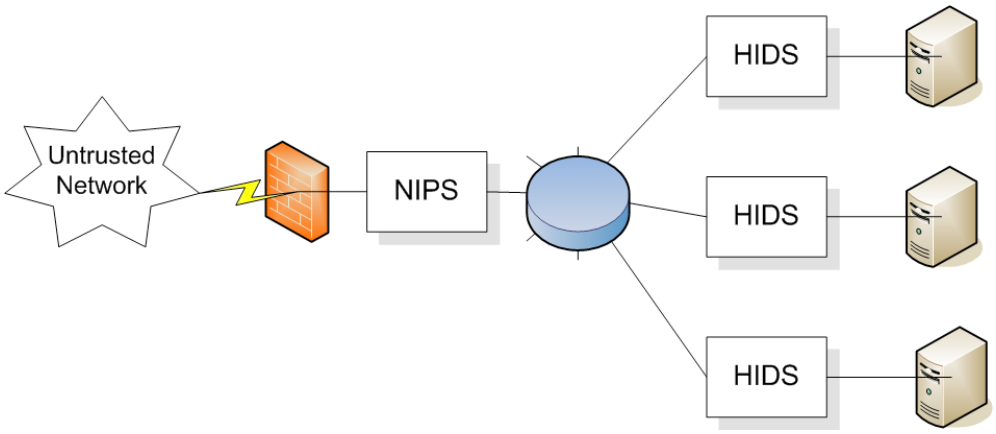
NIPS (Network based IPS)

NIPS yang juga disebut sebagai "In-line proactive protection", menahan semua trafik jaringan dan menginspeksi kelakuan dan kode yang mencurigakan. Karena menggunakan in-line model, performansi tinggi merupakan sebuah elemen krusial dari perangkat IPS untuk mencegah terjadinya bottleneck pada jaringan. Posisi dari NIPS dalam suatu jaringan dapat dilihat pada Gambar 3.50

HIPS (Host based IPS)

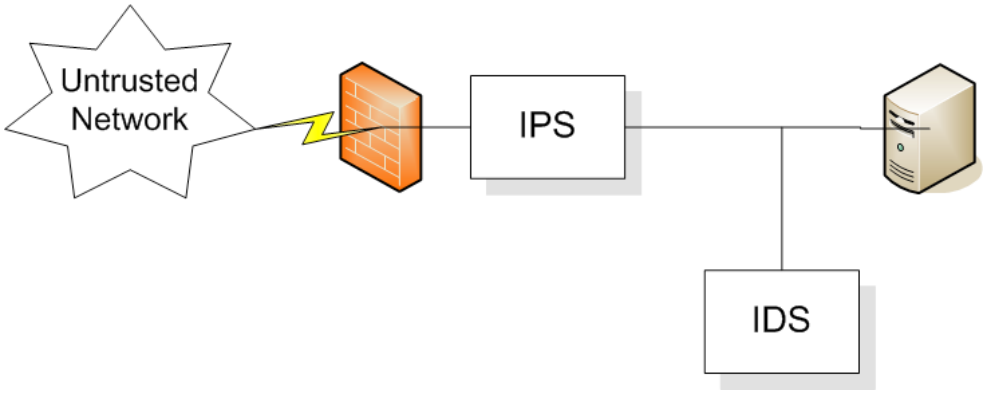
HIPS bekerja dengan memaksa sekelompok perangkat lunak fundamental untuk berkovensi secara konstan. Hal ini disebut dengan Application Binary Interface (ABI). Hampir tidak mungkin untuk

membajak sebuah aplikasi tanpa memodifikasi ABI, karena konvensi ini bersifat universal di antara aplikasi-aplikasi yang dimodifikasi. HIPS merupakan sebuah system pecegahan yang terdiri dari banyak layer, menggunakan packet filtering, inspeksi status dan metode pencegahan intrusi yang bersifat real-time untuk menjaga host berada di bawah keadaan dari efisiensi performansi yang layak. Mekanisme kerjanya yaitu dengan mencegah kode-kode berbahaya yang memasuki host agar tidak dieksekusi tanpa perlu untuk mengecek threat signature. Posisi dari HIPS dalam suatu jaringan dapat dilihat pada Gambar 3.50



Gambar 3.50 Peletakkan NIPS dan HIPS dalam Sebuah Jaringan

Dari definisi dan klasifikasi IDS dan IPS yang sudah dijabarkan, terdapat perbedaan mendasar antara IDS dan IPS. Perbedaannya adalah IDS tidak berada in-line dalam jaringan, atau dengan kata lain IDS "hanya" memantau jaringan saja dengan cara "terhubung" atau "tap" ke jaringan. Sedangkan IPS berada in-line dalam jaringan. Sehingga ketika terdapat serangan atau akses yang mencurigakan, maka IPS dapat langsung menutup akses tersebut. Perbedaan IDS dan IPS dari peletakannya dalam jaringan ditunjukkan oleh Gambar 3.51



Gambar 3.51 Perbedaan IDS dan IPS Ditinjau dari Peletakannya

Untuk lebih detailnya, perbedaan IDS dan IPS dapat dilihat pada tabel 1.

Tabel 1 Perbedaan IDS dan IPS

	IDS	IPS
Layer OSI	Layer 3	Layer 2, 3 dan 7
Manfaat	Identifikasi dan memeriksa semua paket yang melalui traffic jaringan, jika terdapat anomali, maka IDS akan memberi peringatan	Menggabungkan fungsi firewall, QoS, san IDS. Juga dapat mendeteksi anomali, dapat menyediakan fungsiallow, blosk dan log.
Aktifitas	Mendeteksi serangan hanya ketika serangan sudah masuk ke jaringan, dan tidak dapat melakukan sesuatu untuk menghentikannya	Early detection adalah teknik yang proaktif dapat mencegah serangan masuk dan menghentikanya dengan block
Komponen	Tidak dapat mendeteksi semua aktifitas malciuos code setiap saat, sehingga dapat mengakibatkan false negative yang banyak	Dapat mendeteksi new signature dan behavior attack, sehingga akan menurunkan tingkat false negative
Compatibility	Tidak dapat menggunakan ACL/script dari momponen sistem keamanan lain	Dapat diintegrasikan dengan ACL dan perimeter DMZ lainnya

Metode Deteksi

IDPS memiliki 3 metode untuk melakukan deteksi, yaitu signatured-based, anomaly-based, dan stateful protocol analysis. Ketiga metode ini dapat digunakan sekaligus atau sebagain aja.

Signatured-Based Detection

Metode ini dilakukan dengan membandingkan signature dari setiap paket untuk mengidentifikasi kemungkinan adanya intrusi. Metode ini efektif bila IDPS mendeteksi ancaman yang sudah di kenal, tetapi tidak efektif bila ancamannya baru atau tidak di kenal oleh IDPS. Pengertian dikenal dalam konteks ini adalah sudah pernah terjadi sebelumnya.. Metode ini merupakan metode yang paling sederhana, karena hanya membandingkan paket data, lalu di daftarkan menggunakan operasi perbandingan. Kelemahannya adalah metode ini tidak dapat melacak kejadian yang terjadi pada komunikasi yang lebih kompleks.

Anomaly-Based Detection

Metode ini digunakan dengan membandingkan kegiatan yang sedang di pantau dengan kegiatan yang di anggap normal untuk mendeteksi adanya penyimpangan. Pada metode ini, IDPS memiliki profil yang mewakili perilaku yang normal dari user, host, koneksi jaringan dan aplikasi. Profil tersebut didapat dari hasil pemantauan karakteristik dari suatu kegiatan dalam selang waktu tertentu. Kelebihan dari metode ini adalah efektif dalam mendeteksi ancaman yang belum dikenal, contohnya ketika jaringan diserang oleh tipe intrusi yang baru. Sedangkan kekurangan dari metode ini adalah dalam beberapa kasus, akan sulit untuk mendapatkan deteksi yang akurat dalam komunikasi yang lebih kompleks.

Stateful Protocol Analysis

Metode ini sebenarnya menyerupai anomaly-based, yaitu membandingkan profil yang sudah ada dengan kegiatan yang sedang berlangsung untuk mengidentifikasi penyimpangan. Namun, tidak seperti Anomaly-Based Detection yang menggunakan profil host, Stateful Protocol Analysis

menggunakan profil yang lebih luas yang dapat merinci bagaimana sebuah protokol yang istimewa dapat digunakan atau tidak. Arti "Stateful" disini adalah sistem di IDPS ini bisa memahami dan melacak situasi pada protokol network, transport dan application. Kelebihan dari metode ini adalah dapat mengidentifikasi rangkaian perintah yang tidak terduga seperti mengeluarkan perintah yang sama berulang – ulang. Sedangkan kekurangannya adalah kemungkinan terjadinya bentrokan antara protokol yang digunakan oleh IDPS dengan protokol umum yang digunakan oleh sistem operasi, atau dengan kata lain sulit membedakan implementasi client dan server pada interaksi protokol.

Komponen

Ada beberapa tipe komponen pada IDPS seperti yang dijelaskan berikut ini:

Sensor atau Agent

Berfungsi memantau dan menganalisis kegiatan. Sensor biasanya digunakan IDPS untuk memantau jaringan, termasuk di dalamnya teknologi Network-Based, Wireless, dan Network Behavior Analysis. Sedangkan Agent biasanya diguakan IDPS untuk teknologi Host-Based.

Management Server

Management Server adalah perangkat terpusat yang berfungsi menerima informasi dari Sensor atau Agent dan kemudian mengelolanya. Dalam konteks ini terdapat istilah korelasi, yaitu menyamakan informasi yang berasal dari Sensor atau Agent yang bertingkat, seperti menemukan kejadian yang disebabkan oleh IP address yang sama. Management Server tersedia dalam bentuk hardware maupun software.

Database Server

Database Server adalah tempat menyimpan informasi yang di catat oleh Sensor, Agent, maupun Management Server.

Console

Console adalah program yang menyediakan interface ke user dan administrator IDPS. Beberapa Console digunakan hanya untuk hal administrasi IDPS saja, seperti menkonfigurasi Sensor atau Agent. Namun juga terdapat beberapa Console yang digunakan untuk administrasi maupun memonitor.

Protokol lain memiliki representasi stateful yang serupa. Saat IDS memantau lalu lintas, IDS akan membangun representasi status yang serupa, mencocokkan lalu lintas dengan sifat pertukaran yang diharapkan. Protokol yang berbeda dengan status dan kondisi transisi yang berbeda dikalikan dengan jumlah instance (misalnya, jumlah koneksi TCP bersamaan yang dibuat setiap saat), membuat pembukuan IDS memang kompleks.

3.8.2 Sistem Pencegahan Intrusi

Sistem deteksi intrusi bekerja terutama untuk mendeteksi serangan setelah dimulai, dan tentu saja, pemilik sistem atau jaringan ingin mencegah serangan sebelum terjadi. Pikirkan pencuri rumah. Anda memasang kunci untuk mencegah intrusi, tetapi itu benar-benar tidak menghentikan pencuri

yang benar-benar berdedikasi yang akan menghancurkan dan masuk melalui jendela atau membuat lubang di atap jika motivasinya cukup kuat. Seperti kata pepatah, di mana ada kemauan, di situ ada jalan. Kesulitan kedua adalah Anda tidak pernah tahu kapan penyerang akan menyerang, apakah penyerang akan sendirian atau dalam kelompok yang terdiri dari ribuan orang, apakah penyerang akan menjadi seseorang atau pasukan semut terlatih, atau apakah kelompok kuningan berbaris melewatinya. merupakan bagian dari serangan. Anda dapat memasang alarm rumah yang mendeteksi gerakan, tekanan, panas tubuh, atau karakteristik penyerang lainnya, sehingga terlepas dari cara penyerang masuk, Anda atau polisi akan diberitahu tentang penyusupan tersebut, tetapi bahkan alarm mengandaikan penyerang akan menjadi seseorang, padahal sebenarnya itu mungkin robot atau drone. Lebih jauh lagi, alarm semacam itu tunduk pada kesalahan positif, karena hewan peliharaan rumah tangga atau balon yang bergerak tertiup angin dapat memicu alarm.

Demikian pula, sistem komputer tunduk pada banyak kemungkinan serangan, dan mencegah semuanya hampir tidak mungkin. Mengalahkan penyerang, sebenarnya semua penyerang, juga hampir tidak mungkin. Menambah kesulitan ini adalah membedakan serangan dari perilaku jinak tapi tidak biasa. Mendeteksi serangan menjadi lebih mudah saat serangan terungkap, ketika menjadi lebih jelas bahwa motifnya jahat dan bahaya sudah dekat atau benar-benar sedang berlangsung. Jadi, saat bukti meningkat, deteksi menjadi lebih pasti; mampu mendeteksi hal-hal buruk sebelum mereka menyebabkan terlalu banyak kerusakan adalah premis yang menjadi dasar sistem deteksi intrusi.

Sebaliknya, sistem pencegahan intrusi, atau IPS, mencoba untuk memblokir atau menghentikan bahaya. Faktanya, ini adalah sistem deteksi intrusi dengan kemampuan respons bawaan. Tanggapannya tidak hanya meningkatkan alarm; tanggapan otomatis termasuk memotong akses pengguna, menolak semua lalu lintas dari alamat a.b.c.d, atau memblokir akses semua pengguna ke file atau program tertentu. Semua yang telah dikatakan tentang sistem deteksi intrusi juga berlaku untuk sistem pencegahan intrusi. Di bagian berikutnya kami mempertimbangkan beberapa tindakan yang dapat dilakukan IPS setelah mendeteksi kemungkinan serangan.

Respons Intrusi

Deteksi intrusi bersifat probabilistik. Bahkan dalam menghadapi pola yang jelas, seperti sejumlah besar paket ping, mungkin ribuan orang kebetulan ingin menguji apakah server hidup pada saat yang sama, meskipun kemungkinan itu sangat kecil. Dalam mengambil tindakan, terutama jika alat menyebabkan tindakan secara otomatis, administrator jaringan harus mempertimbangkan konsekuensi tindakan terhadap kemungkinan bahwa tidak ada serangan.

Menanggapi Alarm

Apa pun jenisnya, sistem deteksi penyusupan akan membunyikan alarm saat menemukan kecocokan. Alarm dapat berkisar dari sesuatu yang sederhana, seperti menulis catatan di log audit, hingga sesuatu yang signifikan, seperti memanggil administrator keamanan sistem. Implementasi tertentu memungkinkan pengguna untuk menentukan tindakan apa yang harus diambil sistem pada peristiwa apa. Apa tanggapan yang mungkin? Jangkauannya tidak terbatas dan dapat berupa apa saja yang dapat dibayangkan (dan diprogram) oleh administrator.

Secara umum, respons terbagi dalam tiga kategori utama (salah satu atau semuanya dapat digunakan dalam satu respons):

- Memantau, mengumpulkan data, mungkin menambah jumlah data yang dikumpulkan.
- Protect, bertindak untuk mengurangi eksposur.
- Memberi tanda peringatan ke komponen proteksi lainnya.
- Panggil manusia.

Pemantauan sesuai untuk serangan dengan dampak sederhana (awal). Mungkin tujuan sebenarnya adalah untuk melihat penyusup untuk melihat sumber daya apa yang sedang diakses atau upaya serangan apa yang dicoba. Kemungkinan pemantauan lainnya adalah merekam semua lalu lintas dari sumber tertentu untuk analisis di masa mendatang. Pendekatan ini harus tidak terlihat oleh penyerang. Melindungi dapat berarti meningkatkan kontrol akses dan bahkan membuat sumber daya tidak tersedia (misalnya, mematikan koneksi jaringan atau membuat file tidak tersedia). Sistem bahkan dapat memutuskan koneksi jaringan yang digunakan penyerang. Berbeda dengan pemantauan, melindungi mungkin sangat terlihat oleh penyerang. Akhirnya, memanggil manusia memungkinkan diskriminasi individu. IDS dapat mengambil tindakan defensif awal, mungkin terlalu kuat, segera sementara juga menghasilkan peringatan kepada manusia, yang mungkin membutuhkan beberapa detik, menit, atau lebih lama untuk merespons tetapi kemudian menerapkan tindakan balasan yang lebih rinci dan spesifik.

Alarm

Tindakan paling sederhana dan paling aman untuk IDS adalah dengan membuat alarm ke administrator yang kemudian akan menentukan langkah selanjutnya. Manusia paling tepat untuk menilai tingkat keparahan situasi dan memilih di antara tindakan pencegahan. Selain itu, manusia dapat mengingat situasi masa lalu dan terkadang mengenali koneksi atau kesamaan yang mungkin tidak terdeteksi oleh IDS.

Sayangnya, menghasilkan alarm mengharuskan manusia selalu tersedia untuk merespons alarm itu dan responsnya tepat waktu dan tepat. Jika beberapa sensor menghasilkan alarm pada saat yang sama, manusia dapat menjadi kelebihan beban dan melewatkan alarm baru atau terlalu terlibat dengan salah satu sehingga alarm kedua tidak ditangani dengan cukup cepat. Lebih buruk lagi, alarm kedua dapat mengalihkan perhatian atau membingungkan manusia, sehingga tindakan pada alarm pertama terancam. Di Studi Kasus 3-25 kami membahas berapa banyak alarm yang menyebabkan pelanggaran serius.

Studi Kasus 3-25 : Target Corp. Kewalahan oleh Terlalu Banyak Alarm

Pada akhir 2013 yang sekarang ternyata adalah peretas Rusia yang menyusup ke jaringan Target, pengecer besar di Amerika Serikat. Para penyerang menanam kode untuk mengumpulkan nomor kartu kredit dan debit pembeli, termasuk nomor verifikasi yang akan membuat nomor tersebut cukup berharga di pasar gelap. Secara keseluruhan, 40 juta nomor dicuri dalam beberapa minggu menjelang Natal, biasanya periode belanja tersibuk pengecer tahun ini. Target telah berinvestasi dalam teknologi deteksi penyusupan, yang berfungsi dan melihat aktivitas mencurigakan. Perangkat lunak memberi tahu pusat pemantauan keamanan Target yang menganalisis situasi dan meningkatkan peringatan ke pusat operasi keamanan perusahaan pada 30

November dan lagi pada 2 Desember. Staf pusat operasi keamanan tidak melakukan apa-apa.

"Seperti perusahaan besar lainnya, setiap minggu di Target ada sejumlah besar peristiwa teknis yang terjadi dan dicatat. Melalui penyelidikan kami, kami mengetahui bahwa setelah para penjahat ini memasuki jaringan kami, sejumlah kecil aktivitas mereka dicatat dan muncul ke tim kami, "kata juru bicara Target Molly Snyder melalui email. "Kegiatan itu dievaluasi dan ditindaklanjuti."

Ancaman yang diterima diklasifikasikan sebagai "malware.binary" yang berarti bagian dari kode berbahaya yang tidak dikenal dengan jenis, sumber, atau kemampuan yang tidak diketahui. Beberapa ahli mengatakan mereka memperkirakan tim keamanan Target menerima ratusan peringatan ancaman seperti itu setiap hari.

Kisah ini menunjukkan kesulitan menggunakan teknologi IDS: Kecuali seseorang bertindak berdasarkan alarm yang dihasilkan, meningkatkan peringatan tidak ada nilainya. Namun terkadang jumlah alarm sangat banyak sehingga tim respons kewalahan. Dengan terlalu banyak alarm, responden bisa menjadi puas diri, mengabaikan situasi serius karena penyelidikan peringatan sebelumnya ternyata kosong.

Perilaku Adaptif

Karena keterbatasan manusia ini, IDS terkadang dapat dikonfigurasi untuk mengambil tindakan untuk memblokir serangan atau mengurangi dampaknya. Berikut adalah beberapa jenis tindakan yang dapat dilakukan IDS:

- Terus pantau jaringan.
- Blokir serangan dengan mengarahkan lalu lintas serangan ke host pemantauan, membuang lalu lintas, atau menghentikan sesi.
- Konfigurasi ulang jaringan dengan membawa host lain online (untuk meningkatkan kapasitas) atau menyesuaikan penyeimbang beban.
- Sesuaikan performa untuk memperlambat serangan, misalnya dengan menjatuhkan sebagian lalu lintas masuk.
- Tolak akses ke host atau layanan jaringan tertentu.
- Matikan sebagian jaringan.
- Matikan seluruh jaringan.

Serangan balik

Tindakan terakhir yang dapat diambil pada deteksi serangan adalah untuk melakukan pelanggaran, untuk menyerang balik. Tindakan ofensif harus dilakukan dengan sangat hati-hati karena beberapa alasan:

- Penyerang yang terlihat mungkin bukan penyerang yang sebenarnya. Menentukan sumber yang benar dan pengirim lalu lintas Internet tidak mudah. Mengambil tindakan terhadap pihak yang salah hanya memperburuk keadaan.

- Sebuah serangan balik dapat menyebabkan pertempuran real-time di mana baik pertahanan dan pelanggaran harus dilaksanakan dengan sedikit waktu untuk menilai situasi.
- Pembalasan dalam kemarahan belum tentu dipikirkan dengan matang.
- Legalitas bisa bergeser. Tindakan yang terukur dan perlu untuk melindungi sumber daya seseorang adalah prinsip hukum yang mapan. Mengambil tindakan ofensif membuka satu untuk bahaya hukum, sebanding dengan penyerang.
- Memprovokasi penyerang dapat menyebabkan eskalasi. Penyerang dapat menganggap serangan balik sebagai tantangan.

3.7.2 Tujuan untuk Sistem Deteksi Intrusi (IDS)

Dua gaya deteksi intrusi — pencocokan pola dan heuristik — mewakili pendekatan yang berbeda, yang masing-masing memiliki kelebihan dan kekurangan. Produk IDS yang sebenarnya sering memadukan kedua pendekatan tersebut. Idealnya, IDS harus cepat, sederhana, dan akurat, sekaligus lengkap. Itu harus mendeteksi semua serangan dengan penalti kinerja yang dapat diabaikan.

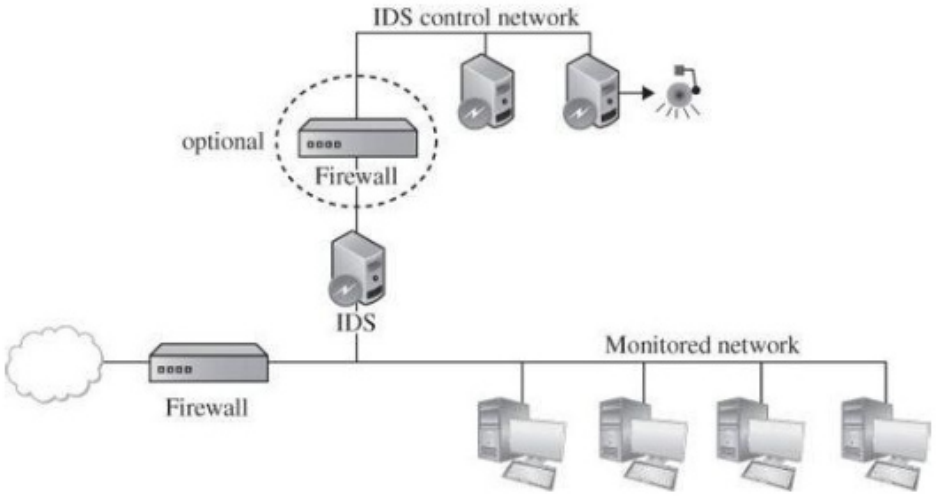
IDS dapat menggunakan beberapa—atau semua—pendekatan desain berikut:

- Filter pada header paket.
- Filter pada isi paket.
- Pertahankan status koneksi.
- Gunakan tanda tangan multipaket yang kompleks.
- Gunakan jumlah tanda tangan minimal dengan efek maksimal.
- Filter secara real time, online.
- Sembunyikan keberadaannya.
- Gunakan ukuran jendela waktu geser yang optimal untuk mencocokkan tanda tangan.

Stealth Mode

IDS adalah perangkat jaringan (atau, dalam kasus IDS berbasis host, program yang berjalan pada perangkat jaringan). Setiap perangkat jaringan berpotensi rentan terhadap serangan jaringan. Seberapa berguna IDS jika diFlooding i dengan serangan penolakan layanan? Jika penyerang berhasil masuk ke sistem dalam jaringan yang dilindungi, bukankah mencoba menonaktifkan IDS menjadi langkah selanjutnya?

Untuk mengatasi masalah tersebut, sebagian besar IDS berjalan dalam Stealth Mode (mode siluman), di mana IDS memiliki dua antarmuka jaringan: satu untuk jaringan (atau segmen jaringan) yang dipantau dan yang lainnya untuk menghasilkan peringatan dan mungkin melakukan kebutuhan administratif lainnya. IDS menggunakan antarmuka yang dipantau sebagai input saja; itu tidak pernah mengirim paket keluar melalui antarmuka itu. Seringkali, antarmuka dikonfigurasi sehingga perangkat tidak memiliki alamat yang dipublikasikan melalui antarmuka yang dipantau; yaitu, tidak ada router yang dapat merutekan apa pun secara langsung ke alamat tersebut karena router tidak mengetahui keberadaan perangkat tersebut. Ini adalah penyadapan pasif yang sempurna. Jika IDS perlu menghasilkan peringatan, ia hanya menggunakan antarmuka alarm pada jaringan kontrol yang benar-benar terpisah. Arsitektur seperti itu ditunjukkan pada Gambar 3-66.



Gambar 3-66 Jaringan Kontrol IDS

Accurate Situation Assessment

Sistem deteksi intrusi tidak sempurna, dan kesalahan adalah masalah terbesar mereka. Meskipun IDS mungkin mendeteksi penyusup dengan benar hampir sepanjang waktu, ia mungkin tersandung dalam dua cara berbeda: dengan membunyikan alarm untuk sesuatu yang sebenarnya bukan serangan (disebut kesalahan positif palsu, atau kesalahan tipe I dalam komunitas statistik) atau tidak. membunyikan alarm untuk serangan nyata (negatif palsu, atau kesalahan tipe II). Terlalu banyak kesalahan positif, seperti pelanggaran Target, berarti administrator akan kurang percaya diri dengan peringatan IDS, mungkin menyebabkan alarm yang sebenarnya diabaikan. Tapi negatif palsu berarti bahwa serangan nyata melewati IDS tanpa tindakan. Kami mengatakan bahwa tingkat positif palsu dan negatif palsu mewakili sensitivitas sistem. Sebagian besar implementasi IDS memungkinkan administrator untuk menyesuaikan sensitivitas sistem untuk mencapai keseimbangan yang dapat diterima antara positif palsu dan negatif.

3.9 Manajemen Jaringan

Selanjutnya, kami memperkenalkan beberapa konsep yang relevan dengan keamanan dalam mengelola, mengelola, dan menyetel jaringan. Administrator dapat mengambil tindakan untuk memilih satu aliran lalu lintas jaringan daripada yang lain, baik untuk mempromosikan penggunaan sumber daya yang adil atau untuk memblokir aliran lalu lintas berbahaya sehingga komunikasi yang tidak berbahaya dapat berjalan. Untuk melakukan penyetelan semacam ini, administrator memerlukan Gambaran akurat tentang status jaringan. Alat yang disebut informasi keamanan dan perangkat manajemen peristiwa mengumpulkan indikasi status dari berbagai produk—termasuk firewall, IDS, router, penyeimbang beban—dan menyatukan aliran data terpisah ini ke dalam tampilan terpadu.

3.9.1 Manajemen untuk Memastikan Layanan

Jaringan bukanlah jenis sistem yang diatur dan dilupakan; karena aktivitas jaringan bersifat dinamis, administrator perlu memantau kinerja jaringan dan menyesuaikan karakteristik yang diperlukan.

Di bagian ini kami mencantumkan beberapa jenis manajemen yang dibutuhkan jaringan. Ketahuilah, bagaimanapun, bahwa sebagian besar informasi ini berguna untuk administrator jaringan yang tanggung jawab utamanya adalah menjaga agar jaringan tetap berjalan dengan lancar, bukan melindungi dari serangan penolakan layanan. Langkah-langkah ini melawan kasus kinerja suboptimal biasa, tetapi bukan serangan bersama. Di bagian ini kami hanya menyebutkan topik-topik ini; untuk detailnya, Anda harus berkonsultasi dengan referensi administrasi jaringan yang komprehensif.

Perencanaan Kapasitas

Salah satu penyebab jinak dari penolakan layanan adalah kapasitas yang tidak mencukupi: terlalu banyak data untuk terlalu sedikit kemampuan. Biasanya tidak dilihat sebagai masalah keamanan, perencanaan kapasitas melibatkan pemantauan beban lalu lintas jaringan dan kinerja untuk menentukan kapan harus meningkatkan aspek mana.

Jaringan atau komponen yang berjalan pada atau mendekati kapasitas memiliki sedikit margin untuk kesalahan, yang berarti bahwa lonjakan lalu lintas yang sedikit tetapi normal dapat menempatkan jaringan di atas dan menyebabkan penurunan layanan yang signifikan.

Situs web sangat rentan terhadap masalah kapasitas yang tidak terduga. Situs berita dapat berjalan dengan baik selama waktu normal hingga peristiwa penting terjadi, seperti kematian orang terkenal atau gempa bumi, kecelakaan pesawat, atau serangan teroris, setelah itu banyak orang menginginkan detail terbaru tentang peristiwa tersebut. Meluncurkan produk baru dengan iklan juga dapat menyebabkan kelebihan beban; acara seperti pembukaan penjualan tiket untuk konser populer atau acara olahraga telah memFlooding i situs web.

Penyeimbang Beban

Situs web populer seperti Google, Microsoft, dan New York Times tidak dijalankan di satu komputer saja; tidak ada satu komputer pun yang memiliki kapasitas untuk mendukung semua lalu lintas yang diterima situs-situs ini sekaligus. Sebaliknya, tempat-tempat ini mengandalkan banyak komputer untuk menangani volume.

Publik tidak mengetahui beberapa server ini, misalnya, saat menggunakan URL www.nytimes.com, yang dapat menjadi server1.nytimes.com atau www3.nytimes.com. Faktanya, pada kunjungan berturut-turut ke situs web, aktivitas pengguna dapat ditangani oleh server yang berbeda. Penyeimbang beban adalah alat yang mengarahkan lalu lintas ke server yang berbeda saat bekerja untuk memastikan bahwa semua server memiliki beban kerja yang kira-kira setara.

Penyetelan Jaringan

Demikian pula, insinyur jaringan dapat menyesuaikan lalu lintas pada segmen jaringan individu. Jika dua klien di satu segmen bertanggung jawab atas sebagian besar lalu lintas, mungkin lebih baik menempatkan mereka di segmen terpisah agar beban lalu lintas merata. Insinyur dapat memasang tautan baru, merestrukturisasi segmen jaringan, atau meningkatkan konektivitas untuk memastikan kinerja jaringan yang baik. Penyetelan jaringan bergantung pada data solid yang diperoleh dengan memantau lalu lintas jaringan dari waktu ke waktu.

Dalam serangan nyata, administrator jaringan dapat menyesuaikan alokasi bandwidth ke segmen, dan mereka dapat memantau lalu lintas masuk, secara selektif menjatuhkan paket yang tampaknya berbahaya. (Catatan: Menjatuhkan paket secara berlebihan berisiko kecil; protokol TCP mendeteksi paket yang hilang dan mencari transmisi ulang, dan protokol UDP tidak menjamin pengiriman. Kehilangan sebagian kecil lalu lintas yang sah sambil menangkis serangan penolakan layanan adalah perdagangan yang dapat diterima -mati.)

Pembatasan laju adalah tindakan balasan yang mengurangi dampak serangan. Dengan pembatasan tarif, volume lalu lintas yang diizinkan ke alamat tertentu berkurang. Router dapat mengirim sinyal quench kembali ke router lain yang meneruskan lalu lintas; sinyal tersebut menginformasikan router pengirim bahwa router penerima kelebihan beban dan tidak dapat mengikuti, oleh karena itu meminta pengirim untuk menahan transmisi data. Quench dapat bekerja kembali melalui jaringan ke sumber serangan, selama serangan itu berasal dari satu titik.

Pengalamatan Jaringan

Masalah yang melekat dalam pengalamatan Internet (IPv4) adalah bahwa paket apa pun dapat mengklaim berasal dari alamat mana pun: Sebuah sistem di alamat A dapat mengirim paket yang menunjukkan alamat B sebagai sumbernya. Pernyataan itu membutuhkan sedikit elaborasi karena spoofing alamat bukan hanya masalah mengisi bagian yang kosong di halaman web. Sebagian besar pengguna berinteraksi dengan Internet melalui aplikasi tingkat yang lebih tinggi, seperti browser dan penangan surat, yang membuat aliran komunikasi dan meneruskannya ke penangan protokol, seperti pengikatan dan kaus kaki. Penangan protokol melakukan interaksi jaringan, memasok data yang akurat dalam aliran komunikasi. Dengan demikian, seseorang dapat memalsukan alamat hanya dengan mengesampingkan penangan protokol ini, yang memerlukan hak istimewa dalam sistem operasi. Alat peretas dapat melakukan interaksi itu, dan peneliti Beverly dan Bauer [BEV05] melaporkan eksperimen di mana mereka memalsukan transmisi dari seperempat alamat Internet.

Penyedia layanan Internet, ISP, dapat berbuat lebih banyak untuk memastikan validitas alamat dalam paket. Dengan susah payah, penyedia dapat membedakan antara lalu lintas dari pelanggan mereka sendiri—yang alamatnya memblokir penyedia yang harus diketahui dan dapat diverifikasi—dan lalu lintas dari pihak luar. Memiliki alamat sumber yang andal akan membatasi serangan penolakan layanan tertentu, tetapi desain protokol Internet tidak menyertakan mekanisme untuk mendukung keaslian alamat.

Shunning

Dengan alamat sumber yang andal, administrator jaringan dapat mengatur router tepi untuk menjatuhkan paket yang terlibat dalam serangan penolakan layanan. Praktik ini, yang disebut pengucilan, pada dasarnya menyaring semua lalu lintas dari alamat yang terlibat. Pemantauan real-time yang mendeteksi serangan menentukan alamat dari mana serangan itu datang dan bertindak cepat untuk memblokir alamat tersebut. Firewall dapat menerapkan pengucilan alamat tertentu

Namun, menghindari memiliki sisi negatifnya. Jika penyerang dapat mendeteksi bahwa sebuah situs menerapkan shunning, penyerang dapat mengirim lalu lintas serangan yang dipalsukan agar tampak dari sumber yang sah. Artinya, penyerang mungkin membuat seolah-olah serangan itu berasal dari google.com atau facebook.com, misalnya; menghindari serangan yang nyata memiliki hasil negatif berupa penolakan lalu lintas yang sah dari Google atau Facebook.

Blacklisting dan Sinkholing

Dalam kasus ekstrim, administrator jaringan dapat memutuskan untuk secara efektif memutuskan sistem yang ditargetkan. Administrator dapat memasukkan alamat target ke daftar hitam, artinya tidak ada lalu lintas yang masuk ke alamat itu, dari sumber yang sah atau berbahaya. Atau, administrator dapat mengarahkan lalu lintas ke alamat yang valid di mana lalu lintas masuk dapat dianalisis; proses ini disebut sinkholing.

Kedua tindakan pencegahan ini dapat diterapkan di tepi jaringan, sebelum volume lalu lintas yang berlebihan diizinkan untuk memFlooding i subjaringan internal. Jika tidak, lalu lintas yang berlebihan dapat memFlooding i semua subnetwork internal, sehingga menolak atau menurunkan layanan ke semua host di subnetwork, bukan hanya satu host yang menjadi target serangan.

Semua tindakan administratif ini membawa potensi risiko. Pemantauan jaringan mempengaruhi kinerja jaringan karena mencegat, menganalisis, dan meneruskan lalu lintas membutuhkan waktu dan karenanya menimbulkan penundaan. Dalam operasi normal, penundaannya kecil, tetapi pada saat serangan, penundaan ini, yang memengaruhi lalu lintas baik maupun jahat, semakin memperlambat sistem yang sudah tertekan. Lebih jauh lagi, manajemen yang baik memerlukan analisis yang terperinci, untuk melihat, misalnya, tidak hanya bahwa lalu lintas adalah paket SYN tetapi paket SYN berasal dari alamat a.b.c.d, yang merupakan alamat yang sama dengan 250 paket SYN yang baru-baru ini berasal. Mengenali paket SYN dapat dilakukan secara instan; mengenali alamat a.b.c.d yang terlibat dalam 250 serangan sebelumnya memerlukan analisis data historis yang disimpan. Inspeksi yang lebih tepat menghasilkan informasi yang lebih berguna tetapi juga membutuhkan lebih banyak waktu untuk inspeksi.

Peralatan jaringan seperti firewall, router, sakelar, dan penyeimbang beban sering kali menyediakan data bagi orang untuk menganalisis dan mengelola jaringan. Terlalu banyak informasi dapat memFlooding i administrator jaringan manusia, terutama seseorang yang keterampilan keamanannya terbatas. Dengan demikian, penanggulangan manajemen lebih tepat untuk jaringan yang besar atau cukup penting untuk memiliki staf keamanan yang berpengalaman dengan sumber daya yang memadai.

Untuk semua jaringan, dengan atau tanpa tim keamanan yang mumpuni, sebagian beban pemantauan dan pendeteksian serangan penolakan layanan dapat ditangani oleh perangkat lunak. Pada bagian berikutnya kami menjelaskan sistem deteksi dan pencegahan intrusi, perangkat komputer yang melakukan pemantauan semacam itu.

3.9.2 Security Information and Event Management (SIEM)

Dalam bab ini, kita telah membahas produk jaringan dan keamanan, termasuk router, switch, VPN, dan berbagai jenis firewall, IDS, dan IPS. Sebuah perusahaan besar dapat memiliki ratusan atau bahkan ribuan produk seperti itu, seringkali dengan merek dan model yang berbeda, serta puluhan ribu server dan workstation, yang semuanya perlu dipantau oleh personel keamanan. Di bagian ini, kita membahas alat yang memungkinkan tim keamanan kecil untuk memantau dan menanggapi masalah keamanan dari seluruh perusahaan tersebut.

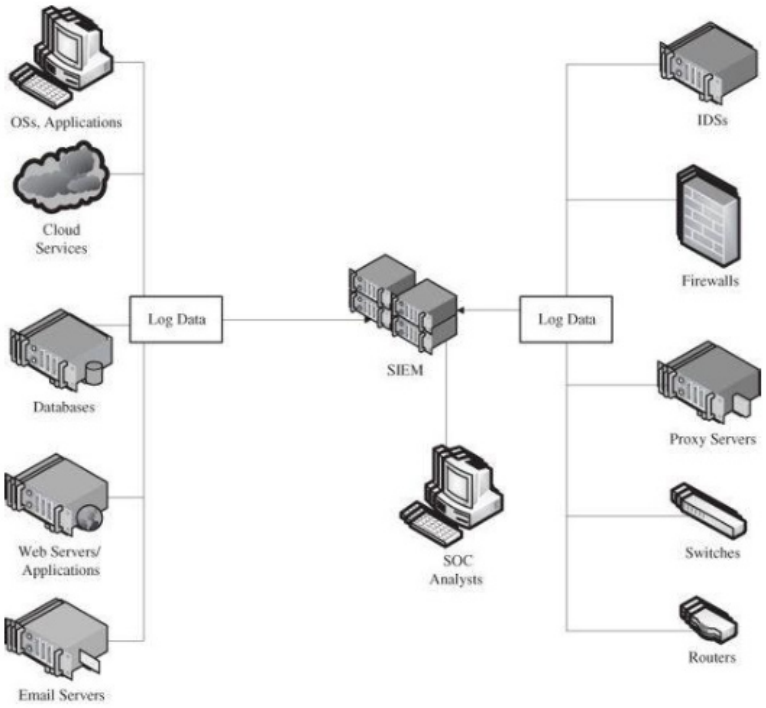
Pusat Operasi Keamanan (SOC)

Sebagai contoh, mari kita bayangkan jaringan toko grosir ritel bernama SiC Groceries, dengan kantor pusat di New York City dan 1.000 toko di seluruh Amerika Serikat. Semua toko SiC perlu memelihara komunikasi jaringan dengan kantor pusat untuk mengoordinasikan inventaris toko, penjualan, karyawan, dan masalah logistik lainnya, sehingga setiap toko memerlukan koneksi Internet serta koneksi VPN yang selalu aktif ke kantor pusat. Koneksi internet di toko ritel pada dasarnya sama dengan yang Anda dapatkan dari ISP lokal, dan mereka rentan terhadap serangan seperti yang lain. Lebih buruk lagi, penyerang yang menembus jaringan toko ritel akan memiliki akses orang dalam ke kantor pusat perusahaan melalui VPN. Itu berarti bahwa setiap toko dalam rantai merupakan vektor serangan potensial untuk membobol jaringan utama perusahaan dan harus dilindungi seperti itu. Alih-alih memiliki satu firewall dan satu IDS yang melindungi koneksi Internet perusahaan, SiC Groceries membutuhkan 1.000 firewall dan IDS yang melindungi 1.000 koneksi Internet.

Tetapi siapa yang akan memantau firewall dan IDS tersebut untuk memastikan mereka berfungsi dengan baik, menanggapi peringatan mereka, dan menyelidiki kemungkinan insiden keamanan di toko ritel? Tentu tidak ekonomis untuk memiliki staf keamanan di setiap toko, karena keahlian semacam itu mahal, dan setiap toko jarang memiliki masalah keamanan yang memerlukan intervensi manual.

Sebagai gantinya, SiC Groceries akan membuat *Security Operations Center* (SOC) di satu lokasi, mungkin markas mereka. SOC adalah tim personel keamanan yang didedikasikan untuk memantau jaringan untuk insiden keamanan dan menyelidiki serta memulihkan insiden tersebut.

Agar SOC-nya efektif, SiC Groceries harus mengizinkan akses jarak jauh tim SOC untuk memantau semua jaringan dan produk keamanan di seluruh perusahaannya, termasuk semua tokonya. Personel keamanan dapat secara manual masuk ke setiap perangkat untuk memeriksa status dan mencari peringatan, tetapi opsi itu tidak berskala, dan membuatnya sulit untuk mengidentifikasi bahkan pola serangan sederhana, seperti serangan terhadap banyak toko yang berasal dari alamat sumber yang sama. Sebaliknya, SiC membutuhkan bantuan otomatis untuk meningkatkan kemampuan tim keamanan, dan di sinilah alat informasi keamanan dan manajemen acara, atau SIEM, berperan. SIEM adalah sistem perangkat lunak yang mengumpulkan data yang relevan dengan keamanan dari berbagai produk perangkat keras dan perangkat lunak untuk membuat dasbor keamanan terpadu (seperti Gambar yang ditunjukkan pada Gambar 3-67 untuk personel SOC.



Gambar 3-67 Dasbor SIEM

Produk keamanan modern, peralatan jaringan, dan sistem operasi biasanya melaporkan data yang relevan dengan keamanan ke file log berbasis teks. Misalnya, ketika pengguna memasukkan kata sandi yang salah saat mencoba masuk ke Windows, Windows dapat menulis peristiwa "kegagalan logon" ke log keamanannya. Selain peristiwa masuk, sistem operasi umumnya dapat mencatat peristiwa manajemen pengguna (misalnya, menambahkan pengguna atau mengubah izin pengguna) dan akses file atau aplikasi. Alat keamanan, baik berbasis host maupun berbasis jaringan, dapat melaporkan hasil pemindaian malware, upaya penyusupan yang terdeteksi, dan koneksi yang diblokir. SIEM dapat mengumpulkan file log tersebut secara teratur dari seluruh perusahaan, memperbarui personel SOC tentang status keamanan perusahaan setiap beberapa detik.

Tetapi SIEM tidak hanya mengumpulkan informasi; mereka dapat melakukan banyak hal untuk membantu analis SOC memahaminya. Dengan semua data peristiwa keamanan di satu tempat, analis dapat mencari pola di seluruh perusahaan dan dalam jangka waktu yang mencakup bulan atau bahkan tahun. Banyak SIEM memungkinkan analis untuk mengatur data dengan cara menarik yang tak terhitung jumlahnya. Misalnya, seorang analis SOC mungkin melihat lonjakan peristiwa login di tengah malam dan ingin menyelidikinya. SIEM akan memungkinkan analis untuk mencari semua aktivitas login antara jam, katakanlah, 1:00–4:00 AM Waktu Bagian Timur, dan kemudian melanjutkan untuk menyelidiki berdasarkan faktor lain, seperti alamat IP, negara sumber yang jelas, target sistem, atau nama pengguna yang ditargetkan. Kemampuan untuk menjalankan pencarian seperti ini dan dengan cepat menyelidiki firasat di semua sistem perusahaan adalah terobosan mendasar untuk analisis keamanan yang hampir real-time.

Fitur pencarian ini memberikan kemampuan seperti IDS untuk deteksi berbasis tanda tangan. Sebagian besar SIEM memungkinkan pengguna untuk membuat dan berbagi pencarian yang diatur untuk berjalan secara berkala dan menghasilkan tanggapan berdasarkan hasil. Misalnya, seorang analis mungkin

membuat aturan yang mencari tiga upaya masuk yang gagal terhadap satu host dalam jangka waktu lima menit. Jika aturan pernah menemukan perilaku seperti itu, analis dapat membuatnya waspada di layar, mengirim email kepadanya, atau bahkan menjalankan skrip khusus. Ini bukan pengganti kemampuan IDS, melainkan pelengkap: SIEM dapat mencocokkan aturan yang lebih kompleks terhadap berbagai sumber log di seluruh perusahaan, tetapi pencocokan aturannya lambat dibandingkan dengan IDS, dan tidak cocok untuk menganalisis jaringan lalu lintas.

Selain mengatur data log dan membuatnya dapat dicari, sebagian besar SIEM juga memiliki fitur untuk mengaktifkan alur kerja analis SOC. Salah satu fitur umum di antara SIEM adalah memungkinkan analis untuk "mengklaim" peristiwa untuk diselidiki, memberi tim SOC cara langsung untuk membagi beban kerja. SIEM ini kemudian memungkinkan analis untuk membubuhi keterangan suatu peristiwa dengan catatan dari penyelidikan dan menempatkan peristiwa tersebut dalam antrian untuk analisis lebih lanjut atau menutup penyelidikan.

Kendala SIEM

Meskipun SIEM telah berkembang menjadi alat keamanan penting untuk perusahaan besar mana pun, mereka adalah sistem kompleks yang sulit untuk diterapkan, dipelihara, dan digunakan. Berikut adalah beberapa masalah yang perlu dipertimbangkan ketika memilih SIEM:

- Biaya. Solusi SIEM komersial untuk perusahaan besar dapat menghabiskan biaya jutaan dolar, tetapi beberapa SIEM open-source gratis. Ukuran organisasi, kompleksitas sistem, persyaratan fungsional dan kinerja, dan selera untuk pengembangan kustom semua harus membantu menginformasikan pilihan SIEM. Sementara jutaan dolar untuk produk perangkat lunak yang tersedia mungkin tampak selangit pada awalnya, pembeliannya akan tergantung pada kebutuhan organisasi. Jika perusahaan perlu membuat fitur khusus dan membayar untuk dukungan ahli, pemeliharaan, dan pelatihan, solusi sumber terbuka gratis mungkin lebih mahal daripada produk komersial yang didukung.
- Portabilitas data. Persyaratan berkembang, dan SIEM yang memenuhi kebutuhan saat ini suatu hari nanti akan perlu diganti. Sebelum memilih SIEM, pastikan Anda akan memiliki cara untuk mengekspor data yang Anda simpan dalam format standar yang dapat dibaca oleh sebagian besar SIEM lainnya. Pengetahuan yang Anda simpan di SIEM, seperti pencarian tersimpan atau visualisasi data, cenderung spesifik untuk SIEM dan Anda mungkin perlu membangun kembali basis pengetahuan tersebut saat Anda berpindah produk.
- Kompatibilitas sumber log. SIEM terus menjadi lebih fleksibel dalam hal jenis data yang dapat mereka impor dan kemudahan yang memungkinkan pengguna menentukan tipe data baru, tetapi beberapa SIEM lebih baik daripada yang lain dalam hal ini. Bergantung pada tipe data dan sistem yang menghasilkan kumpulan log tertentu, SIEM mungkin mengharuskan Anda menginstal agen atau server perantara untuk mengumpulkan log. Setelah mengetahui log yang penting bagi Anda, Anda dapat mengidentifikasi SIEM mana yang telah membaca log data tersebut, yang dapat membaca log tersebut dengan sedikit konfigurasi, dan mana yang memerlukan agen.
- Kompleksitas penerapan. Karena SIEM dapat menyentuh ribuan sistem dalam suatu perusahaan, penerapannya umumnya merupakan tugas yang rumit. Penerapan kemungkinan akan memerlukan berbagai perubahan konfigurasi (misalnya, memperbarui kebijakan audit sistem, dan mengonfigurasi perangkat untuk mengirim log ke IP baru), beberapa di antaranya akan menjadi efek samping yang tidak terduga dari kerumitan lingkungan Anda. Untuk perusahaan

dengan lebih dari 1.000 sistem, penerapan penuh kemungkinan akan memakan waktu berbulan-bulan dan memerlukan koordinasi di berbagai tim.

- Kustomisasi. Vendor SIEM bersaing berdasarkan kedalaman fungsionalitas bawaan dan kemudahan penyesuaian. Beberapa SIEM hadir dengan antarmuka pengguna yang luas dan sejumlah sumber data utama yang bekerja langsung; yang lain hanya menawarkan fungsi dasar, tetapi mudah disesuaikan dengan skrip. Apa pun jenis SIEM yang Anda pilih, pastikan untuk memahami seberapa banyak fungsionalitas yang Anda perlukan yang ada di dalamnya atau mudah diperoleh, dan seberapa banyak Anda harus mengembangkan diri.
- Penyimpanan data. SIEM umumnya membutuhkan penyimpanan dalam jumlah besar, tetapi jumlah pastinya sangat bervariasi sesuai dengan arsitektur sistem dan aktivitas yang akan dipantau. File log yang mencantumkan peringatan IDS relatif jarang, sementara pengambilan paket penuh dapat menghasilkan gigabyte data baru per detik. SIEM sering kali mengelola 100 TB data atau lebih.
- Segregasi dan kontrol akses. Meskipun merupakan kemampuan keamanan yang penting, SIEM juga membawa risiko besar. Mengumpulkan semua data keamanan penting Anda dalam satu sistem berarti menaruh kepercayaan besar pada pengguna dan administrator sistem itu. Seperti yang dikatakan penyair Romawi Juvenal, “Siapa yang mengawasi para penjaga?” SIEM umumnya memiliki pemisahan yang kuat dan kemampuan kontrol akses berbasis peran yang memungkinkan administrator untuk membatasi akses pengguna ke data dan fungsionalitas, tetapi mengurangi risiko orang dalam yang ditimbulkan oleh personel keamanan adalah tantangan terus-menerus.
- Pemeliharaan penuh waktu. Karena mereka berinteraksi dengan begitu banyak sistem yang berbeda, SIEM pada dasarnya kompleks, jadi penerapan, pemeliharaan, dan penyesuaian adalah keterampilan ahli dalam diri mereka sendiri. Jika Anda menjalankan organisasi besar, bersiaplah untuk mengabdikan setidaknya satu anggota staf penuh waktu secara eksklusif untuk tugas-tugas tersebut.
- Pelatihan pengguna. Analis SOC umumnya terlatih dalam deteksi, investigasi, dan respons insiden, tetapi mereka mungkin tidak tahu cara menggunakan alat tertentu yang disebarkan di organisasi Anda. Bersiaplah untuk meminta setiap pengguna menghabiskan sekitar satu minggu pelatihan untuk belajar menggunakan SIEM, dan mengharapkan penurunan produktivitas sementara sambil belajar memigrasi kebiasaan lama dan alur kerja ke sistem baru.

Fungsi SOC seperti halnya pusat kendali lalu lintas udara atau ruang kendali reaktor nuklir: Sejumlah besar data terakumulasi dari berbagai sumber. Staf kontrol harus menggunakan pengalaman dan intuisi untuk memastikan bahwa sistem berjalan dengan baik, sehingga bantuan teknologi apa pun untuk mengatur dan mencerna data membantu staf menjadi lebih efektif. Selama sistem berjalan dengan baik, pemantauan sebagian besar pasif. Namun, ketika anomali terjadi, staf kontrol membutuhkan banyak data latar belakang untuk menentukan apa yang terjadi dan memutuskan apa yang harus dilakukan selanjutnya.

3.10 Kesimpulan

Dalam bab ini kita telah membahas banyak detail komunikasi dan keamanan jaringan. Beberapa materi telah diperluas pada topik sebelumnya (seperti intersepsi dan modifikasi) dalam konteks baru, sementara beberapa tidak seperti topik yang telah kami jelajahi sebelumnya (seperti penolakan layanan terdistribusi). Kami telah menjelajahi teknologi (komunikasi WiFi dan protokol keamanan, DNS, firewall, dan perangkat pendeteksi penyusupan) serta kebijakan dan praktik (manajemen jaringan). Keamanan jaringan sangat penting dalam bidang keamanan komputer yang lebih besar, tetapi ini dibangun di atas banyak blok bangunan yang telah kami buat (enkripsi, identifikasi dan otentikasi, kontrol akses, desain tangguh, hak istimewa paling rendah, kepercayaan, pemodelan ancaman). Jadi, meskipun bab ini telah menyajikan istilah dan konsep baru, sebagian besar materi mungkin tampak seperti perluasan dan perluasan yang wajar dari apa yang sudah Anda ketahui.

Bahan Evaluasi

1. Dalam bab ini kami telah menjelaskan nomor urut antara pengirim dan penerima sebagai cara untuk melindungi aliran komunikasi terhadap serangan substitusi dan replay. Jelaskan situasi di mana penyerang dapat mengganti atau memutar ulang terlepas dari nomor urut. Untuk jenis penomoran urut yang mana—satu aliran umum nomor urut atau aliran terpisah untuk setiap pasangan komunikator—serangan ini efektif?
2. Apakah mesin bensin memiliki satu titik kegagalan? Apakah mobil pemadam kebakaran bermotor? Apakah pemadam kebakaran? Bagaimana masing-masing dari dua yang terakhir mengkompensasi satu titik kegagalan di yang sebelumnya? Jelaskan jawaban Anda.
3. Penyedia dan pengguna jaringan telekomunikasi prihatin dengan satu titik kegagalan dalam “mil terakhir”, yaitu kabel tunggal dari stasiun switching terakhir penyedia jaringan ke tempat pelanggan. Bagaimana pelanggan dapat melindungi dari satu titik kegagalan itu? Beri komentar tentang apakah pendekatan Anda menghadirkan pertukaran biaya-manfaat yang baik.
4. Anda sedang merancang bisnis di mana Anda akan meng-host situs web perusahaan. Masalah apa yang dapat Anda lihat sebagai titik kegagalan tunggal? Daftar sumber daya yang dapat terlibat. Nyatakan cara untuk mengatasi setiap sumber daya yang menjadi satu titik kegagalan.
5. Tubuh manusia menunjukkan ketahanan yang luar biasa. Sebutkan tiga contoh di mana tubuh mengkompensasi kegagalan bagian tubuh tunggal.
6. Bagaimana perangkat keras dapat dirancang untuk toleransi kesalahan? Apakah metode ini berlaku untuk perangkat lunak? Mengapa atau mengapa tidak?
7. "Sakelar" telepon manusia kuno itu aneh tapi sangat lambat. Anda akan memberi isyarat kepada operator dan mengatakan Anda ingin berbicara dengan Jill, tetapi operator, mengetahui Jill mengunjungi Sally, akan menghubungkan Anda ke sana. Selain kelambatan atau inefisiensi, apa dua kelemahan lain dari skema ini?
8. Panggilan telepon (analog) adalah "berbasis sirkuit", yang berarti bahwa sistem memilih jalur kabel dari pengirim ke penerima dan jalur atau sirkuit tersebut didedikasikan untuk panggilan sampai selesai. Apa dua kerugian dari circuit switching?
9. Model OSI tidak efisien; setiap lapisan harus mengambil pekerjaan dari lapisan yang lebih tinggi, menambahkan beberapa hasil, dan meneruskan pekerjaan ke lapisan yang lebih rendah.

Proses ini berakhir dengan hadiah yang setara di dalam tujuh kotak bersarang, masing-masing dibungkus dan disegel. Tentunya pembungkus ini (dan membuka bungkusnya) tidak efisien. (Bukti kelambatan ini adalah bahwa protokol yang mengimplementasikan Internet—TCP, UDP, dan IP—diwakili oleh arsitektur empat lapis.) Dari membaca bab-bab awal buku ini, kutip keuntungan keamanan dari pendekatan berlapis.

10. Jelas, lapisan fisik harus berada di bagian bawah tumpukan OSI, dengan aplikasi di bagian atas. Membenarkan urutan lima lapisan lainnya sebagai bergerak dari abstraksi rendah ke tinggi.
11. Buat daftar masalah keamanan utama yang ditangani pada setiap tingkat tumpukan protokol OSI.
12. Keuntungan keamanan apa yang terjadi dari paket yang berisi NIC sumber? alamat dan bukan hanya alamat NIC tujuan?
13. TCP adalah protokol yang kuat: Pengurutan dan koreksi kesalahan dipastikan, tetapi ada penalti di overhead (misalnya, jika tidak diperlukan pengurutan ulang atau koreksi kesalahan). UDP tidak menyediakan layanan ini tetapi lebih sederhana. Sebutkan situasi spesifik di mana protokol UDP ringan dapat diterima, yaitu, ketika koreksi kesalahan atau pengurutan tidak diperlukan.
14. Asumsikan tidak ada protokol FTP. Anda diminta untuk mendefinisikan fungsi yang analog dengan FTP PUT untuk pertukaran file. Sebutkan tiga fitur atau mekanisme keamanan yang akan Anda sertakan dalam protokol Anda.
15. Skema pengalamatan IP 32-bit memberikan sekitar 4 miliar alamat.
    Bandingkan jumlah ini dengan populasi dunia. Setiap bit tambahan menggandakan jumlah alamat potensial. Meskipun 32 bit menjadi terlalu kecil, 128 bit tampaknya berlebihan, bahkan memungkinkan pertumbuhan yang signifikan. Tetapi tidak semua bit harus didedikasikan untuk menentukan alamat. Mengutip penggunaan keamanan untuk beberapa bit dalam sebuah alamat.
16. Saat domain baru dibuat, misalnya domainanda.com, Tabel di domain .com harus menerima entri untuk domainanda. Serangan keamanan apa yang mungkin dicoba seseorang terhadap pencatat .com (administrator Tabel .com) selama pembuatan domainanda.com?
17. Pemindai port adalah alat yang berguna bagi penyerang untuk mengidentifikasi kemungkinan kerentanan dalam sistem calon korban. Mengutip situasi di mana seseorang yang bukan penyerang dapat menggunakan pemindai port untuk tujuan yang tidak berbahaya.
18. Salah satu argumen dalam komunitas keamanan adalah bahwa kurangnya keragaman itu sendiri merupakan kerentanan. Misalnya, dua browser dominan, Mozilla Firefox dan Microsoft Internet Explorer, digunakan oleh sekitar 95 persen pengguna Internet. Risiko keamanan apa yang diperkenalkan oleh kontrol pasar ini? Misalkan ada tiga (masing-masing dengan pangsa pasar yang signifikan). Akankah tiga meniadakan risiko keamanan itu? Jika tidak, apakah empat? Lima? Jelaskan jawaban Anda.
19. Bandingkan kawat tembaga, microwave, serat optik, inframerah, dan nirkabel (frekuensi radio) dalam ketahanannya terhadap penyadapan pasif dan aktif.
20. Jelaskan mengapa router bawang mencegah node perantara mengetahui sumber dan tujuan komunikasi yang sebenarnya.
21. Perutean bawang tergantung pada node perantara. Apakah cukup hanya ada satu simpul perantara? Justifikasi jawaban Anda.
22. Misalkan node perantara untuk perutean bawang berbahaya, memperlihatkan sumber dan tujuan komunikasi yang diteruskannya. Jelas pengungkapan ini akan merusak kerahasiaan yang dirancang untuk dicapai oleh perutean bawang. Jika node jahat adalah salah satu dari dua di tengah, apa yang akan diekspos. Jika itu salah satu dari tiga, apa yang akan hilang. Jelaskan

jawaban Anda dalam hal node jahat di masing-masing posisi pertama, kedua, dan ketiga. Berapa banyak node tidak berbahaya yang diperlukan untuk menjaga privasi?

23. Masalah dengan pencocokan pola adalah sinonim. Jika direktori saat ini adalah bin, dan . menunjukkan direktori saat ini dan .. induknya, lalu bin, ../bin, ../bin/., ../../bin/../bin semua menunjukkan direktori yang sama. Jika Anda mencoba memblokir akses ke direktori bin dalam skrip perintah, Anda perlu mempertimbangkan semua varian ini (dan lebih banyak lagi). Sebutkan cara algoritma pencocokan pola mengatasi sinonim.
24. Protokol HTTP secara definisi stateless, artinya tidak memiliki mekanisme untuk "mengingat" data dari satu interaksi ke interaksi berikutnya. (a) Sarankan cara untuk mempertahankan status di antara dua panggilan HTTP. Misalnya, Anda dapat mengirimkan halaman buku dan harga yang cocok dengan kueri pengguna kepada pengguna, dan Anda ingin menghindari keharusan mencari harga setiap buku lagi setelah pengguna memilih satu untuk dibeli. (b) Menyarankan suatu cara yang dengannya Anda dapat mempertahankan beberapa gagasan tentang status antara dua akses web yang terpisah beberapa hari. Misalnya, pengguna dapat memilih harga yang dikutip dalam euro daripada dolar, dan Anda ingin menampilkan harga dalam mata uang pilihan lain kali tanpa meminta pengguna.
25. Bagaimana sebuah situs web dapat membedakan antara kurangnya kapasitas dan serangan penolakan layanan? Misalnya, situs web sering mengalami peningkatan volume lalu lintas yang luar biasa tepat setelah iklan yang menampilkan URL situs ditampilkan di televisi selama siaran acara olahraga populer. Lonjakan penggunaan itu adalah hasil dari akses normal yang terjadi secara bersamaan. Bagaimana sebuah situs dapat menentukan kapan lalu lintas tinggi masuk akal?
26. Syn flood adalah hasil dari beberapa pertukaran protokol yang tidak lengkap: Klien memulai pertukaran tetapi tidak menyelesaikannya. Sayangnya, situasi ini juga bisa terjadi secara normal. Jelaskan situasi jinak yang dapat menyebabkan pertukaran protokol menjadi tidak lengkap.
27. Serangan denial-of-service terdistribusi membutuhkan zombie yang berjalan di banyak mesin untuk melakukan bagian dari serangan secara bersamaan. Jika Anda adalah seorang administrator sistem yang mencari zombie di jaringan Anda, apa yang akan Anda cari?
28. Penandatanganan kode seluler adalah pendekatan yang disarankan untuk mengatasi kerentanan kode yang tidak bersahabat. Uraikan apa yang harus dilakukan oleh skema penandatanganan kode.
29. Sistem harus mengontrol akses applet ke sumber daya sistem yang sensitif, seperti sistem file, prosesor, jaringan, dan variabel status internal. Tetapi istilah "sistem file" sangat luas, dan applet yang berguna biasanya memerlukan penyimpanan yang persisten. Sarankan kontrol yang dapat ditempatkan pada akses ke sistem file. Jawaban Anda harus lebih spesifik daripada "izinkan semua membaca" atau "larang semua penulisan." Jawaban Anda pada dasarnya harus membedakan antara apa yang "penting keamanan" dan tidak, atau "berbahaya" dan tidak.
30. Misalkan Anda memiliki koneksi jaringan berkapasitas tinggi yang masuk ke rumah Anda dan Anda juga memiliki titik akses jaringan nirkabel. Juga misalkan Anda tidak menggunakan kapasitas penuh dari koneksi jaringan Anda. Sebutkan tiga alasan Anda mungkin masih ingin mencegah orang luar mendapatkan akses jaringan gratis dengan menyusup ke jaringan nirkabel Anda.
31. Mengapa segmentasi direkomendasikan untuk desain jaringan? Artinya, apa yang membuatnya lebih baik untuk memiliki segmen jaringan terpisah untuk server web, satu untuk pemrosesan backend office, satu untuk menguji kode baru, dan satu untuk manajemen sistem?

32. Untuk aplikasi besar, beberapa situs web menggunakan penyeimbang beban untuk mendistribusikan lalu lintas secara merata di antara beberapa server yang setara. Misalnya, mesin pencari mungkin memiliki database konten dan URL yang sangat besar, dan beberapa prosesor front-end yang merumuskan kueri ke manajer database dan memformat hasil untuk ditampilkan ke klien yang bertanya. Penyeimbang beban akan menetapkan setiap permintaan klien yang masuk ke prosesor front-end yang paling tidak sibuk. Apa keuntungan keamanan menggunakan penyeimbang beban?
33. Dapatkah tautan dan enkripsi ujung-ke-ujung keduanya digunakan pada komunikasi yang sama? Apa yang akan menjadi keuntungan dari itu? Mengutip situasi di mana kedua bentuk enkripsi mungkin diinginkan.
34. Apakah VPN menggunakan enkripsi tautan atau ujung ke ujung? Justifikasi jawaban Anda.
35. Mengapa firewall merupakan tempat yang baik untuk mengimplementasikan VPN? Mengapa tidak mengimplementasikannya di server yang sebenarnya sedang diakses?
36. Apakah VPN menggunakan enkripsi simetris atau asimetris? Jelaskan jawabanmu.
37. Apa tujuan keamanan untuk bidang, seperti nomor urut, dari paket IPsec?
38. Diskusikan trade-off antara sistem respons tantangan manual (di mana pengguna menghitung respons dengan tangan atau mental) dan sistem yang menggunakan perangkat khusus, seperti kalkulator.
39. Token sandi sinkron harus beroperasi pada kecepatan yang sama dengan penerima. Artinya, token harus maju ke nomor acak berikutnya pada saat yang sama penerima maju. Karena ketidaktepatan jam, kedua unit tidak akan selalu bersama dengan sempurna; misalnya, jam token mungkin berjalan 1 detik per hari lebih lambat dari jam penerima. Seiring waktu, perbedaan yang terakumulasi dapat menjadi signifikan. Sarankan sarana yang penerima dapat mendeteksi dan mengkompensasi penyimpangan jam pada bagian dari token.
40. ACL pada router memperlambat throughput sumber daya sistem yang banyak digunakan. Sebutkan dua keuntungan menggunakan ACL. Buat daftar situasi di mana Anda mungkin ingin memblokir (menolak) lalu lintas tertentu melalui ACL pada router; yaitu, situasi di mana hukuman kinerja tidak akan menjadi faktor penentu.
41. Informasi apa yang mungkin ingin diperiksa oleh firewall inspeksi stateful dari beberapa paket?
42. Ingat bahwa penataan ulang paket dan perakitan kembali terjadi pada tingkat transport dari suite protokol TCP/IP. Firewall akan beroperasi pada lapisan yang lebih rendah, baik Internet atau lapisan data. Bagaimana firewall inspeksi stateful dapat menentukan apa pun tentang aliran lalu lintas ketika aliran mungkin rusak atau rusak?
43. Apakah aturan firewall harus simetris? Artinya, apakah firewall harus memblokir jenis lalu lintas tertentu baik masuk (ke situs yang dilindungi) dan keluar (dari situs)? Mengapa atau mengapa tidak?
44. Protokol FTP relatif mudah untuk diproksi; firewall memutuskan, misalnya, apakah orang luar harus dapat mengakses direktori tertentu dalam sistem file dan mengeluarkan perintah yang sesuai ke pengelola file di dalam atau merespons negatif orang luar. Protokol lain tidak layak untuk diproksi. Sebutkan tiga protokol yang akan sangat sulit atau tidak mungkin untuk diproksi. Jelaskan jawabanmu.
45. Bagaimana isi log audit berbeda untuk router penyaringan versus firewall proxy aplikasi?
46. Sebutkan alasan mengapa sebuah organisasi mungkin menginginkan dua atau lebih firewall pada satu jaringan.

47. Firewall adalah target penetrator. Mengapa ada sedikit kompromi dari firewall?
48. Haruskah seorang administrator jaringan meletakkan firewall di depan honeypot (diperkenalkan di Bab 5)? Mengapa atau mengapa tidak?
49. Dapatkah firewall memblokir serangan yang menggunakan skrip server, seperti serangan di mana pengguna dapat mengubah harga item yang ditawarkan oleh situs e-commerce? Mengapa atau mengapa tidak?
50. Mengapa IDS mode siluman memerlukan jaringan terpisah untuk mengomunikasikan alarm dan menerima perintah manajemen?
51. Salah satu bentuk IDS mulai beroperasi dengan menghasilkan peringatan untuk setiap tindakan. Seiring waktu, administrator menyesuaikan pengaturan IDS sehingga aktivitas umum yang tidak berbahaya tidak menimbulkan alarm. Apa kelebihan dan kekurangan desain ini untuk IDS?
52. Dapatkah email terenkripsi memberikan verifikasi kepada pengirim bahwa penerima telah membaca pesan email? Mengapa atau mengapa tidak?
53. Dapatkah kerahasiaan pesan dan perlindungan integritas pesan diterapkan pada pesan yang sama? Mengapa atau mengapa tidak?
54. Apa keuntungan dan kerugian dari program email (seperti Eudora atau Outlook) yang secara otomatis menerapkan dan menghapus perlindungan pesan email antara pengirim dan penerima?

Bab 4
Database

Bab ini akan membahas tentang:

- Istilah dan konsep basis data
- Persyaratan keamanan: C-I-A ; keandalan, jenis integritas
- Kontrol akses; data sensitif, pengungkapan, inferensi, agregasi
- Data Mining dan Big Data

Melindungi data adalah inti dari sistem yang aman, dan banyak pengguna (orang, program, atau sistem) mengandalkan sistem *Database Management System* (DBMS) untuk mengelola perlindungan data terstruktur. Untuk alasan ini, kami mencurahkan bab ini untuk keamanan database dan sistem manajemen database, sebagai contoh bagaimana keamanan aplikasi dapat dirancang dan diimplementasikan untuk tugas tertentu.

Basis data sangat penting bagi banyak organisasi bisnis dan pemerintah, yang menyimpan data yang mencerminkan aktivitas inti organisasi. Seringkali, ketika proses bisnis direkayasa ulang untuk membuatnya lebih efektif dan lebih selaras dengan tujuan baru atau yang direvisi, salah satu sistem pertama yang menerima pengawasan cermat adalah kumpulan database yang mendukung proses bisnis. Dengan demikian, basis data lebih dari sekadar repositori yang berhubungan dengan perangkat lunak. Organisasi dan isinya dianggap sebagai aset perusahaan yang berharga yang harus dilindungi dengan hati-hati.

Namun, perlindungan yang diberikan oleh sistem manajemen basis data memiliki hasil yang beragam. Seiring waktu, kami telah meningkatkan pemahaman kami tentang masalah keamanan basis data, dan beberapa kontrol yang baik telah dikembangkan. Tetapi masih ada lebih banyak masalah keamanan yang tidak tersedia kontrolnya.

Kami memulai bab ini dengan ringkasan singkat tentang terminologi database. Kami kemudian mempertimbangkan persyaratan keamanan untuk sistem manajemen basis data. Dua masalah keamanan utama—integritas dan kerahasiaan—dijelaskan dalam konteks database. Kami melanjutkan bab ini dengan mempelajari dua masalah keamanan database yang penting (tetapi terkait), masalah inferensi dan masalah bertingkat. Kedua masalah itu kompleks, dan tidak ada solusi langsung. Namun, dengan memahami masalah, kami menjadi lebih sensitif terhadap cara mengurangi potensi ancaman terhadap data. Akhirnya, kami menyimpulkan bab ini dengan melihat pengumpulan big data dan data mining, sebuah teknologi untuk menurunkan pola dari satu atau lebih database. Data mining melibatkan banyak masalah keamanan yang kami angkat dalam bab ini.

4.1 Pengenalan Database

Basis Data adalah kumpulan informasi yang disimpan dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari database. Software yang digunakan untuk mengelola dan permintaan panggilan (query) basis data yang disebut sistem manajemen database (database management system, DBMS).

Kita mulai dengan mendeskripsikan database dan mendefinisikan terminologi yang terkait dengan penggunaannya. Kami mengambil contoh dari apa yang disebut database relasional karena ini adalah salah satu tipe yang paling banyak digunakan. Namun, konsep yang dijelaskan di sini berlaku untuk semua jenis database. Kami pertama-tama mendefinisikan konsep dasar dan kemudian menggunakannya untuk mendiskusikan masalah keamanan.

Basis Data memiliki gambaran terstruktur dari jenis fakta yang tersimpan di dalamnya: penjelasan ini disebut skema. Benda Skema menggambarkan yang mewakili database, dan hubungan antara objek-objek tersebut. Ada banyak bagaimana mengatur skema, atau memodelkan struktur database: ini dikenal sebagai model basis data atau model data.

4.1.1 Konsep Database

Konsep dasar dari basis data ialah kumpulan dari sebuah catatan atau sebuah potongan dari pengetahuan. Sebuah basis data memiliki penjelasan testruktur dari jenis fakta yang tersimpan di dalamnya, penjelasan tersebut dengan skema. Skema menggambarkan sebuah objek yang diwakili suatu basis data dan memiliki hubungan diantara objek tersebut. Ada banyak cara untuk mengorganisasi skema atau memodelkan struktur basis data, ini dikenal sebagai model basis data atau model data. Biasanya model yang umum digunakan sekarang ialah model relasional yang istilahnya layman mewakili semua informasi dalam bentuk table-Tabel yang saling berhubungan dimana setiap table terdiri dari baris dan kolom (definisi yang sebenarnya menggunakan terminologi matematika). Dalam model ini hubungan antar table diwakili dengan menggunakan nilai yang sama antar table. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar table.

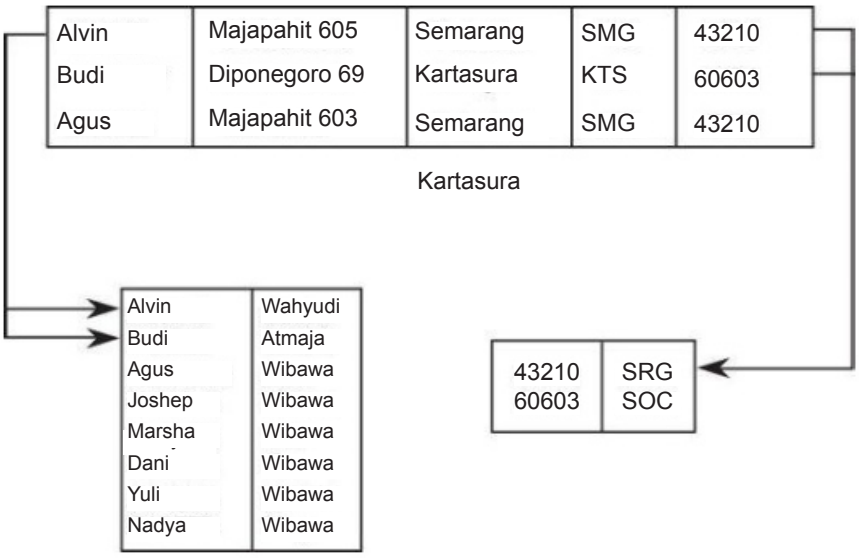
Database dapat dibuat dan diolah dengan menggunakan suatu program komputer, yaitu yang biasa kita sebut dengan software (perangkat lunak). Software yang digunakan untuk mengelola dan memanggil kueri (query) database disebut Database Management System (DBMS) atau jika diterjemahkan kedalam bahasa indonesia berarti “Sistem Manajemen Basis Data”.

DBMS terdiri dari dua komponen, yaitu Relational Database Management System (RDBMS) dan Overview of Database Management System (ODBMS). RDBMS meliputi Interface Drivers, SQL Engine, Transaction Engine, Relational Engine, dan Storage Engine. Sedangkan ODBMS meliputi Language Drivers, Query Engine, Transaction Engine, dan Storage Engine.

4.1.2 Komponen Database

File database terdiri dari record, yang masing-masing berisi satu kelompok data terkait. Seperti ditunjukkan pada contoh di Tabel 4-1, sebuah record dalam file nama dan alamat terdiri dari satu nama dan alamat. Setiap catatan berisi bidang atau elemen, item data dasar itu sendiri. Bidang dalam catatan nama dan alamat adalah NAMA, ALAMAT, KOTA, NEGARA, dan KODE POS. Basis data ini dapat dilihat sebagai Tabel dua dimensi, di mana catatan adalah baris dan setiap bidang catatan adalah elemen Tabel .

Tidak setiap database mudah direpresentasikan sebagai Tabel tunggal yang ringkas. Basis data pada Gambar 4-1 secara logis terdiri dari tiga file dengan kemungkinan kegunaan yang berbeda. Ketiga file ini dapat direpresentasikan sebagai satu Tabel besar, tetapi penggambaran itu mungkin tidak meningkatkan utilitas atau akses ke data.



Gambar 4-1 Database Beberapa Tabel Terkait

Struktur logis dari database disebut skema. Pengguna tertentu mungkin hanya memiliki akses ke sebagian database, yang disebut subskema. Skema keseluruhan database pada Gambar 4-1 dirinci dalam Tabel 4-2. Tiga blok terpisah dari Gambar adalah contoh subskema, meskipun subskema lain dari database ini dapat didefinisikan. Kita dapat menggunakan skema dan subskema untuk menyajikan kepada pengguna hanya elemen-elemen yang mereka ingin atau perlu mereka lihat.

Misalnya, jika Tabel 4-1 mewakili karyawan di sebuah perusahaan, subskema di kiri bawah dapat mencantumkan nama karyawan tanpa mengungkapkan informasi pribadi seperti alamat rumah.

Tabel 4-1 Contoh Database

Alvin	Majapahit 605	Semarang	SMG	43210
Dias	Diponegoro 69	Kartasura	KTS	60603
Agus	Siliwangi 04	Semarang	SMG	43210

Aturan database mengidentifikasi kolom dengan nama. Nama setiap kolom disebut atribut database. Relasi adalah sekumpulan kolom. Sebagai contoh, dengan menggunakan database pada Tabel 4-2, kita melihat bahwa NAMA–KODE POS adalah relasi yang dibentuk dengan mengambil kolom NAMA dan KODE POS, seperti yang ditunjukkan pada Tabel 4-3. Relasi menentukan kelompok nilai data terkait, dengan cara yang hampir sama seperti relasi "ibu dari" menentukan hubungan di antara pasangan manusia. Dalam contoh ini, setiap cluster berisi sepasang elemen, NAMA dan KODE POS. Relasi lain dapat memiliki lebih banyak kolom, sehingga setiap klaster dapat berupa triple, 4-tuple, atau n-tuple (untuk beberapa nilai n) elemen.

Tabel 4-2 Skema Database dari Gambar 4-1

Nama	Lengkap	Alamat	Kota	Kode Kota	Kode Pos	Bandara
Alvin	Wahyudi	Majapahit 605	Semarang	SMG	43210	SRG
Budi	Wahyudi	Majapahit 605	Semarang	KTS	43210	SRG
Dias	Angga	Diponegoro 69	Kartasura	SMG	60603	SOC
Agus	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG
Marsha	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG
Dani	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG
Yuli	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG
Nadya	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG

Tabel 4-3 Relasi dalam Database

Nama	Kode Pos
Alvin	43210
Dias	60603
Agus	43210

Kueri

Pengguna berinteraksi dengan manajer database melalui perintah ke DBMS yang mengambil, memodifikasi, menambah, atau menghapus bidang dan catatan database. Perintah disebut query. Sistem manajemen basis data memiliki aturan sintaksis yang tepat untuk kueri. Sebagian besar bahasa kueri menggunakan notasi seperti bahasa Inggris, dan banyak yang didasarkan pada SQL, bahasa kueri terstruktur yang awalnya dikembangkan oleh IBM. Kami telah menulis pertanyaan

contoh dalam bab ini menyerupai kalimat bahasa Inggris sehingga mudah dimengerti. Misalnya, pertanyaan

PILIH NAMA = 'WAHYUDI'

mengambil semua catatan yang memiliki nilai WAHYUDI di bidang NAMA.

Hasil dari mengeksekusi query adalah subskema. Salah satu cara untuk membentuk subskema database adalah dengan memilih record yang memenuhi kondisi tertentu. Sebagai contoh, kita mungkin memilih record di mana KODE POS=43210, menghasilkan hasil yang ditunjukkan pada Tabel 4-4.

Tabel 4-4 Hasil dari Kueri Pemilihan

Nama	Lengkap	Alamat	Kota	Kode Kota	Kode Pos	Bandara
Alvin	Wahyudi	Majapahit 605	Semarang	SMG	43210	SRG
Budi	Wahyudi	Majapahit 605	Semarang	KTS	43210	SRG
Agus	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG
Marsha	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG
Dani	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG
Yuli	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG
Nadya	Wibawa	Siliwangi 04	Semarang	SMG	43210	SRG

Kriteria seleksi lainnya yang lebih kompleks dimungkinkan, dengan operator logika seperti dan(^) dan atau (v), dan perbandingan seperti kurang dari (<). Contoh kueri pemilihan adalah

```
PILIH (ZIP='43210') (NAMA='WAHYUDI')
```

Setelah memilih catatan, kami dapat memproyeksikan catatan ini ke satu atau lebih atribut. Operasi pilih mengidentifikasi baris tertentu dari database, dan operasi proyek mengekstrak nilai dari bidang (kolom) tertentu dari catatan tersebut. Hasil dari operasi proyek pilih adalah kumpulan nilai atribut yang ditentukan untuk catatan yang dipilih. Sebagai contoh, kita dapat memilih record yang memenuhi kondisi KODE POS=43210 dan memproyeksikan hasilnya ke atribut NAME dan LENGKAP, seperti pada Tabel 4-5. Hasilnya adalah daftar nama depan dan belakang orang yang alamatnya memiliki kode pos 43210.

Tabel 4-5 Hasil dari Select–Project Query

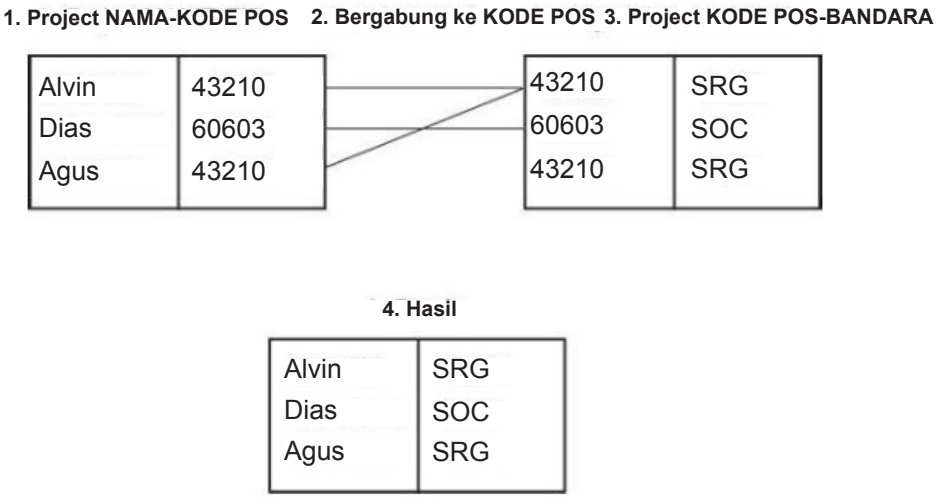
Alvin	Wahyudi
Budi	Wahyudi
Agus	Wibawa
Marsha	Wibawa
Yuli	Wibawa
Dani	Wibawa
Nadya	Wibawa

Perhatikan bahwa kita tidak harus memproyeksikan ke atribut yang sama di mana pemilihan dilakukan. Misalnya, kita dapat membuat kueri menggunakan ZIP dan NAMA tetapi memproyeksikan hasilnya ke PERTAMA:

```
TAMPILKAN PERTAMA DI MANA (ZIP='43210') (NAMA='WAHYUDI')
```

Hasilnya adalah daftar nama depan orang yang nama belakangnya adalah WAHYUDI dan KODE POS 43210.

Kami juga dapat menggabungkan dua subskema pada elemen umum dengan menggunakan kueri gabungan. Hasil dari operasi ini adalah subskema yang recordnya memiliki nilai yang sama untuk elemen yang sama. Sebagai contoh, Gambar 4-2 menunjukkan bahwa subskema NAMA–KODEP POS dan subskema ZIP–BANDARA dapat digabungkan pada bidang umum KODE POS untuk menghasilkan subskema NAMA–BANDARA.



Gambar 4-2 Hasil dari Select–Project–Join Query

4.1.3 Keuntungan Menggunakan Database

Ide logis di balik database adalah: Database adalah kumpulan data tunggal, disimpan dan dipelihara di satu lokasi pusat, yang dapat diakses oleh banyak orang sesuai kebutuhan. Namun, implementasi sebenarnya mungkin melibatkan beberapa pengaturan atau akses penyimpanan fisik lainnya. Inti dari database yang baik adalah bahwa pengguna tidak menyadari pengaturan fisik; pengaturan logis terpadu adalah semua yang mereka lihat. Akibatnya, database menawarkan banyak keuntungan dibandingkan sistem file sederhana:

- akses bersama, sehingga banyak pengguna dapat menggunakan satu kumpulan data yang umum dan terpusat
- akses terkontrol, sehingga hanya pengguna yang berwenang yang diizinkan untuk melihat atau mengubah nilai data
- redundansi minimal, sehingga pengguna individu tidak perlu mengumpulkan dan memelihara kumpulan data mereka sendiri
- konsistensi data, sehingga perubahan nilai data mempengaruhi semua pengguna nilai data
- integritas data, sehingga nilai data terlindungi dari perubahan yang tidak disengaja atau berbahaya

Anda akan melihat banyak konsep keamanan yang familiar dalam daftar ini. Meskipun keamanan hanyalah salah satu dari beberapa motivasi untuk menggunakan database, beberapa pengguna menghargai memiliki tingkat akses yang aman.

Sebuah DBMS dirancang untuk memberikan keuntungan ini secara efisien. Namun, seperti yang sering terjadi, tujuan dapat saling bertentangan. Secara khusus, seperti yang akan kita lihat, kepentingan keamanan dapat bertentangan dengan kinerja. Bentrokan ini tidak mengherankan, karena tindakan yang diambil untuk menegakkan keamanan sering kali meningkatkan ukuran atau kompleksitas sistem komputasi. Namun, yang mengejutkan adalah bahwa kepentingan keamanan juga dapat mengurangi kemampuan sistem untuk menyediakan data kepada pengguna dengan membatasi kueri tertentu yang tampaknya tidak berbahaya.

4.2 Persyaratan Keamanan Basis Data

Persyaratan keamanan dasar sistem database tidak berbeda dengan sistem komputasi lain yang telah kita pelajari. Masalah dasar—kontrol akses, pengecualian data palsu, otentikasi pengguna, dan keandalan—sejauh ini telah muncul dalam banyak konteks dalam buku ini. Berikut ini adalah daftar persyaratan untuk keamanan database.

- Integritas database fisik. Data database kebal dari masalah fisik, seperti listrik padam, dan seseorang dapat merekonstruksi database jika hancur karena bencana.
- Integritas basis data logis. Struktur database dipertahankan. Dengan integritas logis database, misalnya, modifikasi nilai satu bidang tidak memengaruhi bidang lain.
- Integritas elemen. Data yang terkandung dalam setiap elemen akurat.
- Kemampuan audit. Dimungkinkan untuk melacak siapa atau apa yang telah mengakses (atau memodifikasi) elemen dalam database.
- Kontrol akses. Seorang pengguna hanya diperbolehkan untuk mengakses data resmi, dan pengguna yang berbeda dapat dibatasi untuk mode akses yang berbeda (seperti membaca atau menulis).
- Otentikasi pengguna. Setiap pengguna diidentifikasi secara positif, baik untuk jejak audit maupun untuk izin mengakses data tertentu.
- Ketersediaan. Pengguna dapat mengakses database secara umum dan semua data yang mereka otorisasi.

Kami secara singkat memeriksa masing-masing persyaratan ini.

Integritas Basis Data

Jika database berfungsi sebagai pusat penyimpanan data, pengguna harus dapat mempercayai keakuratan nilai data. Kondisi ini menyiratkan bahwa administrator database harus yakin bahwa pembaruan dilakukan hanya oleh individu yang berwenang. Ini juga menyiratkan bahwa data harus dilindungi dari korupsi, baik oleh tindakan program ilegal dari luar atau oleh kekuatan luar seperti kebakaran atau kegagalan daya. Dua situasi dapat mempengaruhi integritas database: ketika seluruh database rusak (seperti yang terjadi, misalnya, jika media penyimpanannya rusak) atau ketika item data individual tidak dapat dibaca.

Integritas database secara keseluruhan adalah tanggung jawab DBMS, sistem operasi, dan manajer sistem komputasi (manusia). Dari perspektif sistem operasi dan manajer sistem komputasi, database dan DBMS adalah file dan program, masing-masing. Oleh karena itu, salah satu cara untuk melindungi database secara keseluruhan adalah dengan membuat cadangan semua file di sistem secara berkala. Pencadangan berkala ini dapat menjadi kontrol yang memadai terhadap kegagalan katastropik.

Terkadang seorang administrator perlu dapat merekonstruksi database pada titik kegagalan. Misalnya, ketika listrik padam tiba-tiba, klien bank mungkin sedang melakukan transaksi atau siswa mungkin mendaftar online untuk kelas mereka. Dalam kasus ini, pemilik ingin dapat memulihkan sistem ke titik stabil tanpa memaksa pengguna untuk mengulang transaksi yang baru saja selesai.

Untuk menangani situasi ini, DBMS harus memelihara log transaksi. Sebagai contoh, misalkan sistem perbankan dirancang agar pesan dibuat dalam log (elektronik atau kertas atau keduanya) setiap kali transaksi diproses. Jika terjadi kegagalan sistem, sistem dapat memperoleh saldo akun yang akurat dengan mengembalikan ke salinan cadangan database dan memproses ulang semua transaksi selanjutnya dari log.

Integritas Elemen

Integritas elemen database adalah kebenaran atau akurasinya. Pada akhirnya, pengguna yang berwenang bertanggung jawab untuk memasukkan data yang benar dalam database. Namun, pengguna dan program membuat kesalahan dalam mengumpulkan data, menghitung hasil, dan memasukkan nilai. Oleh karena itu, DBMS terkadang mengambil tindakan khusus untuk membantu menangkap kesalahan saat dibuat dan untuk memperbaiki kesalahan setelah dimasukkan.

Basis data mencapai integritas basis data, strukturnya, dan elemen individualnya.
Tindakan korektif ini dapat dilakukan dengan tiga cara: dengan pemeriksaan lapangan, melalui kontrol akses, dan dengan log perubahan.

Pertama, DBMS dapat menerapkan pemeriksaan lapangan, aktivitas yang menguji nilai yang sesuai dalam suatu posisi. Bidang mungkin diperlukan berupa angka, huruf besar, atau salah satu dari serangkaian karakter yang dapat diterima. Pemeriksaan memastikan bahwa nilai berada dalam batas yang ditentukan atau tidak lebih besar dari jumlah nilai dalam dua bidang lainnya. Pemeriksaan ini mencegah kesalahan sederhana saat data dimasukkan. (Studi Kasus 4-1 menunjukkan pentingnya integritas elemen.)

Studi Kasus 4-1 : Kegagalan Integritas Elemen Menghancurkan Jaringan

Crocker dan Bernstein [CRO89] mempelajari kegagalan bencana dari apa yang kemudian dikenal sebagai ARPANET, pendahulu Internet saat ini. Beberapa kegagalan datang dari masalah dengan Tabel routing yang digunakan untuk mengarahkan lalu lintas melalui jaringan.

Sebuah kesalahan tahun 1971 disebut "lubang hitam." Kegagalan perangkat keras menyebabkan satu node menyatakan bahwa itu adalah jalur terbaik ke setiap node

lain dalam jaringan. Node ini mengirimkan deklarasi ini ke node lain, yang segera menyebarkan posting yang salah ke seluruh jaringan. Node ini segera menjadi lubang hitam jaringan karena semua lalu lintas diarahkan ke sana tetapi tidak pernah sampai ke tujuan sebenarnya.

ARPANET menggunakan Tabel sederhana, bukan sistem manajemen basis data berfitur lengkap, jadi tidak ada pemeriksaan nilai baru sebelum dipasang di Tabel perutean terdistribusi. Seandainya ada database, perangkat lunak pemeriksa integritas dapat memeriksa kesalahan dalam nilai yang baru didistribusikan dan menaikkan tanda untuk ditinjau oleh manusia.

Tindakan integritas kedua diberikan oleh kontrol akses. Untuk mengetahui alasannya, pertimbangkan hidup tanpa basis data. File data mungkin berisi data dari beberapa sumber, dan data yang berlebihan dapat disimpan di beberapa tempat berbeda. Misalnya, alamat surat siswa dapat disimpan di banyak file kampus yang berbeda: di kantor registrar untuk korespondensi formal, di kantor layanan makanan untuk hak istimewa ruang makan, di toko buku untuk pembelian, dan di kantor bantuan keuangan untuk akuntansi. Memang, siswa mungkin tidak menyadari bahwa setiap kantor yang terpisah memiliki alamat yang tercatat. Jika siswa berpindah dari satu tempat tinggal ke tempat tinggal lainnya, masing-masing file terpisah memerlukan koreksi.

Tanpa database, Anda dapat membayangkan risiko terhadap integritas data. Pertama, pada waktu tertentu, beberapa file data dapat menunjukkan alamat lama (belum diperbarui) dan beberapa secara bersamaan memiliki alamat baru (sudah diperbarui). Kedua, selalu ada kemungkinan bahwa seseorang salah memasukkan bidang data, lagi-lagi mengarah ke file dengan informasi yang salah. Ketiga, siswa bahkan mungkin tidak mengetahui beberapa file, sehingga dia tidak tahu untuk memberi tahu pemilik file tentang memperbarui informasi alamat. Masalah-masalah ini diselesaikan oleh database. Mereka memungkinkan pengumpulan dan kontrol data ini di satu sumber pusat, memastikan siswa dan pengguna memiliki alamat yang benar.

Namun, sentralisasi lebih mudah diucapkan daripada dilakukan. Siapa yang memiliki file pusat bersama ini? Siapa yang berwenang memperbarui elemen mana? Bagaimana jika dua orang menerapkan modifikasi yang saling bertentangan? Bagaimana jika modifikasi diterapkan dari urutan? Bagaimana catatan duplikat terdeteksi? Tindakan apa yang diambil ketika duplikat ditemukan? Ini adalah pertanyaan kebijakan yang harus diselesaikan oleh administrator basis data.Studi Kasus 4-2 menjelaskan bagaimana masalah ini ditujukan untuk mengelola konfigurasi program; Proses formal yang serupa diperlukan untuk mengelola perubahan database.

Insinyur perangkat lunak harus menangani kontrol akses ketika mereka mengelola konfigurasi sistem komputer besar. Kode sistem utama dan perubahan kode dari waktu ke waktu sebenarnya adalah database. Dalam banyak kasus, banyak pemrogram membuat perubahan pada sistem pada saat yang bersamaan; database manajemen konfigurasi harus membantu memastikan bahwa perubahan yang benar dan terbaru disimpan.

Proliferasi versi dan rilis dapat dikontrol dalam tiga cara utama.

- File terpisah: File terpisah dapat disimpan untuk setiap versi atau rilis yang berbeda. Misalnya, versi 1 mungkin ada untuk mesin yang menyimpan semua data di memori utama, dan versi 2 adalah untuk mesin yang harus mengeluarkan beberapa data ke disk. Misalkan fungsi umum adalah sama di kedua versi, berada di komponen C1 sampai Ck, tetapi manajemen memori dilakukan oleh komponen M1 untuk versi 1 dan M2 untuk versi 2. Jika fungsi baru akan ditambahkan ke rutinitas manajemen memori, pertahankan keduanya versi saat ini dan yang benar mungkin sulit; hasilnya harus sama dari sudut pandang pengguna.
- Delta: Satu versi sistem dianggap sebagai versi utama, dan semua versi lainnya dianggap sebagai variasi dari versi utama. Basis data hanya melacak perbedaan, dalam file yang disebut file delta. Delta berisi perintah yang "diterapkan" ke versi utama untuk mengubahnya menjadi versi alternatif. Pendekatan ini menghemat ruang penyimpanan tetapi bisa menjadi berat.
- Kompilasi bersyarat: Semua versi ditangani oleh satu file, dan pernyataan bersyarat digunakan untuk menentukan pernyataan mana yang berlaku dalam kondisi mana. Dalam hal ini, kode yang dibagikan hanya muncul sekali, jadi hanya satu koreksi yang diperlukan jika ditemukan masalah. Tetapi kode dalam satu file ini bisa sangat kompleks dan sulit dipelihara.

Dalam salah satu dari tiga kasus ini, akses terkontrol ke file konfigurasi sangat penting. Dua programmer yang memperbaiki masalah yang berbeda terkadang perlu membuat perubahan pada komponen yang sama. Jika mereka tidak mengoordinasikan akses, pemrogram kedua dapat secara tidak sengaja membatalkan (atau lebih buruk lagi, merusak) perubahan pemrogram pertama, yang mengakibatkan tidak hanya terulangnya masalah awal tetapi juga munculnya masalah tambahan. Untuk alasan ini, file dikendalikan dalam beberapa cara, termasuk dikunci saat perubahan dilakukan oleh satu programmer, dan tunduk pada sekelompok orang yang disebut papan kontrol konfigurasi yang memastikan bahwa tidak ada file yang diubah dimasukkan kembali ke dalam produksi tanpa pemeriksaan yang tepat. dan pengujian. Shari Lawrence Pfleeger dan Joanne Atlee menulis tentang teknik ini.

Cara ketiga untuk menyediakan integritas basis data adalah memelihara log perubahan untuk basis data. Log perubahan mencantumkan setiap perubahan yang dilakukan pada database; itu berisi nilai asli dan nilai yang dimodifikasi. Dengan menggunakan log ini, administrator database dapat membatalkan perubahan yang dibuat karena kesalahan. Misalnya, denda perpustakaan mungkin keliru diposting terhadap Charles W. Robertson, bukan Charles M. Robertson, menandai Charles W. Robertson sebagai tidak memenuhi syarat untuk berpartisipasi dalam atletik universitas. Setelah menemukan kesalahan ini, administrator database memperoleh nilai kelayakan asli Charles W. dari log dan mengoreksi database.

Kemampuan Audit

Untuk beberapa aplikasi, administrator mungkin ingin membuat catatan audit dari semua akses (baca atau tulis) ke database. Catatan semacam itu dapat membantu menjaga integritas basis data, atau setidaknya untuk menemukan setelah fakta siapa yang memengaruhi nilai apa dan kapan. Keuntungan kedua, seperti yang akan kita lihat nanti, adalah bahwa pengguna dapat mengakses data yang dilindungi secara bertahap; yaitu, tidak ada akses tunggal mengungkapkan data yang dilindungi, tetapi satu set akses berurutan dilihat bersama-sama mengungkapkan data, seperti menemukan petunjuk dalam novel detektif. Dalam hal ini, jejak audit dapat mengidentifikasi petunjuk mana yang telah diberikan kepada pengguna, sebagai panduan apakah akan memberi tahu pengguna lebih banyak.

Seperti yang kita catat di Bab 2, granularity dapat menjadi hambatan dalam audit. Peristiwa yang diaudit dalam sistem operasi adalah tindakan seperti file terbuka atau prosedur panggilan; mereka jarang spesifik seperti menulis record 3 atau mengeksekusi instruksi I. Agar berguna untuk menjaga integritas, jejak audit database harus menyertakan akses pada record, field, dan bahkan level elemen. Detail ini menjadi penghalang untuk sebagian besar aplikasi database.

Selanjutnya, sistem manajemen basis data dapat mengakses catatan tetapi tidak melaporkan data kepada pengguna, seperti ketika pengguna melakukan operasi pilih. Misalnya, penasihat asrama mungkin ingin menghitung semua siswa yang gagal bahasa Prancis dasar, dan sistem manajemen basis data melaporkan 462. Untuk mendapatkan nomor itu, sistem harus memeriksa semua catatan siswa dan mencatat mereka yang nilainya gagal, dan itu dilakukan pencarian ini atas nama penasihat yang terdaftar dengan tepat di log sebagai penerima data. Jadi, dalam arti tertentu, pembimbing mengakses semua nilai siswa tersebut, meskipun dari angka 462 pembimbing tidak dapat menentukan nilai setiap siswa. (Mengakses catatan atau elemen tanpa mentransfer ke pengguna data yang diterima disebut masalah pass-through.) Dengan demikian, log dari semua catatan yang diakses secara langsung dapat melebih-lebihkan dan mengecilkan apa yang sebenarnya dipelajari pengguna. Masalahnya bahkan lebih bernuansa daripada apa yang kami jelaskan di sini, dan kami mempertimbangkan beberapa seluk-beluk pengungkapan nanti dalam bab ini.

Kontrol Akses

Database sering dipisahkan secara logis oleh hak akses pengguna. Misalnya, semua pengguna dapat diberikan akses ke data umum, tetapi hanya departemen personalia yang dapat memperoleh data gaji dan hanya departemen pemasaran yang dapat memperoleh data penjualan. Basis data berguna karena memusatkan penyimpanan dan pemeliharaan data. Akses yang terbatas merupakan tanggung jawab dan keuntungan dari sentralisasi ini.

Administrator database menentukan siapa yang harus diizinkan mengakses data mana, pada tampilan, relasi, bidang, catatan, atau bahkan tingkat elemen. DBMS harus menegakkan kebijakan ini, memberikan akses ke semua data tertentu atau tidak ada akses jika dilarang. Apalagi jumlah moda aksesnya bisa banyak. Pengguna atau program mungkin memiliki hak untuk membaca, mengubah, menghapus, atau menambahkan nilai, menambah atau menghapus seluruh bidang atau catatan, atau mengatur ulang seluruh database.

Secara dangkal, kontrol akses untuk database tampak seperti kontrol akses untuk sistem operasi atau komponen lain dari sistem komputasi. Namun, masalah database lebih rumit, seperti yang kita lihat di seluruh bab ini. Objek sistem operasi, seperti file, adalah item yang tidak terkait, sedangkan catatan, bidang, dan elemen terkait. Meskipun pengguna mungkin tidak dapat menentukan isi satu file dengan membaca yang lain, pengguna mungkin dapat menentukan satu elemen data hanya dengan membaca yang lain. Masalah memperoleh nilai data dari orang lain disebut inferensi, dan kami akan membahasnya secara mendalam nanti dalam bab ini.

Penting untuk diperhatikan bahwa Anda dapat mengakses data dengan inferensi tanpa memerlukan akses langsung ke objek aman itu sendiri. Membatasi inferensi dapat berarti melarang jalur tertentu untuk mencegah kemungkinan inferensi. Namun, membatasi akses untuk mengontrol inferensi juga membatasi kueri dari pengguna yang tidak bermaksud mengakses nilai secara tidak sah. Selain itu, upaya untuk memeriksa akses yang diminta untuk kemungkinan inferensi yang tidak dapat diterima sebenarnya dapat menurunkan kinerja DBMS.

Akhirnya, ukuran atau granularity berbeda antara objek sistem operasi dan objek database. Sebuah sistem operasi dapat dengan mudah mengontrol akses ke file, seperti yang kami jelaskan di Bab 5. Namun daftar kontrol akses dari beberapa ratus file jauh lebih mudah diimplementasikan daripada daftar kontrol akses untuk database dengan beberapa ratus file yang masing-masing mungkin seratus bidang. Ukuran mempengaruhi efisiensi pemrosesan. Sistem operasi biasanya tidak "melihat ke dalam" file untuk mengontrol akses ke item di dalam file.

Otentikasi Pengguna

DBMS dapat memerlukan otentikasi pengguna yang ketat. Misalnya, DBMS mungkin bersikeras bahwa pengguna melewati kata sandi dan pemeriksaan waktu tertentu. Otentikasi ini melengkapi otentikasi yang dilakukan oleh sistem operasi. Biasanya, DBMS berjalan sebagai program aplikasi di atas sistem operasi. Perancangan sistem ini berarti tidak ada jalur terpercaya dari DBMS ke sistem operasi, sehingga DBMS harus curiga terhadap setiap data yang diterimanya, termasuk identitas pengguna dari sistem operasi. Dengan demikian, DBMS dipaksa untuk melakukan otentikasi sendiri.

Ketersediaan

Sebuah DBMS memiliki aspek dari kedua program dan sistem. Ini adalah program yang menggunakan sumber daya perangkat keras dan perangkat lunak lain, namun bagi banyak pengguna itu adalah satu-satunya aplikasi yang dijalankan. Pengguna sering menganggap DBMS begitu saja, menggunakannya sebagai alat penting untuk melakukan tugas-tugas tertentu. Tetapi ketika sistem tidak tersedia — sibuk melayani pengguna lain atau down untuk diperbaiki atau ditingkatkan — pengguna sangat menyadari ketidaktersediaan DBMS. Misalnya, dua pengguna dapat meminta catatan yang sama,

dan DBMS harus melakukan arbitrase; satu pengguna pasti akan ditolak aksesnya untuk sementara waktu. Atau DBMS dapat menahan data yang tidak dilindungi untuk menghindari pengungkapan data yang dilindungi, membuat pengguna yang meminta tidak senang. Kami memeriksa masalah ini secara lebih rinci nanti dalam bab ini. Masalah seperti ini menghasilkan persyaratan ketersediaan tinggi untuk DBMS.

Integritas/Kerahasiaan/Ketersediaan

Tiga aspek keamanan komputer—integritas, kerahasiaan, dan ketersediaan—jelas berhubungan dengan sistem manajemen basis data. Seperti yang telah kami jelaskan, integritas berlaku untuk elemen individu dari database serta database secara keseluruhan. Integritas juga merupakan properti dari struktur database (elemen dalam satu Tabel sesuai satu dengan yang lain) dan hubungan database (catatan memiliki pengidentifikasi unik yang sama, yang disebut kunci, terkait). Dengan demikian, integritas merupakan perhatian utama dalam desain sistem manajemen database. Kami melihat lebih dekat pada masalah integritas di bagian berikutnya.

Kerahasiaan juga merupakan masalah utama dengan database karena database sering digunakan untuk menerapkan berbagi data sensitif yang terkontrol. Akses ke data bisa langsung (Anda meminta catatan dan database menyediakannya) atau tidak langsung (Anda meminta beberapa catatan dan dari hasil tersebut menyimpulkan atau memberi tahu data lain). Mengontrol akses langsung menggunakan teknik kontrol akses yang kami jelaskan di Bab 2 dan 5. Akses tidak langsung, bagaimanapun, lebih sulit untuk dikendalikan, dan kami mengeksplorasinya lebih mendalam nanti di bab ini.

Terakhir, ketersediaan menjadi penting karena motivasi akses bersama yang mendasari pengembangan basis data. Namun, ketersediaan bertentangan dengan kerahasiaan. Bagian terakhir dari bab ini membahas ketersediaan di lingkungan di mana kerahasiaan juga penting.

4.3 Keandalan dan Integritas

Basis data menggabungkan data dari banyak sumber, dan pengguna mengharapkan DBMS menyediakan akses ke data dengan cara yang andal. Ketika insinyur perangkat lunak mengatakan bahwa perangkat lunak memiliki keandalan, itu berarti perangkat lunak itu berjalan untuk jangka waktu yang sangat lama tanpa gagal. Pengguna tentu mengharapkan DBMS dapat diandalkan, karena data biasanya merupakan kunci untuk kebutuhan bisnis atau organisasi. Selain itu, pengguna mempercayakan data mereka ke DBMS dan mengharapkannya untuk melindungi data dari kehilangan atau kerusakan. Kekhawatiran untuk keandalan dan integritas adalah masalah keamanan umum, tetapi lebih jelas dengan database.

Sebuah DBMS menjaga terhadap kehilangan atau kerusakan dalam beberapa cara, yang kita pelajari di bagian ini. Namun, kontrol yang kami anggap tidak mutlak: Tidak ada kontrol yang dapat mencegah pengguna yang berwenang secara tidak sengaja memasukkan nilai yang dapat diterima tetapi salah.

Kekhawatiran database tentang keandalan dan integritas dapat dilihat dari tiga dimensi:

- Integritas database: perhatian bahwa database secara keseluruhan dilindungi dari kerusakan, seperti dari kegagalan drive disk atau kerusakan indeks database master. Kekhawatiran ini ditangani oleh kontrol integritas sistem operasi dan prosedur pemulihan.
- Integritas elemen: perhatian bahwa nilai elemen data tertentu ditulis atau diubah hanya oleh pengguna yang berwenang. Kontrol akses yang tepat melindungi database dari korupsi oleh pengguna yang tidak sah.
- Akurasi elemen: perhatian bahwa hanya nilai yang benar yang ditulis ke dalam elemen database. Pemeriksaan nilai elemen dapat membantu mencegah penyisipan nilai yang tidak tepat. Selain itu, kondisi kendala dapat mendeteksi nilai yang salah.

4.3.1 Fitur Perlindungan dari Sistem Operasi

Seorang administrator sistem yang bertanggung jawab membuat cadangan file database secara berkala bersama dengan file pengguna lainnya. Selama eksekusi normal, fasilitas kontrol akses standar sistem operasi melindungi file dari akses luar. Akhirnya, sistem operasi melakukan pemeriksaan integritas tertentu untuk semua data sebagai bagian dari operasi baca dan tulis normal untuk perangkat I/O. Kontrol ini memberikan keamanan dasar untuk database, tetapi manajer database harus meningkatkannya.

Pembaruan Dua Fase

Masalah serius bagi pengelola database adalah kegagalan sistem komputasi di tengah modifikasi data. Jika item data yang akan dimodifikasi adalah bidang panjang atau catatan yang terdiri dari beberapa atribut, hanya beberapa data baru yang mungkin telah ditulis ke penyimpanan permanen. Oleh karena itu, file database akan berisi data yang salah yang belum diperbarui. Bahkan jika kesalahan jenis ini terlihat dengan mudah (yang sebenarnya tidak), masalah yang lebih halus terjadi ketika beberapa bidang diperbarui dan tidak ada satu bidang pun yang tampak jelas dalam kesalahan. Solusi untuk masalah ini, diusulkan pertama kali oleh Lampson dan Sturgis [LAM76] dan diadopsi oleh sebagian besar DBMS, menggunakan pembaruan dua fase.

Perbarui Teknik

Selama fase pertama, yang disebut fase maksud, DBMS mengumpulkan sumber daya yang dibutuhkan untuk melakukan pembaruan. Ini dapat mengumpulkan data, membuat catatan dummy, membuka file, mengunci pengguna lain, dan menghitung jawaban akhir; singkatnya, ia melakukan segalanya untuk mempersiapkan pembaruan, tetapi tidak membuat perubahan pada database. Fase pertama dapat diulang dalam jumlah yang tidak terbatas karena tidak memerlukan tindakan permanen. Jika sistem gagal selama eksekusi fase pertama, tidak ada kerusakan yang dilakukan karena semua langkah ini dapat dimulai ulang dan diulang setelah sistem melanjutkan pemrosesan.

Peristiwa terakhir dari fase pertama, yang disebut commit, melibatkan penulisan sebuah flag commit ke database. Flag commit berarti bahwa DBMS telah melewati point of no return: Setelah melakukan commit, DBMS mulai membuat perubahan permanen.

Fase kedua membuat perubahan permanen. Selama fase kedua, tidak ada tindakan dari sebelum komit yang dapat diulang, tetapi aktivitas pembaruan fase dua juga dapat diulang sesering yang diperlukan. Jika sistem gagal selama fase kedua, database mungkin berisi data yang tidak lengkap, tetapi sistem dapat memperbaiki data ini dengan melakukan semua aktivitas fase kedua. Setelah tahap kedua selesai, database kembali lengkap.

Contoh Pembaruan Dua Fase

Pertimbangkan database yang berisi inventaris perlengkapan kantor perusahaan. Ruang penyimpanan pusat perusahaan menyimpan kertas, pena, klip kertas, dan sejenisnya, dan berbagai departemen meminta item yang mereka butuhkan. Perusahaan membeli dalam jumlah besar untuk mendapatkan harga terbaik. Setiap departemen memiliki anggaran untuk perlengkapan kantor, jadi ada mekanisme pengisian di mana biaya perlengkapan dipulihkan dari departemen. Juga, ruang penyimpanan pusat memantau jumlah persediaan yang ada untuk memesan persediaan baru ketika persediaan menipis.

Misalkan proses dimulai dengan permintaan dari departemen akuntansi untuk 50 kotak klip kertas. Asumsikan ada 107 kotak dalam persediaan dan pesanan baru ditempatkan jika jumlah persediaan turun di bawah 100. Berikut adalah langkah-langkah yang diikuti setelah ruang persediaan menerima permintaan.

1. Ruang penyimpanan memeriksa database untuk menentukan bahwa ada 50 kotak klip kertas. Jika tidak, permintaan ditolak dan transaksi selesai.
2. Jika klip kertas tersedia dalam jumlah yang cukup, ruang penyimpanan mengurangi 50 dari angka inventaris dalam database (107 – 50 = 57).
3. Ruang persediaan membebankan anggaran persediaan akuntansi (juga dalam database) untuk: 50 kotak klip kertas.
4. Ruang persediaan memeriksa sisa jumlah yang ada (57) untuk menentukan apakah jumlah yang tersisa di bawah titik pemesanan ulang. Karena itu, pemberitahuan untuk memesan lebih banyak klip kertas dibuat, dan item tersebut ditandai sebagai "sesuai pesanan" di database.
5. Sebuah pesanan pengiriman disiapkan, memungkinkan 50 kotak klip kertas untuk dikirim ke akuntansi.

Kelima langkah ini harus diselesaikan dalam urutan yang tercantum agar database akurat dan transaksi diproses dengan benar.

Misalkan kegagalan terjadi saat langkah-langkah ini sedang diproses. Jika kegagalan terjadi sebelum langkah 1 selesai, tidak ada salahnya terjadi, karena seluruh transaksi dapat dimulai kembali. Namun, selama langkah 2, 3, dan 4, perubahan dilakukan pada elemen dalam database. Jika terjadi kegagalan, maka nilai dalam database tidak konsisten. Lebih buruk lagi, transaksi tidak dapat diproses ulang karena permintaan akan dipotong dua kali atau departemen akan dikenakan biaya dua kali atau dua pesanan pengiriman akan disiapkan.

Ketika komit dua fase digunakan, nilai bayangan dipertahankan untuk titik data utama. Nilai data bayangan dihitung dan disimpan secara lokal selama fase maksud, dan disalin ke database aktual selama fase komit. Operasi pada database akan dilakukan sebagai berikut untuk komit dua fase.

Intent:

1. Periksa nilai COMMIT-FLAG di database. Jika diatur, fase ini tidak dapat dilakukan. Menghentikan atau mengulang, memeriksa COMMIT-FLAG sampai tidak disetel.
2. Bandingkan jumlah kotak klip kertas di tangan dengan nomor yang diminta; jika lebih banyak yang diminta daripada yang ada, hentikan.
3. Hitung TCLIPS = ONHAND – REQUISITION.
4. Dapatkan BUDGET, anggaran persediaan saat ini yang tersisa untuk departemen akuntansi. Hitung TBUDGET = ANGGARAN – BIAYA, di mana BIAYA adalah biaya 50 kotak klip.
5. Periksa apakah TCLIPS berada di bawah titik pemesanan ulang; jika demikian, atur TREORDER = TRUE; jika tidak, atur TREORDER = FALSE.

Commit:

1. Atur COMMIT-FLAG di database.
2. Salin TCLIPS ke CLIPS di database.
3. Salin TBUDGET ke BUDGET di database.
4. Copy TREORDER ke REORDER di database.
5. Siapkan pemberitahuan untuk mengirimkan klip kertas ke departemen akuntansi. Tunjukkan transaksi selesai di log.
6. Hapus COMMIT-FLAG.

Dengan contoh ini, setiap langkah dari fase maksud hanya bergantung pada nilai yang tidak dimodifikasi dari database dan hasil sebelumnya dari fase maksud. Setiap variabel yang dimulai dengan T adalah variabel bayangan yang hanya digunakan dalam transaksi ini. Langkah-langkah dari fase maksud dapat diulang berkali-kali tanpa mempengaruhi integritas database.

Setelah DBMS memulai fase commit, ia menulis flag COMMIT. Saat tanda ini disetel, DBMS tidak akan melakukan langkah apa pun dari fase maksud. Langkah-langkah maksud tidak dapat dilakukan setelah melakukan karena nilai database dimodifikasi dalam fase komit. Perhatikan, bagaimanapun, bahwa langkah-langkah fase komit dapat diulang dalam jumlah yang tidak terbatas, sekali lagi tanpa efek negatif pada kebenaran nilai dalam database.

Satu kelemahan yang tersisa dalam logika ini terjadi jika sistem gagal setelah menulis pesan “transaction complete” di log tetapi sebelum mengosongkan flag commit di database. Ini adalah masalah sederhana untuk bekerja mundur melalui log transaksi untuk menemukan transaksi selesai yang flag komitnya masih disetel dan untuk menghapus flag tersebut.

4.3.2 Redundansi/Konsistensi Internal

Banyak DBMS memelihara informasi tambahan untuk mendeteksi inkonsistensi internal dalam data. Informasi tambahan berkisar dari beberapa bit cek hingga bidang duplikat atau bayangan, tergantung pada pentingnya data.

Deteksi Kesalahan dan Kode Koreksi

Salah satu bentuk redundansi adalah deteksi kesalahan dan kode koreksi, seperti bit paritas, kode Hamming, dan pemeriksaan redundansi siklik. Kode-kode ini dapat diterapkan ke bidang tunggal, catatan, atau seluruh database. Setiap kali item data ditempatkan dalam database, kode cek yang sesuai dihitung dan disimpan; setiap kali item data diambil, kode cek serupa dihitung dan dibandingkan dengan nilai yang disimpan. Jika nilainya tidak sama, mereka menandakan ke DBMS bahwa telah terjadi kesalahan dalam database. Beberapa kode ini menunjukkan tempat kesalahan; yang lain menunjukkan dengan tepat apa nilai yang benar seharusnya. Semakin banyak informasi yang diberikan, semakin banyak ruang yang dibutuhkan untuk menyimpan kode.

Bidang Bayangan

Seluruh atribut atau seluruh catatan dapat diduplikasi dalam database. Jika data tidak dapat direproduksi, salinan kedua ini dapat memberikan penggantian segera jika kesalahan terdeteksi. Jelas, bidang yang berlebihan membutuhkan ruang penyimpanan yang besar.

Pemulihan

Selain proses koreksi kesalahan ini, DBMS dapat memelihara log akses pengguna, khususnya perubahan. Jika terjadi kegagalan, database dimuat ulang dari salinan cadangan dan semua perubahan selanjutnya diterapkan dari log audit.

Konkurensi/Konsistensi

Sistem basis data seringkali merupakan sistem multiuser. Akses oleh dua pengguna yang berbagi database yang sama harus dibatasi sehingga tidak ada yang mengganggu yang lain. Penguncian sederhana dilakukan oleh DBMS. Jika dua pengguna mencoba membaca item data yang sama, tidak ada konflik karena keduanya mendapatkan nilai yang sama.

Jika kedua pengguna mencoba untuk memodifikasi item data yang sama, kita sering berasumsi bahwa tidak ada konflik karena masing-masing tahu apa yang harus ditulis; nilai yang akan ditulis tidak bergantung pada nilai item data sebelumnya. Namun, anggapan ini kurang tepat.

Untuk melihat bagaimana modifikasi bersamaan dapat membuat kita mendapat masalah, misalkan database terdiri dari reservasi kursi untuk penerbangan maskapai tertentu. Agen A, memesan kursi untuk penumpang Mock, mengajukan pertanyaan untuk menemukan kursi apa yang masih tersedia. Agen tahu bahwa Mock lebih suka kursi di lorong kanan, dan agen menemukan bahwa kursi 5D, 11D, dan 14D terbuka. Pada saat yang sama, Agen B mencoba memesan kursi untuk keluarga yang terdiri dari tiga orang yang bepergian bersama. Sebagai tanggapan atas kueri, database menunjukkan bahwa 8A–B–C dan 11D–E–F adalah dua grup tersisa dari tiga kursi yang belum ditetapkan yang berdekatan. Agen A mengirimkan perintah pembaruan

```
SELECT  (SEAT-NO = ‘11D’)
ASSIGN ‘MOCK,E’ TO PASSENGER-NAME
```

sementara agen B mengirimkan urutan pembaruan

```
SELECT (SEAT-NO = ‘11D’)
ASSIGN ‘EDWARDS,S’ TO PASSENGER-NAME
```

serta perintah untuk kursi 11E dan 11F. Kemudian dua penumpang telah dipesan ke kursi yang sama (yang akan membuat tidak nyaman, untuk sedikitnya).

Kedua agen telah bertindak dengan benar: Masing-masing mencari daftar kursi kosong, memilih satu kursi dari daftar, dan memperbarui database untuk menunjukkan kepada siapa kursi itu diberikan. Kesulitan dalam situasi ini adalah waktu tunda antara membaca nilai dari database dan menulis modifikasi nilai itu. Selama waktu tunda, pengguna lain telah mengakses data yang sama.

Untuk mengatasi masalah ini, DBMS memperlakukan seluruh siklus pembaruan kueri sebagai operasi atom tunggal. Perintah dari agen sekarang harus menyerupai "baca nilai kursi PASSENGER-NAME saat ini untuk kursi 11D; jika 'UNASSIGNED', ubah menjadi 'MOCK,E' (atau 'EDWARDS,S'). Siklus baca-modifikasi harus diselesaikan sebagai item yang tidak terputus tanpa mengizinkan pengguna lain mengakses bidang PASSENGER-NAME untuk kursi 11D.

Permintaan agen kedua untuk memesan tidak akan dipertimbangkan sampai setelah agen pertama selesai; pada saat itu, nilai PASSENGER-NAME tidak lagi menjadi 'UNASSIGNED'.

Masalah terakhir dalam akses konkuren adalah baca-tulis. Misalkan satu pengguna memperbarui nilai ketika pengguna kedua ingin membacanya. Jika pembacaan dilakukan saat penulisan sedang berlangsung, pembaca dapat menerima data yang hanya diperbarui sebagian. Akibatnya, DBMS mengunci semua permintaan baca sampai penulisan selesai.

4.4 Database Disclosure

Seperti yang kami jelaskan di Bab 9, lebih banyak data yang dikumpulkan tentang lebih banyak orang daripada sebelumnya. Di masa lalu, satu perusahaan, organisasi, atau kantor pemerintah hanya tahu tentang klien atau pelanggannya; ada sedikit berbagi antara organisasi. Dan jumlah atau jenis tempat yang mengumpulkan data sedikit. Ya, kami berharap kantor menyimpan catatan tentang kami, tetapi tidak termasuk setiap jalan, mesin kasir, dan situs web yang kami kunjungi. Komputer, tentu saja, telah memungkinkan tidak hanya pengumpulan tetapi juga pembagian data dalam jumlah besar ini.

Basis data berisi pemikiran, preferensi, pendapat, aktivitas (atau deskripsinya), fantasi, teman, dan koneksi. Dari database ini orang dapat menarik kesimpulan yang mungkin akurat atau salah: Jamie adalah teman Anda. Jamie suka katak. Ergo, kamu suka katak. Jelas, ini adalah logika yang salah, meskipun mungkin juga benar. Pada bagian berikutnya kita mengeksplorasi bagaimana orang dan komputer menganalisis database tersebut untuk koneksi data yang mengarah pada pengungkapan data yang tidak dapat diterima.

4.4.1 Data Sensitif

Beberapa database berisi apa yang disebut data sensitif. Sebagai definisi kerja, mari kita katakan bahwa data sensitif adalah data yang tidak boleh dipublikasikan. Menentukan item dan bidang data mana yang sensitif bergantung pada database individual dan makna yang mendasari data tersebut. Jelas, beberapa database, seperti katalog perpustakaan umum, tidak berisi data sensitif; database

lain, seperti yang terkait dengan pertahanan, sepenuhnya sensitif. Kedua kasus ini—tidak ada yang sensitif dan semua yang sensitif—adalah yang paling mudah ditangani, karena dapat dicakup oleh kontrol akses ke database secara keseluruhan. Seseorang adalah atau bukan pengguna resmi. Kontrol ini dapat disediakan oleh sistem operasi.

Masalah yang lebih sulit, yang juga lebih menarik, adalah kasus di mana beberapa tetapi tidak semua elemen dalam database sensitif. Mungkin ada berbagai tingkat sensitivitas. Misalnya, database universitas mungkin berisi data mahasiswa yang terdiri dari nama, bantuan keuangan, asrama, penggunaan narkoba, jenis kelamin, denda parkir, dan ras.. Nama dan asrama mungkin yang paling tidak sensitif; bantuan keuangan, denda parkir, dan penggunaan narkoba paling banyak; seks dan ras di antara keduanya. Artinya, banyak orang mungkin memiliki akses yang sah ke nama, beberapa jenis kelamin dan ras, dan relatif sedikit untuk bantuan keuangan, denda parkir, atau penggunaan narkoba. Memang, pengetahuan tentang keberadaan beberapa bidang, seperti penggunaan narkoba, mungkin sensitif. Dengan demikian, keamanan tidak hanya menyangkut elemen data tetapi juga konteks dan maknanya.

Selanjutnya, kita harus memperhitungkan tingkat sensitivitas yang berbeda. Misalnya, meskipun semua bidang sangat sensitif, bidang bantuan keuangan, denda parkir, dan bidang penggunaan narkoba mungkin tidak memiliki jenis pembatasan akses yang sama. Persyaratan keamanan kami mungkin menuntut agar beberapa orang diizinkan untuk melihat setiap bidang, tetapi tidak seorang pun diizinkan untuk melihat ketiganya. Tantangan dari masalah kontrol akses adalah membatasi akses pengguna sehingga mereka hanya dapat memperoleh data yang mereka miliki aksesnya yang sah. Atau, masalah kontrol akses memaksa kami untuk memastikan bahwa data sensitif tidak diberikan kepada orang yang tidak berwenang.
Beberapa faktor dapat membuat data menjadi sensitif.

- Inheren sensitif. Nilai itu sendiri mungkin sangat terbuka sehingga sensitif. Contohnya adalah lokasi rudal pertahanan atau pendapatan rata-rata tukang cukur di kota dengan hanya satu tukang cukur.
- Dari sumber yang sensitif. Sumber data mungkin menunjukkan perlunya kerahasiaan. Contohnya adalah informasi dari seorang informan yang identitasnya akan dikompromikan jika informasi tersebut diungkapkan.
- Dinyatakan sensitif. Administrator basis data atau pemilik data mungkin telah menyatakan bahwa data tersebut sensitif. Contohnya adalah data militer rahasia atau nama donor anonim dari sebuah karya seni.
- Bagian dari atribut atau catatan sensitif. Dalam database, seluruh atribut atau catatan dapat diklasifikasikan sebagai sensitif. Contohnya adalah atribut gaji dari database personel atau catatan yang menjelaskan misi luar angkasa rahasia.
- Sensitif dalam kaitannya dengan informasi yang diungkapkan sebelumnya. Beberapa data menjadi sensitif dengan adanya data lain. Misalnya, koordinat bujur dari tambang emas rahasia mengungkapkan sedikit, tetapi koordinat bujur dalam hubungannya dengan koordinat lintang menunjukkan tambang tersebut.

Semua faktor ini harus dipertimbangkan ketika sensitivitas data sedang ditentukan.

Jenis Disclosure

Kita semua tahu bahwa beberapa data sensitif. Namun, terkadang karakteristik data pun sensitif. Pada bagian ini, kita melihat bahwa bahkan informasi deskriptif tentang data (seperti keberadaan mereka atau apakah mereka memiliki elemen yang bukan nol) adalah bentuk dari penyingkapan.

Data yang Tepat

Pengungkapan yang paling serius adalah nilai pasti dari item data sensitif itu sendiri. Pengguna mungkin mengetahui bahwa data sensitif sedang diminta, atau pengguna dapat meminta data umum tanpa mengetahui bahwa sebagian data sensitif. Pengelola basis data yang salah bahkan dapat mengirimkan data sensitif secara tidak sengaja, tanpa diminta oleh pengguna. Dalam semua kasus ini, hasilnya sama: Keamanan data sensitif telah dilanggar.

Bounds (Batas)

Eksposur lain adalah mengungkapkan batas pada nilai sensitif, yaitu, menunjukkan bahwa nilai sensitif, y, berada di antara dua nilai, L dan H. Terkadang, dengan menggunakan teknik penyempitan yang tidak berbeda dengan pencarian biner, pengguna dapat menentukan terlebih dahulu bahwa L y H dan kemudian lihat apakah L y H/2, dan seterusnya, sehingga memungkinkan pengguna untuk menentukan y ke presisi yang diinginkan. Dalam kasus lain, hanya mengungkapkan bahwa nilai seperti anggaran beasiswa atletik atau jumlah agen CIA melebihi jumlah tertentu mungkin merupakan pelanggaran keamanan yang serius.

Terkadang, bagaimanapun, batas adalah cara yang berguna untuk menyajikan data sensitif. Adalah umum untuk melepaskan batas atas dan bawah untuk data tanpa mengidentifikasi catatan tertentu. Misalnya, sebuah perusahaan mungkin mengumumkan bahwa gajinya untuk programmer berkisar dari $50.000 hingga $82.000. Jika Anda seorang programmer yang berpenghasilan $79.700, Anda akan mengira Anda cukup kaya, sehingga Anda memiliki informasi yang Anda inginkan; namun, pengumuman tersebut tidak mengungkapkan siapa programmer dengan bayaran tertinggi dan terendah.

Hasil Negatif

Kadang-kadang kita dapat menyusun kata untuk menentukan hasil negatif. Artinya, kita dapat mempelajari bahwa z bukan nilai y. Misalnya, mengetahui bahwa 0 bukanlah jumlah total hukuman kejahatan untuk seseorang mengungkapkan bahwa orang tersebut dihukum karena kejahatan. Perbedaan antara 1 dan 2 atau 46 dan 47 kejahatan tidak sesensitif perbedaan antara 0 dan 1. Oleh karena itu, pengungkapan bahwa suatu nilai bukan 0 dapat menjadi pengungkapan yang signifikan. Demikian pula, jika seorang siswa tidak muncul dalam daftar penghargaan, Anda dapat menyimpulkan bahwa nilai rata-rata orang tersebut di bawah 3,50. Namun, informasi ini tidak terlalu terbuka, karena rentang nilai rata-rata dari 0,0 hingga 3,49 agak lebar.

Eksistensi

Dalam beberapa kasus, keberadaan data itu sendiri merupakan bagian dari data yang sensitif, terlepas dari nilai sebenarnya. Misalnya, pemberi kerja mungkin tidak ingin karyawan mengetahui bahwa penggunaan telepon mereka sedang dipantau. Dalam kasus ini, menemukan bidang NUMBER PANGGILAN TELEPON PRIBADI dalam file personalia akan mengungkapkan data sensitif.

Probable Value

Akhirnya, dimungkinkan untuk menentukan probabilitas bahwa elemen tertentu memiliki nilai tertentu. Untuk melihat caranya, misalkan Anda ingin mengetahui apakah presiden Amerika Serikat terdaftar di partai Tory. Mengetahui bahwa presiden ada di database, Anda mengirimkan dua kueri ke database: Manajer database dapat mengontrol akses dengan kueri langsung; pengungkapan dapat terjadi dengan cara yang lebih halus yang lebih sulit dikendalikan.

```
Count(Residence="1600 Pennsylvania Avenue") =4
Count(Residence="1600 Pennsylvania Avenue" AND Tory=TRUE)
= 1
```

Dari pertanyaan ini Anda menyimpulkan ada kemungkinan 25 persen bahwa presiden adalah Tory yang terdaftar.

Serangan inferensi

Sebuah Inference Attack adalah teknik data mining yang dilakukan dengan menganalisis data dalam rangka untuk secara tidak sah mendapatkan pengetahuan tentang topik atau database.

Informasi sensitif sebuah subjek dapat dianggap sebagai bocoran jika musuh dapat menyimpulkan nilai sebenarnya dengan kepercayaan diri tinggi. Ini adalah contoh dari keamanan informasi yang dilanggar. Sebuah serangan Inference terjadi ketika pengguna dapat menyimpulkan dari informasi sepele menjadi informasi yang lebih kuat tentang database tanpa langsung mengaksesnya.Tujuan dari serangan Inferensi adalah untuk mengumpulkan informasi pada satu tingkat keamanan untuk menentukan fakta yang harus dilindungi pada tingkat keamanan yang lebih tinggi.

Penanggulangan

Kontrol inferensi keamanan komputer adalah upaya untuk mencegah penggunamenyimpulkan informasi rahasia dari potongan berhak diakses informasi dengan klasifikasi rendah. Profesional keamanan komputer menginstal protokol ke dalam database untuk mencegah serangan inferensi oleh software.Teknik untuk mendeteksi dan menghapus saluran inferensi dapat dibagi dalam dua kategori. Kategori pertama meliputi teknik yang mendeteksi saluran inferensi selama database didesain. Saluran inferensi dihapus dengan memodifikasi desain database atau dengan meningkatkan tingkat klasifikasi beberapa item data. Teknik dalam kategori kedua berusaha untuk menghilangkan pelanggaran saluran inferensi selama waktu pemrosesan query. Jika saluran inferensi terdeteksi, query baik ditolak atau dimodifikasi untuk menghindari serangan. saat keamanan desain-waktu lebih mudah untuk mengelola dan melaksanakan, pendekatan permintaan time memungkinkan lebih banyak ketersediaan data daripada pendekatan desain - waktu karena informasi lebih lanjut (masa lalu dan pertanyaan ini / jawaban) dapat digunakan untuk pengungkapan inference.

Disclosure monitor (Dimon) mendeteksi dan menghilangkan saluran kesimpulan berdasarkan database constraints.A pengungkapan inference Mesin (DiIE) diusulkan, yang menghasilkan semua informasi yang diungkapkan dapat didasarkan pada pengguna hasil query sebelumnya, hasil query saat ini, dan satu set Horn - klausul constraints.Dynamic Memantau Pengungkapan (D2Mon) menjamin kerahasiaan data dan ketersediaan maksimal bahkan di hadapan kesimpulan dan update . Ini adalah tujuanya untuk melengkapi mekanisme kontrol akses yang ada (misalnya, DAC, MAC, RBAC) untuk mengatasi masalah inferensi .

MAC : Dalam Mandatory Access Control (MAC) model, pengguna diberi hak akses ke sumber daya oleh administrator. Hanya Administrator yang dapat memberikan izin atau hak untuk benda dan sumber daya. Akses terhadap sumber daya didasarkan pada tingkat keamanan obyek, sedangkan pengguna diberikan izin keamanan. Hanya administrator yang dapat memodifikasi label keamanan obyek atau izin keamanan pengguna.

Pengungkapan dinamis Memantau Arsitektur menggunakan MAC ini untuk mendeteksi pelanggaran keamanan langsung, jika ada pelanggaran keamanan langsung terdeteksi query ditolak saat itu juga, jika tidak melewati mesin inferensi untuk mengungkapkan data yang dapat diinferensi. Semua data diungkapkan diperoleh dari mesin inferensi yang sedang dikirim ke MAC untuk mendeteksi serangan. Jikapelanggaran keamanan langsung terdeteksi, permintaan ditolak.

DAC : Dalam Discretionary Access Control (DAC) model, akses ke sumber daya didasarkan pada identitas pengguna. Seorang pengguna diberikan hak akses ke sumber daya dengan ditempatkan pada daftar kontrol akses (ACL) yang berhubungan dengan sumber daya. Entri pada ACL sumber daya yang dikenal sebagai Access Control entri (ACE). Ketika seorang pengguna (atau kelompok) adalah pemilik dari suatu obyek dalam model DAC, pengguna dapat memberikan izin kepada pengguna lain dan kelompok Model DAC didasarkan pada kepemilikan sumber daya .

RBAC : Dalam Access Control (RBAC) Model Peran Berbasis, akses ke sumber daya didasarkan pada peran yang ditugaskan kepada pengguna. Dalam model ini, administrator memberikan pengguna untuk peran yang memiliki hak yang telah ditentukan tertentu dan hak istimewa. Karena asosiasi pengguna dengan peran, pengguna dapat mengakses sumber daya tertentu dan melakukan tugas-tugas tertentu. RBAC juga dikenal sebagai Non - Discretionary Access Control . Peran ditugaskan untuk pengguna dikelola secara terpusat.

Kami telah menyajikan beberapa teknik dimana sistem manajemen database menyeimbangkan penggunaan dan kontrol akses. Di area lain, seperti sistem operasi, kontrol akses adalah biner: Akses diberikan atau ditolak ke suatu objek. Namun, sistem manajemen basis data mencoba mengambil pendekatan yang lebih bernuansa; pendekatan ya/tidak yang ketat akan menyebabkan pembatasan akses yang ekstrem (jika ada satu catatan sensitif dari satu juta, akses ke seluruh basis data dapat diblokir) atau kelemahan akses yang ekstrem (akses diizinkan terlepas dari potensi inferensi atau agregasi). Dengan demikian, sistem manajemen basis data dan administratornya mencoba menemukan jalan tengah yang masuk akal.

Selanjutnya kita beralih ke penggunaan database dalam apa yang disebut data mining dan big data.

4.5 Data Mining Dan Big Data

Di bagian terakhir ini kami mempertimbangkan dua topik terkait. Yang pertama, data mining, melibatkan orang dan program yang mencari dan menyaring kumpulan data untuk mendapatkan data. Ya, Anda membantah, untuk itulah database dan apa yang telah kami pertimbangkan sepanjang sisa bab ini. Data mining, bagaimanapun, menyiratkan pencarian pola dan koneksi yang sebelumnya tidak diketahui dan bahkan mungkin tidak dapat diprediksi. Tahukah Anda bahwa orang kidal lebih cenderung memilih telur goreng daripada telur rebus? (Kami tidak tahu apakah itu benar.) Sebuah tim peneliti dengan database yang berisi dominasi tangan dan preferensi telur dapat menemukan itu dan korelasi statistik menakjubkan lainnya, melalui keajaiban data mining. Komunitas data mining telah berkembang tanpa banyak pertimbangan keamanan, jadi kami mencantumkan beberapa masalah keamanan yang siap untuk dipertimbangkan.

Data mining terkait erat dengan konsep big data, yang melibatkan pengumpulan data dalam jumlah besar, seringkali tidak dimaksudkan sebagai basis data atau terstruktur seperti itu. Penekanannya adalah pada istilah "besar", misalnya, kumpulan semua entri indeks untuk mesin pencari. Ketika mesin pencari melaporkan bahwa ia memiliki 17 juta halaman yang menjawab pertanyaan Anda, itu mungkin benar, meskipun kegunaan dari tautan ke-17 juta mungkin tidak tinggi. Sebagian besar dari kita menemukan apa yang kita inginkan di beberapa hasil pertama atau kita menyimpulkan bahwa kueri tidak mendapatkan hasil yang kita inginkan dan mengajukan pertanyaan yang berbeda.

Di bagian selanjutnya kita mengeksplorasi data mining dan big data, menunjukkan aspek keamanan dari keduanya.

4.5.1 Data Mining

Pengelolaan informasi pada saat ini menjadi hal yang sangat penting. Dimana, setiap data akan sangat berpengaruh dalam hal pengambilan sebuah keputusan yang akan diambil oleh pengguna. Istilah yang cukup dikenal saat ini oleh praktisi di dunia IT adalah data mining.

Data Mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, machine learning untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar (Turban dkk. 2005). Data mining bisa disebut *knowledge discovery* adalah proses pengambilan pola pada data yang akan di proses lalu output tersebut berupa informasi yang sangat penting. Terdapat beberapa istilah lain yang memiliki makna sama dengan data mining, yaitu *Knowledge discovery in databases* (KDD), ekstraksi pengetahuan (*knowledge extraction*), Analisa data/pola (*data/pattern analysis*), kecerdasan bisnis (*business intelligence*) dan *data archaeology* dan *data dredging* (Larose, 2005).

Perbedaan antara database dan aplikasi data mining menjadi kabur; Anda mungkin dapat melihat bagaimana Anda dapat menerapkan teknik ini dalam kueri basis data biasa. Umumnya, kueri basis data bersifat manual, sedangkan data mining lebih otomatis. Anda dapat mengembangkan kueri basis data untuk melihat produk lain apa yang dibeli oleh orang-orang yang membeli kamera digital dan Anda mungkin melihat lebih banyak pemutar MP3 dalam hasilnya, tetapi Anda harus mengamati hubungan itu sendiri. Alat data mining akan menghadirkan hubungan yang signifikan, tidak hanya

antara kamera dan pemutar MP3, tetapi juga antara bagel, tiket pesawat, dan sepatu lari. (Sekali lagi, kami tidak tahu apakah ada hubungan seperti itu.) Manusia harus menganalisis korelasi ini dan menentukan apa yang signifikan.

Data mining menyajikan kemungkinan hubungan, tetapi ini belum tentu hubungan sebab dan akibat. Misalkan Anda menganalisis data dan menemukan korelasi antara penjualan es krim dan kematian karena tenggelam. Anda tidak akan menyimpulkan bahwa menjual es krim menyebabkan tenggelam (atau sebaliknya). Perbedaan ini menunjukkan mengapa manusia harus terlibat dalam data mining untuk menginterpretasikan output: Hanya manusia yang dapat membedakan bahwa lebih banyak variabel yang terlibat (misalnya, waktu dalam setahun atau tempat di mana kerucut dijual).

Keuntungan keamanan komputer dari data mining. Data mining banyak digunakan untuk menganalisis data sistem, misalnya, log audit, untuk mengidentifikasi pola yang terkait dengan serangan. Menemukan prekursor serangan dapat membantu mengembangkan alat dan teknik pencegahan yang baik, dan melihat tindakan yang terkait dengan serangan dapat membantu menentukan kerentanan untuk mengontrol kerusakan yang mungkin terjadi. (Salah satu karya awal di bidang ini adalah oleh Lee dan Stolfo, dan seluruh konferensi telah dikhususkan untuk topik penting dan matang ini.)

Namun, di bagian ini, kami ingin memeriksa masalah keamanan yang melibatkan data mining. Tiga serangkai kerahasiaan, integritas, dan ketersediaan kami yang sekarang dikenal memberi kami petunjuk tentang apa masalah keamanan ini. Masalah kerahasiaan dimulai dengan privasi tetapi juga mencakup data kepemilikan dan sensitif secara komersial serta melindungi nilai kekayaan intelektual: Bagaimana kita mengontrol apa yang diungkapkan atau diturunkan? Untuk integritas, masalah penting adalah kebenaran—data yang salah tidak berguna dan berpotensi merusak, tetapi kita perlu menyelidiki cara mengukur dan memastikan kebenarannya. Pertimbangan ketersediaan berkaitan dengan kinerja dan struktur: Menggabungkan database yang awalnya tidak dirancang untuk digabungkan akan memengaruhi apakah hasil dapat diperoleh secara tepat waktu atau bahkan sama sekali.

Privasi dan Sensitivitas

Karena tujuan data mining adalah hasil ringkasan, bukan item data individual, Anda tidak akan mengharapkan masalah dengan sensitivitas item data individual. Sayangnya, itu tidak benar.

Privasi individu dapat mengalami masalah inferensi dan agregasi yang sama seperti yang kami pelajari untuk database. Karena privasi, khususnya melindungi apa yang dianggap seseorang sebagai informasi pribadi, merupakan topik penting yang terkait dengan banyak bidang keamanan komputer.

Namun, tidak hanya privasi individu yang terpengaruh: Korelasi dengan agregasi dan inferensi dapat memengaruhi perusahaan, organisasi, dan pemerintah juga. Ambil contoh, masalah yang melibatkan ban Firestone dan kendaraan Ford Explorer. Pada bulan Mei 2000, Administrasi Keselamatan Lalu Lintas Jalan Raya Nasional AS (NHTSA) menemukan insiden kegagalan ban yang tinggi pada Ford Explorer yang dilengkapi dengan ban Firestone. Dalam kondisi tertentu tapak ban Firestone lepas; dalam kondisi tertentu Ford Explorer terbalik, dan ketika tapak terpisah, Ford lebih mungkin untuk

terbalik. Konsumen telah mengeluh kepada Ford dan Firestone sejak tak lama setelah kombinasi ban dan kendaraan ditempatkan di pasar pada tahun 1990, tetapi masalah mulai meningkat setelah perubahan desain pada tahun 1995. Kedua perusahaan memiliki beberapa bukti masalah, tetapi tinjauan NHTSA tentang data gabungan lebih baik menunjukkan korelasi. Mempertahankan data tentang kualitas produk adalah praktik manajemen standar. Tetapi sensitivitas data dalam basis data ini akan menghalangi banyak berbagi. Bahkan jika pihak netral yang dapat dipercaya dapat ditemukan untuk menambang data, pemiliknya akan cukup khawatir dengan apa yang mungkin terungkap. Sejumlah besar kegagalan satu produk dapat menunjukkan potensi kelemahan pasar, atau serangkaian data dalam jumlah kecil dapat mengungkapkan aktivitas pemasaran uji coba kepada pihak luar.

Data tentang suatu entitas (seseorang, perusahaan, organisasi, badan pemerintah) mungkin tidak berada di bawah kendali entitas tersebut. Supermarket mengumpulkan data produk dari pembeli mereka, baik dari kunjungan tunggal atau, yang lebih berguna, di semua pembelian untuk pelanggan yang menggunakan kartu “loyalitas pelanggan”. Secara agregat, data menunjukkan hasil pemasaran yang berguna bagi produsen, biro iklan, peneliti kesehatan, lembaga makanan pemerintah, lembaga keuangan, peneliti, dan lain-lain. Tetapi hasil ini dikumpulkan oleh supermarket yang sekarang dapat melakukan apa saja dengan hasilnya, termasuk menjualnya ke pesaing produsen, misalnya.

Kebenaran dan Integritas Data

"Menghubungkan titik-titik" adalah frasa yang sedang populer saat ini: Ini mengacu pada penarikan kesimpulan dari hubungan antara bit data yang terpisah. Tetapi sebelum kita dapat menghubungkan titik-titik, kita perlu melakukan dua hal penting lainnya: mengumpulkan dan memperbaikinya. Penyimpanan data dan teknologi komputer memungkinkan untuk mengumpulkan lebih banyak titik daripada sebelumnya. Tetapi jika nama atau alamat Anda pernah salah muncul di milis, Anda tahu bahwa tidak semua titik yang dikumpulkan akurat.

Memperbaiki Kesalahan dalam Data

Mari kita ambil milis sebagai contoh. Tetangga Anda membawakan katalog perlengkapan dapur untuk Anda di sebuah alamat dengan nama Anda tetapi alamat bukan dilingkunagn anda; jelas seseorang melakukan kesalahan memasukkan alamat Anda. Anda menghubungi tempat persediaan dapur, dan mereka dengan senang hati mengubah alamat Anda di catatan mereka karena kepentingan mereka untuk memastikan katalog sampai ke orang-orang yang tertarik pada mereka. Tetapi mereka membeli nama dan alamat Anda bersama dengan orang lain dari milis, dan mereka tidak memiliki insentif untuk menghubungi pemilik daftar untuk memperbaiki catatan utama Anda. Jadi katalog tambahan terus muncul dengan tetangga Anda. Anda dapat melihat ke mana arah cerita ini—alamat yang salah tidak pernah mati.

Data mining memperburuk situasi ini. Basis data memerlukan kunci unik untuk membantu struktur dan pencarian. Tetapi database yang berbeda mungkin tidak memiliki kunci bersama, sehingga mereka menggunakan beberapa bidang data seolah-olah itu adalah kunci. Dalam contoh kasus kami, bidang data bersama ini mungkin alamat, jadi sekarang alamat tetangga Anda dikaitkan dengan memasak (bahkan jika tetangga Anda membutuhkan resep untuk membuat teh). Untungnya, contoh ini adalah konsekuensi kecil.

Pertimbangkan teroris, namun. Badan intelijen pemerintah mengumpulkan data tentang aktivitas mencurigakan. Tetapi nama-nama orang yang mencurigakan itu asing, ditulis dalam alfabet yang berbeda. Ketika diubah menjadi alfabet pemerintah, transformasinya tidak teratur: Satu agen menulis "Doe", "Do" lainnya, dan "Dho" lainnya. Mencoba menggunakan nama-nama ini sebagai kunci umum sangat sulit. Salah satu pendekatannya adalah fonetik. Anda mengelompokkan istilah yang terdengar mirip. Namun, dalam kasus ini, Anda mungkin juga membawa "Jo", "Cho", "Toe", dan "Tsiao", sehingga melibatkan orang yang tidak bersalah dalam pencarian teroris. Dengan asumsi seorang analis manusia dapat memisahkan semua ini dengan benar dan ingin memperbaiki database Doe/Do/Doh, masih ada dua masalah. Pertama, analis mungkin tidak memiliki akses ke database asli yang dimiliki oleh lembaga lain. Bahkan jika analis bisa mendapatkan aslinya, analis mungkin tidak akan pernah belajar di mana lagi database asli ini telah disalin.

Salah satu tujuan penting dari database adalah untuk memiliki catatan di satu tempat sehingga satu koreksi melayani semua kegunaan. Dengan data mining, hasilnya adalah agregat dari beberapa basis data. Tidak ada cara alami untuk bekerja mundur dari hasil ke database gabungan untuk menemukan dan memperbaiki kesalahan.

Menggunakan Data Sebanding (*Comparable Data*)

Semantik data adalah pertimbangan penting lainnya saat menambang data. Pertimbangkan dua database geografis dengan data pendapatan keluarga. Kecuali satu database memiliki pendapatan per dolar, dan database lainnya memiliki data dalam ribuan euro. Bahkan jika nama bidangnya sama, menggabungkan data mentah akan menghasilkan statistik yang sangat terdistorsi. Pertimbangkan atribut lain yang dinilai tinggi/sedang/rendah dalam satu database dan pada skala numerik 1-5 di database lain. Haruskah tinggi/sedang/rendah diperlakukan sebagai 1/3/5? Bahkan jika analis menggunakan transformasi itu, komputasi dengan beberapa presisi 3 poin dan beberapa 5 poin mengurangi kualitas hasil. Atau bagaimana Anda bisa secara bermakna menggabungkan satu database yang memiliki atribut tertentu dengan yang lain yang tidak?

Mengurangi False Matches

Seperti yang telah kami jelaskan sebelumnya, kebetulan bukanlah korelasi atau sebab-akibat; karena dua hal terjadi bersamaan tidak berarti salah satu menyebabkan yang lain. Data mining mencoba untuk menyoroti koneksi yang tidak jelas dalam data, tetapi aplikasi data mining sering menggunakan logika fuzzy untuk menemukan koneksi ini. Pendekatan ini akan menghasilkan positif palsu (kecocokan palsu) dan koneksi yang tidak terjawab (negatif palsu). Kita harus peka terhadap ketidakakuratan yang melekat pada pendekatan data mining dan menjaga agar tidak terlalu percaya pada output aplikasi data mining hanya karena "komputer mengatakan demikian." Hasil yang benar dan interpretasi yang benar dari hasil tersebut adalah masalah keamanan utama untuk data mining.

Ketersediaan Data

Interoperabilitas di antara basis data yang berbeda adalah masalah keamanan ketiga untuk data mining. Seperti yang baru saja kami jelaskan, database harus memiliki struktur dan semantik yang kompatibel untuk memungkinkan data mining. Data yang hilang atau tidak dapat dibandingkan dapat membuat hasil data mining menjadi salah, jadi mungkin alternatif yang lebih baik adalah tidak memberikan hasil. Tetapi tidak ada hasil yang tidak sama karena tidak ada korelasi. Seperti halnya

database tunggal, aplikasi data mining harus berurusan dengan banyak sensitivitas. Mencoba menggabungkan database pada atribut dengan nilai yang lebih sensitif dapat menyebabkan tidak ada data dan karenanya tidak ada kecocokan. Menggabungkan Tabel data tanpa bidang umum (kunci) yang alami dan akurat menyebabkan banyak hasil yang salah.

4.5.2 Big Data

Istilah big data berarti analisis sejumlah besar data, sering kali dikumpulkan dari berbagai sumber. Secara tradisional, toko kelontong mungkin mengira bisa menjual 100 kepala selada dalam seminggu; jika tebakannya terlalu rendah, beberapa pembeli meninggalkan toko tanpa selada, tetapi jika tebakannya terlalu tinggi, penjual mungkin harus menurunkan harga selada pada akhir minggu untuk mengeluarkan stok sebelum rusak. Melihat prakiraan cuaca jangka panjang, toko kelontong mungkin melihat gelombang panas yang diprediksi dan memesan selada tambahan, karena orang menginginkan makanan ringan seperti salad dalam cuaca panas. Atau toko kelontong mungkin mengurangi pesanan selama waktu-waktu tertentu ketika banyak pelanggan akan meninggalkan kota pada hari libur. Semua analisis ini bersifat ad hoc, tergantung pada pemahaman dan pengetahuan pedagang tentang pasar. Perusahaan-perusahaan besar menerapkan logika yang sama untuk memutuskan berapa banyak mobil atau kemeja yang akan diproduksi, atau apakah akan berinvestasi di pabrik dan peralatan baru, meskipun dengan keputusan yang lebih mahal hukuman karena salah menebak lebih berat.

Analisis pedagang kelontong dibatasi oleh data yang tersedia. Membuka toko kedua di wilayah yang lebih sadar akan kesehatan dapat membuat toko kelontong menyimpan lebih banyak selada dan lebih sedikit donat. Penjual kelontong mungkin berhipotesis hubungan antara jumlah mobil model tertentu yang melewati toko dan jumlah keju mahal yang dijual, tetapi menghitung mobil tidak mungkin. Mengetahui bahwa jumlah kasus flu yang dilaporkan ke dokter lokal meningkat mungkin mempengaruhi berapa banyak kotak tisu yang harus disimpan, tetapi toko kelontong tidak memiliki akses ke data dokter. Parahnya, penjual kelontong tidak tahu apa yang dipikirkan pelanggan.

Data pelanggan sudah tersedia jika hanya seseorang yang memiliki kemampuan untuk mengumpulkan, menyimpan, dan menganalisisnya. Kekuatan komputasi, diukur dengan komputasi per detik, telah meningkat secara eksponensial sejak diperkenalkannya komputer, dan jumlah penyimpanan yang akan dibeli dengan jumlah uang tertentu juga meroket.

Dimulai pada pertengahan 2000-an, perusahaan mulai mengumpulkan dan menganalisis jenis informasi baru. Perusahaan besar seperti Amazon, GE, dan AT&T banyak berinvestasi dalam mempelajari lebih banyak tentang pelanggan mereka. Satu wilayah geografis memiliki penggemar setia tim olahraga, yang lain memiliki lebih banyak pendukung satu partai politik secara tidak proporsional, dan yang lain memiliki pendapatan yang dapat dibelanjakan tinggi. Studi Kasus 4-3 memberikan contoh penggunaan big data untuk memprediksi kejahatan.

Studi Kasus 4-3 : Polisi Menggunakan Data Mining untuk Memprediksi (dan Mencegah) Kejahatan

Kepala Polisi Rodney Monroe dari Departemen Kepolisian Charlotte-Mecklenburg, Carolina Utara menggunakan teknologi untuk mengurangi kejahatan. Dia berpendapat bahwa pengukuran dan analisis adalah kunci untuk pekerjaan polisi yang lebih cerdas.

Sebagai Kepala Polisi di Richmond, Virginia, pada tahun 2005 ia memperkenalkan analisis data dari sumber yang tampaknya tidak terkait, seperti cuaca, volume lalu lintas, hari dalam seminggu, hari gajian, serta data polisi yang lebih umum seperti laporan kejahatan dan panggilan tanggap darurat. Dia menemukan (tidak mengherankan) bahwa perampokan cenderung melonjak pada hari gajian dari majikan besar di daerah di mana lebih sedikit penduduk memiliki rekening bank dan dengan demikian menggunakan bisnis pencairan cek etalase. Orang-orang yang berjalan-jalan dengan uang tunai dalam jumlah besar akan menjadi target yang menarik. Dengan menganalisis data tersebut dan menargetkan aktivitas kepolisian ke situasi yang berpotensi tinggi, Departemen Kepolisian Richmond melihat kejahatan turun 21 persen pada 2005–2006 dan 19 persen lainnya pada 2006–2004. Sejak pindah ke Charlotte, Chief Monroe telah menerapkan pendekatan serupa dan telah melihat penurunan 20 persen dalam kejahatan kekerasan dan pengurangan 30 persen dalam kejahatan properti, menurut sebuah laporan di Computerworld (24 Oktober 2013).

Departemen kepolisian lainnya menggunakan alat seperti PredPol, yang dikembangkan di University of California Los Angeles. PredPol didasarkan pada algoritma yang sama yang digunakan untuk memprediksi lokasi gempa susulan dari gempa bumi. Alat ini mengembangkan prediksi untuk kotak persegi berukuran 500 kaki (150 m), menunjukkan kepada petugas daerah mana yang paling berisiko terhadap jenis kejahatan tertentu; polisi kemudian dapat fokus berpatroli di tempat-tempat yang paling mungkin.

PredPol bukan tanpa masalah, seperti yang ditunjukkan Jennifer Bachner. Ketergantungan yang berlebihan pada teknologi, memisahkan petugas dari masyarakat (sehingga petugas berinteraksi dengan program alih-alih mengembangkan petunjuk dengan berbicara dengan anggota masyarakat), masalah privasi, kebenaran data, dan keamanan dalam mengelola data adalah semua keterbatasan dalam menggunakan analisis untuk menyarankan titik rawan bagi polisi perhatian.

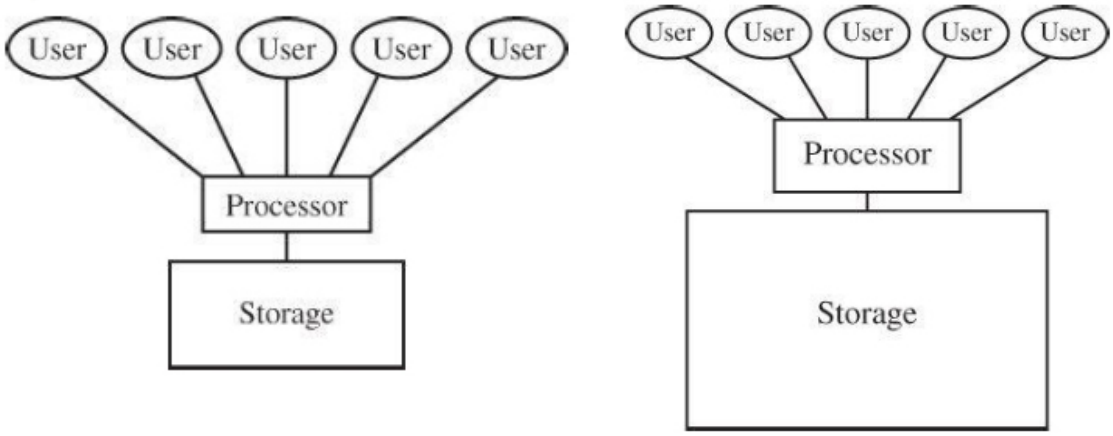
Departemen kepolisian menggunakan strategi dan teknik yang sama seperti pengecer, produsen, dan bisnis lainnya. Dihadapkan dengan sumber daya yang terbatas, kapten polisi ingin mengerahkan petugas di mana dan kapan mereka dapat melakukan yang terbaik—mencegah kejahatan daripada mencari penjahat setelah kejadian. Menyimpulkan pola dari data masa lalu membantu manajer membuat pilihan berdasarkan informasi.

Penggunaan data dalam jumlah besar dari berbagai sumber sering disebut sebagai big data. Big data berbeda dari data "kecil" yang lebih konvensional karena big data berasal dari sumber di luar perusahaan dan tidak dihasilkan hanya oleh sistem internal organisasi itu sendiri. Itu bisa berasal dari sensor dan media sosial serta rekaman video dan audio. Itu bisa berasal dari database pemerintah, analisis pasar, dan laporan pelanggan. Setiap kueri penelusuran membantu menentukan tren yang bernilai tinggi bagi perusahaan. Media sosial memberi perusahaan wawasan tentang produk, layanan, dan preferensi apa yang dibagikan orang dengan teman mereka.

Semua penggunaan data untuk memprediksi perilaku ini sangat berharga karena perusahaan memutuskan bagaimana mengalokasikan sumber daya. Jadi sekarang toko kelontong tidak hanya melihat ramalan cuaca sepuluh hari tetapi menggunakan lebih banyak data untuk memprediksi dengan lebih akurat berapa banyak selada yang akan dijual. Big data menjadi masalah besar. Dan seperti banyak teknologi yang sangat populer, fungsionalitasnya melebihi keamanan. Di bagian berikutnya kami menjelaskan secara singkat pendekatan big data, menunjukkan beberapa keterbatasan keamanannya, dan menyarankan tindakan pencegahan potensial.

Kerangka Aplikasi Big Data: Apache Hadoop

Seperti yang ditunjukkan pada Gambar 4-5, model komputasi konvensional memiliki pengguna yang berinteraksi dengan prosesor yang dapat mengakses penyimpanan. Untuk memperluas sistem seperti itu, seperti yang ditunjukkan pada Gambar 4-6, penyimpanan dapat ditingkatkan dengan menambahkan lebih banyak disk ke array disk, misalnya. Namun, ada batasan implisit tentang seberapa jauh penyimpanan dapat tumbuh tanpa mengalami penundaan kinerja yang serius: Pada titik tertentu, perangkat terbesar di pasar akan terisi dan arsitektur yang berbeda diperlukan. Prosesor baru dapat memberikan kecepatan yang lebih besar, tetapi sekali lagi, teknologi yang ada memiliki batasnya, dan kinerja yang lebih tinggi cenderung lebih mahal secara tidak proporsional. Selanjutnya, satu prosesor dan satu array penyimpanan menjadi titik potensial kegagalan bencana. Big data membutuhkan arsitektur yang dapat dengan mudah menskalakan ke kapasitas yang hampir tidak terbatas.



Gambar 4.5 dan 4-6 Arsitektur Komputasi Konvensional dan Skala Arsitektur Konvensional dengan Menambahkan komponen yang Lebih Banyak, Lebih Besar, atau Lebih Cepat

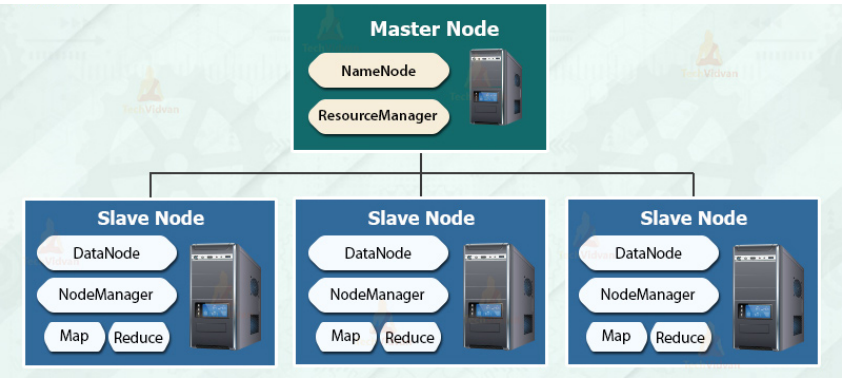
Kerangka kerja Apache Hadoop (lihat http://hadoop.apache.org) adalah lingkungan perangkat lunak untuk menjalankan proyek big data. Pengguna seperti Yahoo!, LinkedIn, dan Twitter menggunakan cluster Hadoop untuk mengelola data yang mereka kumpulkan.Hadoop digadang-gadang mampu

menyelesaiakan permasalahan yang berkaitan dengan data dengan jumlah yang sangat besar atau Big Data. Dengan banyaknya aliran data dalam perkembangan internet saat ini, Hadoop dapat menjadi solusi saat diperlukan model penyimpanan dan pengelolaan data dalam jumlah yang sangat besar. Selain itu, dengan adanya variasi data yang sangat banyak serta kebutuhan akses data yang harus cepat pula, Hadoop diharapkan dapat menyelesaikan permasalahan tersebut.

Beberapa perusahaan besar menggunakan Hadoop untuk mengelola data mereka dalam jumlah sangat besar. Perusahaan tersebut diantaranya Yahoo! dan Facebook dengan klaim bahwa Facebook memiliki Cluster Hadoop terbesar di dunia, yakni per 13 Juni 2012 mereka memiliki 100 petabyte dan per tanggal 8 November 2012, penggunaan data di Facebook naik kurang lebih setengah Petabyte per hari. 1 Petabyte setara dengan 1.000.000 Gigabyte. Tidak heran, karena jumlah pengguna Faecbook di seluruh dunia hampir mencapai 2 milyar. Bayangkan jumlah data yang mengalir dan disimpan setiap harinya.

Hadoop mendukung penyimpanan dan pemrosesan data terdistribusi, berbagai platform komputasi dari berbagai jenis, redundansi, dan akses bersamaan. Ini awalnya dibangun untuk proyek yang melibatkan perayap web, agen otonom yang melintasi Internet dan membangun indeks untuk mesin pencari web. Seperti yang dapat Anda bayangkan, jumlah halaman web dan deskriptor konten pada halaman tersebut sangat banyak. Berbeda dengan database relasional yang sangat terstruktur, data pada konten web cenderung memiliki sedikit interkoneksi dan struktur sederhana yang oleh sebagian orang disebut datar. Dalam situasi seperti itu, memberikan beberapa hasil dengan cepat lebih penting daripada memberikan jawaban yang paling komprehensif secara perlahan.

Sebuah model grafis dari arsitektur Hadoop ditunjukkan pada Gambar 4-4. NameNode bertanggung jawab untuk mereplikasi data dan melacak tempat item data disimpan. Replikasi data mendukung toleransi kesalahan dan integritas.



Gambar 4-7 Arsitektur Hadoop

Hadoop melibatkan tahap yang disebut peta-reduksi, di mana data pertama dipetakan untuk menemukan data umum dan kemudian dikurangi sesuai dengan bagian umum. Hadoop kemudian mendukung penggunaan terdistribusi dari data yang dikurangi itu. Komputasi tidak memerlukan biaya apa pun saat mesin dalam keadaan idle; yaitu, mesin yang digunakan 85 persen dari waktunya memiliki 15 persen waktu tidak terpakai yang dapat digunakan tanpa biaya tambahan. Pendekatan Hadoop menggerakkan komputasi di lingkungan terdistribusi untuk memanfaatkan sumber daya komputasi yang kurang dimanfaatkan.

Model Hadoop dikembangkan untuk lingkungan data terbuka yang dibagikan oleh semua orang. Dengan demikian, ia tidak memiliki mekanisme untuk kontrol akses, pemeriksaan kebenaran, privasi, identifikasi atau otentikasi pengguna, pencatatan tindakan, atau hak istimewa terbatas—semua primitif yang mungkin Anda harapkan di lingkungan aman apa pun. Model keamanan awal adalah pemisahan total: Big data diproses di lingkungan yang terpisah dan tepercaya oleh hanya pengguna tepercaya pada mesin khusus. Model penggunaan ini mirip dengan instalasi komputasi mainframe paling awal (1940-an–50-an) dan dengan niat asli (1960-an–70-an) untuk sistem operasi Unix: Lingkungan tertutup di mana semua pengguna tahu dan mempercayai semua orang lain dan tidak perlu mengecualikan beberapa pengguna dari beberapa data. Seiring waktu, desainer menerapkan keamanan untuk komputer mainframe dan Unix, meskipun menambahkan keamanan itu menantang.

Big data menimbulkan masalah keamanan karena jumlah data yang dipertimbangkan: Melindungi puluhan atau ratusan item data umumnya melibatkan lebih sedikit risiko dan kesulitan daripada miliaran atau triliunan. Namun, banyak masalah perlindungan yang sama seperti di domain lain: Mengamankan data, melindungi privasi, dan memastikan integritas adalah masalah untuk komputer pengguna tunggal, sistem komputasi multipengguna, jaringan, penyedia cloud, dan aplikasi terdistribusi, serta big data prosesor.

Di bagian ini kami mencantumkan masalah keamanan dalam big data, sering kali hanya menunjukkan bahwa masalah tersebut merupakan contoh dari masalah keamanan yang lebih umum yang muncul di bagian lain buku ini. Dan karena big data adalah bidang yang baru muncul, kami mencantumkan masalah dan bukan solusi. Dari contoh otorisasi, kontrol akses, dan sejenisnya yang ditunjukkan di tempat lain dalam buku ini, Anda dapat mengusulkan alat dan teknik yang berlaku untuk masalah ini.

Privacy

Orang dapat berargumen bahwa big data tidak terkait dengan privasi: Pemroses data tidak mengumpulkan data tetapi hanya menyaring data yang ada. Jadi, misalnya, pengguna big data tidak bertanggung jawab atas fakta bahwa sebuah toko melacak pembelian pelanggan melalui kartu loyalitas pelanggan atau bahwa toko tersebut kemudian menjual data pelacakan tersebut kepada orang-orang yang tertarik untuk mempelajari tren. Di sisi lain, kemampuan pengumpulan dan korelasi big data telah memungkinkan penggunaan seperti itu dan bahkan menguntungkan. Terlepas dari itu, masalah privasi muncul. Kami mempelajari privasi secara umum di Bab 5.

Privacy-Preserving Analytics

Anonimisasi adalah metode penting untuk menyeimbangkan masalah privasi dengan tujuan fungsional. Para peneliti ingin tahu, misalnya, apakah merokok berkorelasi dengan kanker paru-paru. Untuk mempelajarinya, peneliti membutuhkan populasi yang terdiri dari perokok dan bukan perokok, beserta status kanker paru-paru mereka. Siapa subjeknya tidak penting. Secara teori, kumpulan besar sejarah kasus membuatnya tidak mungkin untuk menghubungkan subjek dengan identitas yang sebenarnya. Dalam hal itu, big data harus meningkatkan privasi dengan meningkatkan kumpulan subjek secara signifikan, sehingga meningkatkan jumlah subjek dan identitas.

Sayangnya, big data juga berkontribusi pada masalah karena menyediakan lebih banyak data yang dapat mengidentifikasi individu tertentu: Lebih banyak istilah data mengurangi jumlah orang yang

cocok dengan semua atribut. Siapa pasien kanker yang tinggal di Maple Street, 55 tahun, di rumah dengan dua anjing, berlangganan majalah Bicycling, yang sering menelepon ke Rio de Janeiro?

Seperti yang dijelaskan sebelumnya dalam bab ini, menambahkan noise dan menghapus data pengenal dapat membantu menjaga privasi. Kebisingan mungkin termasuk data palsu: satu kucing dan tidak ada anjing, misalnya; menghapus usia mungkin juga mempersulit untuk menyimpulkan identitas orang tersebut dari data. Namun, pendekatan yang membatasi data adalah solusi yang tidak lengkap.

Kontrol Akses Granular (Granular Access Control)

Big data sering menggunakan kumpulan data tidak terstruktur, datar, Tabel dua dimensi. Kontrol akses, jika ada, dikenakan pada tingkat file: Seluruh file dapat atau tidak dapat diakses oleh pengguna. Pendekatan seperti itu sangat cocok dengan arsitektur big data seperti Hadoop, di mana seluruh file direplikasi dan dianalisis oleh kumpulan DataNodes yang bekerja secara paralel.

Seperti yang pertama kali diangkat di Bab 2 dan diperkuat di sepanjang sisa buku ini, kontrol akses yang halus membantu meningkatkan keamanan (dan privasi) dengan mengizinkan hak istimewa paling rendah: Sebuah proses hanya dapat mengakses objek-objek tersebut atau data spesifik yang konsisten dengan kebijakan keamanan dan diperlukan untuk tugas yang ada.

Keamanan (Security)

Keamanan data adalah tantangan utama kedua dari arsitektur big data. Pertimbangkan kembali triad C-I-A dari Bab 1. Kerahasiaan terkait erat dengan privasi, tetapi ada masalah kerahasiaan lainnya. Big data sering kali melibatkan uang besar: Pengumpul data membayar untuk memanen data yang kemudian mereka jual. Pemanen data menuai keuntungan berkelanjutan dengan mengumpulkan data sekali dan menjualnya berkali-kali ke pembeli yang berbeda. Perusahaan penelusuran seperti Google mengumpulkan data dari istilah penelusuran pengguna (seperti “hotel San Francisco”) yang kemudian dapat mereka jual ke Hilton, Marriott, dan Sheraton, serta maskapai penerbangan, restoran, perusahaan tur, dan sebagainya. Jika Google menjual data ke Hilton dan Hilton kemudian menjualnya kembali ke Marriott dan lainnya, Google kehilangan aliran pendapatan berikutnya. Dengan demikian, Google ingin mengontrol kerahasiaan data miliknya. Integritas data — bahwa itu benar dan utuh — penting, seperti halnya memastikan ketersediaan data. Dengan demikian, ketiga elemen dari triad penting untuk big data.

Di sini kami mencantumkan beberapa masalah keamanan big data.

Secure Data Storage

Dalam model Hadoop, item data direplikasi dan disimpan di lokasi yang nyaman. Jika satu penyimpanan data menjadi sangat penuh sehingga kinerjanya menurun, pengontrol secara otomatis membagi data dan memindahkan beberapa data ke tempat lain. Pengembang aplikasi umumnya tidak tahu, apalagi peduli, di mana data disimpan secara fisik.

Penyedia penyimpanan data mencari biaya terendah, dan hosting data di pusat Kota New York, London, atau Tokyo kemungkinan akan jauh lebih mahal daripada Alcoa, Tennessee, atau Pateley

Bridge, Inggris. Dengan daya dan infrastruktur jaringan yang memadai, tempat mana pun sama bagusnya dengan tempat lain.

Hampir. Misalkan data disimpan di wilayah yang sensitif secara politik, bahkan zona perang: Serangan mortir atau dampak dari serangan rudal tidak diinginkan. Atau misalkan data disimpan di negara yang penguasanya memutuskan untuk menasionalisasi semua aset asing. Atau pertimbangkan lokasi di mana warga yang lapar menyerbu instalasi untuk mencuri apa pun yang nantinya dapat mereka jual kembali sebagai barang bekas untuk membeli makanan. Dalam bab lain kami telah menjelaskan kontrol akses dalam hal kartu magnetik dan pembaca sidik jari untuk orang yang sopan, tetapi kontrol akses secara global juga harus menangani masalah fisik dan politik.

Log Transaksi dan Audit

Log aktivitas penting untuk memantau siapa yang melakukan apa. Tinjauan log audit dapat membantu administrator mengatur izin akses, dan log juga membantu menentukan tingkat kerusakan jika terjadi kesalahan atau pelanggaran keamanan. Namun, menentukan apa yang akan dilacak itu sulit: Pelacakan yang terlalu sedikit dapat membatasi kegunaan log akses, tetapi terlalu banyak data dapat membebani manusia dan teknologi, sehingga sulit untuk menemukan jarum pepatah di tumpukan jerami akses. Granularitas data memengaruhi volume data pelacakan. Mungkin tidak ada gunanya mengetahui bahwa pengguna A mengakses database D pada hari Senin, ketika yang akan sangat membantu adalah mengetahui bahwa A memodifikasi catatan 2 dan 17, atau bahkan A mengubah bidang alamat di 2, dan bidang gaji 14. data pelacakan yang tersedia bergantung pada perincian akses yang direkam untuk data tersebut.

Dalam aplikasi big data, unit akses sering berupa file, jadi log hanya akan mencatat bahwa aplikasi mengakses file F, bukan data spesifik dalam F.

Pemantauan Keamanan Real-time

Seperti dijelaskan dalam Bab 6, deteksi intrusi, dan sistem perlindungan memungkinkan administrator untuk memantau aktivitas, mungkin mendeteksi perilaku atau serangan anomali, dan menerapkan tindakan pencegahan saat insiden sedang berlangsung. Arsitektur big data melibatkan pergerakan data dan komputasi yang gesit, tetapi jaringan penghubung mungkin berupa jaringan bersama yang besar, seringkali Internet. Pemantauan keamanan waktu nyata tidak dimaksudkan untuk arsitektur jaringan yang kompleks, dibagi, dan lancar.

Integritas

Akhirnya, integritas layak mendapat pertimbangan tersendiri terpisah dari kerahasiaan dan ketersediaan, karena kebenaran, akurasi, dan keandalan sangat penting bagi pengguna data. Di bagian ini kami mengidentifikasi beberapa masalah integritas yang relevan untuk big data.

Akurasi Data

Anda mungkin pernah mengalami beberapa bagian data pribadi yang salah dimasukkan ke dalam database. Kami telah menyajikan contoh sebelumnya di bab 510 Thames Street ini, bukan 519. Bagian yang membuat frustrasi adalah, coba semau Anda, Anda sering kali tidak akan pernah bisa melacak tempat terakhir di mana nomornya salah. Jadi 510 terus bermunculan selama bertahun-tahun. Kami menggunakan contoh sebelumnya sebagai aspek integritas elemen.

Kumpulan big data memiliki masalah dengan integritas, seperti yang baru saja kami jelaskan. Namun, karakteristik lain dari penggunaan big data memperumit situasi. Big data sering menggunakan banyak aliran data yang dikumpulkan dari banyak sumber, misalnya, pengenalan foto pelat nomor mobil, transkripsi manusia dari catatan publik tertulis, pengenalan suara dari rekaman, dan input dari formulir tulisan tangan, yang semuanya rentan terhadap kesalahan.

Basis data yang kaya dan terstruktur sering kali memiliki satu atau lebih kunci pengenal, seperti nomor telepon, nomor asuransi nasional, nomor rekening, tanggal lahir, atau beberapa item data solid lainnya untuk menggabungkan dua kumpulan data. Kumpulan big data cenderung tidak memiliki kunci yang kuat, sehingga mereka digabungkan pada atribut yang lebih lemah, seperti nama (yang dapat disajikan dalam beberapa bentuk dan salah eja dalam lebih banyak cara). Keakuratan hasil dari gabungan tersebut lebih rendah.

Untuk banyak kegunaan, akurasi tinggi big data tidak penting: Apakah 90, 100, atau 110 orang dari 500 di lingkungan memiliki hewan peliharaan kurang penting daripada kepemilikan hewan peliharaan dalam kisaran 15 persen hingga 25 persen. Namun, pengguna perlu menghargai tingkat akurasi yang terbatas ini.

Sumber Asal (Source Provenance)

Big data biasanya melibatkan pengumpulan dan analisis data dari beberapa sumber. Seperti yang baru saja kami tunjukkan, kumpulan data akan memiliki tingkat kualitas yang berbeda tergantung dari mana data itu berasal. Aplikasi big data harus mengontrol variabilitas tersebut, meskipun model big data tidak selalu memberikan aplikasi cara untuk mempelajari sumber data yang tepat, atau bahkan sifat sumbernya. Dengan demikian, penulis aplikasi tidak dapat dengan mudah menjelaskan asal data dalam hasil yang mereka hasilkan.

Pemfilteran End-point dan Validasi

Terakhir, setelah aplikasi memproses data dari kumpulan big data, aplikasi mungkin ingin memfilter dan memvalidasi hasilnya. Kerangka kerja big data saat ini tidak mendukung revisi dan manipulasi data semacam itu.

Tambahan Keamanan untuk Aplikasi Big Data

Seperti yang dijelaskan dalam bab lain, menambahkan keamanan ke produk atau sistem yang ada jarang merupakan strategi yang berhasil. Namun demikian, terkadang itu adalah satu-satunya pendekatan yang tersedia; tidak hanya spesifikasi dan desainnya yang lengkap, tetapi satu atau lebih versi produk sedang digunakan. Seperti halnya dengan Hadoop dan kerangka kerja aplikasi big data berpemilik lainnya: Desain dan implementasi produk selesai sebelum keamanan dipertimbangkan secara serius. Insinyur keamanan sekarang telah merekomendasikan perubahan dan penambahan pada Hadoop untuk mendukung alat dan teknik keamanan yang terkenal.

Mode aman Hadoop dijelaskan di situs web Hadoop https://hadoop.Apache.org/docs/current/hadoop-project-dist/hadoop-common/SecureMode.html dan dalam buku putih [OMA09] dari Yahoo! Tim, dipimpin oleh Owen O'Malley. Proposal untuk fungsi keamanan ini menyarankan fitur keamanan untuk disertakan dalam kerangka Hadoop. Buku putih mengidentifikasi dua lubang keamanan untuk diatasi: kurangnya otentikasi pengguna (dan identifikasi) dan kurangnya kontrol akses ke blok data.

Untuk mengatasi kesalahan ini, tim mengusulkan mode aman yang melibatkan ekstensi berikut ke Hadoop:

- otentikasi untuk perangkat web pengguna akhir
- otentikasi timbal balik (pengguna–proses–layanan Hadoop) dengan Kerberos
- kontrol akses ke file dalam sistem file Hadoop
- token delegasi untuk otentikasi berkelanjutan antara klien internal dan layanan internal
- token pekerjaan untuk mendistribusikan otorisasi akses ke beberapa platform terdistribusi yang secara kolektif menerapkan pencarian data di seluruh penyimpanan data yang berbeda
- Enkripsi SSL untuk lalu lintas jaringan

Kekuatan pendorong dalam desain ini adalah kinerja: Pengembang memutuskan bahwa fungsi keamanan tidak dapat mengurangi kinerja lebih dari 3 persen. Pada tahun 2008, Yahoo! jaringan pengembang melaporkan bahwa cluster Hadoop telah berhasil mengurutkan satu terabyte data dalam 209 detik. Perhatikan bahwa untuk melindungi data, tim O'Malley tidak mau memperlambat hasil itu dengan waktu yang lebih singkat daripada yang Anda perlukan untuk membaca kalimat ini. (209 detik mengalahkan rekor sebelumnya 297 detik, pengurangan lebih dari 26 persen.)

Andrew Becherer mempresentasikan makalah [BEC10] di BlackHat 2010 yang mencatat beberapa kekurangan ekstensi keamanan Hadoop tahun 2009. Dia mengkritik tingkat perlindungan default, diukur dengan kekuatan kriptografi yang digunakan secara default bagi pengguna yang memilih untuk menggunakan keamanan perangkat tambahan. Lebih jauh, dia menunjukkan bahwa desainnya melibatkan pendistribusian kunci yang sama ke ratusan atau ribuan server yang berpotensi di lokasi terdistribusi. Jumlah besar salinan dari kunci yang sama meningkatkan risiko bahwa satu salinan kunci mungkin jatuh ke tangan penyerang. Model keamanan juga mengotentikasi proses tertentu berdasarkan alamat IP di mana mereka (tampaknya) di-host. Seperti dijelaskan dalam Bab 6, alamat IP dapat dipalsukan, dan kerentanan itu tidak dapat diabaikan ketika sistem Hadoop di-host di Internet.

Peningkatan keamanan untuk Hadoop jelas merupakan langkah maju yang disambut baik, tetapi karena terlalu banyak masalah keamanan yang telah kami jelaskan dalam buku ini, solusi terbatas jika keamanan ditambahkan setelah fakta, dikorbankan untuk kinerja, dan tidak memiliki desain dan analisis yang luas sebelum pendekatan keamanan dikodifikasi.

4.6 Kesimpulan

Dalam bab ini kita telah mengeksplorasi perlindungan data. Selain itu, kami telah meninjau beberapa masalah keamanan komputasi awan, nama baru untuk penyimpanan dan pemrosesan data yang didistribusikan secara luas.

Melindungi data sangat rumit karena pengguna dapat mengumpulkan dan mengumpulkan data di luar sistem komputasi. Jadi, meskipun kami mungkin menyiapkan pendekatan kontrol akses yang solid dan pelacakan lengkap dari data apa yang diakses setiap individu, tindakan di luar sistem sepenuhnya di luar kendali kami.

1. Kerjakan soal-soal berikut ini!
   (a) Dalam lingkungan di mana beberapa pengguna berbagi akses ke satu database, dapatkah satu pengguna memblokir akses orang lain untuk jangka waktu yang tidak terbatas? (Situasi ini disebut penundaan tidak terbatas.)
   (b) Jelaskan skenario di mana dua pengguna dapat menyebabkan penundaan tidak terbatas satu sama lain.
   (c) Jelaskan skenario di mana satu pengguna dapat menyebabkan penundaan yang tidak terbatas dari semua pengguna.
2. Dengan menggunakan komitmen dua langkah yang disajikan di awal bab ini, jelaskan bagaimana menghindari menetapkan satu kursi untuk dua orang, seperti dalam contoh maskapai penerbangan. Yaitu, buatlah daftar dengan tepat langkah-langkah mana yang harus diikuti oleh pengelola basis data dalam menetapkan penumpang ke tempat duduk.
3. Misalkan seorang manajer basis data mengizinkan penyarangan satu transaksi di dalam transaksi lainnya. Artinya, setelah memperbarui bagian dari satu catatan, DBMS akan memungkinkan Anda untuk memilih catatan lain, memperbaruinya, dan kemudian melakukan pembaruan lebih lanjut pada catatan pertama. Apa efek bersarang pada integritas database? Sarankan mekanisme yang memungkinkan pembuatan sarang.
4. Dapatkah database berisi dua catatan identik tanpa efek negatif pada integritas database? Mengapa atau mengapa tidak?
5. Beberapa sistem operasi melakukan buffered I/O. Dalam skema ini, permintaan keluaran diterima dari pengguna dan pengguna diberitahu tentang penyelesaian I/O normal. Namun, operasi tulis fisik yang sebenarnya dilakukan kemudian, pada waktu yang sesuai dengan sistem operasi. Diskusikan efek buffered I/O pada integritas dalam DBMS.
6. Transaksi database mengimplementasikan perintah "set STATUS ke 'CURRENT' di semua catatan di mana BALANCE-OWED = 0." (a) Jelaskan bagaimana transaksi itu akan dilakukan dengan komit dua langkah yang dijelaskan dalam bab ini. (b) Misalkan relasi dari mana perintah itu dibentuk adalah (ID-PELANGGAN,STATUS) dan (ID-PELANGGAN,BALANCE-OWED). Bagaimana transaksi akan dilakukan? (c) Misalkan hubungan dari mana perintah itu dibentuk adalah (ID-PELANGGAN,STATUS), (ID KREDIT,ID-PELANGGAN), (ID-KREDIT, BALANCE-OWED). Bagaimana transaksi akan dilakukan?
4. Tunjukkan bahwa jika paritas longitudinal digunakan sebagai kode deteksi kesalahan, nilai dalam database masih dapat dimodifikasi tanpa deteksi. (Paritas longitudinal dihitung untuk bit ke-n dari setiap byte; yaitu, satu bit paritas dihitung dan dipertahankan untuk semua bit di posisi ke-0, bit paritas lain untuk semua bit di posisi ke-1, dll.)
8. Misalkan kueri Q1 memperoleh median ml dari kumpulan nilai S1, dan kueri Q2 memperoleh median m2 dari subset S2 dari S1. Jika m1 < m2, apa yang dapat disimpulkan tentang S1, S2, dan elemen-elemen S1 yang tidak ada dalam S2?
9. Pengungkapan jumlah semua bantuan keuangan untuk siswa di asrama Smith tidak sensitif karena tidak ada siswa individu yang dikaitkan dengan jumlah. Demikian pula, daftar nama siswa yang menerima bantuan keuangan tidak sensitif karena tidak ada jumlah yang ditentukan. Namun, kombinasi dari dua daftar ini mengungkapkan jumlah untuk seorang siswa jika hanya

satu siswa di asrama Smith yang menerima bantuan. Perhitungan apa yang harus dilakukan oleh sistem manajemen basis data untuk menentukan bahwa daftar nama mungkin mengungkapkan data sensitif? Catatan apa yang harus dipertahankan oleh sistem manajemen basis data tentang apa yang diketahui pengguna berbeda untuk menentukan bahwa daftar nama mungkin mengungkapkan data sensitif?

10. Salah satu pendekatan yang disarankan untuk memastikan privasi adalah penolakan hasil kecil, di mana sistem menolak (tidak mengembalikan hasil dari) permintaan apa pun, yang hasilnya berasal dari sejumlah kecil, misalnya, lima catatan. Perlihatkan cara mendapatkan data sensitif dengan hanya menggunakan kueri yang berasal dari enam rekaman.
11. Respon “nilai sensitif; respon ditekan” itu sendiri merupakan pengungkapan. Sarankan cara di mana sistem manajemen basis data dapat menekan respons yang mengungkapkan informasi sensitif tanpa mengungkapkan bahwa respons terhadap kueri tertentu bersifat sensitif.
12. Sebutkan situasi di mana sensitivitas suatu agregat lebih besar daripada nilai-nilai penyusunnya. Mengutip situasi di mana sensitivitas agregat kurang dari nilai-nilai konstituennya.

Bab 5

Kerahasiaan Privasi

Sampai saat ini masalah privacy masih menjadi pro dan kontra. Pihak pro yang diwakili oleh ahli hukum dan masyarakat yang peduli atas hak – hak mereka, mengambil sudut pandang privasi sebagai sebuah kekayaan intelektual atau hak milik pribadi. Pemerintah sebagai pihak yang kontra mengedepankan alasan – alasan keamanan negara sebagai pembenaran terhadap aktivitas pengawasan dinamika masyarakat beserta atribut informasi yang melekat padanya.

Wujud konkritnya adalah disahkannya peraturan yang melindungi hak privasi individu. Misalnya, Amerika Serikat memiliki antara lain Privacy Act (1974), Electronic Communications Privacy Act (1986), dan Childrens's Online Privacy Protection (1994), sementara Uni Eropa memiliki European Privacy Directive 8 (1998). Dampak signifikannya terhadap perlindungan privasi baru akan terasa bilamana hukum menjadi rujukan utama dalam penanganan masalah sosial

Komputer tidak menciptakan atau bahkan menyebabkan masalah privasi; kami memilikinya jauh sebelum komputer dan mungkin bahkan sebelum bahasa tertulis. Tetapi kemampuan pemrosesan dan penyimpanan data serta transmisi berkecepatan tinggi komputer memungkinkan pengumpulan dan korelasi data yang memengaruhi privasi. Karena privasi adalah bagian dari kerahasiaan, itu adalah aspek keamanan komputer.

Privasi adalah hak asasi manusia, meskipun orang dapat secara sah tidak setuju tentang kapan atau sejauh mana privasi layak; ketidaksepakatan ini mungkin memiliki akar budaya, sejarah, atau pribadi. Tetapi pada dasarnya, hak atas privasi bergantung pada situasi di mana privasi diinginkan, kepemilikan dan kegigihan data, serta hak dan tanggung jawab hukum dari pihak-pihak yang terkena dampak. Selain itu, seperti halnya kerahasiaan, integritas, dan ketersediaan dapat bertentangan, demikian juga privasi dan aspek keamanan lainnya.

Privasi adalah kemampuan satu atau sekelompok individu untuk mempertahankan kehidupan dan urusan personalnya dari publik, atau untuk mengontrol arus informasi mengenai diri mereka. Karena privasi sebagai topik lebih luas daripada implikasinya terhadap keamanan, kami membatasi diri dalam bab ini hanya untuk masalah privasi yang terkait erat dengan keamanan komputer. Jadi, dalam bab ini kita melihat arti dari privasi informasi. Kami meninjau kembali identifikasi dan otentikasi, dua aspek komputasi yang sudah dikenal yang memiliki implikasi privasi yang signifikan. Kami mempelajari bagaimana privasi berhubungan dengan Internet, khususnya dalam email dan akses web. Akhirnya, kami menyelidiki beberapa teknologi berbasis komputer yang muncul di mana privasi penting.

Privasi adalah kebebasan pribadi. Sebagai contoh, ketika anda mengirim surat kepada teman, tentu dengan amplop tertutup bukan? Walaupun mungkin isinya tidak penting. Mengapa anda tidak mengirimkannya dengan kartu pos? Nah dari contoh ini, anda tentu sudah bisa menarik kesimpulan yang intinya, anda tidak ingin orang lain membaca surat anda.

5.1 Konsep Privasi

Konsep privasi menjadi semakin penting dibicarakan di era seperti sekarang, karena kemunculan teknologi yang mampu merekam dan menyimpan bentuk baru dari informasi pribadi, contohnya sidik jari, wajah dan bahkan retina mata seseorang. Proses merekam dan menyimpan tersebut tidak hanya dilakukan dalam skala kecil tapi juga skala besar. Bayangkan berapa banyak pengguna ponsel seperti iPhone yang menggunakan sidik jari atau pengguna Facebook yang mengidentifikasi atau tag wajah seseorang (lihat ilustrasi di atas). Penggunaan internet yang digabungkan dengan pemakaian ponsel semakin menyebar ke berbagai lini kehidupan juga melancarkan jalan tersebut, bahkan dengan kemampuan penyebaran yang lebih luas dari yang pernah kita bayangkan.

5.1.1 Aspek- Aspek Privacy Information (Security)

Sebuah informasi harus aman, dalam arti hanya diakses oleh pihak – pihak yang berkepentingan saja sesuai dengan sifat dan tujuan dari informasi tersebut. Privasi informasi memiliki tiga aspek: data sensitif, pihak yang terpengaruh, dan pengungkapan yang terkendali. Sebenarnya, aspek-aspek ini mirip dengan tiga elemen kontrol akses dari Bab 2: subjek, objek, dan hak akses. Kami membahas ketiganya secara bergantian.

Pengungkapan yang Terkendali

Apa itu privasi? Definisi kerja yang baik adalah bahwa privasi adalah hak untuk mengontrol siapa yang mengetahui aspek-aspek tertentu tentang Anda, komunikasi Anda, dan aktivitas Anda. Dengan kata lain, Anda secara sukarela memilih siapa yang dapat mengetahui hal-hal apa tentang Anda. Orang mungkin menanyakan nomor telepon Anda: montir mobil Anda, pegawai toko, otoritas pajak Anda,

kontak bisnis baru, atau teman baru. Dalam setiap kasus, Anda mempertimbangkan mengapa orang tersebut menginginkan nomor tersebut dan kemudian memutuskan apakah akan memberikannya. Tetapi poin kuncinya adalah Anda memutuskan. Jadi privasi adalah sesuatu di mana Anda dapat memiliki pengaruh yang cukup besar.

Namun, Anda tidak memiliki kendali penuh. Setelah Anda memberikan nomor Anda kepada seseorang atau suatu sistem, kendali Anda akan berkurang karena sebagian bergantung pada apa yang dilakukan orang atau sistem tersebut dengan informasi tersebut. Dalam memberikan nomor Anda, Anda mentransfer atau menyerahkan otoritas dan kendali kepada seseorang atau sesuatu yang lain. Anda dapat mengatakan "jangan berikan nomor saya kepada orang lain", "gunakan kebijaksanaan", atau "Saya sensitif tentang privasi saya", tetapi Anda tidak mengontrol orang atau sistem lain. Anda harus mempercayai orang atau sistem untuk memenuhi keinginan Anda, apakah Anda menyatakan keinginan itu secara eksplisit atau tidak. Masalah ini mirip dengan masalah propagasi keamanan komputer: Siapapun yang memiliki akses ke suatu objek dapat menyalin, mentransfer, atau menyebarkan objek itu atau isinya kepada orang lain tanpa batasan. Dan bahkan jika Anda menentukan bahwa objek tersebut harus dihapus atau dihancurkan setelah jangka waktu tertentu, Anda tidak memiliki cara untuk memverifikasi bahwa sistem atau orang tersebut benar-benar menghancurkan konten tersebut.

Data Sensitif

Seseorang menanyakan ukuran sepatu Anda. Anda dapat menjawab, “Saya orang yang sangat tertutup dan tidak dapat membayangkan mengapa Anda ingin mengetahui detail yang begitu intim” atau Anda dapat mengatakan “10C”; beberapa orang menemukan bahwa item data lebih sensitif daripada yang lain. Beberapa informasi biasanya dianggap sensitif, seperti status keuangan, data kesehatan tertentu, kejadian buruk di masa lalu seseorang, dan sejenisnya. Jadi, jika Anda mempelajari sesuatu yang Anda anggap sensitif tentang seseorang, Anda cenderung merahasiakannya, kecuali jika ada argumen yang kuat untuk mengungkapkannya. Misalnya, di banyak tempat, profesional kesehatan (tertarik pada identifikasi penyakit, penahanan, dan pencegahan) diwajibkan untuk melaporkan contoh penyakit yang sangat menular atau mematikan, bahkan jika orang yang terkena tidak ingin informasi itu dipublikasikan. Tetapi kebanyakan dari kita tidak terlalu sensitif tentang ukuran sepatu kita, jadi kita biasanya tidak melindungi informasi itu jika ditanya, atau jika kita mempelajarinya tentang orang lain. Dalam kebanyakan kasus, kami menghormati permintaan untuk melindungi informasi sensitif seseorang.
Berikut adalah contoh (tanpa urutan tertentu) jenis data yang dianggap pribadi oleh banyak orang.

- Identitas: nama, informasi identitas, kepemilikan data pribadi dan kemampuan untuk mengontrol pengungkapannya
- Komunikasi istimewa: dengan para profesional seperti pengacara, akuntan, dokter, konselor, dan pendeta
- Kesehatan: kondisi medis, penggunaan narkoba, DNA, kecenderungan genetik terhadap penyakit
- Informasi akademik dan pekerjaan: catatan sekolah, peringkat pekerjaan
- Keuangan: peringkat dan status kredit, detail bank, pinjaman terutang, catatan pembayaran, informasi pajak

- Pendapat, preferensi, dan keanggotaan: catatan pemungutan suara, pendapat yang diungkapkan, keanggotaan dalam organisasi advokasi, agama, partai politik, preferensi seksual, kebiasaan membaca, penjelajahan web, hiburan favorit, teman dekat
- Hukum: catatan kriminal, riwayat pernikahan, gugatan perdata
- Biometrik: karakteristik fisik, hasil poligraf, sidik jari
- Bukti dokumenter: surat permukaan, buku harian, puisi, korespondensi, pemikiran yang direkam
- Data lokasi: rencana perjalanan umum, lokasi saat ini, pola perjalanan
- Jejak digital: email, panggilan telepon, spam, pesan instan, tweet, dan bentuk interaksi elektronik lainnya, riwayat jejaring sosial

Privasi juga dipengaruhi oleh siapa Anda. Ketika Anda berada di sebuah ruangan dengan orang yang tidak Anda kenal, mungkin di resepsi, seseorang mungkin mendatangi Anda dan berkata, “Jadi, Anda adalah orang yang membuat kue yang indah di sana; Saya sangat menghargai keahlian Anda sebagai koki kue.” Senang rasanya mendapat pengakuan seperti itu. Sebaliknya, jika Anda seorang penyiar berita di televisi lokal setiap malam, Anda mungkin lebih suka makan malam di rumah daripada pergi ke restoran; Anda mungkin bosan dengan orang asing yang terburu-buru untuk mengatakan, "Saya melihat Anda sepanjang waktu di TV." (Banyak tokoh publik menghargai sedikit privasi yang mereka pertahankan.) Atlet juara dunia tidak dapat menghindari hasil mereka dipublikasikan, sedangkan Anda mungkin tidak ingin semua orang tahu seberapa buruk Anda menyelesaikan pertandingan atletik terakhir Anda. Budaya juga mempengaruhi apa yang orang anggap sensitif; misalnya, mendiskusikan hubungan seksual atau informasi gaji mungkin diperbolehkan di satu budaya tetapi tidak di budaya lain.

Secara umum, ekspektasi privasi seseorang bergantung pada konteks: siapa yang terpengaruh, bagaimana perasaan orang tersebut tentang publisitas, dan norma privasi yang berlaku.

Subyek yang Membutuhkan Privasi

Individu, grup, perusahaan, organisasi, dan pemerintah semuanya memiliki data yang mereka anggap sensitif. Kami menggunakan istilah seperti "subjek" dan "pemilik" untuk membedakan antara orang atau entitas yang dijelaskan oleh data dan orang atau entitas yang menyimpan data. Sejauh ini kami telah mengGambarkan privasi dari sudut pandang pribadi, di mana subjeknya adalah seseorang. Tetapi organisasi publik dan swasta juga tertarik pada privasi. Perusahaan mungkin memiliki data yang mereka anggap pribadi atau sensitif: rencana produk, pelanggan utama, margin keuntungan, dan teknologi yang baru ditemukan, sebagai contoh. Untuk perusahaan swasta, privasi biasanya berkaitan dengan mendapatkan dan mempertahankan keunggulan dalam persaingan. Organisasi lain, seperti sekolah, rumah sakit, atau badan amal, mungkin perlu melindungi data pribadi tentang siswa, pasien, atau donor mereka. Banyak organisasi juga melindungi informasi yang terkait dengan reputasi mereka; mereka mungkin ingin mengontrol berita negatif atau waktu rilis informasi yang dapat mempengaruhi harga saham atau keputusan hukum. Sebagian besar pemerintah menganggap masalah militer dan diplomatik sensitif, tetapi mereka juga mengakui tanggung jawab mereka untuk memberikan informasi yang menginformasikan wacana nasional. Pada saat yang sama, pemerintah memiliki tanggung jawab untuk melindungi dan menjaga kerahasiaan data yang mereka kumpulkan dari warga negara, seperti informasi pajak.

Privasi adalah aspek kerahasiaan. Seperti yang telah kita pelajari di seluruh buku ini, tiga tujuan keamanan kerahasiaan, integritas, dan ketersediaan dapat bertentangan, dan kerahasiaan terkadang bertentangan dengan ketersediaan. Misalnya, jika Anda memilih untuk tidak mencantumkan nomor telepon Anda dalam direktori, maka beberapa orang mungkin tidak dapat menghubungi Anda melalui telepon. Atau menolak untuk mengungkapkan data pribadi ke toko dapat mencegah Anda menerima diskon yang sering berbelanja. Jadi, penting untuk mempertimbangkan privasi tidak hanya sebagai cara untuk melindungi informasi tetapi juga sebagai penghalang yang mungkin untuk tujuan penting dan positif lainnya.

5.1.2 Masalah Privasi dalam Program Komputer

Anda mungkin memperhatikan bahwa banyak jenis data sensitif dan banyak poin tentang privasi tidak ada hubungannya dengan komputer. Anda benar sekali: Sensitivitas dan masalah ini sudah ada sebelum komputer. Komputer dan jaringan hanya memengaruhi kelayakan, kecepatan, dan jangkauan beberapa pengungkapan yang tidak diinginkan. Kantor arsip publik telah lama terbuka bagi orang untuk mempelajari data yang disimpan di sana, tetapi kapasitas penyimpanan dan kecepatan komputer telah memberi kita kemampuan untuk mengumpulkan, mencari, dan menghubungkan lebih cepat dan lebih efektif daripada sebelumnya. Dengan mesin pencari, kita dapat menemukan satu item data dari miliaran, setara dengan menemukan satu lembar kertas dari gudang yang penuh dengan kotak kertas. Selain itu, keterbukaan jaringan dan portabilitas teknologi (seperti laptop, tablet, ponsel, dan perangkat berkemampuan WiFi) telah sangat meningkatkan risiko pengungkapan yang memengaruhi privasi.

Rezgui dkk. membuat daftar delapan dimensi privasi (khususnya terkait dengan web, meskipun definisi terbawa secara alami ke jenis komputasi lainnya).

- Pengumpulan informasi: Data dikumpulkan hanya dengan pengetahuan dan persetujuan eksplisit.
- Penggunaan informasi: Data hanya digunakan untuk tujuan tertentu tertentu.
- Penyimpanan informasi: Data disimpan hanya untuk jangka waktu tertentu.
- Pengungkapan informasi: Data diungkapkan hanya kepada sekelompok orang yang berwenang.
- Keamanan informasi: Mekanisme yang tepat digunakan untuk memastikan perlindungan data.
- Kontrol akses: Semua mode akses ke semua bentuk data yang dikumpulkan dikendalikan.
- Pemantauan:Semua Logs (catatan) dipertahankan untuk menunjukkan semua akses ke data.
- Perubahan kebijakan: Kebijakan yang tidak terlalu ketat tidak pernah diterapkan setelah fakta pada data yang sudah diperoleh.

Berikut adalah masalah privasi yang muncul melalui penggunaan komputer.

Pengumpulan data (Data Collection)

Seperti yang telah kami katakan, kemajuan dalam penyimpanan komputer memungkinkan untuk menyimpan dan memanipulasi sejumlah besar catatan. Disk pada perangkat konsumen biasa diukur dalam gigabyte (109 atau 1 miliar byte), terabyte (1012 atau 1 triliun byte), petabyte (1015 atau 1 kuadriliun byte) dan exabyte (1018 atau 1 triliun byte). Pada 2012, Ngo [NGO12] melaporkan bahwa Seagate mencapai tonggak sejarah dalam kepadatan penyimpanan: satu terabyte per inci,

memungkinkan produksi hard drive 60-terabyte. Plafke [PLA13] menyoroti tim dari Pusat Mikro-Fotonik Swinburne University yang “telah mengembangkan teknik yang dapat meningkatkan kapasitas penyimpanan DVD dari standar 4,7 gigabyte (GB) hingga 1 petabyte (PB). Tekniknya tidak mengubah ukuran atau bentuk cakram, melainkan mengubah laser yang digunakan untuk membaca data cakram.” Dan Solar [SOL10] melaporkan bahwa alternatif media elektronik menjanjikan perangkat penyimpanan yang lebih besar. Dia menggambarkan sebuah kelompok di Chinese University of Hong Kong yang berhasil menggunakan bakteri umum sebagai perangkat penyimpanan yang aman. “Berdasarkan prosedur yang diuji, mereka memperkirakan kemampuan untuk menyimpan sekitar 900.000 gigabyte dalam satu gram sel bakteri. Itu setara dengan 450 hard drive, masing-masing berkapasitas 2 terabyte (2000 GB).” Untuk menempatkan angka-angka ini dalam perspektif, pertimbangkan bahwa para ilmuwan memperkirakan kapasitas otak manusia antara satu terabyte dan satu petabyte.

Pada saat yang sama kemampuan kita untuk menyimpan data berkembang, demikian juga jumlah data yang ingin kita simpan. IBM memberi tahu kita bahwa 2,5 exabyte data baru dibuat setiap hari. Statistik itu menakjubkan; itu berarti "bahwa 90% dari data di dunia saat ini telah dibuat dalam dua tahun terakhir saja." (http://www-01.ibm.com/software/data/bigdata/what-is-big-data.html)

Pusat Superkomputer San Diego memiliki penyimpanan online sebesar satu petabyte dan arsip offline sebesar tujuh petabyte, dan perkiraan data yang disimpan Google juga berada dalam kisaran beberapa petabyte. Sedangkan ruang fisik terbatas menyimpan (dan menemukan) sejumlah besar data tercetak, data elektronik mengambil sedikit ruang relatif terhadap hard copy.

Storageservers.com (http://storageservers.wordpress.com/2013/07/17/facts-and-stats-of-worlds-largest-data-centers/) memperkirakan kapasitas dan penggunaan berikut di situs yang sudah dikenal:

- Google memiliki 17 pusat data pada tahun 2014, terhitung 0,01 persen dari total penggunaan energi dunia.
- Server Facebook memproses sekitar 2,4 miliar keping konten dan 750 terabyte data setiap hari, dan penggunanya mengakses sekitar 7 petabyte penyimpanan foto setiap bulan.
- Server Amazon memiliki lebih dari 17 juta pengunjung bulanan yang mengakses 410 terabyte data dari platformnya. “Sekitar 30 juta pengguna Amazon melakukan streaming sekitar 40 petabyte video per bulan.”
- Microsoft memiliki lebih dari satu miliar pengguna dan lebih dari 100.000 server.

Pemberitahuan dan Persetujuan (Notice and Consent)

Dari mana semua byte ini berasal? Meskipun beberapa berasal dari sumber publik dan komersial (seperti surat kabar, halaman web, audio digital, dan rekaman video) dan lainnya berasal dari transfer data yang disengaja (misalnya, pengembalian pajak, pernyataan kepada polisi setelah kecelakaan, formulir survei pembaca). , kertas sekolah), yang lain dikumpulkan tanpa pengumuman. Perusahaan telepon mencatat tanggal, waktu, durasi, sumber, dan tujuan setiap panggilan telepon. ISP melacak situs yang dikunjungi. Beberapa situs menyimpan alamat IP setiap pengunjung situs (walaupun alamat IP biasanya tidak unik untuk individu tertentu). Pengguna belum tentu mengetahui kategori pengumpulan data ketiga ini dan dengan demikian tidak dapat dikatakan telah memberikan persetujuan atas pengumpulan data tersebut.

Kami dapat diinformasikan tentang pengumpulan dan penggunaan data dalam banyak cara. Misalnya, masuk ke situs web mungkin memerlukan pengakuan “ketentuan penggunaan”, yang menjelaskan apa yang dikumpulkan, mengapa, dan jalan apa yang Anda miliki jika memilih untuk tidak mengumpulkan sesuatu. Ketentuan penggunaan juga dapat memberi tahu apa yang dapat dilakukan jika menemukan kesalahan atau ketidaksesuaian dalam mengumpulkan, menyimpan, atau menggunakan data. Demikian pula, saat menggunakan aplikasi di perangkat seluler, mungkin diberi tahu bahwa beberapa item data, seperti lokasi atau daftar kontak, akan digunakan oleh aplikasi dalam melakukan beberapa tugas.

Selain pemberitahuan, persetujuan terkadang diperlukan. Artinya, Anda secara eksplisit dimintai izin untuk mengumpulkan dan menggunakan informasi. Misalnya, program atau aplikasi pemetaan mungkin meminta izin Anda untuk mengumpulkan lokasi secara otomatis; jika menolak, Anda tidak dapat melanjutkan menggunakan program, atau harus memasukkan lokasi setiap kali menginginkan peta atau petunjuk arah. Pemberitahuan dan persetujuan adalah prinsip penting dalam penyediaan dan perlindungan privasi. Namun, terkadang masalah dengan pemberitahuan dan persetujuan tidak terlihat seperti yang seharusnya atau seharusnya. Studi Kasus 5-1 menjelaskan peristiwa baru-baru ini di mana toilet di sebuah pusat konvensi diklaim dapat menangkap informasi untuk kepentingan publik. Meski akhirnya terungkap sebagai hoax, aksi tersebut mengingatkan kita semua bahwa data seringkali diambil tanpa izin dan bahkan tanpa sepengetahuan kita.

Studi Kasus 5-1 : Sensor Toilet Tanpa Persetujuan?

Itu di berita malam di Amerika Serikat dan di tempat lain: Pada konferensi internasional tentang komputer dan interaksi manusia yang berlangsung di Toronto, pengguna toilet disambut oleh tanda yang ditunjukkan pada Gambar 5-1. Ini memberi tahu pengguna bahwa perilaku di toilet sedang direkam untuk dianalisis.



Gambar 5-1 Notifikasi Pengambilan Data

Pengunjung ke URL yang disediakan, quantifiedtoilets.com, diberitahu, “Kami bangga menjadi bagian dari Inisiatif Bangunan Sehat Toronto, dan bersemangat untuk menyebarkan infrastruktur awal di seluruh struktur sipil utama kota. Bersama dengan mitra kami, kami memanfaatkan big data yang dikumpulkan dari aktivitas sehari-hari bangunan dan penghuninya. Diakui bukan sumber data terseksi, kami menganalisis proses limbah biologis bangunan untuk membuat ruang yang lebih baik dan orang yang lebih bahagia. Kami menggunakan data ini untuk merampingkan jadwal kru pembersihan, menginformasikan kotamadya tentang penggunaan sumber daya, dan membantu bangunan dan kota merencanakan untuk warga yang lebih sehat dan lebih bahagia ... Menggunakan teknologi penginderaan canggih dan sistem pengumpulan data limbah terpusat yang canggih, kami dapat diam-diam menangkap data dari setiap toilet individu. Aktivitas di setiap toilet menciptakan

tanda tangan unik yang memungkinkan kami melacak penggunaan dan menganalisis detail dari setiap toilet di sebuah gedung. Perangkat lunak inovatif kami kemudian dapat membuat katalog data untuk analisis kesehatan multifaset yang saat ini tidak tersedia melalui cara tradisional." Situs web tersebut juga menunjukkan umpan data langsung, yang ditunjukkan pada Gambar 5-2, yang menunjukkan jenis analisis apa yang sedang dilakukan.

Time	Toilet ID	Sex	Deposit	Odor	alcohol	detected	Pregnancy	Infections
10:39:42 AM	T203	male	270ml	acidic	0.002%	yes	no	none
10:39:28 AM	T314	female	145ml	neutral	0.002%	no	no	none
10:39:17 AM	T201	female	110ml	neutral	0.11%	no	no	none
10:38:12 AM	T314	female	80ml	neutral	0.000%	no	no	none
10:38:09 AM	T101	female	170ml	acidic	0.028%	no	no	none
10:37:43 AM	T113	male	260ml	neutral	0.000%	no	no	none
10:37:22 AM	T305	female	265ml	nutty	0.001%	no	no	none
10:37:06 AM	T313	female	270ml	nuetral	0.001%	yes	no	none
10:37:00 AM	T211	female	115ml	neutral	0.000%	no	no	none
10:36:42 AM	T314	female	250ml	neutral	0.000%	no	no	none
10:36:38 AM	T212	male	205ml	neutral	0.071%	no	no	none

Gambar 5-2 Contoh Pengambilan Data

Mengapa ini Hoaks? Salah satu pelaku menjelaskan bahwa, "Data wajah kami tersedia secara gratis untuk ditangkap kamera CCTV setiap hari… Data lain apa yang kami berikan tanpa benar-benar memikirkannya yang dapat digunakan dengan cara yang cukup invasif atau tidak etis?" Meskipun sensor toilet Toronto tidak nyata (belum?), teknologi lain yang digunakan dapat menangkap informasi pribadi. Misalnya, sistem IntelliMat dari Tactonic Technologies (http://www.tactonic.com/index.html) memiliki permukaan peka tekanan untuk menangkap detail tentang cara seseorang berjalan melintasinya. Seperti yang kita lihat di Bab 2, gaya berjalan seseorang dapat digunakan sebagai biometrik untuk mengidentifikasi siapa yang hadir, di mana, dan kapan.

Pengumpulan data ini, bahkan jika disetujui oleh kebanyakan orang karena kepentingan publik yang jelas, menghadirkan masalah serius untuk pemberitahuan dan persetujuan. Bagaimana seharusnya seseorang diberi tahu setiap kali Gambar diambil, gaya berjalan dikenali, atau kehadiran bahan kimia dicatat? Dalam pengaturan seperti toilet, di mana ada harapan privasi, seberapa sering pemberitahuan harus diberikan, dan dalam berapa banyak bahasa? Bagaimana seseorang bisa memilih keluar? Mungkin sulit atau tidak mungkin.

5.1.3 Kontrol dan Kepemilikan Data

Dalam banyak kasus, Anda diminta untuk memberikan data (dengan pemberitahuan yang tepat) dan Anda setuju untuk melakukannya, secara eksplisit atau implisit. Tapi apa yang terjadi ketika data ditransfer ke orang atau sistem yang meminta? Setelah mengumpulkan data dengan izin Anda, orang lain dapat menyimpan data yang Anda berikan kepada mereka; Anda telah menyerahkan kendali (dan terkadang kepemilikan, tergantung pada hukum di wilayah Anda) atas salinan data tersebut kepada mereka. Misalnya, ketika Anda memesan barang dagangan secara online, Anda tahu bahwa Anda baru saja merilis nama, alamat, data pembayaran, dan deskripsi barang yang Anda beli. Demikian pula, ketika Anda menggunakan kartu loyalitas pelanggan di toko atau online, Anda tahu bahwa pedagang dapat mengaitkan identitas Anda dengan barang-barang yang Anda telusuri

atau beli. Setelah mengambil data Anda, pedagang kemudian dapat menyimpan data tanpa batas waktu, serta mendistribusikan kembali data tersebut ke orang atau sistem lain. Kebiasaan menjelajah, praktik pembelian, dan preferensi Anda untuk merek hotel, jenis kamar hotel, maskapai penerbangan, atau agen perjalanan dapat dijual ke hotel lain. Anda memiliki sedikit kendali atas penyebaran (atau penyebaran ulang) data Anda. Dan begitu datanya hilang, Anda tidak bisa mendapatkannya kembali.

Kami tidak selalu menghargai konsekuensi dari kehilangan kendali ini. Misalkan pada saat marah Anda melontarkan nada yang kuat kepada seseorang. Meskipun 100 tahun yang lalu Anda akan menulis catatan di atas kertas dan 50 tahun yang lalu Anda dapat menyuarakan komentar melalui telepon, sekarang Anda memposting pesan tersebut ke halaman media sosial. Jika Anda berubah pikiran dan ingin menarik kembali komentar marah Anda, pertimbangkan bagaimana Anda akan menangani ketiga bentuk komunikasi ini. Untuk catatan tertulis, Anda menulis surat permintaan maaf, penerima Anda merobek catatan Anda, dan tidak ada jejak yang tersisa. (Anda bahkan mungkin dapat meyakinkan operator pos untuk mengembalikan surat Anda sebelum dikirimkan, jadi tidak perlu meminta maaf dan tidak ada kerugian yang dilakukan.) Dalam kasus kedua, Anda menelepon untuk meminta maaf dan yang tersisa hanyalah kenangan (dengan asumsi panggilan asli tidak direkam).

Sedangkan untuk komunikasi elektronik, Anda dapat menghapus postingan Anda. Namun, dalam waktu antara pembuatan dan penghapusan, beberapa orang lain mungkin telah melihat postingan asli Anda (atau versi cache) dan menyalinnya ke blog atau situs web lain yang tidak Anda kontrol. Mesin pencari mungkin telah menemukan yang asli, versi yang di-cache, atau salinan. Dan orang lain mungkin telah mengambil kata-kata Anda dan mengedarkannya dalam email. Jadi, dengan surat kertas, kita biasanya dapat menghapus sesuatu yang ingin kita tarik kembali, dan dengan panggilan telepon, kita dapat meminta maaf dan menebus kesalahan. Tapi begitu sesuatu elektronik di luar kendali Anda di web, itu mungkin tidak akan pernah dihapus; memang, itu bisa berkembang biak dan dengan cepat menjadi masalah serius. (Pikirkan, misalnya, video YouTube yang diinginkan politisi tidak pernah diposting, terutama setelah videonya menjadi viral.)

Situasi serupa menyangkut sesuatu yang ditulis tentang Anda. Orang lain telah memposting sesuatu di web yang bersifat pribadi tentang Anda, dan Anda ingin menghapusnya. Bahkan jika pembuatnya setuju, Anda mungkin tidak dapat menghapus semua jejaknya. Keinginan untuk menghapus informasi lama yang mungkin memalukan ini menjadi fokus upaya Uni Eropa untuk menegakkan “hak untuk dilupakan”. Selain itu, beberapa orang menemukan bahwa mereka mengungkapkan lebih dari yang seharusnya di situs-situs seperti Facebook dan Instagram. Calon karyawan ditolak untuk pekerjaan karena hal-hal yang mereka tulis yang tersedia secara online. Dan paparan data ini dapat memengaruhi sebagian besar aspek kehidupan Anda. Misalnya, anggaplah sebuah perusahaan menyimpan data tentang Anda, dan catatan perusahaan itu terekspos dalam serangan komputer. Perusahaan mungkin tidak bertanggung jawab untuk mencegah bahaya pada Anda, memberikan kompensasi kepada Anda jika Anda dirugikan, atau bahkan memberi tahu Anda tentang kejadian tersebut.

Hukum Amerika Serikat didasarkan pada prinsip kebebasan bicara yang dilindungi: Di bawah amandemen pertama Konstitusi A.S., setiap orang dijamin bahwa Kongres tidak dapat mengesahkan undang-undang “yang meringkas kebebasan berbicara.” Klausul terpisah dari amandemen melindungi

hak individu untuk mengekspresikan pendapat secara bebas dalam publikasi, yang telah mengarah pada praktik mengizinkan jurnalis untuk menjaga anonimitas orang-orang yang pendapatnya dilaporkan dalam artikel. Kebebasan berbicara dan kebebasan pers menimbulkan pertanyaan tentang penerapannya pada media digital, seperti yang dijelaskan dalam Kasus 5-2.

Web adalah arsip sejarah yang hebat, tetapi karena arsip, cache, dan situs cermin, hal-hal yang diposting di web mungkin tidak akan pernah hilang. Seperti yang dilaporkan NBC News, “Sebagian dari Anda ada di seluruh Internet. Jika Anda telah masuk ke Google dan mencari, menyimpan file di folder Dropbox Anda, melakukan panggilan telepon menggunakan Skype, atau baru bangun di pagi hari dan memeriksa email Anda, Anda meninggalkan jejak remah-remah digital. Orang yang memiliki akses ke access informasi ini—perusahaan yang mendukung email dan pencarian Web Anda, pengiklan yang secara strategis mengarahkan iklan kepada Anda—dapat membangun Gambaran tentang siapa Anda, apa yang Anda sukai, dan apa yang mungkin akan Anda lakukan selanjutnya … Agen federal dan operator lain dapat menggunakan data ini , juga.”

Studi Kasus 5-2 : Apakah Tweet Terlindungi dari Kebebasan Berbicara?

Pada bulan Februari 2011, reporter Dana Hedgpeth menulis di Washington Post bahwa pemerintah AS berusaha untuk mendapatkan informasi pribadi dari akun Twitter tiga orang yang terkait dengan penyelidikan WikiLeaks. Pengacara pemerintah meminta nama layar, alamat surat, nomor telepon, rekening bank dan informasi kartu kredit, dan alamat IP. Namun, pengacara terdakwa bersikeras bahwa informasi ini dilindungi oleh Amandemen Pertama konstitusi AS.

Meski kasus di pengadilan membahas dokumen WikiLeaks, permintaan pemerintah ini menimbulkan pertanyaan penting: Data apa yang bisa disita pemerintah dari jejaring sosial? Kita telah melihat dalam bab ini bahwa pengguna memiliki pilihan dalam aplikasi tentang melindungi privasi data mereka di jejaring sosial. Tapi bisakah pemerintah mengesampingkan pengaturan itu? Salah satu pengacara terdakwa mencatat bahwa "data pengguna akan memberi pemerintah peta orang-orang yang terkait dengan WikiLeaks dan pada dasarnya menghentikan kebebasan berbicara secara online." Pengacara pemerintah menunjukkan bahwa ini adalah permintaan standar, dan mereka tidak tahu apakah Twitter bahkan mengumpulkan semua item data yang diminta.

Salah satu masalah yang diangkat dalam permintaan ini adalah apakah undang-undang saat ini berlaku untuk teknologi Internet. “Para ahli mengatakan mereka dimaksudkan untuk menangani catatan telepon, bukan teknologi yang berkembang seperti email dan tweet.” Seorang pelobi untuk American Civil Liberties Union mencatat, "Kami menggunakan alat untuk mengakses informasi di email, situs jejaring sosial yang tidak pernah terpikirkan." Dalam blog 28 Januari 2011, perwakilan Twitter menjelaskan posisi perusahaan: "kebebasan berekspresi disertai dengan mandat untuk melindungi hak pengguna kami untuk berbicara secara bebas dan mempertahankan kemampuan mereka untuk bersaing dengan mengungkapkan informasi pribadi mereka."

Di beberapa negara, seperti di Uni Eropa, Anda memiliki data Anda dan harus memberikan izin sebelum dapat digunakan dalam berbagai cara. Namun di negara lain, seperti Amerika Serikat, pemegang data adalah pemiliknya—salah satu alasan mengapa membiarkan salinan lolos ke seseorang atau di tempat lain adalah masalah. Tetapi bahkan jika undang-undang berubah untuk memungkinkan kita masing-masing memiliki data kita, lalu bagaimana? Berapa banyak dari kita yang ingin menghabiskan sebagian besar hari kita dengan memberikan izin kepada kamera lalu lintas, situs web, dan penyedia email untuk menggunakan data kita?

Masalah ini—pengumpulan data, pemberitahuan dan persetujuan, serta kontrol dan kepemilikan data—memiliki implikasi privasi yang signifikan. Salah satu cara kami mengatasi masalah semacam ini adalah dengan kebijakan: pernyataan praktik tertulis yang menginformasikan semua pihak yang terkena dampak tentang hak dan tanggung jawab mereka. Di bagian berikutnya kami menyelidiki kebijakan privasi untuk komputasi.

5.2 Prinsip dan Kebijakan Privasi

Dengan aplikasi internet yang begitu masif dalam mengumpulkan informasi, kunci dari konsep privasi adalah kontrol pengguna/warga terhadap informasi pribadinya, penghormatan batas privasi dan perlindungan terhadap informasi tersebut. Ibaratnya seorang teman baik yang dipercaya untuk menyimpan rahasia dan menjamin rahasia yang disimpan tidak akan disebarkan. Para penyedia layanan, baik pihak swasta maupun pemerintah, harus melindungi informasi yang kita berikan, bagaimanapun caranya.

Di Amerika Serikat, ketertarikan pada privasi elektronik dan database komputer setidaknya dimulai pada awal 1970-an. Perhatian publik terhadap privasi telah bervariasi selama bertahun-tahun. Pada awal 1970-an, komite yang disponsori pemerintah federal mengembangkan seperangkat prinsip privasi, yang disebut Praktik Informasi yang Adil, yang tidak hanya memengaruhi undang-undang dan peraturan AS, tetapi juga meletakkan dasar bagi undang-undang privasi di negara lain.

5.2.1 Praktik Informasi yang Adil

Pada tahun 1973 Willis Ware dari RAND Corporation mengetuai sebuah komite untuk menasihati Sekretaris Departemen Kesehatan, Pendidikan, dan Kesejahteraan AS (sekarang disebut Layanan Kesehatan dan Kemanusiaan) tentang masalah privasi. Laporan (mengusulkan seperangkat prinsip berdasarkan praktik informasi yang adil:

- Batasan koleksi. Data harus diperoleh secara sah dan adil.
- Kualitas data. Data harus relevan dengan tujuannya, akurat, lengkap, dan terkini.
- Spesifikasi tujuan. Tujuan penggunaan data harus diidentifikasi dan data dimusnahkan jika tidak lagi diperlukan untuk tujuan tersebut.
- Penerapan batasan. Penggunaan untuk tujuan selain yang ditentukan hanya diizinkan dengan persetujuan subjek data atau oleh otoritas hukum.
- Perlindungan keamanan. Prosedur untuk menjaga dari kehilangan, korupsi, perusakan, atau penyalahgunaan data harus ditetapkan.

- Keterbukaan. Harus dimungkinkan untuk memperoleh informasi tentang pengumpulan, penyimpanan, dan penggunaan sistem data pribadi.
- Partisipasi individu. Subyek data biasanya memiliki hak untuk mengakses dan menantang data yang berkaitan dengan mereka.
- Akuntabilitas. Pengontrol data harus ditunjuk dan bertanggung jawab untuk mematuhi langkah-langkah untuk mempengaruhi prinsip-prinsip.

Prinsip-prinsip ini menggambarkan hak individu, bukan persyaratan kolektor; yaitu, prinsip-prinsip tersebut tidak memerlukan perlindungan terhadap data yang dikumpulkan.

Ware menyatakan ada beberapa masalah penting, termasuk penautan data dalam banyak file dan penggunaan kunci yang berlebihan, seperti nomor jaminan sosial, yang tidak pernah dimaksudkan untuk digunakan sebagai pengidentifikasi catatan. Dan meskipun dia melihat bahwa masyarakat dapat bergerak ke arah penggunaan nomor identitas universal, dia takut bahwa gerakan itu akan terjadi tanpa rencana (dan karenanya tanpa kendali). Dia benar, meskipun dia tidak dapat memperkirakan jumlah pertukaran data selama 40 tahun kemudian.

Menurut Rein Turn dan Willis Ware melindungi item data itu sendiri, mengakui bahwa kumpulan data akan menjadi target yang menarik untuk serangan akses yang tidak sah. Mereka menyarankan empat cara untuk melindungi data yang disimpan:

- Kurangi keterpaparan dengan membatasi jumlah data yang dipelihara, hanya menanyakan apa yang diperlukan dan menggunakan sampel acak alih-alih populasi lengkap.
- Kurangi sensitivitas data dengan menukar item data atau menambahkan kesalahan halus pada data (dan memperingatkan penerima bahwa data telah diubah).
- Anonimkan data dengan menghapus atau memodifikasi item data pengidentifikasi.
- Enkripsi data.

Hukum Privasi AS Privacy

Ware dan komitenya mengharapkan prinsip-prinsip ini berlaku untuk semua kumpulan data pribadi tentang individu, tetapi kenyataannya jauh dari tujuan ini. Sebaliknya, laporan komite Ware mengarah pada Undang-Undang Privasi 1974 (5 USC 552a), yang mewujudkan sebagian besar prinsip-prinsip ini, meskipun undang-undang itu hanya berlaku untuk data yang dikumpulkan dan dikelola oleh pemerintah AS. Namun demikian, Undang-Undang Privasi adalah undang-undang yang luas, mencakup semua data yang dikumpulkan oleh pemerintah. Ini adalah undang-undang privasi AS yang terkuat karena luasnya: Ini berlaku untuk semua data pribadi yang disimpan di mana saja di pemerintah federal.

Amerika Serikat kemudian mengesahkan undang-undang yang melindungi data yang dikumpulkan dan disimpan oleh organisasi lain, tetapi undang-undang ini berlaku sedikit demi sedikit, menurut tipe data individu. Misalnya, kredit konsumen dibahas dalam Fair Credit Reporting Act, informasi kesehatan dalam Health Insurance Portability and Accountability Act (HIPAA), organisasi jasa keuangan dalam Gramm–Leach–Bliley Act (GLBA), akses web anak-anak di Children's Online Privacy Protection Act (COPPA), dan catatan siswa di Federal Educational Rights and Privacy Act. Tidak mengherankan, undang-undang terpisah ini tidak konsisten dalam melindungi privasi.

Amerika Serikat juga mengizinkan pemerintah negara bagian untuk mengatur aspek privasi tertentu. Misalnya, Smith secara teratur menerbitkan kompilasi undang-undang privasi negara bagian dan federal; bagian utama mengGambarkan Amerika Serikat, dan lampiran membahas undang-undang privasi di Kanada dan provinsinya. Undang-undang negara bagian dapat sangat bervariasi, terkadang menyulitkan seseorang untuk mematuhi undang-undang privasi di setiap negara bagian. Misalnya, di Nevada, perekam kotak hitam tidak boleh dipasang di mobil tanpa persetujuan pemilik atau penyewa mobil. (Status Rev. Nevada bagian 484 638). Demikian pula, di New Hampshire, pabrikan harus mengungkapkan kepada pemilik keberadaan perekam data kejadian di mobil baru (New Hampshire Rev. Statute ann. Sec. 357-G:1). Namun, baik di negara bagian New York dan North Dakota, ada batasan lebih lanjut pada jenis data yang dapat ditangkap oleh perekam; misalnya, di North Dakota, data hanya dapat digunakan untuk memperbaiki mobil atau untuk meningkatkan keselamatan. (Kode Cent. Dakota Utara det. 51-04.28).

Perlindungan Data Pribadi di Indonesia

Indonesia akhirnya memiliki aturan soal perlindungan data pribadi di era digital.Aturan itu dituangkan dalam bentuk Peraturan Menteri (Permen) No 20 Tahun 2016 tentang Perlindungan Data Pribadi (PDP) ditetapkan 7 November 2016, diundangkan dan berlaku sejak 1 Desember 2016. Dari dokumen yang diunduh, di aturan itu dinyatakan Data Pribadi adalah data perseorangan tertentu yang disimpan, dirawat, dan dijaga kebenaran serta dilindungi kerahasiaannya.

Pemilik Data Pribadi adalah individu yang padanya melekat Data Perseorangan Tertentu.Setiap Penyelenggara Sistem Elektronik harus mempunyai aturan internal perlindungan Data Pribadi untuk melaksanakan proses.Setiap Penyelenggara Sistem Elektronik harus menyusun aturan internal perlindungan Data Pribadi sebagai bentuk tindakan pencegahan untuk menghindari terjadinya kegagalan dalam perlindungan Data Pribadi yang dikelolanya. Perolehan dan pengumpulan Data Pribadi oleh Penyelenggara Sistem Elektronik wajib berdasarkan Persetujuan atau berdasarkan ketentuan peraturan perundang-undangan.

Data Pribadi yang disimpan dalam Sistem Elektronik harus Data Pribadi yang telah diverifikasi keakuratannya. Data Pribadi yang disimpan dalam Sistem Elektronik harus dalam bentuk data terenkripsi.Data Pribadi wajib disimpan dalam Sistem Elektronik sesuai dengan ketentuan peraturan perundang-undangan yang mengatur kewajiban jangka waktu penyimpanan Data Pribadi pada masing-masing Instansi Pengawas dan Pengatur Sektor atau paling singkat lima tahun, jika belum terdapat ketentuan peraturan perundang-undangan yang secara khusus mengatur untuk itu.Aturan data center. Hal yang menarik di aturan ini adalah ketentuan Pusat data(data center) dan pusat pemulihan bencana (disaster recovery center) Penyelenggara Sistem Elektronik untuk pelayanan publik yang digunakan untuk proses perlindungan wajib ditempatkan dalam wilayah negara Republik Indonesia.

Dalam aturan ini ditegaskan sistem elektronik yang dapat digunakan dalam proses perlindungan data pribadi adalah sistem elektronik yang sudah tersertifikasi dan mempunyai aturan internal tentang perlindungan data pribadi yang wajib memperhatikan aspek penerapan teknologi, sumber daya manusia, metode, dan biayanya. Pemilik data pribadi, berhak atas kerahasiaan data miliknya; berhak mengajukan pengaduan dalam rangka penyelesaian sengket data pribadi; berhak mendapatkan akses untuk memperoleh historis data pribadinya; dan berhak meminta pemusnahan data perseorangan tertentu miliknya dalam sistem elektronik.

Penyelenggara system Elektronik wajib memberikan akses atau kesempatan kepada Pemilik Data Pribadi untuk mengubah atau memperbarui Data Pribadinya tanpa mengganggu sistem pengelolaan Data Pribadi, kecuali ditentukan lain oleh ketentuan peraturan perundang-undangan; memusnahkan Data Pribadi sesuai dengan ketentuan dalam Peraturan Menteri ini atau ketentuan peraturan perundang-undangan lainnya yang secara khusus mengatur di masing-masing Instansi Pengawas dan Pengatur Sektor untuk itu; dan menyediakan narahubung (contact person) yang mudah dihubungi oleh Pemilik Data Pribadi terkait pengelolaan Data Pribadinya.

Apabila pemilik data pribadi merupakan kategori anak-anak, pemberian persetujuan sebagaimana yang di maksud dalam permen ini dilakukan oleh orang tua atau wali anak yang bersangkutan. Untuk penyelenggara sistem elektronik yang telah menyediakan, menyimpan, dan mengelola data pribadi sebelum Permen ini berlaku, wajib tetap menjaga kerahasiaan data pribadi yang telah ada. Bagi yang melanggar aturan hanya dikenai sangsi administratif berupa:

(a) Peringatan lisan;
(b) Peringatan Tertulis;
(c) Pengentian semetara kegiatan dan / atau; pengumuman di situs dalam jaringan, yang tata caranya akan diatur dengan Peraturan Menteri.

Permen ini adalah satu dari 21 Permen yang merupakan turunan dari Peraturan Pemerintah (PP) No 82 / 2012 tentang Penyelenggaraan Sistem dan Transaksi Elektronik (PSTE) yang diundangkan dan berlaku sejak 15 Oktober 2012. *Sumber : https://kominfo.go.id/content/detail/8621/indonesia-sudah-miliki-aturan-soal-perlindungan-data-pribadi/0/sorotan_media*

Aturan Privasi di Eropa

Pada tahun 1981, Dewan Eropa (sebuah badan internasional yang terdiri dari 46 negara Eropa, didirikan pada tahun 1949) mengadopsi Konvensi 108 untuk perlindungan individu sehubungan dengan pemrosesan otomatis data pribadi. Ketika sistem otomatis menjadi lebih meresap, Uni Eropa (E.U.) pada tahun 1995 mengadopsi Directive 95/46/EC tentang pemrosesan data pribadi. Directive 95/46/EC, sering disebut European Privacy Directive, mengharuskan hak privasi individu dipertahankan dan data tentang mereka disimpan

- diproses secara adil dan sah.
- dikumpulkan untuk tujuan tertentu, eksplisit, dan sah dan tidak diproses lebih lanjut dengan cara yang tidak sesuai dengan tujuan tersebut (kecuali perlindungan yang tepat melindungi privasi).
- memadai, relevan, dan tidak berlebihan dalam kaitannya dengan tujuan pengumpulan atau pemrosesan lebih lanjut.
- akurat dan, jika perlu, selalu up to date. Memang, jika data ditemukan tidak akurat atau tidak lengkap, setiap langkah yang wajar harus diambil untuk memastikan bahwa data tersebut dihapus atau diperbaiki, sehubungan dengan tujuan pengumpulannya atau pemrosesan lebih lanjut.
- disimpan dalam bentuk yang memungkinkan identifikasi subjek data tidak lebih lama dari yang diperlukan untuk tujuan pengumpulan data atau tujuan mereka diproses lebih lanjut.

Selain itu, individu memiliki hak untuk mengakses data yang dikumpulkan tentang mereka, untuk mengoreksi data yang tidak akurat atau tidak lengkap, dan untuk meminta koreksi tersebut dikirimkan kepada mereka yang telah menerima data tersebut. Anda dapat melihat bahwa aturan ini mencerminkan Praktik Informasi yang Adil yang dijelaskan dalam laporan Ware. Selain itu, E.U. arahan privasi menambahkan tiga prinsip lagi:

- Perlindungan khusus untuk data sensitif. Harus ada pembatasan yang lebih besar pada pengumpulan dan pemrosesan data yang melibatkan "data sensitif." Di bawah E.U. arahan perlindungan data, informasi bersifat sensitif jika melibatkan "asal ras atau etnis, pendapat politik, keyakinan agama, persuasi filosofis atau etis … [atau] kesehatan atau kehidupan seksual."
- Transfer data. Prinsip ini secara eksplisit membatasi pengguna yang berwenang atas informasi pribadi untuk mentransfer informasi tersebut kepada pihak ketiga tanpa izin dari subjek data.
- Pengawasan independen. Entitas yang memproses data pribadi tidak hanya harus bertanggung jawab tetapi juga harus tunduk pada pengawasan independen. Dalam kasus pemerintah, prinsip ini memerlukan pengawasan oleh kantor atau departemen yang terpisah dan independen dari unit yang menangani pengolahan data. Di bawah arahan perlindungan data, pengawas independen harus memiliki wewenang untuk mengaudit sistem pemrosesan data, menyelidiki keluhan yang diajukan oleh individu, dan memberlakukan sanksi atas ketidakpatuhan.

Ringkasan singkat dari undang-undang yang lebih panjang ini memberi Anda Gambaran tentang pendekatan privasi yang berbeda dan lebih komprehensif yang diambil oleh Uni Eropa. Uni Eropa. Persyaratan arahan perlindungan data berlaku untuk pemerintah, bisnis, dan organisasi lain yang mengumpulkan data pribadi. Sejak arahan asli tahun 1995 diterbitkan, Uni Eropa telah memperluas cakupannya ke sistem telekomunikasi dan membuat perubahan lain untuk beradaptasi dengan kemajuan teknologi. Anda dapat menemukan arahan lengkap dan keputusan yang relevan di http://ec.europa.eu/justice/data protection/law/index_en.htm.

Hukum yang Bertentangan

Hukum yang berbeda di yurisdiksi yang berbeda pasti akan berbenturan. Hubungan antara Uni Eropa dan Amerika Serikat telah tegang karena privasi karena Uni Eropa. undang-undang melarang berbagi data dengan perusahaan atau pemerintah di negara-negara yang undang-undang privasinya tidak sekuat undang-undang E.U. (Amerika Serikat dan Uni Eropa telah menyetujui seperangkat prinsip "pelabuhan aman" yang memungkinkan perusahaan AS berdagang dengan negara-negara Eropa meskipun mereka tidak memenuhi semua undang-undang privasi Eropa.) Di Studi Kasus 5-3 Anda dapat melihat bagaimana perbedaan ini hukum dapat mempengaruhi perdagangan dan, pada akhirnya, hubungan diplomatik.

Studi Kasus 5-3 : Saat Prinsip Privasi Berbenturan

Privasi adalah bisnis yang serius. Perdagangan, perjalanan, atau komunikasi dapat berhenti ketika data dibagikan di antara organisasi atau negara dengan prinsip privasi yang berbeda. Misalnya, dalam upaya mengamankan perbatasannya setelah serangan 11 September 2001, Amerika Serikat membuat program untuk menyaring penumpang maskapai penerbangan untuk kemungkinan hubungan teroris. Program ini menggunakan informasi di Penumpang Name Record (PNR): data yang dikumpulkan

oleh maskapai penerbangan saat Anda memesan penerbangan dari satu tempat ke tempat lain. PNR mencakup 34 kategori informasi: tidak hanya nama dan detail penerbangan Anda, tetapi juga nomor telepon, informasi kartu kredit, preferensi makanan, alamat, dan banyak lagi. Karena orang Eropa merupakan kelompok pengunjung terbesar ke Amerika Serikat (13,25 juta pada tahun 2013, menurut Statista.com), orang Amerika meminta maskapai Eropa untuk menyediakan data PNR dalam waktu 15 menit dari keberangkatan pesawat ke Amerika Serikat.

Ingatlah bahwa Arahan Privasi Eropa melarang penggunaan data untuk tujuan selain tujuan pengumpulannya. Permintaan AS jelas melanggar arahan. Setelah negosiasi yang cukup panjang, Komisi Eropa dan Dewan Eropa mencapai kesepakatan pada Mei 2004 untuk mengizinkan maskapai penerbangan memberikan data ke Amerika Serikat.

Namun, Parlemen Eropa keberatan, dan pada tanggal 30 Mei 2006, Pengadilan Eropa, pengadilan tertinggi di Uni Eropa, memutuskan bahwa Komisi Eropa dan Dewan Eropa tidak memiliki wewenang untuk membuat kesepakatan semacam itu dengan Amerika Serikat. Prinsip-prinsip privasi bukanlah dasar utama dari keputusan tersebut, tetapi mereka memiliki dampak yang besar: “Secara khusus, pengadilan mengatakan catatan penumpang dikumpulkan oleh maskapai penerbangan untuk penggunaan komersial mereka sendiri, sehingga Uni Eropa tidak dapat secara hukum setuju untuk memberikannya kepada maskapai penerbangan. otoritas Amerika, bahkan untuk tujuan keamanan publik atau penegakan hukum.” [CLA06] Seorang juru bicara Departemen Keamanan Dalam Negeri AS membantah bahwa privasi bukanlah masalah, karena data dapat diminta dari setiap penumpang yang tiba di Amerika Serikat.

Tanpa data yang diminta, Amerika Serikat secara teori dapat menolak hak pendaratan untuk maskapai yang tidak berpartisipasi. Hampir setengah dari semua perjalanan udara asing ke Amerika Serikat adalah trans-Atlantik, sehingga gangguan tersebut dapat menelan biaya jutaan untuk semua ekonomi yang terlibat. Benturan prinsip privasi ini diselesaikan dengan membuat serangkaian praktik "Safe Harbor" yang memastikan perlindungan yang memadai bagi individu yang datanya sedang ditransfer. Kerangka kerja Safe Harbor juga telah dibuat antara Amerika Serikat dan Swiss (bukan anggota Uni Eropa) untuk alasan yang sama. Rincian setiap kerangka kerja dapat ditemukan di http://export.gov/safeharbor/.

5.2.2 Tindakan Individu untuk Melindungi Privasi

Sejauh ini, kita telah membahas cara bagi pemerintah dan perusahaan untuk mengumpulkan, menyimpan, dan berbagi informasi pribadi. Tetapi ada tindakan yang dapat Anda ambil sebagai individu untuk melilindungi privasi Anda sendiri. Salah satu caranya adalah dengan menjaga identitas Anda. Tidak setiap konteks mengharuskan kita masing-masing untuk mengungkapkan identitas kita, dan ada cara bagi sebagian orang untuk memakai bentuk topeng elektronik.

Anonimitas

Terkadang orang mungkin ingin melakukan sesuatu secara anonim. Misalnya, seorang bintang rock yang membeli rumah pantai mungkin ingin menghindari perhatian yang tidak diinginkan dari tetangga, atau seseorang yang memposting ke daftar kencan mungkin ingin melihat balasan sebelum membuat kencan.

Deirdre Mulligan mencantumkan beberapa alasan mengapa orang mungkin lebih memilih aktivitas anonim di web. Dia menjelaskan bahwa beberapa orang menyukai anonimitas web karena mengurangi ketakutan akan diskriminasi. Keadilan dalam perumahan, pekerjaan, dan pergaulan lebih mudah dipastikan ketika dasar untuk potensi diskriminasi disembunyikan. Juga, orang yang meneliti apa yang mereka anggap sebagai masalah pribadi, seperti masalah kesehatan atau orientasi seksual, mungkin lebih cenderung mencari informasi terlebih dahulu dari apa yang mereka anggap sebagai sumber anonim, beralih ke manusia ketika mereka mengetahui lebih banyak tentang situasi mereka.

Anonimitas, selain memiliki manfaat, juga dapat menimbulkan masalah. Jika Anda mencoba untuk menjadi anonim, bagaimana Anda membayar sesuatu? Anda dapat menggunakan pihak ketiga yang tepercaya (misalnya, agen real estat atau pengacara) untuk menyelesaikan penjualan dan menjaga anonimitas Anda. Tapi kemudian pihak ketiga tahu siapa Anda. David Chaum, mempelajari masalah ini dan merancang seperangkat protokol yang dengannya pembayaran semacam itu dapat terjadi tanpa mengungkapkan pembeli kepada penjual.

Beberapa Identitas—Tertaut atau Tidak

Kebanyakan orang sudah memiliki banyak identitas. Bagi bank Anda, Anda adalah nomor rekening Anda. Untuk biro kendaraan bermotor Anda, Anda adalah nomor SIM Anda. Dan bagi perusahaan kartu kredit Anda, Anda adalah nomor kartu kredit Anda. Untuk tujuan tertentu, nomor-nomor ini adalah identitas Anda; fakta bahwa masing-masing dapat (atau mungkin tidak) diadakan atas nama Anda tidak relevan. Nama menjadi penting jika digunakan sebagai cara untuk menghubungkan angka-angka ini dan catatan terkaitnya. Berapa banyak orang yang membagikan nama Anda? Bisakah (atau haruskah) ada nilai kunci untuk menautkan basis data terpisah ini? Dan komplikasi apa yang muncul ketika kami mempertimbangkan salah eja dan beberapa bentuk nama Anda yang valid (dengan dan tanpa inisial tengah, dengan nama tengah lengkap, dengan salah satu dari dua nama tengah jika Anda memiliki dua, dan seterusnya)?

Selain itu, bagaimana jika Anda memiliki nama yang umum digunakan, atau nama Anda berubah sewaktu-waktu? Misalkan Anda mengubah nama Anda secara legal tetapi tidak pernah mengubah nama di kartu kredit Anda. Maka nama Anda tidak dapat dengan mudah digunakan sebagai kunci untuk ditautkan. Anda dapat mencoba menggunakan karakteristik sekunder sebagai pemverifikasi, seperti alamat. Namun, alamat menghadirkan risiko lain: Mungkin seorang penjahat tinggal di rumah Anda sebelum Anda membelinya. Anda tidak harus mempertahankan reputasi Anda karena penghuni sebelumnya yang tidak terkait. Kami juga bisa mencocokkan tanggal, jadi kami hanya menghubungkan orang-orang yang benar-benar tinggal di rumah pada waktu yang sama. Tapi kemudian rumah kelompok atau teman sekamar menimbulkan masalah tambahan. Sebagai ilmuwan komputer, kita tahu bahwa memprogram semua kemungkinan ini adalah mungkin, tetapi memerlukan pertimbangan yang cermat dan memakan waktu tentang potensi masalah sebelum merancang

solusi. Sayangnya, kita tahu bahwa terlalu sering keanehan yang tidak biasa tetapi kritis seperti itu tidak dipertimbangkan sampai setelah kode dikembangkan dan dipasang, dan kemudian setiap kasus luar biasa dianggap sendiri dan sering kali dengan tergesa-gesa. Kita bisa melihat potensi penyalahgunaan dan ketidaktepatan.

Menautkan identitas dengan benar untuk membuat berkas dan menghancurkan anonimitas menciptakan risiko privasi, tetapi menautkannya secara tidak benar menciptakan risiko yang jauh lebih serius untuk penggunaan data dan privasi orang yang terkena dampak. Jika kita berpikir dengan hati-hati, kita dapat menentukan banyak cara sistem seperti itu akan gagal—pendekatan yang mungkin efektif tetapi berpotensi mahal dan memakan waktu. Godaan untuk bertindak cepat tetapi tidak tepat juga akan mempengaruhi privasi.

Nama samaran

Terkadang, kami tidak menginginkan anonimitas penuh. Anda mungkin ingin memesan umbi bunga tetapi tidak ditempatkan di banyak milis untuk perlengkapan berkebun. Tetapi Anda juga ingin dapat melakukan pemesanan serupa lagi, meminta tulip warna yang sama seperti sebelumnya. Situasi ini membutuhkan nama samaran, pengidentifikasi unik yang dapat digunakan untuk menghubungkan catatan dalam database server tetapi tidak dapat digunakan untuk melacak kembali ke identitas asli.

Banyak identitas juga bisa nyaman; misalnya, Anda mungkin memiliki akun email profesional dan akun sosial. Demikian pula, identitas sekali pakai (yang Anda gunakan untuk sementara waktu dan kemudian berhenti digunakan) bisa nyaman. Ketika Anda mendaftar untuk sesuatu dan Anda tahu alamat email Anda selanjutnya akan dijual berkali-kali, Anda mungkin mendapatkan alamat email baru untuk digunakan hanya sampai email yang tidak diminta menjadi menindas. Seigneur dan Jensen [SEI03] membahas penggunaan alias email untuk menjaga privasi. Penggunaan ini, yang disebut nama samaran, melindungi privasi kami karena kami tidak perlu membocorkan apa yang kami anggap sebagai data sensitif.

Rekening bank Swiss memberikan contoh klasik nama samaran. Setiap nasabah hanya memiliki nomor untuk mengidentifikasi dan mengakses rekening, dan hanya beberapa pegawai bank terpilih yang boleh mengetahui identitas Anda; semua karyawan lain hanya melihat nomor rekening Anda. Pada laporan rekening, tidak ada nama yang muncul: Hanya nomor rekening atau nama samaran yang dicetak. “Hanya dalam kasus investigasi kriminal untuk pelanggaran narkoba, pendanaan terorisme atau kejahatan berat lainnya, identitas pemilik manfaat akan diungkapkan kepada pihak berwenang … Anda dilindungi oleh undang-undang kerahasiaan bank Swiss.” (http://swiss- banking-law.com/faq/)

Beberapa orang mendaftarkan nama samaran dengan penyedia email, sehingga mereka memiliki kotak drop anonim untuk email. Yang lain menggunakan nama samaran di ruang obrolan atau dengan layanan kencan online. Kami meninjau kembali gagasan nama samaran nanti dalam bab ini, ketika kami mempelajari privasi untuk email.

5.2.3 Pemerintah dan Privasi

Pemerintah mengumpulkan dan menyimpan data tentang warga, penduduk, dan pengunjung. Pada saat yang sama, pemerintah juga memfasilitasi dan mengatur perdagangan dan mengawasi kegiatan pribadi seperti perawatan kesehatan, pekerjaan, pendidikan, dan perbankan. Dalam peran tersebut, pemerintah merupakan enabler atau pengatur privasi sekaligus pengguna data pribadi. Pada bagian ini, kami mempertimbangkan beberapa implikasi dari akses pemerintah ke data pribadi.

Autentikasi

Pemerintah memainkan peran yang kompleks dalam otentikasi pribadi. Banyak instansi pemerintah (seperti biro kendaraan bermotor) menggunakan pengenal untuk melakukan pekerjaan mereka: mengautentikasi siapa Anda (misalnya, dengan paspor atau dokumen tempat tinggal) dan mengeluarkan dokumen autentikasi terkait (seperti SIM). Pemerintah juga dapat mengatur bisnis yang menggunakan materi identifikasi dan otentikasi. Dan terkadang pemerintah memperoleh data berdasarkan materi tersebut dari pihak lain (misalnya, pemerintah dapat membeli informasi laporan kredit dari perusahaan swasta untuk membantu menyaring daftar penumpang maskapai penerbangan untuk teroris). Dalam peran ganda ini, selalu ada potensi pemerintah untuk menyalahgunakan data dan melanggar hak privasi.

Risiko Akses Data

Menyadari risiko ini dalam akses pemerintah ke data pribadi, Menteri Pertahanan AS menunjuk sebuah komite untuk menyelidiki dan mendokumentasikan sifat risiko dalam pengumpulan data tersebut. Komite Penasihat Teknologi dan Privasi, yang diketuai oleh Newton Minow, mantan ketua Komisi Komunikasi Federal, membuat laporannya di 2004. Meskipun awalnya diminta untuk meninjau privasi dan pengumpulan data hanya di dalam Departemen Pertahanan, komite merasa tidak mungkin untuk memisahkan Departemen Pertahanan dari pemerintah lainnya. Akibatnya, deskripsinya berlaku untuk pemerintah Federal secara keseluruhan.

Di antara risiko yang diakui ketika pemerintah memperoleh data dari pihak lain adalah sebagai berikut:

- kesalahan data: berkisar dari kesalahan transkripsi hingga analisis yang salah
- penautan yang tidak akurat: dua atau lebih item data benar tetapi ditautkan secara salah oleh elemen yang dianggap sama
- perbedaan bentuk dan isi: presisi, akurasi, format, dan kesalahan semantik
- sengaja salah: dikumpulkan dari sumber yang dengan sengaja memberikan data yang salah, seperti kartu identitas palsu atau alamat palsu yang diberikan untuk menyesatkan
- tuduhan palsu: kesimpulan yang salah atau kedaluwarsa bahwa pemerintah tidak memiliki data untuk memverifikasi atau menolak, misalnya, tunggakan pembayaran pajak negara
- misi merayap: data diperoleh untuk satu tujuan yang mengarah ke penggunaan yang lebih luas karena data akan mendukung misi itu
- tidak terlindungi dengan baik: data dengan integritas yang meragukan karena cara pengelolaan dan penanganannya

Langkah-Langkah untuk Melindungi dari Kehilangan Privasi

Komite merekomendasikan beberapa langkah yang dapat diambil pemerintah untuk membantu menjaga data pribadi:

- Minimisasi data. Dapatkan data paling sedikit yang diperlukan untuk tugas tersebut. Misalnya, jika tujuannya adalah untuk mempelajari penyebaran suatu penyakit, hanya kondisi, tanggal, dan lokasi yang tidak jelas (kota atau kabupaten) yang cukup; nama atau informasi kontak pasien mungkin tidak diperlukan.
- Anonimisasi data. Jika memungkinkan, ganti informasi pengenal dengan kode yang tidak dapat dilacak (seperti nomor catatan). Namun pastikan kode tersebut tidak dapat ditautkan ke database lain yang mengungkapkan data sensitif.
- Audit. Catat siapa yang telah mengakses data dan kapan, baik untuk membantu mengidentifikasi pihak yang bertanggung jawab jika terjadi pelanggaran dan untuk mendokumentasikan tingkat kerusakan.
- Keamanan dan akses terkontrol. Cukup melindungi dan mengontrol akses ke data sensitif.
- Pelatihan. Pastikan bahwa orang yang mengakses data memahami apa yang harus dilindungi dan bagaimana melakukannya.
- Kualitas. Mempertimbangkan tujuan pengumpulan data, cara penyimpanan, usia, dan faktor serupa untuk menentukan kegunaan data.
- Penggunaan terbatas. Berbeda dengan pengendalian akses, tinjau semua usulan penggunaan data untuk menentukan apakah penggunaan tersebut konsisten dengan tujuan pengumpulan data dan cara penanganannya (divalidasi, disimpan, dikendalikan).
- Data tertinggal di tempatnya. Jika memungkinkan, tinggalkan data pada pemilik atau kolektor asli. Langkah ini membantu menjaga dari kemungkinan penyalahgunaan data dari misi yang diperluas hanya karena data tersedia.
- Kebijakan. Tetapkan kebijakan yang jelas untuk privasi data. Mencegah pelanggaran kebijakan privasi.

Langkah-langkah ini secara signifikan akan membantu memastikan perlindungan privasi. Juga, Amerika Serikat mulai membahas gagasan untuk menggabungkan banyak undang-undang pelanggaran data berbasis negara bagian menjadi satu undang-undang yang komprehensif. Pada tahun 2002, California meloloskan undang-undang di seluruh negara bagian pertama untuk mengatasi masalah yang berkembang dari pelanggaran keamanan database konsumen informasi pribadi. Oleh karena itu, undang-undang California mewajibkan setiap “lembaga negara bagian, atau orang atau bisnis yang menjalankan bisnis di California, yang memiliki atau melisensikan data terkomputerisasi yang mencakup informasi pribadi, sebagaimana didefinisikan, untuk mengungkapkan dengan cara tertentu, setiap pelanggaran keamanan data, didefinisikan, untuk setiap penduduk California yang informasi pribadinya yang tidak terenkripsi, atau secara wajar diyakini telah, diperoleh oleh orang yang tidak berwenang.” Undang-undang mengizinkan pemberitahuan yang tertunda hanya “jika lembaga penegak hukum memutuskan bahwa itu akan menghambat penyelidikan kriminal.” Ini juga mewajibkan setiap lembaga yang melisensikan informasi tersebut, seperti biro kendaraan bermotor atau departemen urusan regulasi, untuk memberi tahu pemilik atau penerima lisensi informasi tentang pelanggaran keamanan apa pun yang dapat mengancam privasi atau integritas data.

5.2.4 Pencurian identitas

Sesuai dengan namanya, pencurian identitas berarti mengambil atau mengambil identitas orang lain. Misalnya, menggunakan kartu kredit orang lain tanpa izin adalah penipuan. Pada tahun 1998 di Amerika Serikat, dengan Pencurian Identitas dan Undang-Undang Pencegahan Asumsi, mengeluarkan kartu kredit baru atas nama orang lain juga merupakan kejahatan: pencurian identitas. Pencurian identitas telah meningkat sebagai masalah dari masalah yang relatif jarang terjadi di tahun 1970-an menjadi masalah yang mempengaruhi 1 dari 20 konsumen saat ini. Pada tahun 2005, Komisi Perdagangan Federal AS menerima lebih dari 250.000 pengaduan tentang pencurian identitas Tetapi Laporan Penipuan Identitas 2014 Javelin mencatat bahwa pencurian identitas terjadi di Amerika Serikat setiap dua detik. Memang insiden pencurian identitas secara keseluruhan mempengaruhi 5,3 persen konsumen pada tahun 2013, naik dari 4,9 persen pada tahun 2012.

Pencurian identitas terjadi dalam banyak cara: pembukaan akun yang tidak sah atas nama orang lain, mengubah informasi akun untuk memungkinkan pencuri mengambil alih dan menggunakan akun atau layanan orang lain, atau melakukan penipuan dengan mendapatkan dokumen identitas atas nama curian. Sebagian besar kasus pencurian identitas menjadi jelas satu atau dua bulan setelah data dicuri, ketika tagihan atau transaksi penipuan mulai datang atau muncul di file korban. Pada saat itu, pencuri kemungkinan telah mendapat untung dan telah menghapus identitas yang dicuri, beralih ke korban baru.

Memiliki karakteristik identifikasi unik yang relatif sedikit memudahkan pencurian identitas: Seorang pencuri yang mendapatkan satu kunci, seperti nomor identitas nasional, dapat menggunakannya untuk mendapatkan yang kedua, dan keduanya untuk mendapatkan yang ketiga. Setiap kunci memberikan akses ke lebih banyak data dan sumber daya. Beberapa perusahaan atau agensi dibentuk untuk mengajukan pertanyaan otentikasi yang benar-benar diskriminatif (seperti toko grosir tempat Anda sering berbelanja atau kota tempat Anda baru saja membeli tiket pesawat atau digit ketiga pada baris empat dari pengembalian pajak terakhir Anda). Karena ada sedikit kunci otentikasi, kami sering diminta untuk memberikan kunci yang sama (seperti nama gadis ibu) kepada banyak orang, beberapa di antaranya mungkin menjadi kaki tangan paruh waktu dalam pencurian identitas. Departemen Kehakiman A.S. mengelola situs web pencurian identitas, dengan informasi tentang cara mencegah pencurian identitas dan apa yang harus dilakukan jika Anda menjadi korban: http://www.justice.gov/criminal/fraud/websites/idtheft.html.

5.3 Otentikasi dan Privasi

Kami tahu bahwa kata sandi adalah pembeda yang buruk dan jelas bukan pengidentifikasi. Anda tidak akan mengharapkan semua pengguna sistem memilih kata sandi yang berbeda. Yang kita butuhkan hanyalah pasangan ID-sandi menjadi unik. Di ujung lain spektrum, sidik jari dan pola pembuluh darah di retina mata dianggap unik: Mengingat pola sidik jari atau retina, kami berharap hanya mendapatkan satu identitas yang sesuai atau tidak menemukan kecocokan dalam database. Situasi itu mengasumsikan kita bekerja dengan citra yang baik. Jika sidik jari kabur atau tidak lengkap (bukan kontak yang lengkap atau pada permukaan yang sebagian tidak sesuai), kami mungkin mendapatkan beberapa kemungkinan kecocokan. Autentikator lain masih kurang canggih. Geometri tangan atau penampilan wajah tidak membedakan dengan baik. Pengenalan wajah,

khususnya, sangat bergantung pada kualitas gambar wajah: Mengevaluasi foto satu orang yang menatap langsung ke kamera sangat berbeda dengan mencoba mengidentifikasi satu wajah dalam Gambar orang banyak.

Dua tujuan yang berbeda bekerja di sini, meskipun keduanya terkadang membingungkan. Untuk otentikasi, kami memiliki identitas dan beberapa data otentikasi, dan kami menanyakan apakah data otentikasi cocok dengan pola untuk identitas yang diberikan. Artinya, seseorang mengaku sebagai orang X, dan otentikasi memverifikasi bahwa orang tersebut benar-benar X. Untuk identifikasi, kami hanya memiliki data otentikasi, dan kami menanyakan identitas mana yang sesuai dengan autentikator. Artinya, kami tidak memiliki siapa pun yang mengklaim identitas, tetapi kami harus mencari tahu siapa orang itu dari data otentikasi. Pertanyaan kedua ini (siapa ini?) Jauh lebih sulit dijawab daripada yang pertama (apakah ini X?).

Untuk menjawab yang pertama, kami memiliki karakteristik X dalam database kami, kami membandingkan orang tersebut dengan X, dan kami menyatakan cocok atau tidak cocok (atau terkadang probabilitas cocok). Untuk menjawab pertanyaan kedua, kita tidak tahu apakah subjeknya genap dalam database. Jadi, kita harus memeriksa setiap orang yang mungkin dalam database untuk melihat apakah ada kecocokan yang solid. Tetapi bahkan jika kami menemukan beberapa kecocokan parsial yang potensial, kami tidak tahu apakah mungkin ada kecocokan yang lebih baik dengan seseorang yang tidak ada dalam database kami. Selain itu, pada contoh pertama, kami hanya melakukan satu perbandingan: apakah ini X? Dalam contoh kedua, kita membutuhkan n perbandingan, di mana n adalah jumlah orang dalam database.

5.3.1 Apa Arti Otentikasi

Kami sebenarnya menggunakan istilah otentikasi untuk mengartikan tiga hal yang berbeda. Kami mengotentikasi individu, identitas, atau atribut. Seorang individu adalah pribadi yang unik. Mengautentikasi seorang individu adalah apa yang kita lakukan ketika kita mengizinkan seseorang memasuki ruangan yang dikendalikan: Kita hanya ingin manusia itu yang diizinkan masuk. Identitas adalah rangkaian karakter atau deskripsi serupa, tetapi tidak harus sesuai dengan satu orang, juga tidak setiap orang hanya memiliki satu nama. Identitas dapat menggambarkan sekelompok atau kategori orang yang memenuhi deskripsi yang diberikan. Misalnya, divisi penjualan perusahaan dapat didefinisikan sebagai identitas beberapa orang, yang memungkinkan siapa pun dalam grup tersebut untuk merespons di sales@company.com. Demikian pula, kami mengautentikasi identitas ketika kami mengakui bahwa siapa pun (atau apa pun) yang mencoba masuk sebagai admin telah menunjukkan autentikator yang valid untuk akun tersebut. Mengautentikasi identitas di ruang obrolan karena SuzyQ tidak mengatakan apa pun tentang orang yang menggunakan pengenal itu: Mungkin seorang gadis berusia 16 tahun atau sepasang detektif polisi pria paruh baya, yang di lain waktu menggunakan identitas FreresJacques.

Akhirnya, kami mengotentikasi atribut jika kami memverifikasi bahwa seseorang memiliki atribut itu. Atribut adalah karakteristik, seperti sidik jari atau profil DNA. Berikut adalah contoh mengautentikasi atribut. Beberapa bar, restoran, atau pub mengharuskan pengunjung berusia minimal 21 tahun untuk dapat minum alkohol. Penjaga pintu klub memverifikasi usia seseorang dan mencap tangan orang tersebut untuk menunjukkan bahwa pelindungnya berusia di atas 21 tahun. Perhatikan bahwa untuk memutuskan, penjaga pintu mungkin telah melihat kartu identitas yang mencantumkan tanggal

lahir orang tersebut, sehingga penjaga pintu mengetahui usia pasti orang tersebut untuk menjadi 24 tahun, 6 bulan, 3 hari. Sebagai alternatif, penjaga pintu mungkin diberi wewenang untuk melihat wajah seseorang dan memutuskan apakah orang dengan rambut beruban dan kerutan terlihat jauh melampaui usia 21 tahun sehingga tidak perlu diverifikasi. Otentikator stempel hanya menandakan bahwa orang tersebut memiliki atribut berusia 21 tahun atau lebih.

Dalam aplikasi komputasi, kami sering mengotentikasi individu, identitas, dan atribut. Masalah privasi dapat muncul ketika kita mengacaukan autentikasi yang berbeda ini dan apa artinya. Misalnya, nomor jaminan sosial A.S. tidak pernah dimaksudkan sebagai pengenal, tetapi sekarang sering berfungsi sebagai pengenal, autentikator, kunci basis data, atau ketiganya. Ketika satu nilai data melayani dua atau lebih kegunaan, seseorang yang memperolehnya untuk satu tujuan dapat menggunakannya untuk tujuan lain.

Menghubungkan identitas dengan seseorang itu rumit. Dalam Bab 5 kami menceritakan kisah rootkit, perangkat lunak berbahaya yang dengannya orang yang tidak berwenang dapat memperoleh kendali pengawasan komputer. Misalkan polisi menangkap Michel karena pelanggaran ringan dan menyita komputernya. Dengan memeriksa komputer, polisi menemukan bukti yang menghubungkan komputer itu dengan kasus spionase. Polisi menemukan pesan email yang memberatkan dari Michel di komputer Michel dan menagihnya. Dalam pembelaannya, Michel menunjuk ke rootkit di komputernya. Dia mengakui bahwa komputernya mungkin telah digunakan dalam spionase, tetapi dia menyangkal bahwa dia terlibat secara pribadi. Polisi, katanya, telah menarik hubungan yang tidak dapat dibenarkan antara identitas Michel dalam email dan Michel orang tersebut. Rootkit adalah penjelasan yang masuk akal tentang bagaimana orang lain bertindak menggunakan identitas Michel (komputernya). Contoh ini menunjukkan mengapa kita harus hati-hati membedakan antara otentikasi individu, identitas, dan atribut.

Otentikasi Individu

Ada relatif sedikit cara untuk mengidentifikasi individu. Ketika Anda lahir, kelahiran Anda dicatat di kantor catatan pemerintah, dan kantor tersebut menerbitkan akta kelahiran kepada orang tua Anda. Beberapa tahun kemudian, orang tua Anda mendaftarkan Anda di sekolah, menunjukkan akta kelahiran sehingga sekolah dapat mengeluarkan Anda kartu identitas sekolah. Masih nanti, Anda menyerahkan akta kelahiran dan foto untuk mendapatkan paspor atau kartu identitas nasional. Dengan cara yang sama, masing-masing dari kita menerima banyak nomor dan kartu otentikasi lainnya sepanjang hidup.

Proses seumur hidup ini dimulai dengan akta kelahiran bayi. Tetapi deskripsi fisik bayi (tinggi, berat, bahkan warna rambut) akan berubah secara signifikan hanya dalam beberapa bulan. Akta kelahiran mungkin berisi sidik jari bayi, tetapi mencocokkan sidik jari bayi yang baru lahir dengan sidik jari orang dewasa sangat sulit.

Untungnya, di sebagian besar pengaturan, dapat diterima untuk menerima otentikasi yang lemah untuk individu: Seorang teman yang telah mengenal Anda sejak kecil, seorang guru sekolah, tetangga, dan rekan kerja dapat mendukung klaim identitas.

Otentikasi Identitas

Kita semua menggunakan banyak identitas yang berbeda. Ketika Anda membeli sesuatu dengan kartu kredit, Anda melakukannya di bawah identitas pemegang kartu kredit. Di beberapa tempat Anda dapat membayar tol dengan perangkat frekuensi radio di mobil Anda, sehingga sensor mengautentikasi Anda sebagai pemegang perangkat tol tertentu. Anda mungkin memiliki paket makan yang dapat diakses melalui kartu, sehingga kasir mengautentikasi Anda sebagai pemilik kartu. Anda check in ke hotel dan mendapatkan kartu strip magnetik alih-alih kunci, dan pintu ke kamar Anda mengautentikasi Anda sebagai penduduk yang sah selama tiga malam berikutnya. Jika Anda memikirkan hari Anda, Anda mungkin akan menemukan lusinan cara beberapa aspek identitas Anda telah diautentikasi.

Dari sudut pandang privasi, mungkin ada atau tidak ada cara untuk menghubungkan semua identitas yang berbeda ini. Kartu kredit menautkan ke nama dan alamat pembayar kartu, yang mungkin Anda, pasangan Anda, atau siapa pun yang bersedia membayar pengeluaran Anda. Perangkat tol otomatis Anda menautkan ke nama dan mungkin alamat siapa pun yang membayar tol: Anda, pemilik mobil, agen persewaan, atau pemberi kerja. Ketika Anda melakukan panggilan telepon, ada otentikasi ke pemegang rekening telepon, dan sebagainya.

Terkadang kita tidak menginginkan suatu tindakan dikaitkan dengan suatu identitas. Misalnya, tip anonim atau penggunaan saluran telepon "whistle blower" adalah sarana untuk memberikan informasi anonim tentang aktivitas ilegal atau tidak pantas. Jika Anda tahu bos Anda menipu perusahaan, menghadapi atasan Anda mungkin bukan langkah yang baik untuk meningkatkan karier. Anda mungkin bahkan tidak ingin ada catatan yang memungkinkan atasan Anda menentukan siapa yang melaporkan penipuan tersebut. Jadi Anda melaporkannya secara anonim. Anda mungkin mengambil tindakan pencegahan menelepon dari telepon umum sehingga tidak akan ada cara untuk melacak orang yang menelepon. Dalam hal ini, Anda dengan sengaja mengambil langkah-langkah untuk mencegah pengenal umum menautkan Anda ke laporan.

Namun, karena akumulasi data dari waktu ke waktu, penautan masih dapat dilakukan. Saat Anda meninggalkan kantor untuk pergi ke telepon umum, ada catatan lencana yang Anda usap di pintu. Sebuah kamera pengintai menunjukkan Anda berdiri di telepon umum. Catatan kedai kopi termasuk stempel waktu yang menunjukkan saat Anda membeli kopi (menggunakan kartu loyalitas pelanggan Anda) sebelum kembali ke kantor Anda. Waktu perincian ini sesuai dengan waktu tip telepon anonim. Secara abstrak item data ini tidak menonjol dari jutaan lainnya. Tetapi seseorang yang menyelidiki beberapa menit di sekitar waktu tip dapat membuat tautan tersebut. Menghubungkan bisa dilakukan dengan tangan. Tetapi teknologi yang terus berkembang memungkinkan lebih banyak persamaan seperti ini untuk ditarik oleh komputer dari titik data yang tampaknya tidak berhubungan dan tidak menarik.

Oleh karena itu, untuk menjaga privasi kami, kami dapat menggagalkan upaya untuk menautkan catatan. Seorang teman memberi nama fiktif saat mendaftar kartu loyalitas pelanggan di toko. Teman lain membuat reservasi makan malam dengan nama samaran. Di toko tetangga, pegawai toko selalu menanyakan nomor telepon Anda ketika Anda membeli sesuatu, bahkan jika Anda membayar tunai. Anda dapat dengan senang hati memberi mereka satu; itu bukan nomor asli Anda. Banyak situs (lihat http://www.dmoz.org/Computers/Internet/E-mail/Spam/Preventing/Temporary_Addresses/)

menawarkan alamat email sementara untuk penggunaan satu kali, untuk periode validitas terbatas (hingga beberapa bulan), atau sampai alamat dihapus.

Catatan Anonim

Terkadang, elemen data individual tidak sensitif, tetapi hubungan di antara mereka sensitif. Misalnya, seseorang bernama Erin, seseorang memiliki kondisi medis diabetes; tak satu pun dari fakta itu sensitif. Keterkaitan bahwa Erin mengidap diabetes menjadi sensitif.

Peneliti medis ingin mempelajari populasi untuk menentukan kejadian penyakit, faktor umum, tren, dan pola. Untuk menjaga privasi, peneliti sering berurusan dengan catatan anonim: catatan dari mana informasi identitas telah dihapus. Jika catatan tersebut dapat dihubungkan kembali ke informasi identitas, privasi menderita. Jika, misalnya, nama telah dihapus dari catatan tetapi nomor telepon tetap ada, peneliti dapat menggunakan basis data nomor telepon yang berbeda untuk menentukan pasien, atau setidaknya nama yang ditetapkan untuk telepon tersebut. Menghapus informasi yang cukup untuk mencegah identifikasi atau identifikasi ulang sulit dilakukan dan juga dapat membatasi kemungkinan penelitian.

Seperti dijelaskan di Studi Kasus 5-5, Ross Anderson diminta untuk mempelajari database utama yang sedang disiapkan untuk warga Islandia. Basis data akan bergabung dengan beberapa basis data perawatan kesehatan untuk digunakan oleh peneliti dan profesional perawatan kesehatan. Anderson menunjukkan bahwa meskipun catatan telah dianonimkan, masih mungkin untuk menghubungkan catatan tertentu dengan orang individu. Meskipun ada kesulitan privasi yang signifikan, Islandia melanjutkan rencana untuk membangun database gabungan.

Studi Kasus 5-5 : Islandia Menimbang Anonimitas Terhadap Manfaat Publik

Pada tahun 1998, Islandia mengizinkan pembuatan database catatan medis, silsilah, dan informasi genetik warga negara. Seolah-olah, database ini akan memberikan data tentang penyakit genetik kepada para peneliti—profesional medis dan perusahaan obat. Islandia sangat menarik untuk penelitian penyakit genetik karena kumpulan gen tetap stabil untuk waktu yang lama; beberapa orang luar telah pindah ke Islandia, dan beberapa orang Islandia telah beremigrasi. Untuk privasi, semua nama atau nomor pengenal akan diganti dengan nama samaran yang unik. Departemen kesehatan Islandia meminta pakar keamanan komputer Ross Anderson untuk menganalisis aspek keamanan dari pendekatan ini.

Anderson menemukan beberapa kekurangan dengan pendekatan yang diusulkan:

- Pencantuman dalam database silsilah memperumit tugas menjaga anonimitas individu karena fitur keluarga yang khas. Selain itu, bagian dari database silsilah sudah terbuka untuk umum karena informasi tentang individu diterbitkan dalam catatan kelahiran dan kematian mereka. Misalnya, akan lebih mudah untuk mengidentifikasi seseorang dalam keluarga dengan tiga anak yang lahir masing-masing pada tahun 1910, 1911, dan 1929.

- Bahkan riwayat hidup peristiwa medis dapat mengidentifikasi seseorang. Banyak orang mungkin mengetahui identitas seseorang yang kakinya patah saat bermain ski pada suatu musim dingin dan terkena penyakit kulit pada musim panas berikutnya, jika kedua peristiwa itu terjadi tepat pada satu orang dalam database.
- Bahkan batasan sampel kecil pada kueri akan gagal melindungi dari serangan aljabar.
- Untuk menganalisis data genetik, yang menurut sifatnya pasti sangat detail, peneliti perlu membuat pertanyaan yang kompleks dan spesifik. Kemampuan kueri kuat yang sama ini dapat menyebabkan pemilihan kombinasi hasil yang sewenang-wenang.

Untuk alasan ini (dan lainnya), Anderson merekomendasikan untuk tidak melanjutkan pengembangan database publik. Terlepas dari masalah ini, Parlemen Islandia memilih untuk melanjutkan pembangunan dan rilis publik.

Dalam salah satu analisis paling menakjubkan tentang memperoleh identitas, Latanya Sweeney melaporkan bahwa 87 persen populasi Amerika Serikat kemungkinan besar dapat diidentifikasi dengan kombinasi kode pos 5 digit (disebut kode pos di Amerika Serikat) , jenis kelamin, dan tanggal lahir. Statistik itu luar biasa ketika Anda mempertimbangkan bahwa hampir 8.000 penduduk AS harus memiliki tanggal lahir yang sama atau bahwa populasi rata-rata di area kode pos 5 digit adalah 10.000. Sweeney mendukung analisis statistiknya dengan studi kehidupan nyata. Pada tahun 1997 dia menganalisis daftar pemilih Cambridge, Massachusetts, sebuah kota berpenduduk sekitar 50.000 orang, salah satunya adalah gubernur saat itu. Dia mengambil dia sebagai contoh dan menemukan bahwa hanya enam orang yang memiliki tanggal lahirnya, hanya tiga dari mereka adalah laki-laki, dan dia adalah satu-satunya dari tiga yang hidup dalam kode posnya. Sebagai figur publik, ia telah mempublikasikan tanggal lahirnya dalam literatur kampanyenya, tetapi tanggal lahir terkadang tersedia dari catatan publik. Pekerjaan serupa dalam memperoleh identitas dari catatan anonim menunjukkan seberapa besar kemungkinan seseorang untuk menyimpulkan identitas dari data lain yang mudah diperoleh.

Karya Sweeney menunjukkan dengan meyakinkan betapa sulitnya menganonimkan data secara efektif. Banyak catatan medis dikodekan dengan setidaknya jenis kelamin dan tanggal lahir, dan catatan tersebut sering dianggap dapat dirilis untuk tujuan penelitian anonim. Selanjutnya, peneliti medis mungkin menginginkan kode pos untuk menghubungkan kondisi medis dengan geografi dan demografi; misalnya, para peneliti mungkin ingin melacak penyebaran penyakit di seluruh wilayah geografis atau berdasarkan karakteristik pribadi. Hanya sedikit orang yang akan berpikir bahwa menambahkan kode pos akan menyebabkan tingginya tingkat pelanggaran privasi.

Kesimpulan

Seperti yang telah kita lihat, identifikasi dan otentikasi adalah dua aktivitas berbeda yang mudah dikacaukan. Bagian dari kebingungan muncul karena orang tidak membedakan dengan jelas konsep yang mendasarinya. Kebingungan diperkuat ketika item data digunakan untuk lebih dari satu tujuan.

Otentikasi tergantung pada sesuatu yang mengkonfirmasi properti. Dalam kehidupan, hanya ada sedikit autentikator suara, jadi kita cenderung menggunakan yang kita miliki secara berlebihan: nomor identifikasi, tanggal lahir, atau nama keluarga. Namun, seperti yang kami jelaskan, pengautentikasi tersebut terkadang juga digunakan sebagai kunci basis data, dengan konsekuensi negatif terhadap privasi.
Kami juga telah mempelajari kasus-kasus di mana kami tidak ingin diidentifikasi. Anonimitas dan nama samaran berguna dalam konteks tertentu. Tetapi pengumpulan dan korelasi data, dalam skala yang hanya dimungkinkan dengan komputer, dapat mengalahkan anonimitas dan pseudonimitas. Saat kami para profesional komputer memperkenalkan kemampuan komputer baru, kami perlu mendorong debat publik tentang masalah privasi terkait.

Pada bagian selanjutnya kita mempelajari data mining, proses pengambilan data yang melibatkan menghubungkan database.

5.4 Keamanan Privasi Data mining

Dalam Bab 7 kami menjelaskan proses dan beberapa masalah keamanan dan privasi data mining. Di sini kami mempertimbangkan bagaimana menjaga privasi dalam konteks data mining.

Data mining sektor swasta adalah industri yang menguntungkan dan berkembang pesat. Semakin banyak data yang dikumpulkan, semakin terbuka peluang untuk belajar dari berbagai agregasi. Menentukan tren, preferensi pasar, dan karakteristik mungkin baik karena mengarah pada pasar yang efisien dan efektif. Tetapi orang menjadi sensitif atau bahkan mungkin dirugikan jika informasi pribadi mereka diketahui tanpa izin. Lihat Kasus 5-6 untuk contoh sejauh mana pelacakan data dapat mempelajari individu.

Studi Kasus 5-6 : Perusahaan Tahu Lebih Banyak tentang Anda Daripada Anda Sendiri

Kumpulan big data memungkinkan organisasi membuat prediksi tentang Anda, tidak hanya menyesuaikan iklan tetapi juga menyarankan kemungkinan perubahan kesehatan atau perilaku. Misalnya, Duhigg menjelaskan bagaimana Target Corporation mengumpulkan data tentang setiap pelanggan aktual dan potensial. “Selama beberapa dekade, Target telah mengumpulkan sejumlah besar data tentang setiap orang yang secara teratur masuk ke salah satu tokonya. Kapan pun memungkinkan, Target memberikan kode unik kepada setiap pembelanja—dikenal secara internal sebagai nomor ID Tamu—yang mengawasi semua yang mereka beli. 'Jika Anda menggunakan kartu kredit atau kupon, atau mengisi survei, atau mengirimkan pengembalian dana, atau menghubungi saluran bantuan pelanggan, atau membuka email yang kami kirimkan kepada Anda atau mengunjungi Situs Web kami, kami akan mencatatnya dan tautkan ke ID Tamu Anda,'” kata salah satu analis data Target. “Kami ingin tahu semua yang kami bisa.”

Duhigg menjelaskan bagaimana Target menggunakan data ini untuk mengidentifikasi wanita yang kemungkinan besar berada di trimester kedua kehamilan mereka, untuk

menawarkan harga khusus pada barang-barang yang berhubungan dengan bayi. Ayah seorang wanita muda marah ketika iklan Target terkait kehamilan muncul di surat permukaan—hanya untuk mengetahui dari seorang putri yang malu bahwa tes kehamilan mengkonfirmasi apa yang sudah dicurigai Target.

Prediksi ini cukup mengganggu ketika mereka benar, tetapi mereka dapat merusak ketika mereka salah. Orang dapat ditolak kredit, pekerjaan atau hipotek, berdasarkan prediksi tentang kemungkinan perilaku mereka. Seperti yang telah kita lihat dalam bab ini, data bisa salah, prediksi bisa salah, dan orang-orang yang terpengaruh bisa jadi tidak sadar bahwa pilihan mereka dibatasi dengan cara ini.

5.4.1 Data Mining Pemerintah

Terutama yang meresahkan bagi sebagian orang adalah prospek data mining pemerintah. Kami percaya kami dapat menghentikan ekses dan perilaku mengganggu perusahaan swasta dengan menggunakan pengadilan, publisitas yang tidak diinginkan, atau bentuk tekanan lainnya. Jauh lebih sulit untuk menghentikan tindakan pemerintah. Orang takut pemerintah atau penguasa yang telah mengambil retribusi terhadap warga dianggap musuh, dan bahkan demokrasi yang dianggap bertanggung jawab dapat membuat kesalahan dalam menangani data. Banyak pengumpulan dan analisis data pemerintah terjadi tanpa publisitas; beberapa program tidak diumumkan dan yang lainnya sengaja dirahasiakan. Dengan demikian, warga tidak nyaman dengan apa yang bisa dilakukan pemerintah yang tidak terkendali. Dan karena data mining tidak sempurna atau tepat, mengoreksi data yang salah yang dipegang oleh pemerintah dan kesimpulan yang salah yang diambil dari data mining hampir tidak mungkin.

Privacy Preserving Data Mining

Karena data mining dapat mengancam privasi, para peneliti telah mencari cara untuk melindungi privasi selama operasi data mining. Pendekatan yang naif dan tidak efektif mencoba untuk menghapus semua informasi pengenal dari database yang sedang ditambang. Namun, terkadang, informasi pengidentifikasi diperlukan untuk penambangan dan bahkan mungkin menjadi tujuan data mining. Lebih penting lagi, identifikasi dimungkinkan bahkan ketika informasi identitas yang jelas dihapus dari database. Data mining biasanya menggunakan dua pendekatan—korelasi dan agregasi. .

Privasi untuk Korelasi (Correlation)

Korelasi melibatkan penggabungan database pada bidang umum. Seperti halnya melindungi hubungan sensitif antara Erin dan diabetes, pelestarian privasi untuk korelasi mencoba mengendalikan hubungan itu.

John Vaidya dan Chris Clifton membahas gangguan data sebagai cara untuk mencegah korelasi yang membahayakan privasi. Sebagai contoh sederhana, asumsikan dua database hanya berisi tiga record, seperti yang ditunjukkan pada Tabel 5-1. Bidang ID yang menghubungkan basis data ini memudahkan untuk melihat bahwa Riko menderita diabetes.

Name	ID
Riko	1
Anjas	2
Ningsih	3

ID	Condition
1	Diabetes
2	Sehat
3	Ginjal

Tabel 5-1 Contoh untuk Gangguan Data

Salah satu bentuk gangguan data melibatkan pertukaran bidang data untuk mencegah penautan catatan. Mengganti nilai kondisi Riko dan Ningsih (tetapi bukan nilai ID) memutuskan hubungan Riko dengan diabetes. Properti lain dari database dipertahankan: Tiga pasien memiliki nama sebenarnya dan tiga kondisi menggambarkan pasien secara akurat. Mengganti semua nilai data dapat mencegah analisis yang berguna, tetapi pertukaran yang terbatas akan menyeimbangkan privasi dan akurasi. Dengan contoh kami menukar Riko dan Ningsih saja, Anda masih tahu bahwa salah satu peserta menderita diabetes, tetapi Anda tidak dapat mengetahui apakah Ningsih (yang sekarang memiliki ID=1) telah ditukar atau tidak. Pada gilirannya, jika Anda tidak dapat mengetahui apakah suatu nilai telah ditukar, Anda tidak dapat berasumsi bahwa korelasi yang Anda peroleh adalah benar.

Tentu saja, dengan menghancurkan tautan dalam database, kami juga menolak kemampuan peneliti untuk memeriksa data untuk koneksi lain; misalnya, jika Tabel pertama juga berisi usia, peneliti mungkin ingin menganalisis data untuk melihat apakah usia pasien berkorelasi dengan adanya diabetes.

Contoh tiga titik data kami, tentu saja, terlalu kecil untuk aplikasi data mining yang realistis, tetapi kami membangunnya hanya untuk menunjukkan bagaimana pertukaran nilai akan dilakukan. Peluang satu dari tiga untuk mengidentifikasi orang dengan diabetes dengan benar tampaknya cukup tinggi untuk meyakinkan beberapa orang bahwa Ningsih adalah orangnya. Tapi contoh yang lebih realistis akan melibatkan database ribuan titik data, sehingga kemungkinan inferensi yang benar menjadi sangat kecil.

Pertimbangkan contoh yang lebih realistis dari database yang lebih besar. Kita mungkin memiliki alamat alih-alih nama, dan tujuan data mining adalah untuk menentukan apakah ada korelasi antara lingkungan dan penyakit, seperti campak. Bertukar semua alamat akan mengalahkan kemampuan untuk menarik kesimpulan yang benar tentang lingkungan. Bertukar sejumlah kecil tapi signifikan alamat akan menimbulkan ketidakpastian untuk menjaga privasi. Beberapa pasien campak mungkin dipindahkan dari lingkungan dengan insiden tinggi, tetapi pasien campak lainnya juga akan ditukar. Jika lingkungan memiliki insiden yang lebih tinggi daripada populasi umum, pertukaran acak akan menyebabkan lebih banyak kerugian daripada keuntungan, sehingga mengurangi kekuatan korelasinya. Setelah pertukaran nilai, korelasi yang sudah lemah mungkin menjadi sangat lemah sehingga secara statistik tidak signifikan. Tetapi korelasi yang sebelumnya kuat akan tetap signifikan, hanya saja tidak sekuat itu.

Dengan demikian, pertukaran nilai adalah teknik yang dapat membantu menyeimbangkan tujuan privasi dan akurasi dalam data mining.

Privasi untuk Agregasi

Agregasi tidak perlu secara langsung mengancam privasi. Seperti yang ditunjukkan dalam Bab 7, agregat (seperti jumlah, median, atau hitungan) sering kali bergantung pada begitu banyak item data sehingga sensitivitas item kontribusi tunggal disembunyikan. Statistik pemerintah menunjukkan hal ini dengan baik: Data sensus, statistik tenaga kerja, dan hasil sekolah menunjukkan tren dan pola untuk kelompok (seperti lingkungan atau distrik sekolah) tetapi tidak melanggar privasi satu orang pun.

Seperti yang juga kami jelaskan di Bab 7, serangan inferensi dan agregasi bekerja lebih baik di dekat ujung distribusi. Jika ada sangat sedikit atau sangat banyak titik dalam subset basis data, sejumlah kecil persamaan dapat mengungkapkan data pribadi. Rata-rata dari satu nilai data adalah nilai tersebut secara tepat. Dengan tiga nilai data, rata-rata setiap pasangan menghasilkan tiga persamaan dalam tiga tidak diketahui, yang Anda tahu dapat diselesaikan dengan mudah dengan aljabar linier. Pendekatan serupa bekerja untuk himpunan bagian yang sangat besar dari seluruh database. Subset berukuran sedang menjaga privasi dengan cukup baik. Jadi privasi dipertahankan dengan aturan n item, lebih dari k persen.

Gangguan data dapat digunakan untuk mengurangi risiko dari agregasi. Gangguan tidak membatasi kemampuan peneliti untuk bekerja dengan statistik kumpulan data; itu hanya mencegah menghubungkan identitas individu dengan item data tertentu, sehingga menjaga privasi. Seringkali, peneliti dapat menarik kesimpulan dari distribusi dan besarnya populasi, sehingga menjaga privasi tanpa menghalangi penelitian yang valid.

Vaidya dan Clifton juga menjelaskan metode dimana database dapat dipartisi untuk menjaga privasi. Contoh sepele kami di Tabel 9-1 bisa menjadi contoh database yang dipartisi secara vertikal untuk memisahkan asosiasi sensitif nama dan kondisi.

Ringkasan Privasi Data Mining

Seperti yang telah kami jelaskan di bagian ini, data mining dan privasi tidak saling eksklusif: Kami dapat memperoleh hasil dari data mining tanpa mengorbankan privasi. Benar, beberapa akurasi hilang dengan gangguan. Sebuah kontra-argumen adalah bahwa melemahnya kepercayaan dalam kesimpulan paling serius mempengaruhi hasil yang lemah; kesimpulan yang kuat menjadi hanya sedikit kurang kuat. Penelitian tambahan kemungkinan akan menghasilkan teknik tambahan untuk menjaga privasi selama operasi data mining.

Kami dapat memperoleh hasil tanpa mengorbankan privasi, tetapi privasi tidak akan ada secara otomatis. Teknik yang dijelaskan di sini harus diterapkan oleh orang yang memahami dan menghormati implikasi privasi. Jika dibiarkan, data mining berpotensi merusak privasi. Profesional keamanan perlu terus menekan privasi dalam aplikasi data mining.

5.5 Privasi di Web

Internet terkadang dipandang sebagai ancaman terbesar terhadap privasi. Seperti yang dikatakan Bab 7, keuntungan dari Internet, yang juga merupakan kerugian, adalah anonimitas. Seorang pengguna dapat mengunjungi situs web, mengirim pesan, dan berinteraksi dengan aplikasi tanpa mengungkapkan identitas. Setidaknya itulah yang ingin kami pikirkan. Sayangnya, karena hal-hal seperti cookie, adware, spybots, dan kode berbahaya, anonimitas itu dangkal dan sebagian besar sepihak. Aplikasi web yang canggih dapat mengetahui banyak tentang seorang pengguna, tetapi pengguna tersebut mengetahui sedikit tentang aplikasi tersebut.

Topiknya jelas sangat menarik: pencarian Google baru-baru ini menghasilkan lebih dari 7 miliar klik untuk istilah "web" dan "privasi" secara bersamaan, dan 634.000 klik untuk frasa "privasi web". Di bagian ini kami menyelidiki beberapa cara hilangnya privasi pengguna di Internet.

5.5.1 Memahami Lingkungan Online

Internet seperti bazar besar yang tidak diatur. Setiap kata yang Anda ucapkan dapat didengar oleh banyak orang lain. Dan tenda-tenda para pedagang tidak seperti kelihatannya: pedagang rempah-rempah sebenarnya menjalankan sarang perjudian, dan wanita baik hati yang menjual syal adalah tiga saudara bajak laut dan seekor harimau. Anda merogoh saku Anda untuk mendapatkan uang hanya untuk menemukan bahwa dompet Anda telah dikosongkan. Kemudian polisi memberi tahu Anda bahwa mereka akan senang membantu tetapi, sayangnya, tidak ada undang-undang yang berlaku. Penghapus peringatan di excelsis.

Kami sebelumnya telah menjelaskan anonimitas web: Sulit bagi dua pihak yang tidak terkait untuk mengotentikasi satu sama lain. Otentikasi internet paling sering mengkonfirmasi identitas pengguna, bukan server, sehingga pengguna tidak yakin apakah situs web itu sah. Ketidakpastian ini membuat sulit untuk memberikan informed consent untuk pelepasan data pribadi: Bagaimana persetujuan dapat diinformasikan jika Anda tidak tahu kepada siapa Anda memberikannya? Untuk contoh pelacakan dan privasi Internet, lihat Kasus 5-7.

Studi Kasus 5-7 : The Terrorists, Inc., Departemen TI

Pada tahun 2010, distrik sekolah Lower Merion dekat Philadelphia, Pennsylvania, ditemukan melacak siswanya secara online. Sekolah mungkin memiliki alasan yang sah untuk memantau penggunaan Internet oleh siswa, misalnya, saat di sekolah untuk menjauhkan anak-anak dari situs dewasa. Namun, dalam kasus ini, distrik sekolah telah mengeluarkan komputer untuk dibawa pulang oleh siswa dan ingin mempertanggungjawabkannya jika terjadi kehilangan atau pencurian. Tidak, sekolah tidak hanya memantau untuk menentukan lokasi semua komputer milik sekolah yang ditugaskan kepada siswa, tetapi juga secara aktif memantau aktivitas fisik siswa melalui web cam. Seorang siswa mengetahui pelacakan hanya ketika asisten kepala sekolah menuduhnya dengan perilaku yang tidak pantas di rumahnya sendiri dan menunjukkan Gambar web-cam sebagai bukti. (Siswa mengaku makan permen, tidak menggunakan narkoba.)

Distrik sekolah menyatakan bahwa mereka mengaktifkan kamera web dan mengumpulkan Gambar diam hanya untuk membantu melacak komputer yang hilang atau dicuri. Belakangan diketahui bahwa sekolah tersebut telah memperoleh 50.000 Gambar selama periode dua tahun, dan bahwa Gambar-Gambar ini menangkap siapa pun yang ada di depan kamera, tanpa sepengetahuan atau persetujuan. Keluarga siswa menggugat, dengan alasan pelanggaran Computer Fraud and Abuse Act (1986), Electronic Communications Privacy Act (1986), dan berbagai undang-undang Pennsylvania.

Distrik sekolah menyelesaikan dua tuntutan hukum atas insiden tersebut dengan biaya sekitar $600.000. FBI memutuskan untuk tidak mengajukan tuntutan terhadap distrik sekolah karena mereka tidak dapat menetapkan niat kriminal. (Sumber: WHYY News, 12 Okt 2010.) Seperti yang ditunjukkan oleh kasus ini, pelacakan komputer memiliki implikasi hak privasi yang penting.

Kebocoran data seperti ini bukanlah hal baru, tetapi pertumbuhan Internet telah memudahkan untuk menjangkau jutaan orang, seperti yang ditunjukkan oleh postingan WikiLeaks (http://wikileaks.org).

5.5.2 Pembayaran dalam Web

Pelanggan pedagang online harus dapat membayar secara online untuk pembelian. Ada dua pendekatan dasar: Pelanggan memberikan informasi kartu kredit mereka kepada pedagang atau mereka mengatur pembayaran melalui sistem pembayaran online seperti PayPal.

Pembayaran Kartu Kredit

Dengan kartu kredit, pengguna memasukkan nomor kartu kredit, nomor khusus yang tercetak pada kartu (mungkin untuk menunjukkan bahwa pengguna benar-benar memiliki kartu), tanggal kedaluwarsa kartu (untuk memastikan bahwa kartu saat ini aktif), dan alamat penagihan kartu kredit (mungkin untuk melindungi dari pencurian kartu kredit). Semua perlindungan ini ada di pihak pedagang: Mereka menunjukkan bahwa pedagang melakukan upaya terbaik untuk menentukan bahwa penggunaan kartu kredit itu sah. Tidak ada perlindungan kepada pelanggan bahwa pedagang akan mengamankan data ini. Setelah pelanggan memberikan informasi ini kepada satu pedagang, informasi yang sama adalah semua yang diperlukan untuk pedagang lain untuk menerima penjualan yang dibebankan ke kartu yang sama.

Selanjutnya, potongan informasi ini menyediakan banyak kunci statis yang digunakan untuk menghubungkan database. Seperti yang telah kita lihat, nama bisa jadi sulit untuk dikerjakan karena risiko salah eja, variasi dalam penyajian, pemotongan, dan sejenisnya. Nomor kartu kredit merupakan kunci yang sangat baik karena dapat disajikan hanya dengan satu cara dan bahkan ada digit cek yang sepele untuk memastikan bahwa nomor kartu adalah urutan yang valid.

Kartu debit juga dapat digunakan untuk pembayaran online. Meskipun bekerja dengan cara yang sama seperti kartu kredit, mereka biasanya tidak diberikan perlindungan yang sama seperti kartu kredit; ada risiko yang jauh lebih besar bagi pembayar untuk menggunakan debit daripada kredit.

Karena masalah dengan nomor kartu kredit yang dicuri, beberapa bank bereksperimen dengan kartu kredit sekali pakai: kartu yang dapat Anda gunakan untuk satu transaksi atau untuk jangka waktu yang tetap. Dengan begitu, jika nomor kartu dicuri atau disadap, tidak dapat digunakan kembali. Selain itu, memiliki beberapa nomor kartu membatasi kemampuan untuk menggunakan nomor kartu kredit sebagai kunci untuk mengkompromikan privasi melalui data mining.

Skema Pembayaran

Cara lain untuk melakukan pembayaran web adalah dengan skema pembayaran online, seperti PayPal. Anda membayar PayPal sejumlah uang dan menerima nomor rekening dan PIN. Anda kemudian dapat masuk ke situs pusat PayPal, memberikan alamat email dan jumlah yang harus dibayar, dan PayPal mentransfer jumlah tersebut. Karena di Amerika Serikat, PayPal tidak diatur di bawah undang-undang perbankan yang sama seperti kartu kredit, ia menawarkan perlindungan konsumen yang lebih sedikit daripada kartu kredit. Namun, keuntungan privasi adalah bahwa kartu kredit atau detail keuangan pengguna hanya diketahui oleh PayPal, sehingga mengurangi risiko dicuri. Skema serupa, seperti Square, menggunakan ponsel untuk melakukan pembayaran. Sistem lain, seperti Bitcoin, sedang ditetapkan sebagai mata uang virtual, terlepas dari penerbitan pemerintah. Nilai dan kelangsungan hidup mata uang virtual belum ditunjukkan.

5.5.3 Pendaftaran Situs dan Portal

Banyak situs memerlukan pendaftaran untuk digunakan. Situs meminta informasi dari Anda sebagai imbalan untuk memberi Anda akses ke informasi dan layanan situs. Seringkali pendaftarannya gratis; Anda tinggal memilih user ID dan password. Surat kabar dan portal web (seperti Yahoo! atau MSN) sangat menyukai teknik ini, dan penjelasan yang mereka berikan terdengar menenangkan: Mereka ingin melacak perilaku Anda di situs untuk meningkatkan pengalaman menjelajah Anda (apa pun artinya) dan dapat menawarkan konten kepada orang-orang dengan kebutuhan serupa di seluruh dunia. Pada kenyataannya, situs ingin mendapatkan demografi pelanggan, yang kemudian dapat mereka jual ke pemasar atau tunjukkan kepada pengiklan untuk menjamin iklan mereka.

Orang mengalami kesulitan mengingat banyak ID, sehingga mereka cenderung default ke yang sederhana, sering menggunakan variasi pada nama mereka. Dan karena orang-orang kesulitan mengingat ID, situs-situs menjadi lebih mudah: Banyak sekarang meminta Anda untuk menggunakan alamat email Anda sebagai ID Anda. Anda tidak hanya mengorbankan privasi alamat email Anda, Anda memberi situs pengidentifikasi Anda, yang juga pengidentifikasi Anda ke banyak situs lain. Masalah dengan menggunakan ID yang sama di banyak situs adalah bahwa ID itu sekarang menjadi kunci basis data di mana basis data yang sebelumnya terpisah dari situs yang berbeda dapat digabungkan. Parahnya lagi, karena ID atau alamat email seringkali berkaitan erat dengan nama asli individu, tautan ini juga menghubungkan identitas seseorang dengan data lain yang dikumpulkan. Jadi sekarang, agregator data dapat menyimpulkan bahwa V. Putin menelusuri situs web New York Times mencari artikel tentang vodka dan umur panjang, lalu membeli 200 lembar saham di penyulingan Rusia.

Anda tentu saja dapat mencoba mengingat banyak ID yang berbeda. Atau Anda dapat memilih persona sekali pakai, mendaftar untuk akun email gratis dengan nama seperti xxxyyy, dan tidak pernah menggunakan akun untuk apa pun kecuali pendaftaran gratis wajib ini. Dan seringkali

tampak bahwa ketika ada kebutuhan, muncullah suatu pelayanan. Lihat www.bugmenot.com untuk layanan yang akan memberikan ID dan kata sandi acak yang tidak dapat dilacak untuk situs yang memerlukan pendaftaran.

Alasan pendaftaran biasanya tidak ada hubungannya dengan surat kabar atau portal; itu ada hubungannya dengan pengiklan, orang-orang yang membayar agar konten web dapat disediakan. Web menawarkan kemungkinan pelacakan yang jauh lebih rinci daripada media lain. Misalkan Anda melihat papan iklan untuk permen di pagi hari dan iklan yang sama itu tetap ada di pikiran Anda sampai waktu makan siang; jika Anda kemudian membeli permen yang sama saat makan siang, pengiklan sangat senang: Uang iklan telah terbayar. Tetapi pengiklan tidak memiliki cara untuk mengetahui apakah Anda benar-benar melihat iklan (dan jika ya, yang mana). Ada beberapa tindakan kasar: Jika penjualan naik setelah kampanye iklan, kampanye mungkin memiliki beberapa efek. Tetapi pengiklan benar-benar ingin hubungan sebab-akibat yang lebih dekat, hubungan yang mudah diterapkan di web.

Iklan Pihak Ketiga

Anda mengunjungi Yahoo! Halaman web atau aplikasi olahraga, dan Anda mungkin melihat iklan untuk hipotek, perbankan, pinjaman mobil, dan majalah olahraga, tawaran televisi kabel, dan kupon diskon untuk rantai makanan cepat saji. Anda mengklik salah satu tautan, dan Anda langsung membuka formulir "beli di sini sekarang" atau Anda mendapatkan kupon khusus yang bernilai untuk pembelian Anda secara langsung. Iklan web jauh lebih terhubung dengan vendor: Anda melihat iklan, Anda mengkliknya, dan baik pembeli maupun pemilik halaman web tahu bahwa iklan melakukan tugasnya dengan menarik perhatian Anda. (Sebaliknya, pengiklan jarang tahu apakah Anda sedang menonton papan reklame jalan raya atau lalu lintas mereka.) Jika Anda mengklik dan membeli, iklan tersebut benar-benar terbayar. Jika Anda mengeklik dan kemudian menunjukkan kupon, nomor pelacakan pada kupon memungkinkan vendor dan pemilik laman web menautkan pembelian Anda ke iklan di situs web tertentu. Dari sudut pandang vendor, umpan balik langsung dan keterlacakan sangat bagus.

Tetapi apakah Anda ingin pihak-pihak yang terlibat ini mengetahui bahwa Anda menyukai bola basket dan sedang mencari hipotek kedua? Ingatlah bahwa, sejak Anda masuk ke situs portal, mereka sudah memiliki identitas yang dapat ditautkan ke nama Anda yang sebenarnya. Selain itu, Anda mungkin berurusan dengan lebih dari sekadar vendor dan situs web. Banyak jenis pihak ketiga yang dapat terlibat, banyak di antaranya menggunakan informasi untuk memahami kebiasaan dan preferensi Anda dan kemudian memberi Anda iklan bertarget.

Gambar 5-3 adalah cuplikan layar beranda Pearson Higher Education, penerbit buku ini. Pearson menggunakan pelacak, cookie, dan suar untuk menangkap informasi tentang perilaku Anda secara online. Seperti yang diungkapkan oleh program Ghostery dan tercantum dalam kotak di kanan atas, ada tiga pelacak di halaman beranda Pearson: untuk mengumpulkan dan menampilkan data kunjungan halaman (Google Analytics), mengizinkan pengujian presentasi halaman yang berbeda (Optimalkan), dan mengatur penyisipan kode pelacakan pada halaman terpisah (Adobe Tag Manager). Masing-masing panggilan tunggal ini dapat memanggil fungsi lain dari situs mana pun, sehingga ketiga pelacak ini dapat menjadi ujung dari upaya pemantauan yang jauh lebih besar. Nanti di bab ini, kita akan memeriksa beberapa jenis perangkat yang digunakan untuk melacak perilaku Anda secara online, serta strategi untuk membuatnya terlihat dan mengendalikan aktivitas mereka.



Gambar 5-3 Pemberitahuan Data Tracking

Sayembara dan Penawaran

Sulit untuk menolak sesuatu yang gratis. Kami akan mendaftar untuk kesempatan memenangkan hadiah besar, bahkan jika kami hanya memiliki peluang kecil untuk berhasil. Pengiklan tahu itu. Jadi kontes dan penawaran khusus sering meyakinkan orang untuk membocorkan detail pribadi. Pengiklan juga tahu bahwa orang-orang antusias pada saat itu, tetapi antusiasme dan perhatian mereka berkurang dengan cepat; akibatnya, pengiklan bekerja keras untuk "menutup kesepakatan" dengan cepat.

Promosi tipikal menawarkan sesuatu yang kecil secara gratis, untuk memikat Anda agar berkomitmen pada produk atau layanan. Pada hari-hari iklan pisau cukur yang aman, semboyannya adalah, "Beri mereka pisau cukur, lalu jual pisaunya kepada mereka." Saat ini, tawarannya lebih cenderung menjadi "Beri mereka layanan gratis selama satu bulan, lalu daftarkan mereka secara otomatis untuk melanjutkannya." Anda cukup mendaftar, memberikan nomor kartu kredit (yang tidak akan dikenakan biaya hingga bulan depan), dan Anda mendapatkan penggunaan layanan gratis selama satu bulan. Segera setelah Anda mendaftar, nomor kartu kredit dan nama Anda menjadi kunci untuk menautkan ke data lain tentang Anda. Faktanya, jika Anda menuju ke situs vendor melalui aplikasi atau akses web, mungkin sudah ada riwayat tautan dari situs penerusan yang dapat dieksploitasi oleh vendor. Jadi, bahkan jika Anda membatalkan layanan setelah bulan pertama, riwayat tautan tetap ada dan dapat dibagikan dengan dan digunakan untuk tujuan lain oleh banyak tautan dalam rantai.

5.5.4 Tindakan Pencegahan untuk Web Surfing

Kami telah melihat mengapa pemerintah, perusahaan, dan orang-orang ingin melacak aktivitas Anda dan mengumpulkan informasi tentang Anda. Pada bagian ini kami membahas beberapa teknologi yang digunakan untuk melakukan pelacakan dan pengumpulan: cookie dan bug web. Seperti yang telah kami catat, teknologi ini sering digunakan untuk memantau aktivitas tanpa sepengetahuan pengguna.

Cookie

Cookie adalah file data yang ditempatkan oleh situs web. Mereka benar-benar cara yang murah bagi pemilik situs web untuk mentransfer kebutuhan penyimpanannya dari situs webnya ke komputer atau telepon pengguna.

Cookie diformat sebagai file teks, disimpan di komputer pengguna, dan diteruskan oleh browser pengguna ke situs web saat pengguna membuka situs tersebut. Setiap file cookie terdiri dari sepasang item data yang dikirim ke browser web Anda oleh situs web yang dikunjungi: kunci dan nilai. Bersama-sama, pasangan mewakili status sesi saat ini antara pengguna yang berkunjung dan situs web yang dikunjungi. Kuncinya adalah URL situs yang membuat cookie. Nilai cookie dapat dianggap sebagai enam bidang: nama, data persisten, tanggal kedaluwarsa, jalur di server yang akan dikirim, domain server yang akan dikirim, dan persyaratan untuk koneksi yang aman (SSL) yang digunakan untuk mengirimkan cookie. Data persisten, yang sering dienkripsi, adalah sesuatu yang pemilik situs ingin simpan tentang pengguna untuk referensi di masa mendatang, misalnya, pengguna terakhir menelusuri mawar merah bertangkai panjang.

Setelah cookie ditempatkan pada sistem pengguna (biasanya dalam direktori dengan cookie lain), browser terus menggunakannya untuk interaksi selanjutnya antara pengguna dan situs web tersebut. Setiap cookie seharusnya memiliki tanggal kedaluwarsa, tetapi tanggal itu bisa jauh di masa depan—dan dapat dimodifikasi nanti atau bahkan diabaikan.

Misalnya, situs web Wall Street Journal, wsj.com, membuat cookie saat pengguna pertama kali masuk. Dalam transaksi berikutnya, cookie bertindak sebagai pengenal; pengguna tidak lagi memerlukan kata sandi untuk mengakses situs. Situs lain menggunakan pendekatan serupa. The Wall Street Journal memiliki dinding gaji; jika Anda bukan pelanggan berbayar, Anda tidak dapat masuk. The New York Times menggunakan cookie dengan cara yang berbeda; karena memiliki dinding pembayaran sebagian, situs surat kabar menggunakan cookie untuk melacak jumlah akses setiap bulan oleh pengguna tertentu. Jika pengguna melebihi sepuluh akses, dinding pembayaran naik, dan pengguna yang tidak membayar harus menunggu hingga bulan depan untuk dapat membaca lebih dari sekadar berita utama.

Sebuah portal seperti Yahoo! menggunakan cookie untuk memungkinkan pengguna menyesuaikan tampilan halaman web. Misalkan Sadie menginginkan latar belakang yang cerah untuk berita utama, cuaca, dan emailnya; Norman menginginkan latar belakang pastel yang lembut untuk hasil pasar saham, berita tentang film yang sedang diputar di daerahnya, dan hal-hal menarik yang terjadi pada hari ini dalam sejarah. Yahoo! dapat menyimpan semua informasi preferensi ini dalam basis datanya dan dengan mudah menyesuaikan halaman yang dikirimkannya ke dua pengguna ini. Dengan demikian, preferensi untuk Sadie atau Norman disimpan di komputer mereka sendiri dan dikembalikan ke Yahoo! untuk membantu Yahoo! membentuk dan mengirimkan halaman web sesuai dengan preferensi Sadie atau Norman.

Sebuah situs dapat mengatur cookie sebanyak yang diinginkan, dengan nilai sebanyak yang diinginkan. Seperti disebutkan di atas, beberapa situs menggunakan cookie untuk menghindari pelanggan harus masuk pada setiap kunjungan ke situs; cookie ini berisi ID pengguna dan kata sandi. Tetapi cookie juga dapat berisi, misalnya, nomor kartu kredit, nama pelanggan dan alamat pengiriman, tanggal kunjungan terakhir ke situs, jumlah barang yang dibeli atau jumlah dolar pembelian.

Informasi sensitif, seperti nomor kartu kredit atau bahkan nama dan alamat, harus dienkripsi atau dilindungi dalam cookie. Terserah situs untuk menentukan atau menentukan jenis perlindungan apa yang berlaku untuk cookie-nya. Pengguna tidak pernah tahu apakah atau bagaimana data dilindungi.

Bidang jalur dan domain seharusnya melindungi terhadap satu situs yang dapat mengakses cookie lain. Namun, seperti yang kami tunjukkan di bagian berikutnya, satu perusahaan dapat bekerja sama dengan perusahaan lain untuk membagikan data cookie.

Cookie Pihak Ketiga

Saat Anda mengunjungi sebuah situs, servernya meminta browser Anda untuk menyimpan cookie. Saat Anda mengunjungi situs itu lagi, browser Anda meneruskan cookie itu kembali ke situs tersebut. Alur umum adalah dari server ke browser Anda dan kemudian kembali ke tempat asal cookie. Halaman web juga dapat berisi cookie untuk organisasi. Karena cookie ini untuk organisasi selain pemilik halaman web, cookie ini disebut cookie pihak ketiga. Perusahaan pelacakan pihak ketiga menerima laporan dari masing-masing situs dan menghubungkan data untuk memberikan kecerdasan prediktif.

Cookie pihak ketiga mengizinkan agregator untuk menautkan informasi dari kunjungan pengguna ke situs web organisasi yang berbeda.

Misalnya, DoubleClick (anak perusahaan Google) memiliki perjanjian dengan jaringan situs web yang mengirimkan konten: berita, olahraga, makanan, keuangan, perjalanan, dan sebagainya. Perusahaan dalam jaringan setuju untuk berbagi data dengan DoubleClick. Geary menunjukkan bahwa DoubleClick mendapat untung dari tiga aktivitas:

- Penayangan iklan. DoubleClick menampilkan iklan di situs web pelanggan.
- Pengiriman iklan. DoubleClick memungkinkan pengiklan untuk mengontrol seberapa sering iklan ditampilkan dan berapa lama setiap penayangan berlangsung.
- Penargetan perilaku. Untuk satu pemilik situs web, penerbit menyetel kuki untuk mengetahui bagian mana dari situs yang dijelajahi pelanggan. Kemudian DoubleClick mencocokkan iklan dengan minat yang ditunjukkan oleh kebiasaan menjelajah pelanggan. Tetapi DoubleClick juga telah membentuk divisi yang disebut AdSense, yang membentuk jaringan pengiklan yang mengumpulkan informasi yang mereka kumpulkan, memungkinkan setiap anggota jaringan untuk menyempurnakan iklan penargetan.

Jadi, pada dasarnya, DoubleClick tahu di mana Anda berada, ke mana Anda pergi, dan iklan apa yang ditempatkan. Tetapi karena dapat membaca dan menulis cookie-nya, ia dapat merekam semua informasi ini untuk digunakan di masa mendatang.

Kebijakan privasi Google menjelaskan seperti apa tampilan cookie DoubleClick generik:

- time: 01/Jan/2015 12:01:00
- ad_placement_id: 103 (the ID of where the advertisement was viewed on the website)
- ad_id: 1234 (the unique ID of the advertisement)
- userid: 0000000000000001 (the unique number the cookie has given your browser)
- client_ip: 123.45.64.89
- referral_url: http://youtube.com/categories (the page where you saw the advertisement)

Geary menunjukkan, “karena mencatat alamat IP Anda, DoubleClick juga dapat menebak dengan baik negara dan kota/kota Anda juga.”

Berikut adalah contoh hal lain yang dapat dilakukan cookie pihak ketiga:

- Hitung berapa kali browser ini telah melihat halaman web tertentu.
- Lacak halaman yang dilihat pengunjung di dalam situs atau di berbagai situs.
- Hitung berapa kali iklan tertentu muncul.
- Mencocokkan kunjungan ke situs dengan tampilan iklan untuk situs tersebut.
- Cocokkan pembelian dengan iklan yang dilihat seseorang sebelum melakukan pembelian.
- Rekam dan laporkan string pencarian dari mesin pencari.

Tentu saja, semua aktivitas penghitungan dan pencocokan ini menghasilkan statistik yang juga dapat dikirim oleh situs cookie kembali ke situs pusat setiap kali cookie diaktifkan. Dan data yang dikumpulkan ini juga tersedia untuk dikirim ke mitra lain dari jaringan cookie.

Untuk melihat secara detail cara kerja cookie pihak ketiga, anggap Anda mengunjungi halaman investasi pribadi yang, dibiayai oleh iklan, berisi ruang untuk iklan dari empat pialang saham. Mari kita asumsikan juga bahwa delapan kemungkinan pialang dapat mengisi empat slot iklan ini. Saat halaman dimuat, DoubleClick mengambil cookie-nya, melihat bahwa Anda pernah mengunjungi halaman itu sebelumnya, dan juga melihat bahwa Anda pernah mengklik iklan atau tautan broker B5 di masa lalu. Berdasarkan riwayat tersebut, DoubleClick mungkin akan mengatur agar B5 menjadi salah satu dari empat pialang yang ditampilkan kepada Anda kali ini. Jika kuki juga menunjukkan bahwa Anda sebelumnya telah melihat iklan untuk mobil dan perhiasan yang sangat mahal, maka DoubleClick juga dapat menempatkan iklan untuk pialang harga penuh, bukan pialang diskon, di tiga slot lainnya. Tujuan dari layanan ini adalah untuk menyajikan iklan yang paling mungkin menarik minat pelanggan, yang merupakan kepentingan terbaik semua orang.

Namun strategi ini juga memungkinkan DoubleClick membuat dokumen yang kaya tentang kebiasaan menjelajah web Anda. Jika Anda mengunjungi situs perjudian online dan kemudian mengunjungi situs peminjaman uang, DoubleClick mengetahuinya. Jika Anda membeli obat herbal untuk tekanan darah tinggi dan kemudian mengunjungi situs asuransi kesehatan, DoubleClick tahu. DoubleClick mengetahui informasi pribadi apa yang sebelumnya Anda berikan di formulir web, seperti afiliasi politik, masalah seksual, agama, status keuangan atau medis, atau informasi identitas. Bahkan tanpa Anda memberikan data pribadi, hanya dengan membuka halaman web untuk satu partai politik dapat menempatkan Anda dalam daftar ajakan partai tersebut dan daftar musuh partai lain.

Jenis kegiatan ini dikenal sebagai profil online. Setiap bagian data tersedia untuk masing-masing perusahaan yang menyajikan halaman web; DoubleClick mengumpulkan dan mendistribusikan ulang item data terpisah ini sebagai satu paket.

Agaknya semua penjelajahan bersifat anonim. Tetapi seperti yang telah kami tunjukkan sebelumnya, ID login, alamat email, dan detail pengiriman atau penagihan yang disimpan semuanya dapat mengarah pada pencocokan seseorang dengan dokumen ini, jadi ini bukan lagi string cookie yang tidak disebutkan namanya. Pada tahun 1999, DoubleClick membeli Abacus, perusahaan lain yang mengelola database pemasaran. Abacus mengumpulkan data belanja pribadi dari pedagang katalog; dengan akuisisi itu, DoubleClick memperoleh cara untuk menautkan nama dan alamat pribadi yang sebelumnya hanya merupakan pola mesin, bukan orang.

Asosiasi ini mewakili keterkaitan yang sangat mungkin tetapi tidak pasti, karena dua alasan. Pertama, cookie biasanya mengaitkan aktivitas dengan mesin, bukan pengguna. Jika semua anggota keluarga berbagi satu mesin atau jika seorang tamu meminjam mesin tersebut, koneksi yang terlihat akan menjadi palsu. Kedua, karena cookie mengaitkan tindakan pada browser, hasilnya tidak lengkap jika seseorang menggunakan dua atau lebih browser atau akun atau mesin. Anda dapat menggunakan kelemahan ini untuk menginformasikan teknik penghindaran Anda. Namun, seperti dalam banyak aspek privasi lainnya, ketika pengguna tidak mengetahui data apa yang telah dikumpulkan, mereka tidak dapat mengetahui validitas data tersebut.

5.5.5 Bug Web: Apakah ada Alat Pembasminya?

Cookie adalah file teks yang disimpan di komputer Anda. Mereka menyimpan dan mengembalikan data untuk pemilik cookie, tetapi mereka sendiri tidak melakukan tindakan apa pun. Tapi bug web, lebih berbahaya: mereka adalah grafik tak terlihat yang disematkan dalam Gambar yang berada di halaman web. Kadang-kadang disebut GIF yang jelas atau 1 × GIF, bug web adalah satu piksel demi satu piksel, terlalu kecil untuk dideteksi dengan penglihatan normal. Untuk browser web, ukuran bug tidak masalah. Gambar adalah Gambar, terlepas dari ukurannya; browser akan meminta file, seolah-olah untuk menampilkan Gambar itu, dari alamat yang diberikan. File, bagaimanapun, tidak terbatas pada Gambar; itu dapat mencakup musik atau video atau yang lebih penting, itu dapat berisi skrip yang dapat dieksekusi, misalnya, untuk menghidupkan Gambar yang diunduh.

Perbedaan antara cookie dan bug sangat besar. Cookie adalah alat pelacak, menyimpan informasi di mesin Anda yang dapat dibaca nanti oleh server web, tetapi hanya oleh server yang menyetel cookie. Dengan demikian, cookie mengungkapkan tindakan Anda hanya saat berada di satu situs. Cookie adalah objek pelacakan pasif, bertindak sebagai catatan kecil yang menunjukkan di mana Anda berada atau apa yang telah Anda lakukan. Satu-satunya informasi yang dapat mereka kumpulkan adalah apa yang Anda berikan kepada mereka dengan memasukkan data atau memilih objek di halaman web. Karena cookie disimpan di mesin Anda, Anda dapat menghapus cookie sesuka hati untuk mengurangi jumlah data yang dikembalikan ke host web pada kunjungan berikutnya.

Sebaliknya, bug web dapat menjalankan proses yang dapat berasal dari lokasi mana pun, dan bug apa pun dapat memanggil lebih banyak bug dan karenanya lebih banyak kode. Halaman web periklanan biasa mungkin memiliki 20 bug web, mengundang 20 situs lain untuk menjatuhkan Gambar, skrip, atau bug web lainnya ke mesin pengguna. Seperti yang kami jelaskan di Bab 4,

kode yang dapat dieksekusi dapat melakukan tindakan apa pun yang diizinkan oleh pengguna, seperti membaca data dan mengirim item menarik ke luar lokasi. Semua aktivitas ini terjadi tanpa sepengetahuan atau kendali langsung Anda.

Sayangnya, pemusnahan tidak sesederhana melarang Gambar yang lebih kecil dari yang terlihat oleh mata, karena banyak halaman web menggunakan Gambar seperti itu dengan polos untuk membantu menyelaraskan konten. Atau beberapa aplikasi visual khusus mungkin benar-benar menggunakan kumpulan Gambar kecil untuk tujuan yang valid. Jawabannya bukan untuk membatasi Gambar tetapi untuk membatasi tindakan yang dapat dilakukan oleh bug. Namun, membatasi bug web juga membatasi kekayaan tampilan konten (pikirkan Gambar bergerak, musik, tayangan slide, bahkan menu tarik-turun dinamis). Situs web, dan terutama pengiklan, tidak mau melepaskan kemampuan ini, sehingga tindakan bug web kemungkinan tidak akan dibatasi secara signifikan.

Seperti yang kita lihat di bagian berikutnya, spyware jauh lebih kuat daripada bug atau cookie—dan berpotensi lebih berbahaya.

Spyware

Cookie adalah file pasif dan data yang dapat mereka tangkap terbatas. Mereka tidak dapat, misalnya, membaca registri komputer, membaca dengan teliti kotak keluar email, atau menangkap struktur direktori file. Spyware adalah kode aktif yang dapat melakukan semua hal yang tidak dapat dilakukan cookie. Umumnya, spyware dapat melakukan apa saja yang dapat dilakukan oleh suatu program, karena itulah mereka: program.

Spyware adalah kode yang dirancang untuk memata-matai pengguna, mengumpulkan data (termasuk apa pun yang diketik pengguna). Di bagian ini kami menjelaskan berbagai jenis spyware.

Keystroke Logger dan Spyware

Dalam Bab 4 kami menjelaskan keystroke logger, program yang berada di komputer dan merekam setiap tombol yang ditekan. Pencatat yang canggih melakukan diskriminasi, merekam hanya situs web yang dikunjungi atau, bahkan lebih serius, hanya penekanan tombol yang dimasukkan di situs web tertentu (misalnya, ID masuk dan kata sandi ke situs perbankan).

Keystroke logger adalah komputer yang setara dengan penyadapan telepon. Ini adalah program yang mencatat setiap kunci yang diketik. Seperti yang dapat Anda bayangkan, keystroke logger dapat secara serius membahayakan privasi dengan mendapatkan kata sandi, nomor rekening bank, nama kontak, dan argumen pencarian web.

Spyware adalah istilah yang lebih umum yang mencakup keystroke logger dan juga program yang diam-diam merekam aktivitas pengguna dan data sistem, meskipun tidak harus pada tingkat setiap penekanan tombol individu. Pusat Demokrasi dan Teknologi [CDT09] telah menyelidiki ancaman spyware terhadap privasi. CDT menunjukkan bahwa, “Istilah 'spyware' telah diterapkan pada segala hal mulai dari keystroke logger, hingga aplikasi periklanan yang melacak penjelajahan web pengguna, hingga cookie web, hingga program yang dirancang untuk membantu menyediakan patch keamanan langsung kepada pengguna. Baru-baru ini, ada perhatian khusus yang diberikan pada berbagai aplikasi yang mendukung perangkat lunak berbagi file peer-to-peer dan unduhan gratis lainnya

sebagai cara untuk mendapatkan akses ke komputer orang." Laporan CDT membahas secara rinci "aplikasi serupa lainnya, yang semakin menjadi fokus proposal legislatif dan peraturan. Banyak dari aplikasi ini mewakili ancaman privasi yang signifikan, tetapi dalam pandangan kami masalah yang lebih besar yang diangkat oleh program ini adalah transparansi dan kontrol pengguna, masalah yang terkadang diabaikan dalam diskusi tentang masalah tersebut dan sampai batas tertentu dikaburkan oleh istilah 'spyware' itu sendiri."

Tujuan dari spyware umum dapat meluas ke pencurian identitas dan aktivitas kriminal lainnya. Selain dampak privasi, keystroke logger dan spyware terkadang berdampak buruk pada sistem komputasi. Tidak selalu ditulis atau diuji dengan hati-hati, spyware dapat mengganggu program lain yang sah. Selain itu, mesin yang terinfeksi spyware sering kali memiliki beberapa bagian spyware yang berbeda yang dapat saling bertentangan, menyebabkan dampak serius pada kinerja.

Karakteristik umum lainnya dari berbagai jenis spyware adalah sulitnya menghapusnya. Untuk satu produk spyware, Altnet, penghapusan melibatkan setidaknya dua belas langkah, termasuk menemukan file di berbagai folder sistem.

Hijacker

Kategori lain dari spyware adalah perangkat lunak yang membajak program yang diinstal untuk tujuan yang berbeda. Misalnya, perangkat lunak berbagi file biasanya digunakan untuk berbagi salinan file musik atau film. Sebuah program ABC News pada tahun 2006 melaporkan bahwa pembayar pajak menemukan pengembalian pajak mereka di Internet setelah pembayar pajak menggunakan program file-sharing. Layanan berbagi musik seperti KaZaa (tidak lagi beroperasi) dan Morpheus memungkinkan pengguna untuk menawarkan sebagian dari file yang mereka simpan kepada pengguna lain. Menurut Pusat Demokrasi dan Teknologi, ketika pengguna menginstal KaZaa, program kedua, Altnet, juga diinstal. Dokumentasi untuk Altnet mengatakan akan menyediakan daya komputasi yang tidak digunakan pada mesin pengguna untuk mitra bisnis yang tidak ditentukan. Lisensi untuk Altnet memberikan Altnet hak untuk mengakses dan menggunakan daya komputasi dan penyimpanan yang tidak digunakan. Mencari "Altnet spyware" di mesin pencari mengungkapkan penawaran untuk lusinan produk yang mengklaim menghapus Altnet, serta lusinan entri blog yang menjelaskan kesulitan melakukannya.

Masalah privasi untuk layanan seperti Altnet adalah bahwa meskipun pengguna mengizinkan penggunaan daya komputasi cadangan atau berbagi atau file atau sumber daya lainnya, mungkin tidak ada kontrol atas akses ke data sensitif lainnya di komputer pengguna.

Adware

Adware menampilkan iklan yang dipilih di jendela pop-up atau di jendela browser utama. Topik dan karakteristik iklan dipilih sesuai dengan preferensi, deskripsi, dan riwayat pengguna, yang dikumpulkan oleh browser atau program tambahan dengan memantau penggunaan komputasi pengguna dan melaporkan informasi ke pangkalan.

Adware biasanya diinstal sebagai bagian dari perangkat lunak lain tanpa pemberitahuan. Terkubur dalam lisensi panjang pengguna dari perangkat lunak lain adalah referensi ke "perangkat lunak X dan ekstensi," sehingga pengguna bisa dibilang memberikan izin untuk instalasi adware. Perangkat lunak berbagi file adalah target umum adware, tetapi juga pengelola unduhan yang mengambil file

besar dalam beberapa aliran sekaligus untuk unduhan yang lebih cepat. Dan produk yang mengaku sebagai alat keamanan, seperti agen antivirus, telah diketahui menyimpan adware.

Penulis perangkat lunak adware dibayar untuk menampilkan iklan klien mereka di depan pengguna, yang mereka lakukan dengan jendela pop-up, iklan yang menutupi iklan yang sah, atau iklan yang menempati seluruh permukaan layar. Lebih halus, adware dapat menyusun ulang hasil mesin pencari sehingga produk klien mendapatkan penempatan yang lebih tinggi atau menggantikan produk orang lain sepenuhnya.

Zango adalah perusahaan yang menghasilkan iklan pop-up sebagai tanggapan atas situs yang dikunjungi. Ini mendistribusikan perangkat lunak untuk diinstal pada komputer pengguna untuk menghasilkan pop-up dan mengumpulkan data untuk memberi tahu Zango tentang iklan mana yang akan ditampilkan. Pada tahun 2006, Pusat Demokrasi dan Teknologi mengajukan keluhan kepada Komisi Perdagangan Federal tentang Zango, yang akhirnya menuduh bahwa Zango melanggar Undang-Undang Komisi Perdagangan Federal dengan

- menipu gagal untuk mengungkapkan adware
- menginstal adware secara tidak adil
- mencegah uninstall secara tidak adil

Selama bertahun-tahun setelah itu, peneliti keamanan seperti Ben Edelman dari Harvard terus mengklaim bahwa Zango berperilaku buruk: “Zango melanjutkan banyak praktik yang mungkin membingungkan, menipu, atau membahayakan pengguna biasa serta praktik yang secara khusus bertentangan dengan kewajiban Zango berdasarkan penyelesaian November 2006 dengan FTC.” Banyak vendor alat keamanan menghasilkan produk yang ditujukan untuk mencopot pemasangan Zango. Pendiri Zango menyatakan bangkrut dan menutup perusahaan pada tahun 2009.

5.5.6 Belanja di Internet

Pedagang web mengklaim menawarkan harga terbaik untuk suatu produk atau layanan karena banyak pedagang bersaing untuk bisnis Anda, bukan? Belum tentu begitu. Dan spyware sebagian yang harus disalahkan.

Pertimbangkan dua kasus: Anda memiliki toko fisik yang menjual perangkat keras. Salah satu pelanggan Anda, Viva, sangat setia: Dia telah berbelanja di toko Anda selama bertahun-tahun; dia tidak akan berpikir untuk pergi ke tempat lain. Viva juga cukup kaya; dia secara teratur membeli barang-barang mahal dan cenderung membeli dengan cepat. Joan adalah pelanggan baru. Anda menganggap dia pernah ke toko perangkat keras lain tetapi sejauh ini dia belum membeli banyak dari Anda. Joan berjuang dengan keluarga besar, hipotek besar, dan tabungan kecil. Kedua wanita tersebut mengunjungi toko Anda pada hari yang sama untuk membeli palu, yang biasanya Anda jual seharga $20. Berapa harga yang Anda tawarkan masing-masing? Banyak orang mengatakan Anda harus memberi Viva harga yang bagus karena kesetiaannya. Yang lain mengatakan kesetiaannya memberi Anda ruang untuk menghasilkan keuntungan. Dan dia pasti mampu membelinya. Adapun Joan, apakah dia akan menjadi pelanggan tetap? Jika dia pernah ke tempat lain, apakah dia berbelanja berdasarkan harga untuk semuanya? Jika Anda memenangkan bisnisnya dengan harga bagus, bisakah Anda meyakinkannya untuk tetap tinggal? Atau kembali lagi lain waktu? Toko

perangkat keras tidak melalui analisis ini: palu seharga $20 dihargai $20 hari ini, besok, dan minggu depan, untuk semua orang, kecuali ada obral.

Tidak benar secara online. Ingat, online Anda tidak melihat harga di rak; Anda hanya melihat harga yang dikutip untuk Anda di halaman yang menunjukkan palu. Kecuali seseorang yang duduk di depan komputer terdekat sedang melihat palu yang sama, Anda tidak akan tahu apakah orang lain ditawari penawaran harga yang berbeda dari $20.

Menurut sebuah penelitian yang dilakukan oleh Joseph Turow et al. [TUR05] Publik Annenberg Pusat Kebijakan Sekolah Komunikasi Universitas Pennsylvania, diskriminasi harga terjadi dan kemungkinan akan meluas saat pedagang mengumpulkan lebih banyak informasi tentang kami. Contoh yang paling banyak dikutip adalah Amazon.com, yang memberi harga DVD 30 persen, 35 persen, dan 40 persen dari daftar harga secara bersamaan untuk pelanggan yang berbeda. Seorang pelanggan melaporkan menghapus cookie pelacakan Amazon.com dan membuat harga di situs web turun dari $26,00 menjadi $22,00 karena situs web mengira dia adalah pelanggan baru, bukan pelanggan yang kembali. Rupanya, loyalitas pelanggan kurang berharga daripada menemukan target baru. Studi Turow melibatkan wawancara 1500 orang dewasa AS tentang harga web dan masalah pembelian. Di antara temuan penting adalah sebagai berikut:

- Lima puluh tiga persen dengan benar berpikir bahwa sebagian besar pedagang online tidak memberi mereka hak untuk mengoreksi informasi yang salah yang diperoleh tentang mereka.
- Lima puluh persen benar berpikir bahwa sebagian besar pedagang online tidak memberi mereka kesempatan untuk menghapus informasi yang dikumpulkan tentang mereka.
- Tiga puluh delapan persen dengan benar berpikir bahwa sah bagi pedagang online untuk membebankan harga yang berbeda kepada orang yang berbeda pada waktu yang sama.
- Tiga puluh enam persen dengan benar berpikir bahwa adalah legal bagi supermarket untuk menjual data kebiasaan membeli.
- Tiga puluh dua persen benar berpikir bahwa layanan perjalanan belanja harga seperti Orbitz atau Expedia tidak harus menyajikan harga terendah yang ditemukan sebagai salah satu pilihan perjalanan.
- Dua puluh sembilan persen dengan benar berpikir bahwa toko video tidak dilarang menjual informasi tentang video yang disewa pelanggan.

Baru-baru ini, David Streitfeld [STR14] mengGambarkan pertikaian yang sedang berlangsung antara Amazon dan Hachette, penerbit buku besar. “Di antara taktik Amazon melawan Hachette, beberapa di antaranya telah digunakan selama berbulan-bulan, mengenakan biaya lebih untuk buku-bukunya dan menyarankan agar pembaca menikmati buku dari penulis lain. Jika pelanggan karena alasan tertentu tetap bertahan dan tetap membeli buku Hachette, Amazon mengatakan akan memakan waktu berminggu-minggu untuk mengirimkannya.” Dalam hal ini, Amazon berusaha menekan Hachette untuk memberikan persyaratan yang lebih baik kepada Amazon untuk penjualan buku Hachette di situs Amazon.

Pedagang web tidak berkewajiban untuk memberi harga produk yang sama untuk semua pelanggan, atau sama dengan harga penjual lain untuk produk yang sama.

Pasar yang adil terjadi ketika penjual dan pembeli memiliki pengetahuan yang lengkap: Jika keduanya dapat melihat dan setuju dengan dasar keputusan, masing-masing mengetahui pihak lain bermain secara adil. Internet memiliki beberapa aturan seperti itu, namun. Hilangnya privasi Internet dapat menyebabkan keseimbangan kekuatan pengetahuan bergeser kuat ke sisi pedagang.

5.6 Keamanan Email

Dalam bab ini kita kembali ke email, kali ini menganalisis privasi, dan kekurangannya, dalam korespondensi email. Email biasanya terekspos saat bergerak dari node ke node di sepanjang Internet. Selanjutnya, privasi pesan email dapat dikompromikan di sisi pengirim atau penerima, tanpa peringatan.

Pertimbangkan perbedaan antara email dan surat biasa. Surat biasa ditangani oleh sistem pos berbasis permukaan yang oleh undang-undang (di sebagian besar negara dan dalam kebanyakan situasi) dilarang untuk melihat ke dalam surat. Sebuah surat disegel di dalam amplop buram, sehingga hampir mustahil bagi orang luar untuk melihat isinya. Amplop fisiknya sudah rusak, artinya amplop itu menunjukkan kerusakan jika seseorang membukanya. Pengirim dapat menjatuhkan surat di kotak surat apa pun, membuat pengiriman surat menjadi anonim; tidak ada persyaratan untuk alamat pengirim atau tanda tangan pada surat itu. Karena alasan ini, kami mengharapkan privasi yang tinggi dengan surat permukaan. (Pada waktu tertentu dalam sejarah, misalnya, selama perang atau di bawah penguasa otokratis, surat diperiksa secara teratur. Dalam kasus tersebut, sebagian besar warga tahu bahwa surat mereka tidak bersifat pribadi.)

Tetapi harapan untuk privasi ini berbeda dengan email. Di bagian ini kita melihat realitas privasi untuk email.

5.6.1 Ke Mana Email Pergi, dan Siapa yang Dapat Mengaksesnya?

Di bagian ini, kami hanya melihat mekanisme pengiriman email dengan memperhatikan dampak privasi.

Email secara konseptual adalah komunikasi point-to-point. Jika Janet mengirim email ke Scott, komputer Janet membuat koneksi virtual dengan Scott, komputer disinkronkan, dan pesan ditransfer oleh beberapa protokol yang terdefinisi dengan baik, seperti SMTP (protokol transfer surat sederhana). Namun, Scott mungkin tidak online pada saat Janet ingin mengirim pesannya, sehingga pesan untuk Scott disimpan untuknya di server (disebut POP atau server protokol kantor pos). Lain kali Scott sedang online, dia mengunduh pesan itu dari server. Dalam komunikasi point-to-point, pesan Janet bersifat pribadi; dalam versi server, berpotensi terbuka saat duduk di server.

Janet mungkin merupakan bagian dari organisasi besar (seperti perusahaan atau universitas), jadi dia sendiri mungkin tidak memiliki koneksi keluar langsung; sebagai gantinya, emailnya juga dirutekan melalui server organisasinya, di mana privasi pesan bisa terancam. Misalnya, beberapa organisasi menjelaskan kepada karyawan bahwa semua konten di server mereka tunduk pada pemindaian atau pemeriksaan.

Komplikasi email lebih lanjut adalah penggunaan alias dan agen penerusan, yang menambahkan lebih banyak titik tengah pada deskripsi ini. Juga, perutean Internet dapat membuat banyak lompatan dalam model titik-ke-titik konseptual yang inheren.

Apa yang dimulai sebagai kasus surat sederhana dari Janet ke Scott dapat dengan mudah melibatkan setidaknya enam pihak: (a) Janet dan komputernya, (b) server SMTP organisasi Janet, (c) ISP organisasi Janet, (d) ISP yang terhubung ke Server POP Scott, (e) Server POP Scott, dan (f) Scott dan komputernya. Untuk saat ini, kami paling tertarik pada empat pihak tengah: (b), (c), (d), dan (e). Salah satu dari mereka dapat mencatat fakta bahwa pesan itu dikirim atau bahkan dapat menyimpan salinan pesan.

Penyadapan Email

Email memiliki risiko intersepsi yang sama seperti lalu lintas web lainnya: Saat transit di Internet, email terbuka untuk dibaca oleh pencegat mana pun. Email tunduk pada intersepsi dan modifikasi di banyak titik dari pengirim ke penerima.

Secara khusus, S/MIME dan PGP adalah dua program perlindungan email yang banyak digunakan. S/MIME dan PGP tersedia untuk penangan email populer seperti Outlook, Mail (dari Apple), Thunderbird, dan lainnya. Produk ini melindungi email dari stasiun kerja klien melalui agen email, di seluruh Internet, dan ke stasiun kerja penerima. Perlindungan itu dianggap end-to-end, artinya dari pengirim ke penerima. Perlindungan email terenkripsi tunduk pada kekuatan enkripsi dan keamanan protokol enkripsi.

Jaringan pribadi virtual, dapat melindungi data pada koneksi antara workstation klien dan beberapa titik tepi, biasanya router atau firewall, di organisasi tempat klien berada. Untuk karyawan perusahaan atau pemerintah atau mahasiswa, komunikasi dilindungi hanya sampai ke tepi jaringan perusahaan, pemerintah, atau universitas. Jadi, dengan jaringan pribadi virtual, email hanya dilindungi dari pengirim ke kantor pengirim, tidak sampai ke agen surat pengirim, dan tentu saja tidak ke penerima.

Beberapa organisasi secara rutin menyalin semua email yang dikirim dari komputer mereka. Banyak tujuan salinan ini termasuk menggunakan email sebagai bukti dalam urusan hukum dan memantau email untuk konten yang tidak pantas.

Pemantauan Email

Di banyak negara, perusahaan dan lembaga pemerintah dapat secara sah memantau penggunaan email karyawan mereka. Demikian pula, sekolah dan perpustakaan dapat memantau penggunaan komputer siswa atau pelanggan mereka. Administrator jaringan dan ISP dapat memantau lalu lintas untuk tujuan bisnis normal, seperti mengukur pola lalu lintas atau mendeteksi spam. Organisasi biasanya harus memberi tahu pengguna tentang pemantauan ini, tetapi pemberitahuan tersebut dapat berupa Kasus kecil di buku pegangan personel atau cetakan kecil kontrak layanan. Organisasi dapat menggunakan data pemantauan untuk tujuan hukum apa pun, misalnya, untuk menyelidiki kebocoran, untuk mengelola sumber daya, atau untuk melacak perilaku pengguna.

Pengguna jaringan seharusnya tidak mengharapkan privasi dalam email atau penggunaan komputer secara umum.

Email Anonim, Samaran, dan Menghilang

Kami telah menjelaskan anonimitas di pengaturan lain; ada alasan untuk email anonim juga.

Seperti halnya panggilan telepon, karyawan yang mengirimkan tip atau keluhan kepada manajemen mungkin ingin melakukannya secara anonim. Misalnya, konsumen mungkin ingin menghubungi perusahaan komersial—untuk mengajukan keluhan, menanyakan produk, atau meminta informasi—tanpa masuk ke milis atau menjadi target spam. Atau orang yang memulai hubungan pribadi mungkin ingin menyampaikan beberapa informasi tanpa memberikan identitas atau lokasi lengkap mereka. Untuk alasan ini dan banyak lagi, orang ingin dapat mengirim email anonim.

Alamat email gratis sudah tersedia dari Yahoo!, Microsoft Hotmail, dan banyak tempat lainnya, dan beberapa layanan juga menawarkan alamat sekali pakai. Orang dapat memperlakukan alamat ini sebagai sekali pakai: Dapatkan satu, gunakan untuk sementara, dan buang (dengan berhenti menggunakannya).

Remailer

Solusi lain adalah remailer. Remailer adalah pihak ketiga tepercaya kepada siapa Anda mengirim pesan email dan menunjukkan kepada siapa Anda ingin email Anda dikirim. Remailer menghapus nama dan alamat pengirim, memberikan nama samaran anonim sebagai pengirim, dan meneruskan pesan ke penerima yang ditunjuk. Pihak ketiga menyimpan catatan korespondensi antara nama samaran dan nama asli serta alamat. Jika penerima membalas, pengirim ulang menghapus nama dan alamat penerima, menerapkan nama samaran anonim yang berbeda, dan meneruskan pesan ke pengirim asli. Remailer semacam itu mengetahui pengirim dan penerima, sehingga memberikan nama samaran, bukan anonimitas.

Multiple Remailer

Sebuah desain yang lebih rumit diperlukan untuk mengatasi masalah bahwa remailer mengetahui siapa pengirim dan penerima sebenarnya. Pendekatan dasar melibatkan sekumpulan host yang bekerja sama, kadang-kadang disebut remailer mixmaster, yang setuju untuk meneruskan email. Setiap host menerbitkan kunci enkripsi publiknya sendiri.

Pengirim membuat pesan dan memilih beberapa host yang bekerja sama. Pengirim menunjuk penerima akhir (sebut saja node n) dan menempatkan catatan tujuan dengan konten. Pengirim kemudian memilih salah satu host yang bekerja sama (sebut saja node n-1), mengenkripsi paket dengan kunci publik node (n-1) dan menempatkan catatan tujuan yang menunjukkan node (n) dengan paket terenkripsi. Pengirim memilih node lain (n-2), mengenkripsi, dan menambahkan catatan tujuan untuk (n-1). Pengirim dengan demikian membangun paket berlapis-lapis, dengan pesan di dalamnya; setiap lapisan menambahkan lapisan enkripsi lain dan tujuan lain.

Setiap node remailer hanya mengetahui dari mana paket tersebut diterima dan kepada siapa selanjutnya akan dikirimkan. Hanya remailer pertama yang mengetahui penerima sebenarnya, dan hanya remailer terakhir yang mengetahui penerima akhir. Oleh karena itu, tidak ada remailer yang dapat membahayakan hubungan antara pengirim dan penerima.

Meskipun strategi ini masuk akal, overhead yang terlibat menunjukkan bahwa pendekatan ini harus digunakan hanya ketika anonimitas sangat penting. Konsep umum mengarah ke TOR jaringan yang mempertahankan anonimitas yang dijelaskan dalam Bab 6.

Email yang Menghilang

Beberapa layanan mengklaim melindungi privasi Anda dengan mengaktifkan pesan yang menghilang. Artinya, Anda dapat menggunakan layanan tersebut untuk mengirim file, foto, atau pesan yang dihancurkan oleh layanan segera setelah mencapai tujuannya. Seperti yang kami catat sebelumnya, risikonya cukup besar. Wortham [WOR14] menunjukkan bahwa, “apa yang dibagikan melalui web dan melalui perangkat seluler berisiko untuk dicegat atau diambil kembali, bahkan jika perusahaan perangkat keras dan perangkat lunak yang mengirimkannya berjanji sebaliknya. Kerentanan keamanan telah terungkap di bank-bank besar, perusahaan, dan pengecer di seluruh dunia dan di banyak perusahaan baru.”

Layanan seperti Snapchat berjanji untuk menghapus semua jejak dari apa yang Anda kirim, agar konten Anda tidak mengintip. Namun terkadang klaim tidak sesuai dengan kenyataan. Snapchat menjadi sangat sukses—sangat sukses sehingga menolak tawaran bernilai miliaran dolar untuk dibeli oleh Facebook. Pada tahun 2014, Komisi Perdagangan Federal menuduh Snapchat salah mengartikan bagaimana ia melindungi informasi pengguna.

Dalam tanggung jawabnya, FTC mencatat bahwa Snapchat mengklaim bahwa pesannya, yang sering disebut snaps, tidak dapat disimpan. Namun sebenarnya ada beberapa cara untuk menyimpannya, termasuk menggunakan aplikasi pihak ketiga atau solusi yang melibatkan pengambilan tangkapan layar pesan.

Itu bukan satu-satunya pelanggaran privasi. Snapchat juga “mentransmisikan informasi lokasi pengguna dan mengumpulkan data sensitif seperti kontak buku alamat, meskipun dikatakan bahwa itu tidak mengumpulkan informasi tersebut. Komisi mengatakan kebijakan yang longgar tidak mengamankan fitur yang disebut 'Temukan Teman' yang memungkinkan peneliti keamanan untuk mengkompilasi database 4,6 juta nama pengguna dan nomor telepon selama pelanggaran keamanan baru-baru ini."

Pelajaran di sini jelas: Jika Anda berencana untuk menggunakan layanan atau menggunakan produk untuk melindungi privasi Anda, cari dulu bukti tentang bagaimana perlindungan diberikan dan apakah itu benar-benar berfungsi.

Spoofing dan Spamming

Email memiliki perlindungan keaslian yang sangat sedikit. Tidak ada dalam protokol SMTP yang memeriksa untuk memverifikasi bahwa pengirim yang terdaftar (alamat Dari:) akurat atau bahkan sah. Memalsukan alamat sumber pesan email tidaklah sulit. Keterbatasan ini memudahkan pengiriman spam karena tidak mungkin melacak pengirim pesan spam yang sebenarnya. Terkadang pengirim yang terlihat adalah seseorang yang dikenal penerima atau seseorang di milis yang sama dengan penerima. Memalsukan pengirim yang jelas dimaksudkan untuk memberikan kredibilitas pada pesan spam.

Phishing adalah bentuk spam di mana pengirim berusaha meyakinkan penerima untuk mengungkapkan data pribadi, seperti detail perbankan. Pengirim meningkatkan kredibilitas pesan phishing dengan memalsukan alamat sumber yang meyakinkan atau menggunakan nama domain yang menipu.

Pesan email semacam ini memikat pengguna yang mudah tertipu untuk mengungkapkan data pribadi yang sensitif. Karena terbatasnya regulasi Internet, sangat sedikit yang dapat dilakukan untuk mengendalikan ancaman ini. Kesadaran pengguna adalah pertahanan terbaik.

5.7 Dampak Privasi dari Teknologi yang Muncul

Di bagian ini, kita melihat implikasi privasi dari beberapa teknologi yang muncul. Tidak ada yang melekat dalam teknologi yang memengaruhi privasi, tetapi aplikasinya memiliki risiko. Yang pertama adalah teknologi broadcast yang bisa digunakan untuk melacak objek atau orang. Kedua, kelompok teknologi untuk memfasilitasi pemilu. Teknologi ketiga melibatkan perubahan metode untuk menyediakan panggilan telepon tingkat suara. Dan akhirnya, kami membahas masalah privasi tertentu yang terkait dengan komputasi awan.

5.7.1 Radio Frequency Identification (RFID)

Identifikasi frekuensi radio (RFID) adalah teknologi yang menggunakan pemancar radio nirkabel kecil berdaya rendah yang disebut tag RFID. Perangkat bisa sekecil sebutir pasir dan harganya bisa kurang dari satu sen. Tag disetel ke frekuensi tertentu dan masing-masing memiliki nomor ID unik. Ketika sebuah tag menerima sinyalnya dari produk jarak jauh, ia mengirimkan sinyal nomor ID-nya sebagai tanggapan. Banyak tag tidak memiliki catu daya sendiri dan menerima daya untuk mengirim sinyal dari tindakan menerima sinyal. Dengan demikian, perangkat ini dapat menjadi pasif sampai mereka menerima sinyal dari pembaca yang menginterogasi.

Beberapa tag dapat ditanamkan melalui pembedahan di bawah kulit manusia atau hewan. Lainnya dapat disematkan dalam kartu kredit atau lencana identitas, dan lainnya dapat ditempatkan di label pengiriman atau inventaris.

Jarak di mana mereka dapat menerima dan menyiarkan sinyal yang dapat diterima bervariasi dari kira-kira lima sentimeter (paling kuat) hingga beberapa meter (paling kuat). Beberapa pemancar memiliki catu daya sendiri (biasanya baterai, tetapi dapat berupa kolektor surya atau perangkat terkait lainnya) dan dapat mentransmisikan melalui jarak yang lebih jauh. Saat receiver menjadi lebih baik dan catu daya menjadi lebih porTabel , jarak penerimaan akan meningkat.

Kemajuan teknologi telah memungkinkan token RFID yang lebih kecil dari waktu ke waktu. Sebagai contoh, token RFID tertentu sekarang dapat dibuat dalam sebuah utas.

Penggunaan tag RFID saat ini meliputi:
- kartu tarif sistem transit; juga pemungut tarif jalan tol
- catatan pasien dan pelacakan perangkat medis
- waktu acara olahraga
- akses dan penagihan di fasilitas hiburan

- label stok atau inventaris
- deteksi palsu
- paspor dan kartu identitas; juga token identitas yang ditanamkan secara operasi untuk ternak dan hewan peliharaan

Dua aplikasi tag RFID sangat menarik dari sudut pandang privasi, seperti yang kami tunjukkan di bagian selanjutnya.

Produk Konsumer

Asumsikan Anda telah membeli baju baru. Jika produsen telah menyematkan tag RFID di baju, tag akan membantu pedagang dalam memproses penjualan Anda, seperti yang telah dilakukan barcode selama bertahun-tahun. Tetapi barcode pada barang dagangan hanya mengidentifikasi produk pabrikan, seperti kemeja flanel kotak-kotak hijau L.L Bean, ukuran sedang, sehingga mesin kasir otomatis yang terhubung ke pembaca kode batang dapat membebankan harga yang sesuai untuk produk tersebut. Tag RFID dapat mengidentifikasi tidak hanya produk dan ukuran tetapi juga batch dan pengiriman; yaitu, nilai tag menunjukkan kemeja tertentu. ID unik di kaos membantu pedagang melacak stok, mengetahui bahwa kaos ini berasal dari pengiriman yang telah di display penjualan selama 90 hari. Tag juga memungkinkan produsen menentukan dengan tepat kapan dan di mana ia diproduksi, yang mungkin penting untuk kontrol kualitas jika Anda mengembalikan kemeja karena cacat.

Saat Anda berjalan di jalan, kemeja Anda akan merespons penerima mana pun dalam jangkauan yang memancarkan sinyalnya. Dengan tag berdaya rendah yang menggunakan teknologi saat ini, Anda harus melewati cukup dekat ke penerima untuk mendapatkan sinyal Anda, paling banyak beberapa sentimeter. Beberapa ilmuwan berpikir penerimaan ini akan diperpanjang di masa depan, dan yang lain berpikir teknologi ada saat ini untuk pembaca daya tinggi untuk mengambil sinyal satu meter jauhnya. Jika jaraknya beberapa sentimeter, Anda hampir harus menyentuh penerima agar dapat melacak label di baju Anda; pada satu meter, seseorang dapat memiliki pembaca di tepi trotoar saat Anda berjalan melewatinya.

Baju, sepatu, pena, dompet, kartu kredit, ponsel, pemutar media, dan bungkus permen Anda masing-masing mungkin memiliki tag RFID. Pemilik hewan peliharaan bahkan memiliki tag RFID yang ditempatkan di bawah kulit hewan peliharaan mereka, sehingga hewan yang hilang dapat dipertemukan kembali dengan pemiliknya. Saat Anda membawa banyak tag saat berpindah dari satu lokasi ke lokasi lain, Anda mempermudah pelacakan. Salah satu dari tag ini akan memungkinkan pelacakan diam-diam; yang lain memberikan redundansi. Skenario pelacakan yang dulu hanya ditemukan dalam fiksi ilmiah kini mendekati kenyataan.

Bacaan ini terakumulasi saat Anda menjalankan bisnis Anda. Jika sebuah kota dilengkapi dengan pembaca di setiap sudut jalan, adalah mungkin untuk mengumpulkan profil lengkap dari liku-liku Anda; stempel waktu akan muncul saat Anda berhenti sebentar di antara dua penerima. Dengan demikian, dapat dibayangkan dan mungkin layak untuk mengembangkan sistem yang dapat melacak semua gerakan Anda, menentukan kebiasaan Anda, dan dengan demikian memprediksi kapan Anda mungkin paling rentan terhadap kejahatan.

Masalah privasi lainnya adalah apa yang dikatakan tag ini tentang Anda: Satu tag dari ID karyawan mungkin mengungkapkan untuk siapa Anda bekerja, yang lain dari botol obat mungkin mengungkapkan kondisi medis, dan yang lain dari key fob mahal mungkin menyarankan keuangan Anda. Saat ini Anda dapat menyembunyikan objek secara visual seperti ID karyawan Anda di saku Anda; dengan teknologi RFID Anda mungkin harus lebih berhati-hati untuk memblokir sinyal radio yang tidak terlihat. Tag RFID merespon setiap pembaca cukup dekat untuk mengambil sinyal.

Tag RFID untuk Individu

Menandai kemeja adalah masalah kebetulan. Jika Anda membeli jenis kemeja yang tepat, Anda akan memiliki label yang memungkinkan Anda untuk dipantau. Tetapi jika Anda membeli kemeja yang tidak bertanda, atau menemukan dan memotong labelnya, atau menonaktifkan labelnya, atau memutuskan untuk tidak mengenakan kemeja, Anda tidak dapat dilacak.

Beberapa orang memilih untuk dikenali, terlepas dari apa yang mereka kenakan. Beberapa orang dengan kondisi medis yang tidak biasa telah memiliki tag RFID yang ditanamkan secara permanen di tubuh mereka. Dengan cara ini, bahkan jika seorang pasien dibawa tidak sadarkan diri ke rumah sakit, dokter dapat memindai label, menerima nomor unik orang tersebut, dan mencari catatan medis orang tersebut dengan nomor tersebut. Pendekatan serupa sedang digunakan untuk mengizinkan hewan melintasi perbatasan karantina atau untuk mengidentifikasi hewan secara unik seperti hewan peliharaan atau kuda pacu yang berharga.

Dalam contoh ini, individu secara sukarela mengizinkan tag untuk ditanamkan. Tapi ingat bahwa setelah tag ditanamkan, mereka akan merespons penerima yang sesuai, jadi contoh gangguan privasi kami saat berjalan di jalan masih berlaku.

Pendukung RFID segera menunjukkan bahwa teknologi saat ini tidak mengizinkan membaca tag paling sederhana di kejauhan dan penerima sangat mahal sehingga akan menjadi penghalang untuk membangun jaringan yang mampu melacak setiap gerakan seseorang. Seperti yang kami tunjukkan dalam diskusi kami tentang kriptografi dan ulangi dalam presentasi kami tentang perangkat lunak keamanan, Anda tidak boleh mendasarkan keamanan Anda hanya pada apa yang mungkin secara teknis atau ekonomis saat ini.

Masalah Keamanan dan Privasi

Kami telah menjelaskan dua masalah privasi utama RFID: kemampuan untuk melacak individu ke mana pun mereka pergi dan kemampuan untuk membedakan data sensitif tentang orang. Ada masalah terkait lainnya, termasuk kebenaran dan prediksi. Untuk melihat mengapa kebenaran menjadi masalah, pertimbangkan bagaimana sensor pembacaan dapat tidak berfungsi atau ID pemrosesan perangkat lunak mungkin gagal; kedua kasus dapat menyebabkan kesalahan identitas. Bagaimana Anda menentang tuduhan bahwa Anda tidak berada di suatu tempat ketika penerima menunjukkan bahwa Anda berada? Kemungkinan kegagalan lain adalah pemalsuan tag RFID. Di sini sekali lagi sensor akan mengambil pembacaan tag yang terkait dengan Anda. Satu-satunya cara Anda dapat membuktikan bahwa Anda tidak berada di dekat sensor adalah dengan memiliki alibi yang mendukung di mana Anda sebenarnya berada.

5.7.2 Pemungutan Suara Elektronik

Voting adalah area lain di mana privasi itu penting. Kami ingin suara menjadi pribadi, tetapi pada saat yang sama kami ingin cara untuk menunjukkan bahwa semua suara yang dikumpulkan adalah asli. Dengan kontrol kertas suara yang hati-hati, kami sebagian besar dapat memenuhi kedua persyaratan tersebut, tetapi efisiensi sistem tersebut buruk. Kami ingin menggunakan sistem pemungutan suara terkomputerisasi untuk meningkatkan efisiensi tanpa mengorbankan privasi atau akurasi. Di bagian ini kami mempertimbangkan aspek privasi pemungutan suara terkomputerisasi.

Privasi dan Proses Voting

Membuat dan menghitung surat suara adalah langkah yang paling jelas dalam proses pemilihan; membangun dan memelihara daftar pemilih yang memenuhi syarat, mencatat siapa yang telah memilih (dan mencegah satu orang memilih dua kali), mendukung surat suara yang tidak hadir, membantu pemilih di tempat pemungutan suara yang salah, dan mengirimkan hasil pemilihan ke markas pemilihan adalah langkah penting lainnya. Masing-masing memiliki implikasi privasi yang jelas. Misalnya, dalam beberapa budaya politik, mungkin diinginkan untuk menjaga privasi siapa yang telah memilih (untuk mencegah pembalasan terhadap orang-orang yang tidak memilih kandidat yang kuat). Demikian pula, seperti yang kita ketahui dari studi keamanan lainnya, penting untuk melindungi privasi suara dalam transmisi ke markas pemilihan.

Pada tahun 2005, Badan Ilmu Komputer dan Telekomunikasi AS (CSTB) dari Akademi Sains Nasional [NRC05] merilis studinya tentang pemungutan suara elektronik. Laporan tersebut menimbulkan pertanyaan yang harus dijawab dalam debat menyeluruh tentang pemungutan suara elektronik. Misalnya, CSTB menanyakan bagaimana proses pemungutan suara elektronik akan menjamin privasi individu dalam pendaftaran pemilih dan dalam pemungutan suara individu. Selain itu, studi tersebut menekankan bahwa publik harus memiliki kepercayaan dalam proses; jika tidak, publik tidak akan mempercayai hasilnya.

Teknologi Privacy-Preserving

Masalah privasi kritis dengan pemungutan suara adalah memastikan akuntabilitas selain privasi. Dalam banyak pendekatan, misalnya, mengenkripsi suara dengan kunci publik dewan pemilihan, dapat menjaga kerahasiaan. Kesulitannya adalah memastikan bahwa hanya orang yang diberi wewenang yang dapat memilih (yaitu, bahwa setiap suara yang dihitung adalah penyerahan dari satu pemilih yang diberi wewenang), dan bahwa orang yang diberi wewenang hanya dapat memilih satu kali. Karakteristik terakhir ini juga dapat ditangani dengan mudah, jika saja kita tidak perlu memastikan privasi pilihan individu. Apa pun yang mengaitkan suara yang dapat dihitung dengan individu bernama tertentu menghancurkan privasi pemilih.

Finlandia, Estonia, Belanda, dan beberapa negara lain telah menjalankan proyek percontohan pemilihan elektronik. Di Finlandia, eksperimen berjalan buruk karena pemilih tidak dapat menentukan apakah suara mereka telah diterima (yang tidak). Di Estonia, peninjau pemilu independen mengidentifikasi banyak masalah dengan prosedur dan perangkat lunak yang digunakan, meskipun mereka tidak menuduh adanya penghitungan suara yang salah [HAL14]. Sistem elektronik digunakan di Belanda pada tahun 2004 dan 2006, tetapi kontroversi berkembang, merusak kepercayaan publik dalam proses pemungutan suara. Akibatnya, teknologi tersebut ditinggalkan pada tahun

2008. Dalam kasus-kasus yang terdokumentasi ini, privasi suara individu tidak dipertanyakan; sifat keadilan lainnya dari pemilihan menjadi perhatian.

Situasi-situasi ini menunjukkan bahwa dibutuhkan lebih banyak pekerjaan baik pada teknologi maupun persepsi publik tentang pemungutan suara elektronik. Privasi hanyalah satu area yang membutuhkan perkembangan baru yang dapat dipercaya.

5.7.3 VoIP dan Skype

Aspek privasi telepon tradisional cukup dipahami dengan baik: Perusahaan telepon diatur dalam monopoli yang perlu menjaga kerahasiaan komunikasi klien mereka. Pengecualian terjadi dalam keadaan yang ditentukan secara hukum untuk tujuan penegakan hukum dan dalam keadaan darurat. Selain itu, teknologi ini relatif tahan terhadap penyadapan, dengan eksposur terbesar pada titik akhir.

Telepon seluler dan layanan telepon berbasis Internet telah mengubah situasi itu secara signifikan. Voice over IP (VoIP) adalah protokol untuk transmisi lalu lintas suara melalui Internet. Operator VoIP utama termasuk Skype, Google Talk, dan Vonage. Layanan ini mengubah suara analog Anda menjadi sinyal digital yang dikirim melalui Internet; Anda menggunakan handset telepon atau mikrofon dan speaker yang terhubung ke komputer Anda. Untuk menelepon dari London ke Rio, misalnya, Anda akan memanggil aplikasi VoIP, memberinya nomor telepon di Rio. Kantor lokal di Rio akan menghubungi nomor di Rio dan menambal panggilan itu ke server Internetnya. (Prosesnya bahkan lebih mudah jika kedua titik akhir menggunakan VoIP.)

Keuntungan VoIP adalah biaya: Bagi orang yang sudah memiliki koneksi Internet broadband harga tetap, menambahkan VoIP hanya perlu menutupi biaya koneksi lokal di ujung jarak jauh dan biaya perangkat lunak. Tetapi seperti yang telah kita lihat di aplikasi Internet lainnya, privasi dapat dikorbankan. Bahkan jika lalu lintas suara dienkripsi dengan kuat, sumber dan tujuan panggilan telepon akan sedikit terekspos melalui header paket.

5.7.4 Privasi di Cloud

Komputasi awan menjadi dasar bagi banyak model bisnis, mulai dari menghubungkan produk komputer hingga menyediakan penyimpanan cadangan.

Pada tahun 2009, Robert Gellman membuat laporan untuk Forum Privasi Dunia yang meneliti implikasi privasi dari penggunaan cloud. Dia membahas berbagai cara yang, untuk beberapa informasi dan untuk beberapa pengguna bisnis dan pemerintah, berbagi informasi di cloud paling buruk bisa ilegal, kemungkinan besar terbatas, atau bahkan dapat memengaruhi status atau perlindungan untuk informasi yang dibagikan. Roland Trope dan Claudia Ray memberikan contoh ekstensif dari masalah ini untuk pengacara dan hakim. Mereka menjelaskan bagaimana syarat penggunaan untuk banyak penyedia cloud dapat merusak perlindungan kerahasiaan pengacara-klien yang biasanya ditemukan dalam sistem hukum Amerika. Misalnya, dengan meletakkan beberapa dokumen hukum di cloud, seorang pengacara dapat kehilangan kendali atas kontennya; pada kenyataannya, beberapa penyedia cloud menganggap diri mereka sebagai pemilik konten setelah tiba di cloud.

Gellman menjelaskan bagaimana, bahkan ketika tidak ada undang-undang yang melarang pengguna mengungkapkan informasi ke penyedia cloud, pengungkapan tersebut masih dapat menimbulkan konsekuensi. Misalnya, informasi yang disimpan mungkin memiliki perlindungan privasi yang lebih lemah daripada informasi asli yang ada di tangan pembuatnya. Faktanya, "baik lembaga pemerintah maupun pihak yang berperkara swasta mungkin dapat memperoleh informasi dari pihak ketiga dengan lebih mudah daripada dari pembuat informasi tersebut." Selain itu, karena undang-undang privasi berbeda dari satu negara ke negara lain, lokasi server cloud dapat memengaruhi privasi dan kerahasiaan data pengguna cloud.

Gellman mencantumkan beberapa temuan lain yang penting untuk Anda pertimbangkan sebelum Anda menyimpan informasi sensitif di cloud:

- Lokasi informasi di awan mungkin memiliki pengaruh yang signifikan terhadap privasi dan perlindungan kerahasiaan informasi dan kewajiban privasi dari mereka yang memproses atau menyimpan informasi.
- Informasi di cloud mungkin memiliki lebih dari satu lokasi legal pada saat yang sama, dengan konsekuensi hukum yang berbeda.
- Undang-undang dapat mewajibkan penyedia cloud untuk memeriksa catatan pengguna untuk mencari bukti aktivitas kriminal dan hal-hal lain.
- Ketidakpastian hukum mempersulit penilaian status informasi di awan serta perlindungan privasi dan kerahasiaan yang tersedia bagi pengguna.
- Tanggapan terhadap risiko privasi dan kerahasiaan komputasi awan mencakup kebijakan dan praktik yang lebih baik oleh penyedia awan, perubahan undang-undang, dan lebih banyak kewaspadaan oleh pengguna.

Banyak penyedia cloud menawarkan argumen yang meyakinkan bahwa cloud lebih aman daripada komputasi dan penyimpanan konvensional. Tapi seperti yang disarankan Gellman dan Trope dan Ray, caveat emptor juga berlaku untuk cloud. Sebelum Anda meletakkan apa pun di cloud, baca persyaratan layanan dan kebijakan privasi, mengingat bahwa vendor dapat mengubah perjanjian tersebut kapan saja. Memang, Trope dan Ray menunjukkan bahwa, bahkan jika Anda mengakhiri perjanjian Anda dengan vendor cloud, vendor mungkin tetap dapat menyimpan salinan cadangan Anda, berdasarkan persyaratan layanan.

5.7.5 Kesimpulan tentang Emerging Technology

Teknologi terus muncul dan matang, dan kami hanya memberikan beberapa contoh janji teknologi yang hebat tetapi risiko privasi yang cukup besar. Jika Anda berpikir untuk mengadopsi teknologi semacam itu, pastikan untuk mengevaluasi implikasi privasi dan kemudian ikuti dengan cermat seiring perkembangan teknologi.

Pengalaman kami dengan keamanan telah menunjukkan bahwa jika kami mempertimbangkan keamanan di awal kehidupan sistem, opsi yang lebih luas tersedia untuk keamanan. Hal lain yang telah berulang kali ditunjukkan oleh pengalaman adalah bahwa menambahkan keamanan ke sistem yang hampir lengkap itu sulit, jika bukan tidak mungkin. Untuk kedua alasan tersebut, analisis privasi dan keamanan harus dilakukan seiring dengan perkembangan teknologi dan aplikasi.

Sayangnya, untuk semua EmergingTechnologies, tampaknya ada tekanan finansial untuk membuat perangkat atau layanan terlebih dahulu dan kemudian menangani masalah penggunaan dan privasi nanti. Pendekatan ini adalah cara yang salah untuk merancang sistem apa pun. Sayangnya, orang tampaknya mulai dengan teknologi dan bekerja mundur ke sistem yang akan menggunakannya. Pendekatan pengembangan harus bekerja ke depan (menentukan persyaratan yang diperlukan, termasuk pertimbangan privasi, dan mengembangkan sistem untuk mengimplementasikan persyaratan tersebut dengan andal) untuk membangun fitur dan kontrol desain privasi, dan juga bekerja mundur, untuk menyelidiki apa yang mungkin salah dengan privasi dan kemudian menambahkan fitur desain untuk mencegah penyimpangan ini.

5.8 Kesimpulan

Dalam bab tentang privasi ini kita telah memeriksa bagaimana keamanan, privasi, teknologi, dan informasi berinteraksi. Di satu sisi adalah kemampuan baru yang tersedia hanya karena kekuatan dan kapasitas komputer. Di sisi lain adalah hak asasi manusia dan harapan privasi. Seperti yang telah kami tunjukkan, kedua sisi ini tidak harus bertentangan: Privasi dan teknologi tidak selalu bertentangan.

Langkah pertama dalam menetapkan privasi sama dengan bidang keamanan komputer lainnya: Pertama-tama kita harus menentukan kebijakan privasi yang mendokumentasikan privasi apa yang kita perlukan. Pekerjaan awal oleh komite Ware meletakkan prinsip-prinsip dasar yang sangat penting dari privasi informasi.

Selanjutnya, kami melihat interaksi antara individu, identitas, atribut, dan otentikasi, mirip dengan bagaimana kami mempelajari subjek, objek. Contoh spesifik privasi di email dan web menunjukkan bagaimana privasi itu dan saat ini tidak. ditegakkan dalam penanganan informasi yang terkomputerisasi. Akhirnya, topik yang muncul seperti pemungutan suara terkomputerisasi, telepon Internet, RFID, dan cloud menunjukkan kepada kita bahwa dalam teknologi yang berubah dengan cepat, kita perlu memastikan bahwa kepentingan privasi ditegakkan.

Hak privasi memiliki dimensi politik dan teknologi. Teknologi mungkin merupakan bagian yang lebih mudah: Setelah kami memutuskan secara politis hak privasi apa yang ingin kami pertahankan, kami biasanya dapat menyesuaikan teknologi tersebut. Namun studi kami tentang keamanan telah menunjukkan kepada kami bahwa keamanan—atau privasi—tidak mungkin terjadi kecuali kami menuntutnya, merencanakannya, dan mendesainnya ke dalam produk dan proses kami.

Bahan Evaluasi

1. Anda telah diminta untuk berpartisipasi dalam mengembangkan persyaratan kartu identitas berbasis RFID untuk mahasiswa, fakultas, dan afiliasi di universitas. Pertama, buat daftar lima sampai sepuluh kegunaan kartu yang berbeda. Kedua, dari daftar kegunaan itu, perinci data apa yang dibutuhkan kartu untuk disiarkan ke penerima yang akan menyelesaikan penggunaan tersebut. Ketiga, mengidentifikasi kegunaan yang dapat dibuat dari data tersebut oleh penerima nakal yang diam-diam ditanam di sekitar kampus universitas. Akses nakal mana yang mengancam privasi pribadi? Dalam hal apa? Apa tingkat kerusakannya?
2. Anda telah diminta untuk melakukan latihan serupa untuk organisasi rahasia pemerintah. Buat daftar penggunaan kartu secara terbuka dan terselubung, buat daftar data yang perlu disiarkan, dan identifikasi potensi penyalahgunaan data.
3. Jika Anda menyediakan mesin pemungutan suara elektronik untuk pemilihan, apa yang dapat Anda lakukan untuk melanggar hak privasi individu? Artinya, sarankan beberapa cara yang tidak mudah terlihat Anda dapat mencurangi mesin untuk memungkinkan menentukan setelah pemilihan siapa yang telah memilih kandidat mana.
4. Misalkan sebuah perusahaan telepon menyimpan catatan pada setiap panggilan telepon yang ditanganinya, menunjukkan nomor telepon yang menelepon, nomor telepon yang dipanggil, dan waktu, tanggal, dan durasi panggilan. Apa kegunaan yang mungkin dibuat oleh perusahaan telepon dari catatan-catatan itu? Apa kegunaan yang mungkin dilakukan pemasar komersial? Apa kegunaan yang mungkin dibuat oleh perusahaan telepon saingan? Apa kegunaan yang mungkin dibuat oleh pemerintah? Manakah dari penggunaan tersebut yang melanggar hak privasi individu?
5. Diskusikan aljabar otentikasi dan implikasinya terhadap privasi. Artinya, asumsikan situasi dengan otentikasi dua faktor, dan panggil faktor A dan B. Pertimbangkan empat kasus di mana masing-masing kuat atau lemah. Kesimpulan apa yang akan Anda tarik tentang hasilnya: A lemah dan B lemah; lemah A dan kuat B; kuat A dan lemah B; kuat A dan kuat B? Apakah pesanan itu penting? Apakah penting jika kedua faktor memiliki tipe yang sama (apa yang Anda ketahui, apa yang Anda miliki, siapa diri Anda)? Apa yang terjadi jika Anda menambahkan faktor ketiga, C?
6. Anda lupa kata sandi Anda, jadi Anda mengklik "lupa kata sandi saya" untuk mendapatkan kata sandi baru yang dikirim melalui email. Terkadang situs memberi tahu Anda apa kata sandi Anda; di lain waktu, ia mengirimi Anda kata sandi baru (biasanya sementara). Apa implikasi privasi dari setiap pendekatan?
4. Sampaikan argumen yang mendukung dan menentang memiliki apa yang disebut fungsi penuaan untuk data Internet pribadi. Artinya, beberapa posting mungkin dihapus secara otomatis setelah satu bulan, yang lain setelah satu tahun, yang lain setelah satu dekade. Apakah ini cara yang layak untuk memastikan privasi? Mengapa atau mengapa tidak?
8. Apakah etis bagi sekolah untuk membuat video siswa menggunakan komputer yang disediakan sekolah di luar kelas? Prinsip etika apa yang membenarkan pemantauan seperti itu? Apakah sekolah juga dibenarkan dalam merekam semua perilaku web? Semua penekanan tombol? Dukung argumen Anda dengan prinsip-prinsip etika, bukan hanya pendapat pribadi.
9. Gambarkan situasi di mana sumber informasi lebih sensitif daripada informasi itu sendiri. Jelaskan mengapa jumlah data sensitif mungkin juga sensitif.

10. Sarankan desain untuk filter yang akan membedakan kueri yang mengungkapkan data sensitif tentang penyelidik dari kueri yang tidak mengungkapkan apa pun. Kualitas apa yang mungkin menunjukkan bahwa kueri sensitif?
11. Temukan tiga situs web yang mempublikasikan kebijakan privasi mereka. Bandingkan kebijakan mereka. Mana yang menawarkan privasi paling banyak, dan dalam situasi apa? Yang menawarkan paling sedikit? Bagaimana privasi dapat ditingkatkan di setiap situs? Bagaimana Anda bisa tahu bahwa kebijakan privasi yang disebutkan telah diterapkan secara lengkap dan benar?
12. Apakah perlindungan hukum merupakan tindakan pencegahan yang efektif untuk gangguan privasi? Jelaskan kesulitan atau kemanjuran menggunakan hukum untuk memberikan perlindungan privasi.

Bab 6

Kriptografi dalam Keamanan Komputer

Bab ini akan membahas tentang:

- Kriptologi, kriptanalisis
- Enkripsi simetris: DES, AES, dan RC2, RC4, RC5, dan RC6
- Enkripsi asimetris: RSA
- Intisari pesan: SHA
- Tanda tangan digital: Kriptosistem kurva elips, El Gamal, dan DSA/DSS
- Kriptografi kuantum

Kriptografi merupakan sebuah ilmu yang mempelajari bagaimana cara menjaga agar data atau pesan tetap aman saat dikirimkan, dari pengirim ke penerima tanpa mengalami gangguan dari pihak ketiga. Naskah asli yang diacak memanfaatkan sebuah kunci enkripsi sehingga naskah asli tersebut berubah menjadi naskah yang sulit dibaca (chipertext) oleh pihak yang tidak memiliki kunci dekripsi. Bagi banyak orang, manual pengguna untuk kriptografi sudah cukup. Tetapi bagi orang-orang yang menginginkan atau membutuhkan detail lebih lanjut tentang topik ini, kami memperluas kriptografi dalam bab ini.

Kita mulai dengan pengenalan kriptanalisis. Kriptanalisis adalah sebuah studi mengenai cipher, ciphertext atau cyrptosystems yang bertujuan menemukan kelemahan dalam sistem penyandian, sehingga dimungkinkan untuk memperoleh plaintext dari ciphertext yang ada, tanpa perlu mengetahui kunci ataupun algoritma pembangun ciphertext tersebut. Cara ini disebut dengan memecahkan cipher, ciphertext atau cryptosystem.

Kriptografi mengacu pada informasi dan teknik komunikasi yang berasal dari konsep matematika dan seperangkat perhitungan berbasis aturan yang disebut algoritma. Algoritma dan konsep perhitungan matematika ini nantinya akan digunakan untuk mengubah sebuah pesan dengan cara yang sulit diuraikan.

Kriptografi telah digunakan oleh Julius Caesar sejak zaman Romawi Kuno. Teknik ini dijuluki Caesar cipher untuk mengirim pesan secara rahasia, meskipun teknik yang digunakannya sangat tidak memadai untuk ukuran kini. Casanova menggunakan pengetahuan mengenai kriptografi untuk mengelabui Madame d'Urfe (ia mengatakan kepada Madame d'Urfe bahwa sesosok jin memberi tahu kunci rahasia Madame d'Urfe kepadanya, padahal ia berhasil memecahkan kunci rahasia berdasarkan pengetahuannya mengenai kriptografi), sehingga ia mampu mengontrol kehidupan Madame d'Urfe secara total. (Kromodimoeljo, 2010).

Contoh algoritma seperti ini sering digunakan untuk pembuatan kunci kriptografi, penandatanganan digital, verifikasi untuk melindungi privasi data, penjelajahan situs web di internet, dan komunikasi pribadi seperti transaksi kartu kredit dan email. Kami menyimpulkan bab ini dengan beberapa aplikasi di mana kriptografi tertanam: kode hash dan tanda tangan digital.

Kriptografi adalah topik khusus yang bergantung pada beberapa bidang matematika dan ilmu komputer teoretis, termasuk teori bilangan, aljabar medan hingga, kompleksitas komputasi, dan logika. Setelah membaca ikhtisar ini, Anda perlu mengembangkan latar belakang yang signifikan untuk mempelajari kriptografi secara mendalam. Dan kami sangat memperingatkan Anda agar tidak mempelajari sedikit kriptografi dan menyimpulkan bahwa Anda dapat merancang sistem kriptografi aman Anda sendiri. Bidang kriptografi dipenuhi dengan pendekatan gagal yang dirancang bahkan oleh para ahli, jadi yang tidak ahli disarankan untuk "menyerahkan kendali kepada para profesional." Lihat, misalnya, Studi Kasus 6-1 tentang bahayanya menciptakan kriptografi Anda sendiri.

Studi Kasus 1-2 : Bos Mafia Menggunakan Enkripsi

Ditangkap di Sisilia pada April 2006, kepala keluarga mafia Italia yang terkenal, Bernardo Provenzano, membuat catatan, pizzini dalam dialek Sisilia. Saat ditangkap, ia meninggalkan sekitar 350 lembar uang kertas. Di restoran pizza dia memberikan instruksi kepada para letnannya tentang orang-orang tertentu.

Alih-alih menulis nama seseorang, Provenzano menggunakan variasi sandi Caesar di mana huruf diganti dengan angka: A dengan 4, B dengan 5, ... Z dengan 24 (hanya ada 21 huruf dalam alfabet Italia). Jadi dalam salah satu catatannya string "… Saya bertemu 512151522 191212154 dan kami sepakat bahwa kami akan bertemu setelah liburan …," mengacu pada Binnu Riina, seorang rekan yang ditangkap segera setelah Provenzano. Polisi mendekripsi catatan yang ditemukan sebelum penangkapan Provenzano dan menggunakan petunjuk di dalamnya untuk menemukan bos yang dicari selama 40 tahun.

Semua catatan tampaknya menggunakan enkripsi yang sama, membuatnya sepele untuk didekripsi begitu polisi melihat polanya.
Saran yang mungkin kami buat untuk Sig. Provenzano: gunakan algoritma enkripsi yang kuat, ubah kunci enkripsi dari waktu ke waktu, dan pekerjakan ahli kriptografi.

6.1 Kriptologi

Pada bagian ini kita mempelajari dua hal terkait: menemukan kode dan memecahkannya. Pemain olahraga mempelajari gerakan ofensif dengan mempelajari manuver defensif, dan sebaliknya. Oleh karena itu, kami menyajikan aspek primitif membuat dan memecahkan kode bersama-sama.

Kriptografi menyebabkan timbulnya kriptanalisis, yaitu ilmu pengetahuan dan seni untuk membongkar data acak. Praktisi dari kriptanalisis disebut kriptanalis. Kriptanalis berkembang sejalan dengan kriptografi. Setiap ada algoritma kriptografi baru yang dibuat oleh kriptografer, langsung diikuti oleh adanya upaya percobaan kriptanalisis. Percobaan kriptanalisis ini disebut attack (serangan).

6.1.1 Kriptanalisis

Kriptanalisis adalah tindakan mempelajari algoritma kriptografi, implementasinya, plaintext, ciphertext, dan informasi lain yang tersedia untuk mencoba merusak perlindungan enkripsi.

Kriptanalisis mencoba megembalikan data jelas tanpa akses ke ke kunci kriptografi. Ukuran keberhasilan suatu upaya kriptanalisis adalah sampai sejauh mana keberhasilan diketahuinya data jelas atau kunci kriptografi. Asumsi dasar dari suatu kriptosistem adalah bahwa seorang kriptanalis mengetahui keseluruhan mekanisme enkripsi kecuali kuncinya.

Ciphertext-only attack

Pada jenis serangan ini, kriptanalis mempunyai ciphertext dari beberapa data yang dienkripsikan dengan algoritma kriptografi yang sama. Tujuan kriptanalis adalah mendapatkan plaintext dari ciphertext atau lebih baik lagi menarik kesimpulan megnenai kunci yang digunakan.

Known-plaintext attack

Pada jenis serangan ini, kriptanalis tidak hanya memiliki ciphertext, tetapi juga plaintext dari ciphertext tersebut. Tujuan kriptanalis adalah menarik kesimpulan mengenai kunci yang digunkan untuk mengenkripsi data atau algoritma untuk mendekripsikan ciphertext.

Chosen-plaintext attack

Pada jenis serangan ini, kriptanalis selain megetahui ciphertext dan plaintext, juga dapat memilih plaintext yang diinginkan yang biasanya memiliki lebih banyak informasi tentang kunci. Tujuan kriptanalis adalah menarik kesimpulan mengenai kunci yang diguankan untuk mengenkripsi data.

Adaptive-chosen-plaintext attack

Pada hal ini kriptanalis tidak hanya dapat memilih plaintext yang telah dienkripsi, tapi ia juga dapat memodifikasi pilihannya tersebut berdasarkan hasil enkripsi sebelumnya. Kriptanalis mengetahui blok plaintext yang lebih kecil dan kemudian memilih yang lainnya berdasarkan hasil enkripsi pertama, kedua dan seterusnya.

Seorang cryptanalyst dapat mencoba melakukan salah satu atau semua dari enam hal yang berbeda:

- Memcahkan (dekripsi) satu pesan.
- Mengenali pola dalam pesan terenkripsi, untuk memecahkan pola berikutnya dengan menerapkan algoritme dekripsi langsung.
- Menyimpulkan beberapa makna bahkan tanpa merusak enkripsi, seperti memperhatikan frekuensi komunikasi yang tidak biasa atau menentukan sesuatu berdasarkan apakah komunikasi itu pendek atau panjang.
- Mudah menyimpulkan kunci untuk memecahkan satu pesan, dan mungkin pesan berikutnya.
- Menemukan kelemahan dalam implementasi atau lingkungan penggunaan enkripsi oleh pengirim.
- Menemukan kelemahan umum dalam algoritme enkripsi, tanpa harus mencegat pesan apa pun.

Selain serangan-serangan ini, satu pendekatan lain yang layak disebut, meskipun tidak secara langsung melibatkan enkripsi itu sendiri. Penyerang dapat mencoba memperoleh data sebelum enkripsi atau setelah dekripsi, misalnya, dengan mengetuk jalur komunikasi atau memodifikasi program atau sistem operasi. Meskipun bukan teknik kriptografi, metode ini menegaskan bahwa penyerang harus diharapkan menggunakan segala cara untuk mendapatkan data yang diinginkan.

Kadang-kadang analis cukup beruntung untuk mendapatkan salinan dari algoritma atau mesin enkripsi, situasi yang dijelaskan di Studi Kasus 6-2. Dalam hal ini analis dapat menghasilkan banyak pesan, menjalankannya melalui mesin, dan melihat hasilnya.

Studi Kasus 6-2 : Kesalahan Manusia Menyebabkan Kode Rusak

Kahn menjelaskan sejarah mesin Enigma, alat mekanis yang digunakan oleh Jerman dalam Perang Dunia II untuk mengacak pesan dan mencegah musuh memahaminya. Enigma didasarkan pada roda berputar, atau rotor, yang dihubungkan bersama dan terhubung ke keyboard mesin tik. Ada begitu banyak cara untuk mengenkripsi pesan sehingga bahkan jika 1.000 analis mencoba empat cara berbeda setiap menit, sepanjang hari, setiap hari, tim akan membutuhkan waktu 1,8 miliar tahun untuk menguji semuanya.

Jadi bagaimana Sekutu memecahkan enkripsi? Pertama, mereka memanfaatkan kemungkinan obrolan melalui kabel tentang peristiwa setiap hari. Dengan menebak bahwa Jerman akan mendiskusikan tempat atau masalah tertentu, Sekutu menemukan bagian teks acak yang dapat mereka hubungkan dengan pesan asli, atau teks jelas. Selanjutnya, mereka berkonsentrasi pada pesan Luftwaffe.

Mengandalkan kemungkinan bahwa pemberi sinyal Luftwaffe tidak terlatih dengan baik seperti yang ada di Angkatan Darat atau Angkatan Laut, Sekutu kemudian mengawasi kesalahan yang meningkatkan kemungkinan memahami pesan

terenkripsi. Misalnya, petugas sinyal Luftwaffe sering menggunakan "nama pacar untuk pengaturan kunci atau memulai pesan kedua dengan pengaturan yang sama seperti yang tersisa di akhir yang pertama." Pengetahuan tersebut memungkinkan Sekutu untuk menentukan beberapa rencana Luftwaffe selama Pertempuran Inggris.

Dengan demikian, teknologi canggih dapat dipalsukan ketika protokol kontrol tidak diikuti dengan hati-hati dan lengkap.

Ciphertext dari Any Plaintext

Analis, setelah menyusup ke proses transmisi pengirim, dapat secara menguntungkan menyebabkan pesan dienkripsi dan dikirim sesuka hati. Serangan ini disebut serangan plaintext terpilih. Misalnya, analis dapat memasukkan catatan ke dalam database dan mengamati perubahan statistik setelah penyisipan. Pemrograman linier terkadang memungkinkan analis semacam itu untuk menyimpulkan data dalam database yang harus dirahasiakan. Sebagai alternatif, seorang analis dapat menyadap kabel dalam jaringan dan memperhatikan efek pengiriman pesan tertentu ke pengguna jaringan tertentu. cryptanalyst mungkin orang dalam atau memiliki rekan orang dalam yang dapat menyebabkan transaksi tertentu tercermin dalam ciphertext; misalnya, orang dalam dapat meneruskan pesan yang dihasilkan dari penerimaan pesanan dalam jumlah besar. Serangan plaintext yang dipilih menguntungkan analis. Situasi lain yang diinginkan adalah bagi analis untuk memaksa musuh memasukkan konten tertentu ke dalam aliran ciphertext, situasi yang dijelaskan di Studi Kasus 6-3.

Studi Kasus 6-3 : Makna Tersembunyi Mengubah Arah Perang Dunia II

Pada musim semi 1942, Amerika Serikat berperang melawan Jepang di Pasifik. Kriptanalisis Amerika telah memecahkan beberapa kode angkatan laut Jepang, tetapi mereka tidak memahami pengkodean tambahan yang digunakan Jepang untuk menggambarkan situs tertentu. Sebuah pesan yang dicegat oleh Amerika Serikat mengatakan kepada petugas Sekutu bahwa “AF” akan menjadi target serangan besar. Angkatan Laut AS menduga bahwa serangan itu akan dilakukan di pulau Midway, tetapi itu perlu dipastikan.

Komandan Joseph Rochefort, kepala pusat kriptografi Angkatan Laut AS di Pearl Harbor, menyusun rencana cerdas untuk menggali arti “AF.” Dia mengarahkan kelompok angkatan laut di Midway untuk mengirim pesan, meminta air segar, mengatakan bahwa penyulingan air telah rusak. Segera, Amerika Serikat mencegat pesan Jepang yang menunjukkan bahwa "AF" kekurangan air— memverifikasi bahwa "AF" memang berarti Midway!

Kelemahan Lainnya

Seorang cryptanalyst bekerja melawan manusia, yang bisa tergesa-gesa, malas, ceroboh, naif, atau kurang informasi. Manusia terkadang gagal mengubah kunci kriptografi saat dibutuhkan, menyiarkan kunci kriptografi secara jelas, atau memilih kunci dengan cara yang dapat diprediksi. Artinya, algoritme mungkin kuat dan implementasinya efektif, tetapi orang yang menggunakannya gagal dalam beberapa cara dan membuka ciphertext untuk dideteksi. Orang-orang diketahui membuang materi sensitif secara sembarangan yang dapat memberikan akses mata-mata ke teks biasa dengan mencocokkan teks sandi yang diketahui. Dan manusia terkadang bisa disuap atau dipaksa.

Tidak hanya orang bisa salah, tetapi juga implementasi perangkat keras dan perangkat lunak. Terkadang perangkat keras gagal dengan cara yang dapat diprediksi, seperti ketika kepala pembaca disk kehilangan keselarasan treknya, sehingga data sensitif yang dianggap terhapus masih ada di disk. Di lain waktu, hal-hal yang tampaknya kecil dapat melemahkan pendekatan yang sebenarnya kuat. Misalnya, dalam satu serangan, analis secara akurat mengukur listrik yang digunakan oleh komputer yang melakukan enkripsi dan menyimpulkan kunci dari perbedaan daya yang digunakan untuk menghitung 1 versus 0.

Masalah-masalah ini terpisah dari masalah algoritma itu sendiri, tetapi mereka menawarkan cara-cara yang dapat dilakukan oleh seorang cryptanalyst untuk memecahkan kode. Ingatlah bahwa satu-satunya aturan yang berlaku untuk penyerang adalah tidak ada aturan. Contoh "apa pun yang berfungsi" dalam kriptanalisis dijelaskan di Studi Kasus 6-4.

Studi Kasus 6-4 : Cold Data data yang benar-benar 'dingin'

Alex Halderman dan tim peneliti di Universitas Princeton menyelidiki cara baru untuk mendapatkan kunci kriptografi. Chip memori komputer, modul memori akses acak dinamis (DRAM), kehilangan isinya setelah kehilangan daya. Untuk semua tujuan praktis, nilai data menghilang saat dimatikan.

Kinerja semikonduktor pada suhu rendah bervariasi dari perilaku mereka pada suhu kamar. Karena sebagian besar semikonduktor digunakan dalam rentang suhu yang sempit (–20°C hingga +40°C, misalnya) efeknya tidak penting. Tetapi kriptanalisis adalah pertempuran "tidak ada larangan".

Namun, seperti yang dijelaskan oleh tim Halderman, kehilangan data DRAM tidak langsung terjadi. Data tetap selama beberapa detik dan kehilangannya bertahap, tidak seketika. Mereka menemukan laporan tentang perangkat semikonduktor yang menyimpan datanya selama seminggu bila dipertahankan dalam nitrogen cair, sekitar –200 °C; mereka menjalankan tes mereka sendiri dan menemukan hanya 0,17 persen kehilangan data setelah 60 menit dalam nitrogen cair tanpa daya di luar komputer.

Kebanyakan penyerang tidak berjalan-jalan dengan wadah nitrogen cair. Tim Halderman menggunakan alternatif yang tersedia: toko elektronik menjual kaleng aerosol "udara kalengan", semprotan untuk meniup debu dan kotoran lain dari

elektronik. Terlepas dari namanya, kaleng ini mengandung gas terkompresi, refrigeran fluorohidrokarbon terkompresi, yaitu sekitar -50 ° C dalam bentuk cair, dan cairan dapat bocor jika kaleng dibalik. Tim Halderman menggunakan produk semacam itu dalam eksperimen mereka, menemukan bahwa mereka dapat menyimpan konten memori selama beberapa menit.

Jadi bagaimana chip komputer mengarah ke kriptanalisis? Enkripsi dan dekripsi keduanya membutuhkan kunci. Jika sebuah aplikasi melakukan kriptografi untuk pengguna, ia harus menyimpan kunci di suatu tempat, biasanya di memori agar tersedia untuk dimasukkan ke dalam rutin enkripsi atau dekripsi. Yang sangat membantu adalah bahwa program enkripsi sering kali melakukan pra-perhitungan bagian dari pekerjaan mereka, memperdagangkan sejumlah ruang memori (untuk menyimpan tabel yang telah dihitung sebelumnya) untuk waktu yang dihemat dengan mengurangi pekerjaan untuk melakukan setiap enkripsi atau dekripsi. Pola tabel yang telah dihitung sebelumnya ini membantu tim menemukan tabel tersebut dalam memori, membuatnya lebih mudah ditemukan daripada, katakanlah, nomor rekening bank. Karena tim Halderman dapat menyimpan konten chip memori, mereka memiliki cukup waktu untuk menyalin seluruh kontennya ke media stabil lain sebelum rusak parah. Setelah data berada pada media yang tidak mudah menguap, analis memiliki waktu luang untuk melakukan analisis forensik terperinci pada memori, mencari petunjuk yang menunjukkan di mana kunci disimpan.

Diakui, contoh ini menunjukkan pendekatan yang sangat kreatif (beberapa orang mungkin mengatakan aneh atau bahkan patologis) untuk mendapatkan data sensitif. Namun demikian, ini mencontohkan dua poin yang awalnya kami buat di Bab 1: Pertama, penyerang dapat menggunakan alat atau teknik apa pun; tidak ada konsep "bermain dengan baik" dalam kriptanalisis (atau dalam keamanan). Kedua, motivasi itu penting; penyerang bermotivasi tinggi memiliki insentif untuk mengembangkan dan menggunakan serangan yang sangat canggih yang dapat menghasilkan hasil yang kaya.

Informasi latar belakang ini telah mempersiapkan Anda untuk mempelajari skema enkripsi yang banyak digunakan saat ini. Menggunakan skema ini cukup mudah, meskipun konstruksi rinci dari algoritma bisa sangat kompleks. Saat Anda mempelajari ketiga algoritme, ingatlah kemungkinan bahwa cryptanalyst juga bekerja untuk mengalahkan enkripsi ini.

6.1.2 Kriptografi Primitif

Saat membuat sistem kriptografi, desainer menggunakan primitif kriptografi sebagai blok bangunan paling dasar mereka. Karena itu, primitif kriptografi dirancang untuk melakukan satu tugas yang sangat spesifik dengan cara yang ditentukan secara tepat dan sangat andal.

Karena primitif kriptografi digunakan sebagai blok bangunan, mereka harus sangat andal, yaitu bekerja sesuai dengan spesifikasinya. Misalnya, jika prosedur enkripsi mengklaim hanya dapat dibobol X jumlah operasi komputer, dan rusak dengan jauh lebih sedikit dari X operasi, maka primitif

kriptografi itu telah gagal. Jika primitif kriptografi ditemukan gagal, hampir setiap protokol yang menggunakannya menjadi rentan. Karena membuat rutinitas kriptografi sangat sulit, dan mengujinya membutuhkan waktu lama, pada dasarnya tidak pernah masuk akal (atau aman) untuk merancang primitif kriptografi baru untuk menyesuaikan dengan kebutuhan sistem kriptografi baru.

Kriptografi melibatkan dua teknik dasar: mengganti dan mengacak. Operasi ini dapat diterapkan dalam pola yang kompleks. Dengan komputer, kompleksitas pola hanya dibatasi oleh waktu yang bersedia dicurahkan untuk proses enkripsi dan dekripsi.

Substitusi dan Transposisi

Cipher Substitusi adalah cipher dengan cara mensubstitusi huruf dengan huruf yang lain sesuai dengan yang ditetapkan. Prinsip utama cipher substitusi adalah menukarkan setiap huruf pada plainteks dengan sesuatu. Cipher substitusi termasuk algoritma kriptografi klasik. Idenya adalah menggantikan sebuah atau lebih huruf pada plainteks dengan sebuah atau lebih huruf pada plainteks dengan aturan tertentu. Aturan tersebut bergantung cara proses enkripsi dan dekripsi.

Dua primitif kriptografi adalah substitusi dan transposisi. Dalam substitusi, satu set bit ditukar dengan yang lain. Jika enkripsi bekerja pada huruf alfabet, setiap huruf dapat diganti dengan yang lain, seperti teka-teki kriptogram yang diterbitkan di beberapa surat kabar. Substitusi juga dapat dilakukan pada byte atau blok data dengan ukuran lain. Substitusi melibatkan pencarian tabel sederhana, sehingga dapat dilakukan dengan cepat, dan prosesor kriptografi perangkat keras dapat dioptimalkan dengan tabel substitusi yang dikodekan dalam memori prosesor.

Sandi substitusi adalah jenis metode enkripsi dimana setiap satuan pada teks terang digantikan oleh teks tersandi dengan sistem yang teratur. Metode penyandian substitusi telah dipakai dari zaman dulu (kriptografi klasik) hingga kini (kriptografi modern).

Langkah pertama adalah membuat suatu tabel substitusi. Tabel substitusi dapat dibuat sesuka hati, dengan catatan bahwa penerima pesan memiliki tabel yang sama untuk keperluan decrypt. Bila tabel substitusi dibuat secara acak, akan semakin sulit pemecahanciphertext oleh orang yang tidak berhak. Metode ini dilakukan dengan mengganti setiap huruf dari teks asli dengan huruf lain sebagai huruf sandi yang telah didefinisikan sebelumnya oleh algoritma kunci.

Contoh:
Metode Penyandian Substitusi Sederhana

```
Caesar Cipher
-------------
PLAINTEXT      : BELAJAR BERSAMA
Alphabet       : A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Deret Inversi  : T E D A B C Z Y X W V U F S R Q P O N M L K J I H G
Kunci          : A = T
---------------------
Encrypt        : EBUTWTO EBONTFT
Decrypt        : BELAJAR BERSAMA
```

Pada penyandian substitusi, setiap unit plainteks digantikan dengan satu unit chiperteks. Satu uni disini bisa saja berupa saty huruf, pasangan huruf, atau sekelompoik huruf (lebih dari 3 huruf). Untuk membaca huruf aslinya kembali dari teks asli, cukup dengan membalik prosesnya. Metode ini dengan mengganti setiap huruf dari teks asli dengan huruf lain sebagai huruf sandi yang telah didefinisikan debelumnya oleh algoritma kunci.

Jenis-jenis CipherSubstitusi
1. Cipher Abjad - Tunggal (Monoalphabetic Cipher)
2. Cipher Substitusi Homofonik (Homophonic substitution cipher)
3. Cipherabjad – majemuk (Polyalpabetic substitution cipher)
4. Cipher Substitusi Poligram (Polygram substitution cipher)

CAESAR CIPHER 1
Plain text: MAHASISWA BARU
Nomor: 12 0 7 0 18 8 18 22 0 1 0 17 20
Transformasi(n=3): 3+Pt
15 3 10 3 21 11 21 25 3 4 3 20 23
Chipertext: PDKDVLVZD EDUX

CAESAR CIPHER 2
Tiap huruf alfabet digeser 3 huruf kekanan

Contoh:
Plainteks : KEAMANAN SISTEM KOMPUTER
Cipherteks : NHDPDQDQ VLVWHP NRPSXWHU

CAESAR CIPHER 3
Dalam praktek, cipherteks dikelompokkan kedalam kelompok n - huruf, misalnya kelompok 4 - huruf:

NHDP DQDQ VLVW HPNR PSXW HU
Atau membuang semua spasi:

NHDPDQDQVLVWHPNRPSXWHU
Tujuannya agar kriptanalisis menjadi lebih sulit

CAESAR CIPHER 4
Misalkan $A = 0, B = 1, \ldots, Z = 25$,
maka secara matematis caesar cipher dirumuskan sebagai berikut:
Enkripsi: $ci = E(pi) = (pi + 3) \bmod 26$
Dekripsi: $pi = D(ci) = (ci - 3) \bmod 26$

Jika pergeseran huruf sejauh k, maka:
Enkripsi: $ci = E(pi) = (pi + k) \bmod 26$
Dekripsi: $pi = D(ci) = (ci - k) \bmod 26$
k = kunci rahasia

Untuk 256 karakter ASCII, maka:

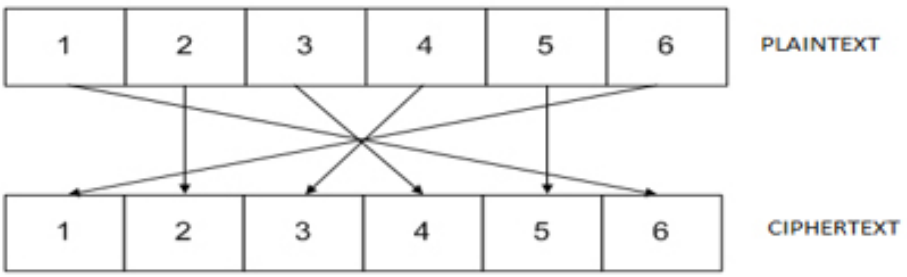
Enkripsi: *ci = E(pi) = (pi + k) mod 256*
Dekripsi: p*i = D(ci) = (ci – k) mod 256*
k = kunci rahasia

Kelemahan substitusi adalah keteraturannya. Dalam substitusi berbasis huruf, jika E selalu diganti dengan p, frekuensi ciphertext p akan sama dengan plaintext E, memberikan petunjuk kepada analis. Untuk alasan ini, substitusi jarang digunakan sendiri. Kadang-kadang tabel substitusi diubah, sehingga, misalnya, E kadang-kadang diganti dengan p dan kadang-kadang oleh w. Penggunaan beberapa penggantian ini membantu memuluskan pola yang terlihat dalam ciphertext keluaran.

Transposisi melibatkan penataan ulang urutan ciphertext untuk memecahkan pola berulang di plaintext yang mendasarinya. Beberapa surat kabar juga menerbitkan teka-teki kata-kata individu, yang huruf-hurufnya telah diacak. Pemecah teka-teki ini mencari pola huruf umum, seperti Q diikuti oleh U, E-D, I-N-G, dan terminal S. Jika transposisi digunakan dengan sendirinya, pola ortografi ini membantu memecahkan teka-teki.

Salah satu teknik enkripsi yang terpenting adalah permutasi atau sering juga disebut transposisi. Teknik ini memindahkan atau merotasikan karakter dengan aturan tertentu. Prinsipnya adalah berlawanan dengan teknik substitusi. Dalam teknik substitusi, karakter berada pada posisi yang tetap tapi identitasnya yang diacak. Pada teknik permutasi, identitas karakternya tetap, namun posisinya yang diacak.

Caranya
Sebelum dilakukan permutasi, umumnya plaintext terlebih dahulu dibagi menjadi blok-blok dengan panjang yang sama.
Plaintext akan dibagi menjadi blok-blok yang terdiri dari 6 karakter, dengan aturan permutasi, sebagai berikut :



Banyak algoritma kriptografi melibatkan substitusi dan transposisi; substitusi memuluskan distribusi keluaran ciphertext dan transposisi memecah pola yang tampak dari unit plaintext berikutnya.

Prinsip Confusion dan Diffusion

Dua konsep penting tambahan terkait dengan jumlah pekerjaan yang diperlukan untuk melakukan enkripsi. Sebuah algoritma enkripsi harus mengambil informasi dari plaintext dan mengubahnya sehingga pencegat tidak dapat dengan mudah mengenali pesan tersebut. Interseptor seharusnya tidak dapat memprediksi apa yang akan terjadi pada ciphertext dengan mengubah satu karakter pada plaintext. Kami menyebutnya ***confusion***. Algoritma yang memberikan kebingungan yang baik memiliki hubungan fungsional yang kompleks antara plaintext/key pair dan ciphertext. Dengan cara ini, sebuah interseptor akan membutuhkan waktu yang lama untuk menentukan hubungan antara plaintext, key, dan ciphertext; oleh karena itu, pencegat akan membutuhkan waktu lama untuk memecahkan kode. Pergantian menghasilkan kebingungan.

Prinsip Confusion menyembunyikan hubungan apapun yang ada antara plainteks, cipherteks, dan kunci. Prinsip Confusion akan membuat kriptanalis frustasi untuk mencari pola-pola statistik yang muncul cipherteks. Confusion yang bagus membuat hubungan statistik antara plainteks, chiperteks, dan kunci menjadi sangat rumit.

Cipher juga harus menyebarkan informasi dari plaintext ke seluruh ciphertext sehingga perubahan pada plaintext mempengaruhi banyak bagian ciphertext. Prinsip ini disebut ***diffusion***, karakteristik pendistribusian informasi dari satu huruf plaintext ke seluruh keluaran. Difusi yang baik berarti bahwa pencegat membutuhkan akses ke sebagian besar ciphertext untuk dapat menyimpulkan algoritma. Prinsip diffusion menyebarkan pengaruh satu bit plainteks atau kunci ke sebanyak mungkin cipherteks. Prinsip Diffusion juga menyembunyikan hubungan statistik antara plainteks, cipherteks, dan kunci dan membuat kriptanalisis menjadi sulit.

One-Time Pad

One Time Pad merupakan salah satu algoritma yang populer dan sering digunakan dalam teknik kriptografi. OTP termasuk kelompok algoritma yang simetris dalam kriptografi dimana kunci enkripsi dan dekripsi dalam bentuk dan panjang yang sama, serta menggunakan operasi XOR (Ignatius et al. 2017). Keunggulan OTP adalah sangat sulit untuk dipecahkan tapi memiliki kekurangan dimana kunci yang digunakan kadang terlalu panjang karena harus menyesuaikan jumlah karakter yang akan dienkripsi (Tornea et al. 2011).Dari semua metode kriptografi yang telah dirancang, OTP adalah metode yang telah terbukti benar-benar aman secara matematis.One Time Pad bisa dikatakan jika memenuhi kondisi berikut (Sari et al. 2016)kunci harus sepanjang plainteks, kunci harus acak seluruhnya atau sepenuhnya berbeda, kunci hanya sekali digunakan setiap melakukan enkripsi, danhanya terdapat dua salinan dari kunci: satu untuk pengirim dan satu untuk penerima(Shukla et al. 2013).

One-Time Pad dianggap sebagai sandi yang sempurna. Ini adalah sandi substitusi murni. Nama tersebut berasal dari metode enkripsi di mana satu set kunci besar yang tidak berulang ditulis pada lembaran kertas, direkatkan menjadi satu pad. Misalnya, jika kuncinya adalah 20 karakter dan pengirim harus mengirimkan pesan sepanjang 300 karakter, pengirim akan merobek 15 halaman kunci berikutnya. Pengirim akan menulis huruf-huruf kunci satu per satu di atas huruf-huruf plaintext dan menyandikan setiap plaintext dengan substitusi yang telah diatur sebelumnya yang melibatkan setiap huruf plaintext dan nilai kunci yang tertulis di atasnya. Pengirim kemudian akan menghancurkan kunci yang digunakan.

Agar enkripsi berfungsi, penerima memerlukan pad yang identik dengan pad pengirim. Setelah menerima pesan, penerima mengambil jumlah kunci yang sesuai dan menguraikan pesan seolah-olah itu adalah substitusi biasa dengan kunci yang panjang. Pada dasarnya, algoritma ini memberikan efek kunci selama jumlah karakter di pad.

Metode one-time pad memiliki dua masalah: kebutuhan akan sinkronisasi mutlak antara pengirim dan penerima, dan kebutuhan akan jumlah kunci yang tidak terbatas. Meskipun menghasilkan sejumlah besar kunci acak tidak masalah, mencetak, mendistribusikan, menyimpan, dan menghitung kunci tersebut adalah masalah.

Random Number (Bilangan Acak)

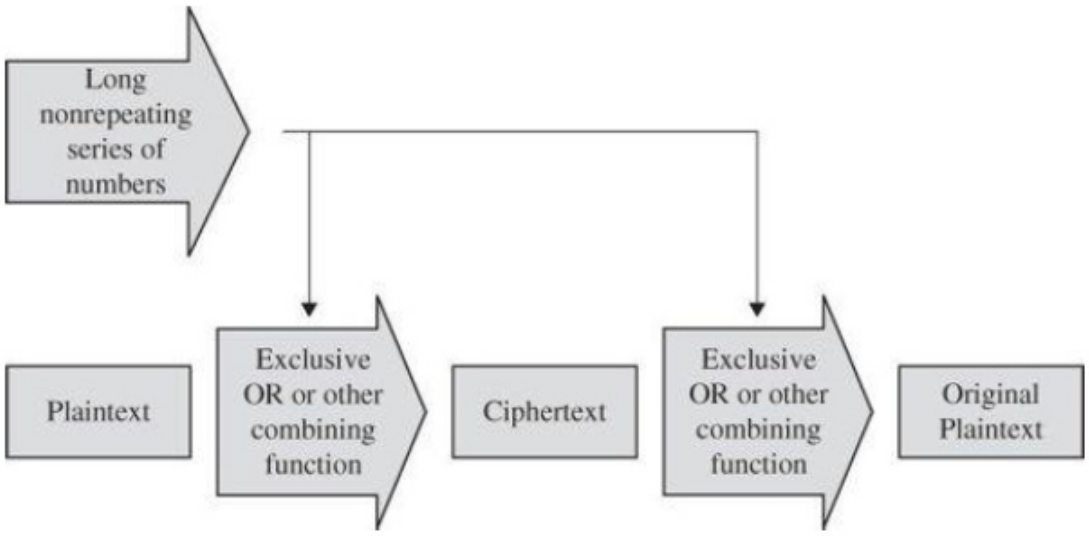
Sebuah pendekatan dekat dari pad satu kali untuk digunakan pada komputer adalah Random Number yang panjang. Faktanya, nomor acak komputer tidak acak; mereka benar-benar membentuk urutan dengan periode yang sangat panjang. Dalam prakteknya, generator dengan periode yang lama dapat diterima untuk waktu yang terbatas atau plaintext.

Telah banyak algoritma generator bilangan acak yang diusulkan dan digunakan hingga saat ini. Algoritma-algoritma tersebut menggunakan berbagai pendekatan berbeda untuk menghasilkan bilangan random seacak mungkin. Hingga saat ini kita tidak dapat membuat suatu generator bilangan acak yang dapat menghasilkan bilangan acak secara murni. Untuk mendapatkan bilangan acak banyak ahli yang mengusulkan berbagai algoritma dari yang sederhana hingga yang paling rumit. Dari penggunaan variabel waktu sebagai faktor pengacak hingga penggunaan teori biologi cellular. Algoritma-algoritma ini masing – masing memiliki kelebihan maupun kekurangan.

Untuk menggunakan generator random number, pengirim dengan pesan 300 karakter akan menginterogasi komputer untuk 300 nomor acak berikutnya dan menggunakan satu bilangan untuk menyandikan setiap karakter pesan teks biasa.

Cipher Vernam

Vernam Cipher ditemukan oleh Gilbert Sandford Vernam (1890-1960) pada tahun 1917. Vernam berkerja di suatu perusahaan bernama Bell Labs, dia memiliki hak paten yang menjelaskan tentang sebuah cipher dengan kunci yang telah disiapkan pada sebuah pita kertas, kemudian dikombinasikan dengan sebuah plaintext untuk menghasilkan sebuah pesan ciphertext yang dapat ditransmisikan. Pita yang sama kemudian digunakan pula untuk mendekripsikan ciphertext. Cipher Vernam kebal dari sebagian besar serangan cryptanalytic. Enkripsi dasar melibatkan urutan angka yang tidak berulang yang dikombinasikan dengan teks biasa. Penemuan Vernam menggunakan pita kertas yang dilubangi dengan panjang sewenang-wenang yang dimasukkan ke dalam mesin teletype. Rekaman itu berisi nomor acak yang digabungkan dengan karakter yang diketik ke dalam teletype. Urutan angka acak tidak berulang, dan setiap pita hanya digunakan sekali. Selama pita kunci tidak berulang atau tidak digunakan kembali, jenis cipher ini kebal dari serangan cryptanalytic karena ciphertext yang tersedia tidak menampilkan pola kunci. Sebuah model dari proses ini ditunjukkan pada Gambar 6-1.



Gambar 6-1 Vernam Cipher

Vernam Cipher merupakan stream cipher yang didefenisikan pada sebuah alfabet. Sebuah pesan *binaryM1, M2, M3, …, Mt* dioperasikan dengan kunci *string binaryK1, K2, K3, …, Kt* yang memilikipanjang yang sama untuk menghasilkan sebuah *string ciphertextC1, C2, C3, …, Ct.*

1. Enkripsi Vernam Cipher

   Secara matematis proses enkripsi plaintext (M) menjadi ciphertext (C) menggunakan kunci rahasia (k) dapat dinyatakan sebagai berikut :

   $C = M + k \pmod n$

   Dimana :

   C = Ciphertext

   M = Message / Plaintext

   k = Key / Kunci

   n = Banyaknya jumlah karakter yang digunakan

2. Dekripsi Vernam Cipher

   Proses dekripsi *ciphertext (C)* menjadi *plaintext (P)* menggunakan kunci rahasia (k) dapat dinyatakan sebagai berikut :

   $M = C - k \pmod n$

   Dimana :

   M = Message / Plaintext

   C = Ciphertext

   k = Key / Kunci

   n = Banyaknya jumlah karakter yang digunakan

   Dimana *Ci* adalah digit binary dari ciphertext, *Pi* adalah digit binary dari plaintext, sedangkan *Ki* adalah bit binary dari kunci, dan + adalah operasi penjumlahan.

Algoritma yang digunakan untuk Vernam Cipher dapat dinyatakan sebagai berikut:

1. Misalkan setiap alfabet plaintext sebagai sebuah urutan nomor yang meningkat, contoh A=0, B=1, …, Z=25.
2. Lakukan hal yang sama untuk setiap karakter input plaintext

3. Tambahkan setiap nomor yang bersangkutan dari alfabet plaintext kepada nomor alfabet kunci yang bersangkutan.
4. Jika hasil yang dikeluarkan melebihi 26, modulo hasilnya dengan 26.
5. Terjemahkan setiap nomor hasil yang didapat tadi kembali kedalam alfabet yang bersangkutan. Hal ini akan memberikan outputciphertext.

Book Ciphers

Sumber lain dari angka yang dianggap "acak" adalah buku, karya musik, atau objek lain yang strukturnya dapat dianalisis. Baik pengirim maupun penerima membutuhkan akses ke objek yang identik. Misalnya, buku catatan satu kali yang mungkin dapat didasarkan pada buku telepon. Pengirim dan penerima mungkin setuju untuk memulai dari halaman 35 dan menggunakan dua digit tengah (ddd- DDdd) dari setiap nomor telepon tujuh digit, mod 26, sebagai huruf kunci untuk sandi pengganti. Mereka menggunakan tabel yang sudah disepakati (tabel Vigenre) yang memiliki 26 huruf.

Book cipher adalah suatu teknik enkripsi dengan menggunakan suatu buku atau dokumen untuk menyembunyikan maknanya. Kode ini menggunakan angka yang menunjukkan urutan dari Buku atau dokumen tersebut. Urutannya dapat kita tentukan sendiri. Dalam dunia modern ini, kita dapat menggunakan banyak hal sebagai kunci untuk memecahkan Kode ini.

Buku apa pun dapat memberikan kunci. Kuncinya dibentuk dari huruf-huruf teks, secara berurutan. Jenis enkripsi ini adalah dasar untuk novel Ken Follett, The Key to Rebecca, di mana film thriller terkenal Daphne du Maurier Rebecca bertindak sebagai sumber kunci untuk mata-mata dalam Perang Dunia II. Jika pengirim dan penerima diketahui menggunakan buku populer, seperti The Key to Rebecca, the bible, atau Security in Computing, akan lebih mudah bagi cryptanalyst untuk mencoba buku yang bertentangan dengan ciphertext, daripada mencari pola dan menggunakan yang canggih. alat. Tentu saja, analis harus menyimpulkan buku yang benar.

Kami ingin menekankan bahwa bantalan satu kali dan bantalan satu kali semu ini tidak dapat diulang. Jika ada pengulangan, pencegat mendapatkan dua aliran ciphertext: satu untuk satu blok plaintext, yang lain untuk plaintext yang berbeda, tetapi keduanya dienkripsi dengan kunci yang sama. Interseptor menggabungkan dua cipherteks sedemikian rupa sehingga tombol-tombolnya saling meniadakan, meninggalkan kombinasi dari dua teks biasa. Interseptor kemudian dapat melakukan analisis lain untuk mengekspos pola di plaintext yang mendasarinya dan memberikan beberapa elemen plaintext yang mungkin. Kasus terburuk adalah bahwa pengguna hanya memulai pad untuk pesan baru, karena pencegat kemudian dapat menentukan cara membagi plainteks dan membuka ritsleting kedua plainteks secara utuh.

6.1.3 Analisis Statistik

Teks dalam bahasa alami apa pun memiliki pola. Pertama adalah frekuensi huruf itu sendiri. Huruf A, E, O, dan T menyumbang sekitar 40 persen dari semua huruf yang digunakan dalam teks bahasa Inggris standar, dan huruf-huruf itu ditambah N dan I mencapai 50 persen. Lebih lanjut, pasangan huruf tertentu, seperti EN, RE, ER, dan NT, dan rangkap tiga, seperti ENT, ION, AND, dan ING, muncul dengan frekuensi tinggi. Bahasa abjad selain bahasa Inggris memiliki hasil yang serupa.

Bahkan plaintext non-bahasa sering mengungkapkan ketidakteraturan dalam distribusi: Bahasa mesin, misalnya, mencerminkan kode operasi instruksi populer, dan kode sumber program mencerminkan kata kunci dari bahasa dan pilihan pemrogram untuk nama variabel.

Bagi cryptanalyst, distribusi frekuensi ini sangat membantu karena mencerminkan ketidakteraturan tertentu dari bahasa dasar yang mungkin ditampilkan dalam ciphertext keluaran.

Apa yang Membuat Algoritma Enkripsi "Aman"?

Algoritma enkripsi berlimpah, termasuk banyak teknik di luar yang kita bahas dalam buku ini. Misalkan Anda memiliki teks untuk dienkripsi. Bagaimana Anda memilih algoritma enkripsi untuk aplikasi tertentu? Untuk menjawab pertanyaan ini, pertimbangkan kembali apa yang telah kita pelajari sejauh ini tentang enkripsi. Kami melihat dua kelas besar algoritma: substitusi dan transposisi. Substitusi "menyembunyikan" huruf-huruf dari plaintext, dan beberapa substitusi menghilangkan frekuensi huruf tinggi untuk mempersulit menentukan bagaimana substitusi dilakukan. Sebaliknya, transposisi mengacak teks sehingga analisis karakter yang berdekatan gagal.

Untuk setiap jenis enkripsi yang kami pertimbangkan, kami menjelaskan kelebihan dan kekurangannya. Tetapi ada pertanyaan yang lebih luas: Apa artinya cipher menjadi "baik"? Arti baik tergantung pada tujuan penggunaan sandi. Sandi yang akan digunakan oleh personel militer di lapangan memiliki persyaratan yang berbeda dari sandi yang digunakan dalam instalasi yang aman dengan dukungan komputer yang substansial. Pada bagian ini, kita melihat lebih dekat pada karakteristik yang berbeda dari cipher.

Karakteristik Sandi "Baik" Menurut Shannon

Pada tahun 1949, Claude Shannon mengusulkan beberapa karakteristik yang mengidentifikasi cipher yang baik.

1. Jumlah kerahasiaan yang dibutuhkan harus menentukan jumlah tenaga kerja yang sesuai untuk enkripsi dan dekripsi.

   Prinsip 1 adalah pengulangan prinsip ketepatan waktu dari Bab 1 dan pengamatan sebelumnya bahwa bahkan sandi sederhana mungkin cukup kuat untuk menghalangi pencegat biasa atau untuk menahan pencegat apa pun untuk waktu yang singkat.

2. Himpunan kunci dan algoritma penyandian harus bebas dari kerumitan.

   Prinsip ini menyiratkan bahwa kita tidak boleh membatasi pilihan kunci maupun jenis teks biasa yang dapat digunakan algoritma. Misalnya, sebuah algoritma yang hanya bekerja pada plaintext yang memiliki jumlah A dan E yang sama tidak berguna. Demikian pula, akan sulit untuk memilih kunci sedemikian rupa sehingga jumlah nilai huruf dari kunci tersebut adalah bilangan prima. Pembatasan seperti ini membuat penggunaan penyandian menjadi sangat rumit. Jika prosesnya terlalu rumit, itu tidak akan digunakan. Selanjutnya, kunci harus ditransmisikan, disimpan, dan diingat, sehingga harus pendek (setidaknya untuk implementasi tangan).

3. Pelaksanaan proses harus sesederhana mungkin.

   Implementasi manual termotivasi prinsip 3: Algoritma yang rumit rentan terhadap kesalahan atau mungkin terlupakan. Dengan perkembangan dan popularitas komputer digital, algoritma yang terlalu rumit untuk implementasi tangan menjadi layak. Namun, masalah kompleksitas itu penting. Orang akan menghindari algoritma enkripsi yang proses implementasinya sangat menghambat transmisi pesan, sehingga merusak keamanan. Dan algoritma yang kompleks lebih mungkin diprogram secara tidak benar.

4. Kesalahan dalam penyandian tidak boleh menyebar dan menyebabkan kerusakan informasi lebih lanjut dalam pesan.

   Prinsip 4 mengakui bahwa manusia membuat kesalahan dalam penggunaan algoritma penyandian. Satu kesalahan di awal proses seharusnya tidak membuang seluruh ciphertext yang tersisa. Misalnya, menjatuhkan satu huruf dalam transposisi membuang seluruh penyandian yang tersisa. Kecuali penerima dapat menentukan di mana surat itu dijatuhkan, sisa pesan tidak akan dapat dipahami. Sebaliknya, membaca baris atau kolom yang salah untuk substitusi berbasis tabel hanya memengaruhi satu karakter—karakter yang tersisa tidak terpengaruh.

5. Ukuran teks yang dienkripsi tidak boleh lebih besar dari teks pesan aslinya.

   Di balik prinsip 5 adalah gagasan bahwa ciphertext yang berkembang secara dramatis dalam ukuran tidak mungkin membawa lebih banyak informasi daripada plaintext, namun memberikan cryptanalyst lebih banyak data untuk menyimpulkan suatu pola. Selain itu, ciphertext yang lebih panjang menyiratkan lebih banyak ruang untuk penyimpanan dan lebih banyak waktu untuk berkomunikasi.

Prinsip-prinsip ini dikembangkan sebelum ketersediaan komputer digital, meskipun Shannon mengetahui komputer dan kekuatan komputasi yang diwakilinya. Dengan demikian, beberapa kekhawatiran yang dia ungkapkan tentang implementasi tangan tidak benar-benar membatasi implementasi berbasis komputer. Misalnya, implementasi cipher di komputer tidak perlu sederhana, asalkan kompleksitas waktu implementasinya dapat ditoleransi. Namun demikian, alasan yang disiratkan oleh prinsip-prinsip ini sebagian besar masih sah.

Properti Sistem Enkripsi "Terpercaya"

Pengguna komersial memiliki beberapa persyaratan yang harus dipenuhi ketika mereka memilih algoritma enkripsi. Jadi, ketika kami mengatakan bahwa enkripsi adalah "tingkat komersial," atau "dapat dipercaya" yang kami maksudkan adalah memenuhi batasan berikut:

- Hal ini didasarkan pada matematika suara. Algoritma kriptografi yang baik tidak hanya ditemukan; mereka berasal dari prinsip-prinsip yang solid.
- Telah dianalisis oleh para ahli yang kompeten dan terbukti benar. Bahkan ahli kriptografi terbaik hanya dapat memikirkan begitu banyak kemungkinan serangan, dan para pengembang mungkin menjadi terlalu yakin akan kekuatan algoritme mereka sendiri. Dengan demikian, tinjauan oleh ahli luar yang kritis sangat penting.

- Itu telah bertahan dalam "ujian waktu". Saat algoritme baru semakin populer, orang-orang terus meninjau fondasi matematisnya dan cara membangunnya di atas fondasi tersebut. Meskipun penggunaan dan analisis yang sukses dalam jangka waktu yang lama bukanlah jaminan dari algoritma yang baik, kekurangan dalam banyak algoritma ditemukan relatif segera setelah dirilis.

Tiga algoritma yang populer di dunia komersial: DES (standar enkripsi data), RSA (Rivest-Shamir-Adelman, dinamai menurut penemunya), dan AES (standar enkripsi lanjutan). Ketiganya (dan juga yang lainnya) memenuhi kriteria kami untuk enkripsi kelas komersial. Pada bagian berikutnya kita membahas DES, AES, dan RSA, tiga algoritma yang diuraikan dalam Bab 2. Dalam bab ini kita mempelajari lebih dalam struktur internal. Kami tidak merinci secara lengkap tentang algoritme ini karena dokumen pendefinisi yang direferensikan melakukannya dengan baik, serta menguraikan pilihan desain dan alasan untuk setiap pembaca yang tertarik.

6.2 Algoritma Enkripsi Simetris

Selama berabad-abad, militer nasional dan dinas diplomatik telah menggunakan kriptografi untuk melindungi rahasia mereka. Faktanya, sejarah kriptografi hampir secara eksklusif melibatkan pemerintah yang melindungi barang-barang dari pemerintah lain.

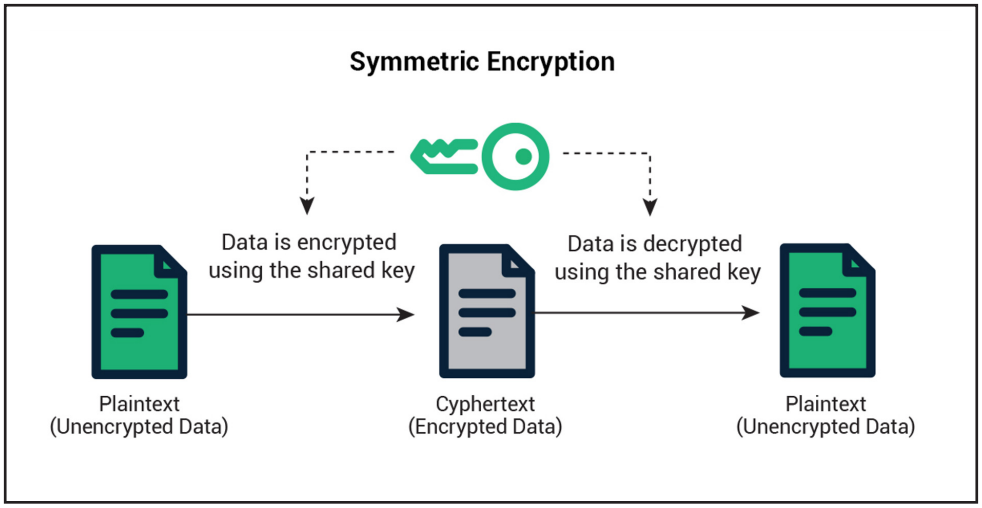
Ketika komputer digital menjadi populer, perusahaan yang menggunakan komputer tersebut merasa perlu untuk melindungi data dari paparan pesaing, serta untuk memastikan privasi pekerja mereka. Dan, ketika jaringan menjadi lebih populer, mereka perlu mencegah masalah dari transmisi yang salah. Jadi, untuk alasan kerahasiaan dan integritas, bisnis dan bahkan beberapa individu mulai mencari enkripsi. Menyadari bahwa kurangnya enkripsi berkualitas tinggi yang andal akan menghambat perdagangan, Departemen Perdagangan A.S., melalui Biro Standar Nasional, mengambil langkah-langkah untuk membuat enkripsi yang solid tersedia untuk industri.

Metode enkripsi simetris menggunakan kunci kriptografi tunggal untuk enkripsi dan dekripsi data. Penggunaan satu kunci untuk kedua operasi membuatnya menjadi proses yang mudah, dan karenanya disebut “simetris.”

Metode enkripsi saat ini sudah tidak menggunakan jenis enkripsi simetris karena dianggap sangat sederhana. Algoritma enkripsi yang digunakan secara luas sangat kompleks sehingga kekuatan komputasi gabungan tidak dapat meretas. Inilah mengapa tidak perlu khawatir untuk melakukan transaksi menggunakan kartu kredit maupun debit secara online.

Fitur enkripsi simetris yang paling menonjol adalah kesederhanaan proses enkripsi dan dekripsi yang menggunakan satu jenis kunci. Hal ini menjadikan teknik enkripsi simetris ini secara signifikan lebih cepat dibandingkan dengan enkripsi asimetris, membutuhkan lebih sedikit daya komputasi serta mengurangi kecepatan internet.

Enkripsi simetris menjadi salah satu enkripsi terbaik ketika sejumlah besar data akan di enkripsi, sehingga tidak membutuhkan proses yang sangat panjang untuk enkripsi dan dekripsi data tersebut. Lihat Gambar 6.2



Gambar 6.2 Enkripsi Symetric

6.2.1 Enkripsi Simetris DES

Diperkenalkan pada tahun 1976, DES (Data Encryption Standard) merupakan salah satu metode enkripsi simetris tertua yang dikembangkan oleh IBM dan difungsikan untuk melindungi data perintahan secara elektronik.

DES menggunakan kunci enkripsi 56-bit, dan itu didasarkan pada Struktur Feistel yang dirancang oleh seorang cryptographer bernama Horst Feistel. Algoritma enkripsi DES termasuk di antara yang termasuk dalam TLS (transport layer security) versi 1.0 dan 1.1.

DES mengubah blok 64-bit dari data plaintext menjadi ciphertext dengan membagi blok menjadi dua blok 32-bit yang terpisah dan menerapkan proses enkripsi untuk masing-masing secara independen. Ini melibatkan 16 putaran berbagai proses – seperti ekspansi, permutasi, penggantian, atau operasi XOR dengan kunci bulat – yang akan dilalui data saat dienkripsi. Pada akhirnya, 64-bit blok teks terenkripsi diproduksi sebagai output.

Saat ini, DES tidak lagi digunakan karena sudah dirusak oleh banyak peneliti keamanan. Pada tahun 2005, DES secara resmi dihentikan dan digantikan oleh algoritma enkripsi AES dan dialihkan juga penggunaan ke TLS 1.2.

Latar Belakang dan Sejarah

Pada awal 1970-an, Biro Standar Nasional AS (National Bureau of Standards / NBS), kemudian berganti nama menjadi the National Institute for Standards and Technology (NIST), mengakui bahwa masyarakat umum membutuhkan teknik enkripsi yang aman untuk melindungi informasi sensitif. Secara historis, Departemen Pertahanan AS dan Departemen Luar Negeri terus tertarik pada sistem enkripsi untuk melindungi rahasia militer dan diplomatik; dianggap bahwa departemen ini adalah rumah bagi keahlian terbesar dalam kriptologi. Namun, justru karena sifat sensitif dari informasi yang mereka enkripsi, departemen tidak dapat merilis pekerjaan mereka. Dengan demikian, tanggung jawab untuk teknik enkripsi yang lebih umum didelegasikan kepada NBS, sebuah badan dari Departemen Perdagangan AS.

Pada saat yang sama, beberapa vendor swasta telah mengembangkan perangkat enkripsi mekanis atau perangkat lunak yang dapat dibeli oleh individu atau perusahaan untuk melindungi komunikasi sensitif mereka. Kesulitan dengan proliferasi komersial teknik enkripsi ini adalah pertukaran: Dua pengguna dengan perangkat yang berbeda tidak dapat bertukar informasi terenkripsi. Selain itu, tidak ada badan independen yang mampu menguji perangkat secara ekstensif untuk memverifikasi bahwa mereka menerapkan algoritme dengan benar, atau bahkan algoritme tersebut layak digunakan.

Segera menjadi jelas bahwa enkripsi sudah sempurna untuk penilaian dan standarisasi, untuk mempromosikan kemampuan pihak yang tidak terkait untuk bertukar informasi terenkripsi dan untuk menyediakan sistem enkripsi tunggal yang dapat diuji secara ketat dan disertifikasi secara publik. Selanjutnya, pada tahun 1972 NBS meminta proposal untuk memproduksi algoritma enkripsi publik. Panggilan tersebut menentukan kriteria yang diinginkan untuk algoritme seperti itu:

- mampu memberikan tingkat keamanan yang tinggi
- ditentukan dan mudah dimengerti
- dapat dipublikasikan, sehingga keamanan tidak bergantung pada kerahasiaan algoritme
- tersedia untuk semua pengguna
- mudah beradaptasi untuk digunakan dalam beragam aplikasi
- ekonomis untuk diterapkan pada perangkat elektronik
- efisien untuk digunakan
- dapat divalidasi
- dapat diekspor

NBS membayangkan menyediakan enkripsi sebagai perangkat keras terpisah. Untuk memungkinkan algoritme publik, NBS berharap untuk mengungkapkan algoritme itu sendiri, mendasarkan keamanan sistem pada kunci (yang akan berada di bawah kendali pengguna).

Beberapa organisasi menanggapi panggilan tersebut, sehingga NBS mengeluarkan pengumuman kedua pada Agustus 1974. Pengajuan yang paling menjanjikan adalah algoritma Lucifer yang telah digunakan IBM selama beberapa tahun. Ide ini telah diterbitkan sebelumnya, sehingga algoritma dasarnya sudah terbuka untuk umum dan telah terbuka untuk pemeriksaan dan validasi. Meskipun panjang, algoritmenya mudah, kandidat alami untuk implementasi berulang dalam program komputer. Lebih jauh lagi, tidak seperti beberapa algoritma yang menggunakan aritmatika pada bilangan biner 500 atau 1.000 digit atau lebih panjang (jauh lebih besar daripada yang dapat ditangani oleh kebanyakan instruksi mesin sebagai kuantitas tunggal), Lucifer hanya menggunakan operasi logika sederhana pada blok data yang relatif kecil. Dengan demikian, algoritma dapat diimplementasikan dengan cukup efisien baik dalam perangkat keras maupun perangkat lunak pada komputer konvensional.

Algoritma enkripsi data yang dikembangkan oleh IBM untuk NBS didasarkan pada Lucifer, dan kemudian dikenal sebagai Data Encryption Standard (DES), meskipun nama aslinya adalah DEA (Data Encryption Algorithm) di Amerika Serikat dan DEA1 (Data Encryption Algorithm-1) di negara Lain. Kemudian, NBS meminta Departemen Pertahanan melalui Badan Keamanan Nasional (NSA) untuk menganalisis kekuatan algoritma enkripsi, dan IBM mengubahnya sedikit. Akhirnya, NBS merilis algoritme untuk pengawasan dan diskusi publik.

DES secara resmi diadopsi sebagai standar federal AS [NBS77] pada November 1976, disahkan oleh NBS untuk digunakan pada semua komunikasi tidak terklasifikasi sektor publik dan swasta. Akhirnya, DES diterima sebagai standar internasional oleh Organisasi Standar Internasional.

Algoritma DES

Algoritme memanfaatkan dua teknik yang diidentifikasi Shannon untuk menyembunyikan informasi: kebingungan dan difusi. Artinya, algoritma menyelesaikan dua hal: memastikan bahwa bit output tidak memiliki hubungan yang jelas dengan bit input dan menyebarkan efek dari satu bit plaintext ke bit lain dalam ciphertext. Substitusi membingungkan, transposisi menyebar. Secara umum, plaintext dipengaruhi oleh serangkaian siklus substitusi diikuti oleh permutasi.

Kami tidak merinci algoritma DES lengkap di sini karena sepenuhnya didokumentasikan dalam dokumen definisi asli. Namun, kami ingin Anda melihat struktur dasar algoritme sehingga Anda dapat menghargai asal mula kekuatan kriptografinya.

Dasar dari DES adalah dua cipher yang berbeda, diterapkan secara bergantian. Shannon mencatat bahwa dua cipher yang lemah tetapi saling melengkapi dapat dibuat lebih aman dengan diterapkan bersama-sama (disebut "produk" dari dua cipher) secara bergantian, dalam struktur yang disebut product cipher.

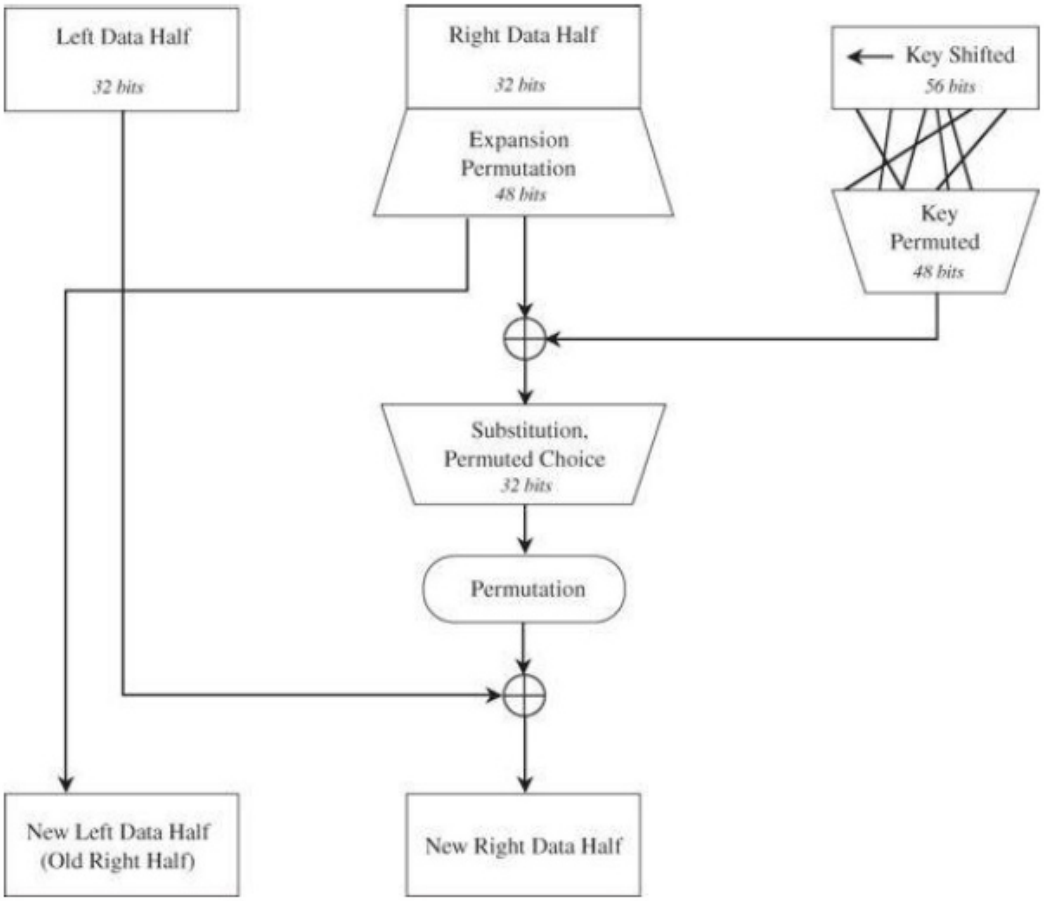
Setelah inisialisasi, algoritma DES beroperasi pada blok data. Ini membagi blok data menjadi dua, mengacak masing-masing setengah secara independen, menggabungkan kunci dengan satu setengah, dan menukar dua bagian. Proses ini diulang sebanyak 16 kali. Ini adalah algoritma iteratif yang hanya menggunakan pencarian tabel dan operasi bit sederhana. Meskipun manipulasi tingkat bit dari algoritme itu rumit, algoritme itu sendiri dapat diimplementasikan dengan cukup efisien. Manipulasi data menggunakan string bit mulai dari 32 hingga 64 bit, hanya menggunakan pencarian tabel, operasi logika (AND, OR, exclusive OR, (XOR)), dan bit shift dan rotasi, membuat prosedur ini ideal untuk diimplementasikan pada komputer dengan 32- atau ukuran kata 64-bit.

Input ke DES dibagi menjadi blok 64 bit. 64 bit data diubah oleh apa yang disebut permutasi awal. Bit data diubah oleh kunci 64-bit (yang hanya digunakan 56 bit). Kuncinya dikurangi dari 64 bit menjadi 56 bit dengan menjatuhkan bit 8, 16, 24, … 64 (di mana bit yang paling signifikan diberi nama bit “1”). Bit yang diabaikan diasumsikan sebagai bit paritas yang tidak membawa informasi dalam kunci.

Selanjutnya dimulai urutan operasi yang dikenal sebagai siklus. 64 bit data permutasi dipecah menjadi setengah kiri dan setengah kanan masing-masing 32 bit. Agar setengah kanan 32-bit digabungkan dengan kunci 64-bit, diperlukan dua perubahan. Pertama, algoritma memperluas setengah 32-bit menjadi 48 bit dengan mengulang bit-bit tertentu, sambil mengurangi kunci 56-bit menjadi 48 bit dengan memilih hanya bit-bit tertentu menurut tabel yang disebut S-box. Dua operasi terakhir ini disebut permutasi ekspansi dan pilihan permutasi.

Kunci digeser ke kiri oleh sejumlah bit dan juga permutasi. Kuncinya digabungkan dengan separuh kanan, yang kemudian digabungkan dengan separuh kiri. Hasil dari kombinasi ini menjadi bagian kanan yang baru; bagian kanan yang lama menjadi bagian kiri yang baru. Urutan kegiatan ini, yang

merupakan siklus, ditunjukkan pada Gambar 6-3. Siklus diulang sebanyak 16 kali. Setelah siklus terakhir adalah permutasi akhir, yang merupakan kebalikan dari permutasi awal.

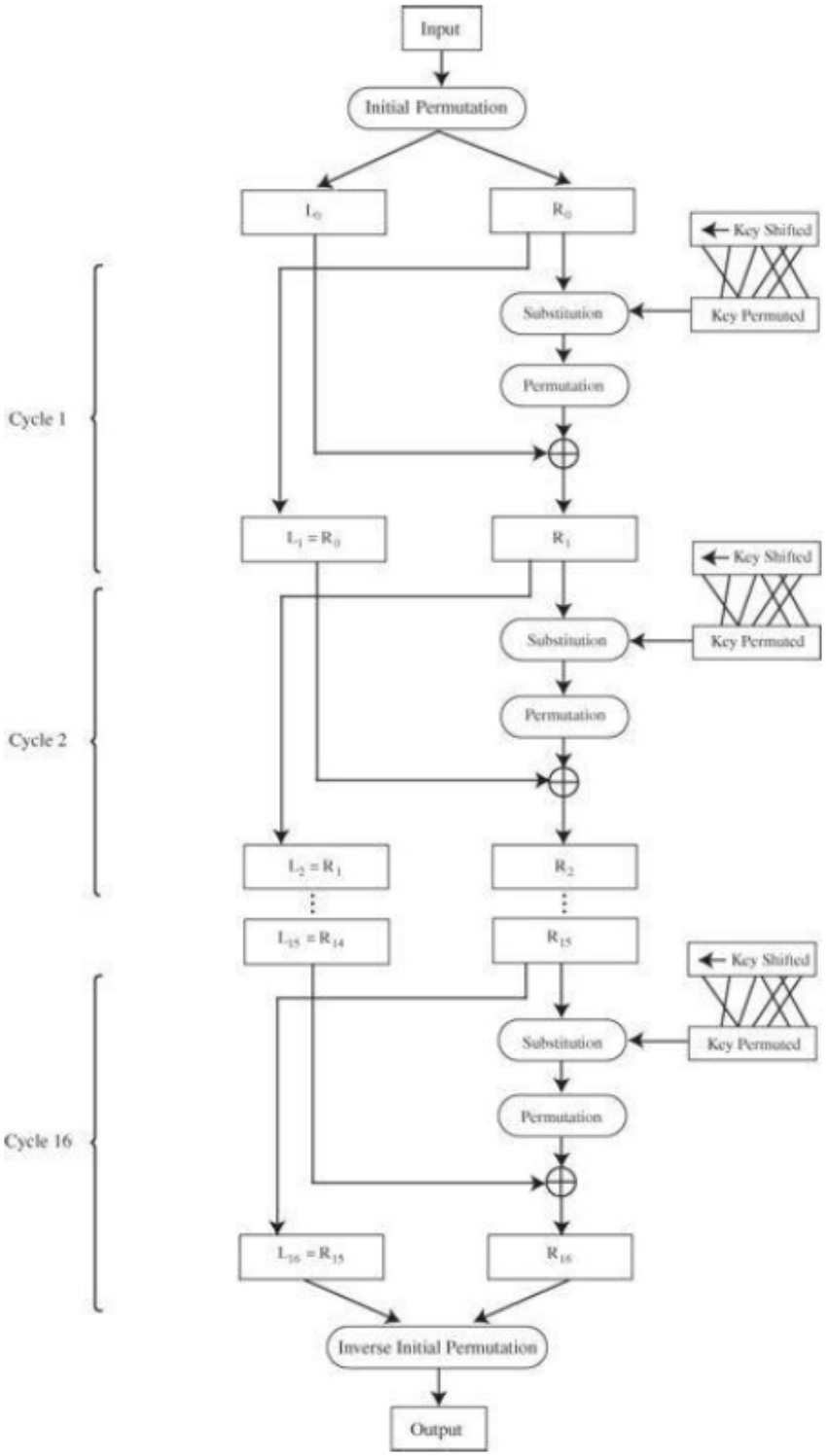


Gambar 6-3 Siklus dalam DES

Siklus ini diulang 16 kali untuk satu blok data 64-bit. Proses yang sama ini diulang secara terpisah untuk setiap blok data plaintext.

Skema DES Lengkap

Sekarang kita dapat menyatukan kembali semua bagian, seperti yang ditunjukkan pada Gambar 6-3. Pertama, kuncinya dikurangi menjadi 56 bit. Kemudian, sebuah blok 64 bit data diubah oleh permutasi awal. Berikut adalah 16 siklus di mana kunci digeser dan di permutasi, setengah dari blok data ditransformasikan dengan fungsi substitusi dan permutasi, dan hasilnya digabungkan dengan setengah sisanya dari blok data. Setelah siklus terakhir, blok data diubah dengan permutasi terakhir.



Gambar 6-3 DES Lengkap

Dekripsi DES

Algoritma DES yang sama digunakan baik untuk enkripsi maupun dekripsi. Hasil ini benar karena siklus j diturunkan dari siklus (j-1) dengan cara sebagai berikut:

$$L_j = R_{j-1} \quad (1)$$

$$R_j = L_{j-1} \oplus f(R_{j-1}, k_j) \quad (2)$$

di mana adalah operasi OR eksklusif dan f adalah fungsi yang dihitung dalam siklus perluasan-pergeseran-pengganti-permutasi. Kedua persamaan ini menunjukkan bahwa hasil setiap siklus hanya bergantung pada siklus sebelumnya.

Dengan menulis ulang persamaan ini dalam bentuk Rj-1 dan Lj-1, kita mendapatkan

$$R_{j-1} = L_j \quad (3)$$

dan

$$L_{j-1} = R_j \oplus f(R_{j-1}, k_j) \quad (4)$$

Mengganti (3) menjadi (4) menghasilkan

$$L_{j-1} = R_j \oplus f(L_j, k_j) \quad (5)$$

Persamaan (3) dan (5) menunjukkan bahwa nilai yang sama dapat diperoleh dari hasil siklus selanjutnya. Properti ini membuat DES prosedur reversibel; kita dapat mengenkripsi string dan juga mendekripsi hasilnya untuk mendapatkan plaintext lagi.

Dengan DES, fungsi yang sama f digunakan maju untuk mengenkripsi atau mundur untuk mendekripsi. Satu-satunya perubahan adalah bahwa kunci harus diambil dalam urutan terbalik (k16, k15, …, k1) untuk dekripsi. Menggunakan satu algoritma baik untuk mengenkripsi atau mendekripsi nyaman untuk implementasi perangkat keras atau perangkat lunak DES.

Chaining

Anda mungkin telah melihat kelemahan dalam deskripsi bagian sebelumnya. DES menggunakan proses yang sama untuk setiap blok 64-bit. Itu berarti bahwa setiap blok data identik yang dienkripsi dengan kunci yang sama akan memiliki output yang sama. Tentu saja, Anda mungkin mengatakan, prosesnya harus teratur dan konsisten agar dekripsi dapat dilakukan, dan itu memang benar.

Sekarang berpikirlah seperti penyerang. Sebuah bank menggunakan DES untuk mengenkripsi aliran data jaringan yang berisi instruksi untuk mentransfer uang dari satu akun ke akun lainnya. Bank memilih kunci yang akan digunakan untuk jangka waktu tertentu, misalnya sehari, karena mengganti kunci tidak praktis. Untuk menyederhanakan argumen ini, asumsikan jumlah dan nomor rekening sama-sama panjangnya 64 bit dan kebetulan muncul pada batas 64-bit dalam pesan transfer. (Argumen serupa akan bekerja terlepas dari panjang dan posisi bidang.) Pada Gambar 6-4 kami menggambarkan bentuk umum dari pesan-pesan ini. Gambar tersebut juga menunjukkan empat contoh, untuk mentransfer (1) $100 dari Annie ke Brian, (2) $500 dari Carole ke Drew, (3) $0,01

dari Evin ke agen jahat kami Zelda, dan (4) jumlah yang sama dari Feng ke Zelda . (Catatan: nilai terenkripsi dalam empat contoh gambar ini adalah fiktif, dibuat hanya untuk penjelasan ini; tidak ada kriptografi nyata yang diterapkan.)

	64 bits				
	Date	From acct	To acct	Trf Num	Amount
	1 Aug	Annie	Brian	0001	100.00
ciphertext	apqrwx	w2z%pr	grd#d#	wenh55	3dhop3
	1 Aug	Carole	Drew	0002	500.00
	apqrwx	df7ynm	gyl615	23opdw	kslw4l
	1 Aug	Evin	Zelda	0003	0.01
	apqrwx	bze4n4	cd4wx7	wenh55	otm4m5
	1 Aug	Feng	Zelda	0004	0.01
	apqrwx	br5hun	cd4wx7	ztpztp	otm4m5

Gambar 6-4 Transfer Pesan

Karena ini adalah komunikasi jaringan, Zelda dapat melihatnya dalam bentuk ciphertext. Meskipun dia tidak bisa menafsirkan isinya, dia bisa mencari kesamaan. Mari kita asumsikan dia tahu di mana mencari bidang yang berbeda. Dia akan melihat dua pesan dengan nomor akun tujuan terenkripsi yang sama dan menyimpulkan bahwa itu mungkin dua transfer ke akunnya. Dengan pengetahuan itu, dia dapat membuat pesan baru, seperti yang ditunjukkan pada Gambar 6-5, untuk mentransfer uang dari Annie dan Carole ke rekeningnya.

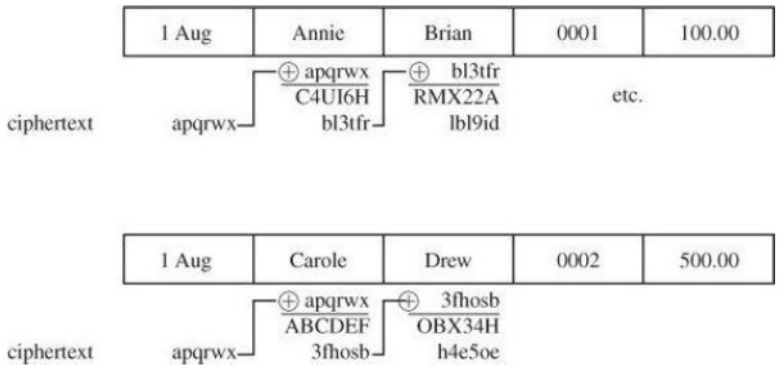
1 Aug	Annie	Zelda	0001	100.00
apqrwx	w2z%pr	cd4wx7	wenh55	3dhop3
1 Aug	Carole	Zelda	0002	500.00
apqrwx	df7ynm	cd4wx7	ztpztp	kslw4l

Gambar 6-5 Pesan Transfer Fabrikasi

Tanpa pernah mengakses plaintext yang mendasarinya, Zelda telah membuat dua pesan terenkripsi yang mungkin lewat. (Dia masih memiliki sedikit pekerjaan yang harus dilakukan untuk menutupi penggunaan kembali nomor transaksi sebelumnya.) Masalah yang dijelaskan di sini terjadi karena setiap blok plaintext dienkripsi secara independen dari semua blok lainnya, sehingga penyerang dapat menyusun ulang dan mengganti blok tanpa terdeteksi.

Solusi untuk masalah ini disebut chaining. Jika enkripsi setiap blok tidak hanya bergantung pada ciphertext langsung dari konten yang berbeda tetapi juga pada konten sebelumnya, urutan dan posisi setiap blok dalam rantai tetap. Perubahan itu akan mencegah Zelda menggunakan kembali blok dari enkripsi sebelumnya.

Pada Gambar 6-6 Anda dapat melihat efek chaining. Pada gambar sebelumnya, nomor akun Annie dienkripsi dua kali sebagai w2z%pr. Namun dalam gambar ini nomor rekening Annie pertama kali digabungkan dengan enkripsi blok tanggal menggunakan fungsi OR (⊕) eksklusif, menghasilkan hasil C4UI6H. C4UI6H itu kemudian dienkripsi, menghasilkan hasil bl3tfr.



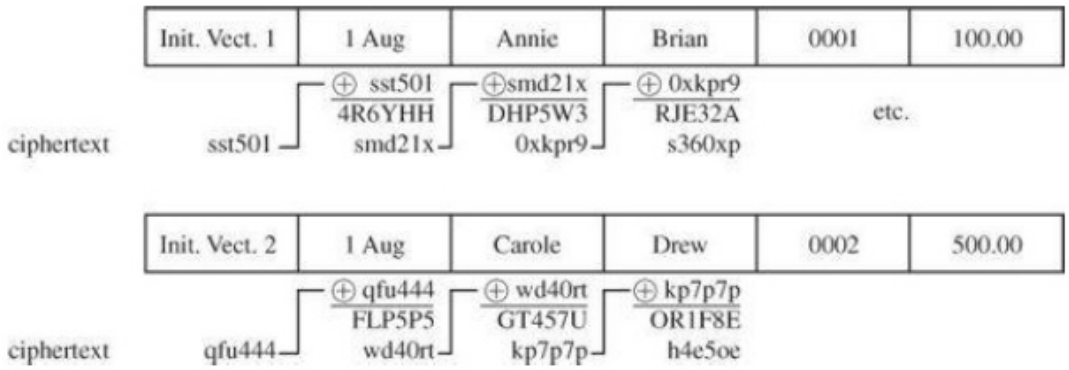
Gambar 6-6 Chained Encryption

Membandingkan bagian atas dan bawah gambar, Anda dapat melihat bahwa bidang pertama, tanggal, mengenkripsi sama di kedua bagian, tetapi bidang berikutnya berbeda, sehingga semua blok berikutnya dipengaruhi oleh perbedaan ini. Nilai terenkripsi dari setiap blok dimasukkan ke dalam plaintext dari blok berikutnya, sehingga setiap blok terenkripsi bergantung pada nilai dari semua blok sebelumnya. Dengan demikian, tidak ada penyerang yang dapat menyusun ulang atau mengganti blok terenkripsi tanpa merusak rantai enkripsi ini. Jika kami telah menunjukkan hasil untuk dua baris terakhir dari gambar asli, Anda akan melihat bahwa kedua blok Zelda akan berbeda, karena masing-masing bergantung pada data sebelumnya yang berbeda.

Initialization Vector

Anda mungkin telah mendeteksi satu kesulitan terakhir dengan contoh rantai. Meskipun blok data di sepanjang rantai berbeda, blok pertama (tanggal) tidak memiliki konteks sebelumnya untuk bergantung, dan dengan demikian akan selalu mengenkripsi sama. Solusi untuk kesulitan ini adalah sedikit perluasan dari pendekatan chaining.

Untuk memulai enkripsi aliran data, Anda terlebih dahulu membuat satu blok tambahan yang berisi nilai apa pun. Nilai pastinya tidak signifikan selama itu berubah untuk setiap enkripsi; generator nomor acak berguna. Kemudian, seperti yang ditunjukkan pada Gambar 6-7 Anda menerapkan enkripsi berantai, dimulai dengan blok pertama (nomor acak), dan menghubungkan blok berikutnya bersama-sama seperti pada contoh sebelumnya. Dengan cara ini, blok data nyata pertama (sebenarnya blok kedua yang dienkripsi) berbeda dari satu waktu ke waktu berikutnya karena vektor inisialisasi acak berbeda.



Gambar 6-7 Enkripsi Berrantai dengan Initialization Vector

Kedua teknik chaining ini berlaku sama dengan sebagian besar cipher simetris.

6.2.1 Pertanyaan Tentang Keamanan DES

Sejak pertama kali diumumkan, ada kontroversi mengenai keamanan yang diberikan oleh DES. Meskipun banyak kontroversi ini telah muncul dalam literatur terbuka, fitur tertentu dari DES belum diungkapkan oleh para desainer atau disimpulkan oleh analis luar.

Desain Algoritma

Awalnya, ada kekhawatiran dengan algoritma dasar itu sendiri. Selama pengembangan algoritme, Badan Keamanan Nasional (NSA) menunjukkan bahwa elemen kunci dari desain algoritme "sensitif" dan tidak akan dipublikasikan. Elemen-elemen ini termasuk alasan di balik transformasi oleh S-box, P-box, dan perubahan kunci. Ada banyak kemungkinan untuk substitusi S-box, tetapi satu set tertentu dipilih untuk DES.

Dua masalah muncul tentang kerahasiaan desain. Yang pertama melibatkan ketakutan bahwa "pintu jebakan" tertentu telah tertanam dalam algoritma DES sehingga cara rahasia dan mudah tersedia untuk mendekripsi pesan terenkripsi DES. Misalnya, pintu jebakan semacam itu akan memberi NSA kemampuan untuk memeriksa komunikasi pribadi.

Setelah penyelidikan Kongres, yang hasilnya diklasifikasikan, ringkasan yang tidak terklasifikasi membebaskan NSA dari keterlibatan yang tidak semestinya dalam desain DES.

Isu kedua membahas kemungkinan bahwa cacat desain akan (atau mungkin telah) ditemukan oleh seorang cryptanalyst, kali ini memberikan pencegat kemampuan untuk mengakses komunikasi pribadi.

Baik Bell Laboratories dan Lexan Corporation meneliti operasi (bukan desain) dari S-box. Tidak ada analisis yang mengungkapkan kelemahan apa pun yang mengganggu berfungsinya S-box. Algoritma DES telah dipelajari secara ekstensif dan, sampai saat ini, tidak ada kekurangan serius yang dipublikasikan.

Menanggapi kritik, NSA merilis informasi tertentu tentang pemilihan S-box.

- Tidak ada S-box yang merupakan fungsi linear atau affine dari inputnya; yaitu, empat bit keluaran tidak dapat dinyatakan sebagai sistem persamaan linier dari enam bit masukan.
- Mengubah satu bit input dari S-box menghasilkan perubahan setidaknya dua bit output; yaitu, kotak-S menyebarkan informasinya dengan baik ke seluruh keluarannya.
- S-box dipilih untuk meminimalkan perbedaan antara jumlah 1s dan 0s ketika setiap bit input tetap konstan; yaitu, memegang satu bit konstan sebagai 0 atau 1 dan mengubah bit di sekitarnya seharusnya tidak menghasilkan banyak 0 atau 1 secara tidak proporsional dalam output.

Jumlah Iterasi

Banyak analis bertanya-tanya apakah 16 iterasi sudah cukup. Karena setiap iterasi menyebarkan informasi dari plaintext ke seluruh ciphertext, tidak jelas apakah 16 siklus menyebarkan informasi secara memadai. Misalnya, dengan hanya satu siklus, satu bit ciphertext hanya dipengaruhi oleh beberapa bit plaintext. Dengan lebih banyak siklus, difusi menjadi lebih besar, jadi idealnya tidak ada satu bit cipherteks yang bergantung pada subset bit plainteks.

Eksperimen dengan DES dan pendahulunya IBM Lucifer dilakukan oleh NBS dan oleh IBM sebagai bagian dari proses sertifikasi algoritma DES. Eksperimen ini telah menunjukkan] bahwa 8 iterasi cukup untuk menghilangkan ketergantungan yang diamati. Dengan demikian, 16 iterasi DES tentunya harus memadai.

Kriptanalisis Diferensial

Pada tahun 1990 Eli Biham dan Adi Shamir mengumumkan sebuah teknik yang mereka namakan kriptanalisis diferensial. Teknik yang diterapkan pada algoritma kriptografi yang menggunakan substitusi dan permutasi. Teknik yang kuat ini adalah yang pertama memiliki efek yang mengesankan terhadap berbagai algoritma jenis ini. Teknik ini menggunakan pasangan plaintext yang dipilih secara hati-hati dengan perbedaan halus dan mempelajari efek dari perbedaan ini pada ciphertext yang dihasilkan. Jika kombinasi bit input tertentu dimodifikasi secara bersamaan, bit perantara tertentu juga kemungkinan besar akan berubah dengan cara tertentu. Teknik ini melihat eksklusif OR (XOR) dari sepasang input; XOR akan memiliki 0 di setiap bit di mana input identik dan 1 di mana mereka berbeda.

Analisis lengkapnya agak rumit, tetapi kami menyajikan sketsanya di sini. S-box mengubah enam bit menjadi empat. Jika S-box benar-benar seragam, orang akan mengharapkan semua output 4-bit memiliki kemungkinan yang sama. Namun, seperti yang ditunjukkan Biham dan Shamir, teks-teks serupa tertentu lebih mungkin menghasilkan keluaran yang serupa daripada yang lain. Misalnya, memeriksa semua pasangan string 6-bit dengan pola XOR 35 dalam notasi heksadesimal (yaitu, string bentuk ddsdsd di mana d berarti nilai bit berbeda antara dua string dan s berarti nilai bitnya sama) untuk S-box S1, peneliti menemukan bahwa pasangan memiliki pola keluaran dsss 14 kali, ddds 14 kali, dan semua pola lainnya memiliki frekuensi berkisar antara 0 dan 8. Artinya, masukan berupa ddsdsd memiliki keluaran sebesar bentuk dsss 14 kali dari 64, dan ddds 14 kali lagi dari 64; masing-masing hasil ini hampir 1/4, yang berlanjut ke babak berikutnya. Biham dan Shamir menyebut masing-masing efek yang dapat dikenali ini sebagai "karakteristik"; mereka kemudian

memperluas hasil mereka dengan menggabungkan karakteristik. Serangan itu memungkinkan mereka menyimpulkan nilai di posisi kunci tertentu. Jika m bit dari kunci k-bit dapat ditemukan, sisa (k-m) bit dapat ditemukan dalam pencarian menyeluruh dari semua 2(k-m) kunci yang mungkin; jika m cukup besar, pencarian lengkap 2(k–m) layak dilakukan.

Dalam [BIH90] penulis menyajikan kesimpulan dari banyak hasil yang telah mereka hasilkan dengan menggunakan kriptanalisis diferensial; mereka menjelaskan rincian hasil ini dalam makalah berikutnya. Serangan terhadap Lucifer, pendahulu DES yang dirancang IBM, hanya berhasil dengan 30 pasangan cipherteks. FEAL adalah algoritma yang mirip dengan DES yang menggunakan sejumlah putaran; versi n-putaran disebut FEAL-n. FEAL-4 dapat dipecahkan dengan 20 item teks biasa yang dipilih, FEAL-8 dengan 10.000 pasang ; dan FEAL-n untuk n≤31 dapat dipecahkan lebih cepat dengan kriptanalisis diferensial daripada dengan pencarian lengkap [BIH91]. Singkatnya, FEAL rentan terhadap kriptanalisis diferensial.

Hasil mengenai DES sangat mengesankan. Memperpendek DES menjadi kurang dari 16 putaran normal memungkinkan kunci ditentukan dari ciphertext yang dipilih dalam waktu kurang dari 256 (sebenarnya, nilai yang diharapkan dari 255) pencarian. Misalnya, dengan 15 putaran, hanya diperlukan 252 tes (yang masih merupakan jumlah tes yang besar); dengan 10 putaran, jumlah tes turun menjadi 235, dan dengan 6 putaran, hanya diperlukan 28 tes. Namun, dengan 16 putaran penuh, teknik ini membutuhkan 258 tes, atau 22 = 4 kali lebih banyak upaya daripada yang dibutuhkan pencarian lengkap.

Akhirnya, penulis menunjukkan bahwa dengan nilai S-box yang dipilih secara acak, DES mudah untuk dipecahkan. Memang, bahkan dengan perubahan hanya satu entri dalam satu S-box, DES menjadi mudah dibobol. Orang mungkin menyimpulkan bahwa desain kotak-S dan jumlah putaran yang dipilih menjadi optimal.

Faktanya, itu benar. Don Coppersmith dari IBM, salah satu tim asli yang mengerjakan Lucifer dan DES, mengakui bahwa teknik kriptanalisis diferensial diketahui oleh tim desain pada tahun 1974 ketika mereka merancang DES. Tim menyusun S-box dan permutasi sedemikian rupa untuk mengalahkan garis serangan itu. Pekerjaan kriptanalisis diferensial menunjukkan bahwa dasar DES memang solid, dan perubahan desain NSA yang tidak dapat dijelaskan hanya memperkuatnya.

6.2.3 Enkripsi AES

Karena kekhawatiran tentang kunci DES berukuran tetap dan fakta bahwa daya komputasi terus meningkat terhadap target stasioner itu, analis keamanan mulai mencari pengganti DES.

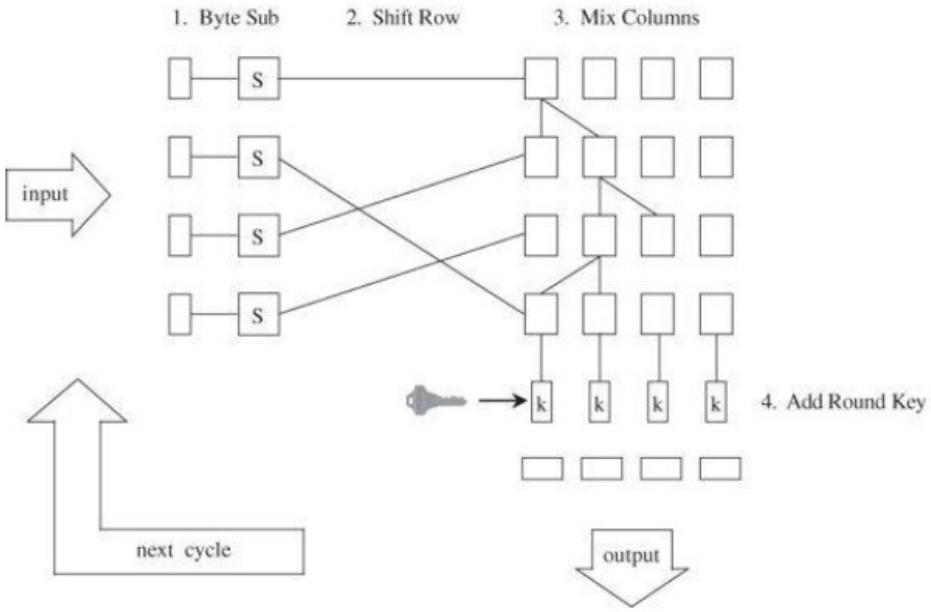
Kontes AES

Pada Januari 1997, NIST meminta kriptografer untuk mengembangkan sistem enkripsi baru. Seperti panggilan untuk kandidat dari mana DES dipilih, NIST membuat beberapa batasan penting. Algoritme harus tidak diklasifikasikan dan diungkapkan kepada publik, dan, untuk mempromosikan penggunaan luas oleh bisnis, NIST menetapkan bahwa algoritme ditawarkan bebas royalti untuk digunakan di seluruh dunia. Penggantian DES juga harus berupa cipher blok simetris yang dapat beroperasi pada blok minimal 128 bit. Akhirnya, untuk mengatasi batasan panjang kunci DES, NIST membutuhkan algoritma baru untuk dapat menggunakan kunci dengan panjang 128, 192, dan 256 bit.

Pada bulan Agustus 1998, lima belas algoritma dipilih dari antara yang diajukan; pada Agustus 1999, bidang kandidat dipersempit menjadi lima finalis. Kelimanya kemudian menjalani pengawasan publik dan swasta yang ekstensif. Pemilihan akhir dibuat atas dasar tidak hanya keamanan tetapi juga biaya atau efisiensi operasi dan kemudahan implementasi dalam perangkat lunak. NIST menggambarkan empat yang tidak dipilih juga memiliki keamanan yang memadai untuk AES—tidak ada kelemahan kriptografi yang diidentifikasi di salah satu dari lima. Dengan demikian, pemilihan didasarkan pada efisiensi dan karakteristik implementasi. Algoritma pemenang, yang diajukan oleh dua kriptografer Belanda, adalah Rijndael (diucapkan RINE dahl); nama algoritme berasal dari nama pembuatnya, Vincent Rijmen dan Joan Daemen.

Algoritme ini didasarkan pada aritmatika di bidang hingga GF(28), tetapi sebagian besar operasi enkripsi dapat dilakukan dengan pencarian tabel, sehingga menyederhanakan implementasi AES. Algoritma terdiri dari 10, 12 atau 14 siklus, masing-masing untuk kunci 128-, 192-, atau 256-bit. Setiap siklus (disebut "putaran" dalam algoritma) terdiri dari empat langkah.

- Substitusi byte. Langkah ini menggunakan substitusi yang menggantikan setiap byte dari blok 128-bit menurut tabel substitusi. Ini adalah operasi difusi lurus.
- Pergeseran baris. Bit tertentu digeser ke posisi lain. Ini adalah operasi kebingungan lurus.
- Kolom campuran. Langkah ini melibatkan pemindahan bit ke kiri dan XOR dengan dirinya sendiri. Operasi ini menerapkan kebingungan dan difusi.
- Tambahkan subkunci. Di sini, sebagian dari kunci unik untuk siklus ini di-XOR dengan hasil siklus. Operasi ini memberikan kebingungan dan menggabungkan kuncinya.



Gambar 6-8 Struktur AES

Setiap putaran melakukan kebingungan dan difusi, serta memadukan kunci ke dalam hasil. Struktur AES ditunjukkan pada Gambar 6-8.

Kekuatan Algoritma Rijndael

Algoritma Rijndael relatif baru, tetapi antara pengajuannya sebagai kandidat untuk AES pada tahun 1997 dan pemilihannya pada tahun 2001, algoritma ini menjalani kriptanalisis ekstensif oleh kriptografer pemerintah dan independen. Penemu Belandanya tidak memiliki hubungan dengan NSA atau bagian lain dari pemerintah AS, jadi tidak ada kecurigaan bahwa pemerintah entah bagaimana melemahkan algoritme atau menambahkan pintu jebakan. Meskipun langkah-langkah dari suatu siklus sederhana untuk dijelaskan dan tampaknya merupakan transformasi bit yang agak acak, transformasi ini memiliki asal mula matematis yang baik.

Ketika pendahulu Rijndael, DES, diadopsi, dua pertanyaan segera muncul:

1. Seberapa kuat, dan khususnya, apakah ada pintu belakang?
2. Berapa lama sampai kode terenkripsi dapat di-crack secara rutin?

Dengan penggunaan sekitar 40 tahun, kecurigaan kelemahan DES (disengaja atau tidak) dan backdoors telah cukup banyak dibatalkan. Analis tidak hanya gagal menemukan kekurangan yang signifikan, tetapi pada kenyataannya penelitian yang dijelaskan sebelumnya dalam bab ini telah menunjukkan bahwa perubahan yang tampaknya tidak signifikan melemahkan kekuatan algoritme—yaitu, algoritma adalah yang terbaik. Pertanyaan kedua, tentang berapa lama DES akan bertahan, tidak terjawab untuk waktu yang lama tetapi kemudian dijawab dengan sangat cepat oleh dua percobaan pada tahun 1997 dan 1998, di mana DES dipecahkan dalam beberapa hari. Jadi, lebih dari 20 tahun setelah peluncuran DES, kekuatan prosesor khusus individu dan pencarian paralel besar-besaran akhirnya mengambil alih ukuran kunci DES yang tetap.

Kita harus menanyakan pertanyaan yang sama tentang AES: Apakah itu memiliki kekurangan, dan untuk berapa lama akan tetap terdengar? Kami belum bisa menjawab pertanyaan tentang kekurangan, selain mengatakan bahwa tim cryptanalyst meneliti desain Rijndael selama periode peninjauan dua tahun tanpa menemukan masalah. Selanjutnya, sejak AES diadopsi pada tahun 2001, satu-satunya tantangan serius terhadap keamanannya adalah sangat terspesialisasi dan teoretis.

Pertanyaan umur panjang lebih sulit, tetapi juga lebih optimis, untuk dijawab untuk AES daripada untuk DES. Ingat bahwa memperpanjang kunci satu bit menggandakan upaya serangan brute force. Algoritma AES seperti yang didefinisikan dapat menggunakan kunci 128-, 192-, atau 256-bit. Jadi, relatif terhadap kunci DES 56-bit, kunci AES 128-bit menghasilkan 72 penggandaan, yang berarti pekerjaan 272 (kira-kira 4*1021) kali lebih keras. Panjang kunci 192 dan 256 bit memperpanjang upaya yang sudah luar biasa ini bahkan lebih. Tetapi karena ada struktur dasar yang jelas untuk AES, bahkan dimungkinkan untuk menggunakan pendekatan umum yang sama pada masalah mendasar yang sedikit berbeda dan mengakomodasi kunci dengan ukuran yang lebih besar. Jadi, tidak seperti DES, AES memungkinkan pengguna berpindah ke kunci yang lebih panjang setiap kali teknologi mengancam untuk memungkinkan seorang analis mengambil alih ukuran kunci saat ini, sehingga dugaan Diffie dan Hellman tahun 1977 tidak mungkin berlaku untuk AES kapan saja di masa mendatang. Selain itu, panjang kunci yang diperpanjang ini membangun margin keamanan jika serangan pintar membagi upaya dalam serangan brute force.

Namun demikian, cryptanalyst terus mengeksplorasi AES. Sejak diperkenalkan pada tahun 2001, ada sedikit paparan kepentingan akademis, tetapi tidak ada yang mengancam keamanan AES. Seorang peneliti menggambarkan salah satu serangan ini: “Meskipun kerumitan ini jauh lebih cepat daripada pencarian menyeluruh, mereka sama sekali tidak praktis, dan tampaknya tidak menimbulkan ancaman nyata terhadap keamanan sistem berbasis AES.” Upaya penelitian itu berlanjut untuk menunjukkan serangan yang mengurangi dari 2256 menjadi 245 upaya untuk menyimpulkan satu kunci, tetapi serangan ini menggunakan versi 10 putaran AES, bukan 14 putaran standar. Serangan seperti itu adalah pengurangan waktu yang nyata, tetapi serangan itu seperti melawan seorang pria yang satu tangannya terikat di belakang punggungnya. Serangan yang lebih adil disajikan oleh Andrey Bogdanov dan rekannya [BOG11], di mana mereka mengurangi kompleksitas penurunan kunci dengan kelipatan empat, yaitu dari 2256 menjadi 2252, dengan hasil yang serupa untuk versi kunci 128- dan 192-bit . Mengurangi pekerjaan dengan faktor empat hampir tidak ada perubahan sama sekali.

Tidak ada serangan hingga saat ini yang menimbulkan pertanyaan serius mengenai kekuatan keseluruhan AES.

6.2.4 RC2, RC4, RC5, dan RC6

Dalam kriptografi,RC2 adalah cipher blok yang dirancang oleh Ron Rivest di 1987.“RC” singkatan dari “Ron’s Code” atau “Rivest Cipher”.cipher lainnya yang dirancang oleh Rivest termasuk RC4,RC5 dan RC6.

Pengembangan RC2 disponsori oleh Lotus,yang sedang mencari kebiasaan sandi yang, setelah evaluasi oleh NSA,dapat diekspor sebagai bagian dari Lotus Notes perangkat lunak. NSA menyarankan beberapa perubahan, yang Rivest dimasukkan.Setelah negosiasi lebih lanjut,cipher telah disetujui untuk ekspor di 1989.Seiring dengan RC4,RC2 dengan 40-bit ukuran kunci dirawat baik di bawah US peraturan ekspor untuk kriptografi.

Awalnya,rincian algoritma dirahasiakan,eksklusif untuk RSA Security,tetapi pada Januari 29,1996,kode sumber untuk RC2 adalah anonim diposting ke Internet pada Usenet forum, sci.crypt.Sebuah pengungkapan yang sama telah terjadi sebelumnya dengan algoritma RC4. Tidak jelas apakah poster itu memiliki akses dengan spesifikasi atau apakah telah reverse engineered.

RC2 adalah 64-bit blok cipher dengan ukuran variabel kunci.18 Its putaran diatur sebagai sumber-berat jaringan Feistel,dengan 16 putaran dari satu jenis (pencampuran) diselingi oleh dua putaran jenis lain (dihaluskan.Sebuah putaran pencampuran terdiri dari empat aplikasi transformasi MIX,seperti ditunjukkan pada diagram.

RC4 juga menggunakan variabel berisi ukuran kunci. Hanya saja algoritma ini menggunakan aliran kode dan digunakan secara luas pada sistem enkripsi keamanan sepeerti protocol SSL (Secure Sockets Layer). Sedangkan RC6, yang merupakan perkembangan dari RC5, adalah golongan blok kode yang merupakan tiga parameter, yaitu ukuran blok (32,64 atau 128), ukuran kunci primer (0-2048 bit),dan round number (0-255). Dua algoritma ini menyediakan berbagai format pengenkripsian suatu pesan.

Cipher RC2, RC4, RC5, dan RC6 semuanya berasal dari Ronald Rivest, salah satu penemu algoritma RSA dan pendiri RSA Laboratories. (RC pada nama dapat berarti “Rivest cipher” atau “Kode Ron”, tergantung pada sumber mana yang Anda yakini.) Cipher secara struktural berbeda, tetapi semuanya memiliki tingkat penggunaan yang sama, jadi kami menjelajahinya secara singkat di sini.

Cipher RC2

RC2 adalah cipher blok yang dirancang sebagai algoritma yang sederhana dan cepat [KNU02]. Meskipun Rivest awalnya bermaksud agar algoritme menjadi hak milik, seseorang merilis deskripsi desainnya di Internet pada tahun 1996.

Karakteristik algoritma yang paling signifikan adalah panjang kuncinya yang kecil: 40 bit. Ketika dirancang, pemerintah AS membatasi kekuatan (diukur dengan panjang kunci) aplikasi kriptografi untuk ekspor tidak lebih dari 40 bit. (Perhatikan bahwa kunci DES 56-bit dipulihkan dalam waktu singkat hanya setahun kemudian, jadi kunci 40-bit pasti dapat dengan mudah dipulihkan oleh organisasi mana pun yang ingin memecahkan enkripsi.) Algoritme ini pertama kali ditujukan untuk penggunaan internasional oleh suite aplikasi kantor Lotus Notes; itu akan menggunakan kunci yang cukup pendek untuk memenuhi pembatasan ekspor AS ke sebagian besar negara, sehingga memastikan Lotus dapat dipasarkan secara internasional.

Faktanya, RC2 dapat mendukung ukuran kunci dari 8 hingga 128 bit, memberikan kekuatan yang melebihi DES terhadap pencarian kunci yang lengkap. Pengoperasiannya cukup mirip dengan DES sehingga dapat menggantikan DES dalam aplikasi, membuat edisi internasional tanpa kesulitan. Dengan relaksasi pembatasan ekspor pada tahun 2000, kebutuhan akan pengganti DES yang lebih pendek telah berkurang.

RC2 terdiri dari dua operasi: pencampuran dan menumbuk. Dalam pencampuran, aliran bit mengalami beberapa transposisi dalam bentuk pergeseran bit dengan substitusi bersamaan melalui operasi biner (AND, OR, NOT) pada bagian bit. Selama setiap putaran pencampuran, pengocokan bit terjadi dari kanan, bergerak ke kiri, dan berputar ke kanan lagi. Ada enam belas putaran pencampuran. Putaran menumbuk adalah substitusi murni. Dua putaran menumbuk dilakukan setelah pencampuran putaran 5 dan 11.

Diciptakan pada tahun 1987, RC2 sudah tua seperti cryptosystems pergi. Tidak ada kelemahan serius yang ditemukan dalam desain, tetapi kunci 40-bit membuat pencarian kunci brute force menjadi sepele.

Cipher RC4

RC4 adalah stream cipher, banyak digunakan dalam jaringan nirkabel (WEP dan WPA, serta di SSL dan berbagai produk. Itu sangat populer sebelum tahun 2000 karena, seperti RC2, menggunakan kunci panjang variabel dan dapat dikonfigurasi untuk menggunakan kunci 40-bit, cukup pendek untuk melewati pembatasan ekspor.

RC4 pada dasarnya adalah generator nomor pseudorandom (PRNG); itu menghasilkan aliran bit dalam urutan yang tidak dapat diprediksi. Untuk enkripsi, aliran bit di-XOR dengan bit plaintext.

Algoritma ini sangat sederhana. Ini menggunakan array 256-elemen A yang berisi masing-masing 256 nilai yang mungkin dari byte 8-bit. Pointer i dan j mengidentifikasi byte dari array yang akan ditukar. Pada setiap langkah, i bertambah 1, j diganti dengan j + A[i], A[i] dan A[j] ditukar, dan byte A[A[i]+A[j]] dihasilkan sebagai keluaran. (Semua penambahan dilakukan mod 256.) Algoritma ini sangat efisien, terutama untuk implementasi perangkat lunak.

Tidak ada kelemahan cryptanalytic serius yang ditemukan dalam algoritma itu sendiri. Namun urutan nomor acak dari stream cipher XOR tidak boleh berulang. Artinya, kunci yang sama tidak boleh digunakan untuk dua plainteks yang berbeda. Untuk mengetahui alasannya, pertimbangkan plaintext p1 dan p2, dienkripsi dengan kunci umum k.
c1 = p1 ⊕ k
c2 = p2 ⊕ k

Penyerang mengambil dua ciphertext dan menghitung
c1 ⊕ c2 = (p1 ⊕ k) (p2 ⊕ k)
= p1 ⊕ p2 (k ⊕ k)
= p1 ⊕ p2

dari mana p1 dan p2 dapat dipulihkan dengan analisis frekuensi, kemungkinan teks biasa, atau teknik lainnya.

Banyak implementasi RC4 memiliki kelemahan itu. Untuk menginisialisasi array A, algoritma dimulai dengan nilai 1 sampai 256 dalam urutan numerik. Kemudian ia bekerja melalui 256 byte, menukar setiap byte dengan byte yang lokasinya bergantung, sebagian, pada satu byte dari kunci: untuk setiap i,

```
j := j + A[i] + kunci[i]
```

dan A[*i*] dan A[*j*] ditukar. Jadi kunci hingga-256-byte mengontrol bagaimana array angka acak diinisialisasi.

Untuk melindungi dari serangan plaintext yang identik, cipher dan terutama stream cipher XOR digunakan dengan vektor inisialisasi, juga disebut nonce; nilai ini bekerja seperti vektor inisialisasi yang kami jelaskan dengan DES. Dalam beberapa implementasi RC4 nonce ditambahkan ke kunci, secara efektif memperluas dan mengacak kunci.

Fluhrer dkk. menganalisis output RC4 untuk semua kunci yang mungkin dan menemukan bahwa outputnya bias, membocorkan informasi tentang kunci tersebut. Jika nonce telah ditambahkan ke kunci, dimungkinkan untuk mempersempit ruang pencarian untuk kunci secara signifikan.

RC4 banyak digunakan untuk enkripsi WEP pada jaringan nirkabel. Titik akses nirkabel dan perangkat jarak jauh menggunakan kunci yang sama tanpa batas waktu hingga dikunci ulang secara manual. Kelemahan Fluhrer et al. diidentifikasi telah memungkinkan enkripsi WEP untuk dipecahkan dalam hitungan menit. Kelemahan ini menyebabkan pengembangan WPA dan WPA2 untuk komunikasi nirkabel, yang terakhir menggunakan enkripsi AES yang jauh lebih kuat.

Pada bulan Maret 2012, tim peneliti di Royal Holloway–University of London dan University of Illinois mengungkapkan serangan baru terhadap TLS (dijelaskan dalam Bab 6 bersama dengan SSL pendahulunya) yang memungkinkan penyerang memulihkan sejumlah teks biasa dari koneksi TLS saat enkripsi RC4 digunakan. Serangan muncul dari kelemahan statistik dalam keystream yang dihasilkan oleh algoritma RC4 yang menjadi jelas dalam ciphertext TLS ketika plaintext yang sama berulang kali dienkripsi. Kemudian pada tahun 2012, pengungkapan dari mantan karyawan NSA menunjukkan NSA dan badan intelijen lainnya mungkin memiliki kemampuan untuk memecahkan enkripsi RC4. Akibatnya, Microsoft merekomendasikan agar pelanggan berhenti menggunakan RC4.

Keamanan TLS dan terutama SSL pendahulunya (enkripsi yang dipanggil di bawah protokol https) juga dapat menggunakan RC4, meskipun browser memberi pengguna opsi untuk memilih atau melarang algoritme tertentu.

RC4 juga telah digunakan dalam produk Microsoft Office Word dan Excel untuk mengenkripsi dokumen yang dilindungi kata sandi. Microsoft membuat kesalahan dengan mengenkripsi setiap versi dokumen di bawah kunci enkripsi yang sama (kata sandi). Serangan seperti serangan aliran XOR yang dijelaskan di atas di mana teks dari dokumen terenkripsi dapat dengan mudah diambil, mengingat dua versi dokumen.

Cipher RC5

RC5 adalah cipher blok berparameter penuh; ini berarti panjang kunci, ukuran blok, dan jumlah siklus dapat divariasikan untuk mengubah keseimbangan antara keamanan dan kerumitan pengoperasian dan penggunaan. RC5 [RIV94] menggunakan desain sederhana yang berfungsi sebagai model untuk kandidat AES RC6.

Blok data di RC5 dibagi dua, separuh kiri dimodifikasi, separuh ditukar, separuh kiri baru (yaitu, separuh kanan lama) dimodifikasi dengan cara yang sama, dan separuh ditukar lagi. Urutan itu merupakan putaran penuh dari algoritma. Modifikasi setiap setengah putaran melibatkan XOR, pergeseran melingkar, dan penambahan sebagian kunci. Dalam putaran yang tidak biasa untuk algoritma kriptografi, jumlah bit yang digeser tergantung pada data input: Setengah kiri digeser oleh jumlah bit dari nilai setengah kanan.

Tidak ada kelemahan signifikan yang ditemukan di RC5. Enkripsi dengan sejumlah kecil (12) putaran telah ditemukan tunduk pada kriptanalisis diferensial, tetapi jumlah putaran dapat diatur secara sewenang-wenang. Karena operasi per putaran sedikit dan sederhana dan kecepatan implementasi bergantung secara linier pada jumlah putaran, menaikkan jumlah putaran di atas 12 tidak memperlambat enkripsi secara signifikan.

Cipher RC6

Untuk bersaing di kompetisi AES, RSA Laboratories melakukan modifikasi kecil pada RC5, menyebutnya RC6. Cipher RC6 adalah produk eksklusif RSA Security, tetapi tampaknya tidak didukung.

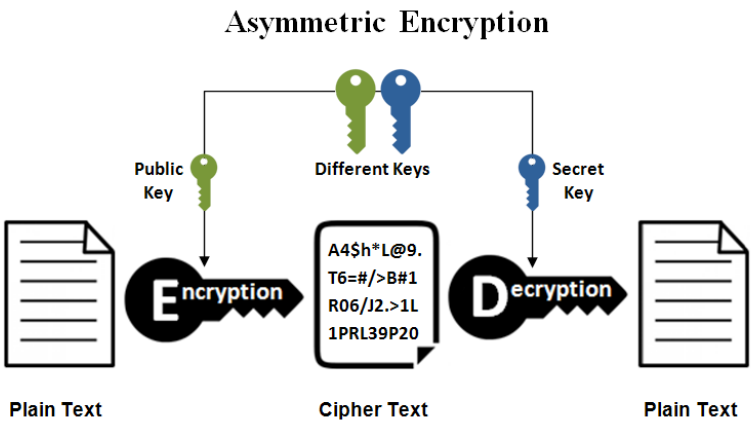
Ada ribuan algoritma enkripsi simetris lainnya. Seperti yang telah kami nyatakan sebelumnya dalam bab ini, kriptologi adalah subjek yang sangat kompleks, tentu saja tidak bisa dianggap enteng. Meskipun seorang teman dari seorang teman mungkin memiliki latar belakang untuk merancang algoritma yang solid, berhati-hatilah terhadap amatir yang mempraktikkan kriptologi (atau operasi otak, dalam hal ini). Anda harus memilih kriptografi dengan hati-hati, dan tentu saja mencocokkan penilaian Anda tentang kekuatan suatu teknik dengan nilai data yang ingin Anda lindungi. Algoritme terkenal yang telah bertahan dalam pengawasan terpadu adalah alat terbaik Anda.

Kami sekarang beralih ke algoritma yang lebih canggih dalam kriptografi asimetris.

6.3 Enkripsi Asimetris dengan RSA

Enkripsi asimetris juga dikenal sebagai kriptografi menggunakan kunci publik (public-key), yang merupakan pengembangan dari metode enkripsi simetris dan lebih menjamin keamanan. Enkripsi asimetris menggunakan dua kunci yaitu kunci publik (public key) dan kunci pribadi (private key). Kunci publik tersedia secara bebas bagi siapa saja yang mungkin ingin mengirimkan pesan kepada Anda, namun Kunci pribadi (kunci kedua) dirahasiakan sehingga hanya Anda yang mengetahuinya.

Pesan yang dienkripsi menggunakan kunci publik hanya dapat didekripsi dengan menggunakan kunci pribadi, sementara pesan yang dienkripsi menggunakan kunci pribadi dapat didekripsi dengan menggunakan kunci publik. Keamanan kunci publik tidak diperlukan karena tersedia untuk umum dan bisa ditransmisikan melalui internet. Kunci asimetris memiliki kekuatan yang jauh lebih baik dalam menjamin keamanan informasi yang ditransmisikan selama komunikasi berlangsung. Lihat Gambar 6.9



Gambar 6.9 Enkripsi Asimetris

Enkripsi asimetris banyak digunakan dalam saluran komunikasi sehari-hari, terutama melalui Internet. Algoritma enkripsi asimetris yang populer digunakan antara lain ElGamal, RSA, DSA, Elliptic, dan PKCS.

Algoritma RSA adalah kriptosistem yang didasarkan pada masalah sulit yang mendasari Algoritma ini diperkenalkan pada tahun 1978 oleh Rivest, Shamir, dan Adelman [RIV78]. RSA telah menjadi subjek kriptanalisis ekstensif. Belum ada kelemahan serius yang ditemukan—bukan jaminan keamanannya tetapi menunjukkan tingkat kepercayaan yang tinggi dalam penggunaannya.

6.3.1 Algoritma RSA

Pada bagian ini, kami menyajikan algoritma RSA dalam dua bagian. Pertama, kami menguraikan RSA, untuk memberi Anda gambaran tentang cara kerjanya. Kemudian, kami mempelajari lebih dalam analisis rinci dari langkah-langkah yang terlibat.

Pengantar Algoritma RSA

Algoritma RSA membutuhkan pencarian istilah yang mengalikan ke produk tertentu. Algoritma enkripsi RSA menggabungkan hasil dari teori bilangan, dikombinasikan dengan kesulitan menentukan faktor prima dari suatu target. Algoritma RSA beroperasi dengan mod aritmatika n, yang membuat faktorisasi menjadi sangat sulit.

Algoritma enkripsi didasarkan pada masalah mendasar dari pemfaktoran bilangan besar, masalah yang algoritma tercepat diketahui adalah eksponensial dalam waktu.

Dua kunci, *d* dan *e*, digunakan untuk dekripsi dan enkripsi. Mereka sebenarnya dapat dipertukarkan. Blok plaintext *P* dienkripsi sebagai $P^e \bmod n$. Karena eksponensial dilakukan *mod n*, memfaktorkan P^e untuk mengungkap plaintext terenkripsi adalah hal yang menakutkan. Namun, kunci dekripsi *d* dipilih dengan hati-hati sehingga $(P^e)d \bmod n = P$. Jadi, penerima sah yang mengetahui *d* cukup menghitung $(P^e)d \bmod n = P$ dan memulihkan P tanpa harus memfaktorkan P^e.

Deskripsi Detail Algoritma Enkripsi

Algoritma RSA menggunakan dua kunci, *d* dan *e*, yang bekerja berpasangan, masing-masing untuk dekripsi dan enkripsi. Pesan plaintext *P* dienkripsi menjadi ciphertext *C* oleh

$$C = P^e \bmod n$$

Plaintext dipulihkan oleh

$$P = C^d \bmod n$$

Karena simetri dalam aritmatika modular C, enkripsi dan dekripsi saling invers dan komutatif. Karena itu,

$$P = C^d \bmod n = (P^e)d \bmod n = (P^d)e \bmod n$$

Hubungan ini berarti bahwa seseorang dapat menerapkan transformasi enkripsi dan kemudian yang mendekripsi, atau yang mendekripsi diikuti oleh yang mengenkripsi.

Menurunkan Pasangan Kunci

Kunci enkripsi terdiri dari pasangan bilangan bulat (*e, n*), dan kunci dekripsi adalah (*d, n*). Titik awal dalam menemukan kunci untuk algoritma ini adalah pemilihan nilai n. Nilai n harus cukup besar, hasil kali dua bilangan prima *p* dan *q*. Baik *p* dan *q* harus besar sendiri. Biasanya, *p* dan *q* masing-masing hampir 100 digit, jadi panjang n kira-kira 200 digit desimal (sekitar 512 bit);

tergantung pada aplikasinya, 768, 1024, atau lebih bit mungkin lebih tepat. Nilai *n* yang besar secara efektif menghambat pemfaktoran *n* untuk menyimpulkan *p* dan *q* (tetapi waktu untuk mengenkripsi meningkat seiring dengan bertambahnya nilai *n*).

Selanjutnya, sebuah bilangan bulat e yang relatif besar dipilih sehingga e relatif prima untuk *(p - 1) * (q - 1)*. (Ingat bahwa "relatif prima" berarti *e* tidak memiliki faktor yang sama dengan (p - 1) * (q - 1).) Cara mudah untuk menjamin bahwa *e* relatif prima ke *(p - 1) * (q - 1*) adalah dengan memilih *e* sebagai bilangan prima yang lebih besar dari keduanya *(p - 1) dan (q - 1)*.

Akhirnya, pilih *d* sedemikian rupa sehingga

$$e * d = 1 \bmod (p - 1) * (q - 1)$$

Contoh

Misalkan $p = 11$ dan $q = 12$, sehingga $n = p * q = 143$ dan $j(n) = (p - 1) * (q - 1) = 10 * 12 = 120$. Selanjutnya, diperlukan bilangan bulat e, dan e harus relatif prima untuk $(p - 1) * (q - 1)$. Pilih $e = 11$.

Kebalikan dari 11 mod 120 juga 11, karena 11 * 11 = 121 = 1 mod 120. Jadi, kunci enkripsi dan dekripsi adalah sama: $e = d = 11$. (Misalnya, $e = d$ bukan masalah, tetapi dalam aplikasi nyata Anda ingin memilih nilai di mana e tidak sama dengan *d*.)

Biarkan *P* menjadi "pesan" untuk dienkripsi. Untuk contoh ini kita menggunakan $P = 7$. Pesan dienkripsi sebagai berikut: 711 mod 143 = 106, sehingga $E(7) = 106$. (Catatan: Hasil ini dapat dihitung dengan cukup mudah dengan menggunakan kalkulator saku biasa. 711 = 79 * 72. Maka 79 = 40 353 607, tetapi kita tidak harus bekerja dengan angka sebesar itu.

Dikarenakan aturan reducibility, $a * b \bmod n = (a \bmod n) * (b \bmod n) \bmod n$. Karena kita akan mengurangi hasil akhir mod 143, kita dapat mengurangi istilah apa pun, seperti 79, yaitu 8 mod 143. Kemudian, 8 * 72 mod 143 = 392 mod 143 = 106.)

Jawaban ini benar karena $D(106) = 10611 \bmod 143 = 7$.

Kekuatan Algoritma RSA

Algoritme RSA memperoleh kekuatan dari fakta bahwa ia didasarkan pada masalah pemfaktoran bilangan secara efisien dalam bidang terbatas, masalah terbuka yang sudah lama ada dalam teori bilangan. Masalahnya tentu saja dapat dipecahkan, dengan menggunakan teknik brute force mencoba semua faktor yang mungkin, tetapi dalam bidang yang besar, yaitu untuk nilai n yang besar, teknik brute force tidak mungkin dilakukan. Trik yang membuat enkripsi RSA bisa diterapkan tergantung pada teknik tersembunyi untuk memilih n.

Dasar Matematika dari Algoritma RSA

Fungsi totien Euler (n) adalah banyaknya bilangan bulat positif yang kurang dari n yang relatif prima terhadap n. Jika p prima, maka

$$\varphi(p) = p - 1$$

Selanjutnya, jika $n = p * q$, di mana p dan q keduanya prima, maka

$$\varphi(n) = \varphi(p) * \varphi(q) = (p - 1) * (q - 1)$$

Euler dan Fermat membuktikan bahwa

$$x^{\varphi(n)} = 1 \bmod n$$

untuk sembarang bilangan bulat *x* jika *n* dan *x* relatif prima.

Misalkan kita mengenkripsi pesan teks biasa *P* dengan algoritma RSA sehingga $E(P) = P^e$. Kami harus yakin bahwa kami dapat memulihkan pesan tersebut. Nilai *e* dipilih, sehingga kita dapat dengan mudah menemukan inversnya *d*. Karena *e* dan *d* adalah invers mod (*n*),

$$e * d \; 1 \bmod \varphi(n)$$

atau

$$e * d = k * \varphi(n) + 1 \; (*)$$

untuk beberapa bilangan bulat *k*.
Karena hasil Euler–Fermat, dengan asumsi *P* dan p relatif prima,

$$P^p - 1 \; 1 \bmod p$$

dan, karena (p-1) adalah faktor dari (n),

$$P^k * \varphi(n) \; 1 \bmod p$$

Mengalikan dengan P menghasilkan

$$P^k * \varphi(n) + 1 \; P \bmod p$$

Argumen yang sama berlaku untuk *q*, jadi

$$P^k * \varphi(n) + 1 \; P \bmod q$$

Menggabungkan dua hasil terakhir ini dengan (a) menghasilkan

$$\begin{aligned} (P^e)^d &= P^{e*d} \\ &= P^{k*\varphi(n)+1} \\ &= P \bmod p \\ &= P \bmod q \end{aligned}$$

yang seperti itu

$$(P^e)d \; P \bmod n$$

dan *e* dan *d* adalah operasi invers.

Penggunaan Algoritma

Pengguna algoritma RSA memilih bilangan prima *p* dan *q*, dari mana nilai *n* = *p* * *q* diperoleh. Selanjutnya, *e* dipilih relatif prima untuk *(p - 1) * (q - 1); e* biasanya bilangan prima yang lebih besar dari (*p* - 1) atau (q - 1). Akhirnya, *d* dihitung sebagai kebalikan dari *e* mod (φ(*n*)).

Pengguna mendistribusikan e dan n dan merahasiakan *d; p, q*, dan (*n*) dapat dibuang (tetapi tidak diungkapkan) pada saat ini. Perhatikan bahwa meskipun *n* diketahui merupakan hasil kali dua bilangan prima, jika keduanya relatif besar (seperti panjang 100 digit), tidak mungkin untuk menentukan bilangan prima *p* dan *q* atau kunci privat *d* dari *e*. Oleh karena itu, skema ini memberikan keamanan yang memadai untuk *d*.

Bahkan tidak praktis untuk memverifikasi bahwa p dan q sendiri adalah bilangan prima, karena itu akan memerlukan pertimbangan pada urutan 1050 faktor yang mungkin. Sebuah algoritma heuristik dari Solovay dan Strassen dapat menentukan probabilitas primality untuk setiap derajat kepercayaan yang diinginkan.

Setiap bilangan prima melewati dua tes. Jika *p* prima dan *r* adalah bilangan yang lebih kecil dari p, maka gcd(p, r) = 1 (di mana gcd adalah fungsi pembagi persekutuan terbesar) dan

$$J(r, p)\ r^{(p-1)/2} \bmod p$$

di mana J(r,p) adalah fungsi Jacobi yang didefinisikan sebagai berikut.

$$J(r, p) = \begin{cases} 1 & \text{if } r = 1 \\ J(r/2, p) * (-1)^{(p^2-1)/8} & \text{if } r \text{ is even} \\ J(p \bmod r, r) * (-1)^{(r-1)*(p-1)/4} & \text{if } r \text{ is odd and } r \neq 1 \end{cases}$$

Jika suatu bilangan diduga prima tetapi gagal dalam salah satu tes ini, itu jelas bukan bilangan prima. Jika suatu bilangan diduga merupakan bilangan prima dan lolos kedua pengujian ini, kemungkinan bilangan tersebut adalah bilangan prima paling sedikit 1/2.

Masalah relatif terhadap algoritma RSA adalah menemukan dua bilangan prima besar *p* dan *q*. Dengan pendekatan Solovay dan Strassen, pertama-tama Anda menebak kandidat prima besar *p*. Anda kemudian menghasilkan angka acak *r* dan menghitung gcd*(p,r)* dan *J(r, p)*. Jika salah satu dari tes ini gagal, *p* bukan bilangan prima, dan Anda menghentikan prosedur. Jika keduanya lolos, kemungkinan p bukan prima paling banyak 1/2. Anda mengulangi proses dengan nilai baru untuk *r* yang dipilih secara acak. Jika *r* kedua ini lolos, kemungkinan *p* nonprima dapat lulus kedua tes paling banyak adalah 1/4. Secara umum, setelah proses diulang sebanyak *k* kali tanpa salah satu pengujian gagal, kemungkinan *p* bukan bilangan prima paling banyak 1/2k.

Kriptanalisis Algoritma RSA

Metode RSA telah diteliti secara intensif oleh para profesional di bidang keamanan komputer dan kriptanalisis. Beberapa masalah kecil telah diidentifikasi dengan itu, tidak ada perhatian yang signifikan; Boneh [BON99] membuat katalog serangan yang diketahui terhadap RSA. Dia mencatat

tidak ada serangan yang berhasil pada RSA itu sendiri, tetapi beberapa serangan serius tetapi tidak mungkin pada implementasi dan penggunaan RSA.

RSA sejauh ini merupakan algoritma enkripsi kunci publik yang paling populer digunakan. Sekarang kita beralih ke dua aplikasi kriptografi: intisari pesan dan tanda tangan digital, yang keduanya kita perkenalkan di Bab 2. Dalam bab ini kita mengeksplorasi algoritme secara lebih rinci.

6.4 Message Digest

Dalam Bab 2 kami memperkenalkan konsep deteksi kesalahan dan kode koreksi. Secara khusus, kami menjelaskan fungsi hash satu arah dan kriptografi, keduanya dirancang untuk melindungi dari upaya jahat untuk mengubah data dan juga menyesuaikan nilai kode agar sesuai dengan data yang dimodifikasi.

Message Digest dirancang untuk melindungi integritas sebuah data atau media untuk mendeteksi perubahan dan perubahan pada bagian mana pun dari pesan. Mereka adalah jenis kriptografi yang memanfaatkan nilai hash yang dapat memperingatkan pemilik hak cipta tentang modifikasi apa pun yang diterapkan pada karya mereka.

Bilangan algoritma hash message digest mewakili file tertentu yang berisi karya yang dilindungi. Satu intisari pesan ditugaskan untuk konten data tertentu. Ini dapat merujuk pada perubahan yang dibuat dengan sengaja atau tidak sengaja, tetapi meminta pemilik untuk mengidentifikasi modifikasi serta individu yang membuat perubahan. Message diggest adalah angka algoritmik.

Istilah ini juga dikenal sebagai nilai hash dan terkadang sebagai checksum.

Message digest tertentu akan berubah jika file berubah. Message digest tidak hanya dapat membantu menentukan perubahan file, tetapi juga dapat membantu menemukan file duplikat.

IMessage digest dapat diproduksi pada sistem UNIX dengan perintah MD5. MD5 disimpan dengan aman di sistem dan dapat mengungkapkan jika pengguna yang tidak sah telah mengakses file. Telah ditunjukkan bahwa MD5 tidak dapat diandalkan dengan masalah yang berkaitan dengan tabrakan (di mana 2 kunci untuk data yang berbeda adalah sama) dan tidak lagi digunakan.

6.4.1 Fungsi Hash

Hampir setiap kali mengakses internet, kriptografi digunakan tanpa sepengetahuan penggunanya. Misalnya untuk berkomunikasi atau bertukar informasi, informasi tersebut akan di enkripsi sebelum dikirimkan dan dideskripsi kembali sebelum di terima oleh penerima. Contoh sederhananya, untuk login ke email saja membutuhkan fungsi hash untuk memeriksa password nya benar atau tidak.
Fungsi hash yang konseptual adalah fungsi yang hasil hash value tidak dapat didekripsi kembali. Hal ini dimanfaatkan untuk kerahasiaan. Ada berbagai macam algoritma untuk melakukan hash. Namun, karena semakin banyak ahli kriptanalisis, banyak algoritma terdahulu dianggap strong ternyata telah ditemukan celahnya.

Hash adalah suatu teknik "klasik" dalam Ilmu Komputer yang banyak digunakan dalam praktek secara mendalam. Hash merupakan suatu metode yang secara langsung mengakses record-record dalam suatu tabel dengan melakukan transformasi aritmatik pada key yang menjadi alamat dalam tabel tersebut. Key merupakan suatu input dari pemakai di mana pada umumnya berupa nilai atau string karakter. Pelacakan dengan menggunakan Hash terdiri dari dua langkah utama, yaitu:

1. Menghitung Fungsi Hash. Fungsi Hash adalah suatu fungsi yang mengubah key menjadi alamat dalam tabel. Fungsi Hash memetakan sebuah key ke suatu alamat dalam tabel. Idealnya, key-key yang berbeda seharusnya dipetakan ke alamat-alamat yang berbeda juga. Pada kenyataannya, tidak ada fungsi Hash yang sempurna. Kemungkinan besar yang terjadi adalah dua atau lebih key yang berbeda dipetakan ke alamat yang sama dalam tabel. Peristiwa ini disebut dengan collision (tabrakan). Karena itulah diperlukan langkah berikutnya, yaitu collision resolution (pemecahan tabrakan).
2. Collision Resolution. Collision resolution merupakan proses untuk menangani kejadian dua atau lebih key di-hash ke alamat yang sama. Cara yang dilakukan jika terjadi collision adalah mencari lokasi yang kosong dalam tabel Hash secara terurut. Cara lainnya adalah dengan menggunakan fungsi Hash yang lain untuk mencari lokasi kosong tersebut.

6.4.2 One-Way Hash Function

One-Way Hash Function adalah konstruksi kriptografi dengan banyak kegunaan. Mereka digunakan bersama dengan algoritma kunci publik untuk enkripsi dan tanda tangan digital. Mereka digunakan dalam pemeriksaan integritas. Mereka digunakan dalam otentikasi. Mereka digunakan dalam protokol komunikasi. Lebih dari sekadar algoritma enkripsi, fungsi hash satu arah adalah pekerja keras kriptografi modern.

One-Way Hash Function mencegah pihak luar mengambil hasil hash yang ada dan menentukan nilai data lain yang cocok dengan hasil hash tersebut. Jadi, Hector mungkin menerima pesan yang mengatakan "Saya rela memberikan resep kue bolu emas saya yang berharga kepada Hector" dan beberapa hal lainnya. Hector tentu saja dapat mengubah "resep kue bolu" menjadi "koleksi bullion" tetapi kemudian Hector terjebak: Dia perlu membuat perubahan lain pada pesannya, tetapi dia perlu mengetahui konten lain yang akan menghasilkan nilai hash asli. Dengan One-Way Hash Function dia bisa menebak “file resep”, “kotak potongan string yang terlalu pendek untuk digunakan”, dan sebagainya. Tetapi dia harus menemukan setiap frasa seperti itu dan mengujinya. Akan lebih mudah jika dia bisa menjalankan fungsi hash secara terbalik dan mendapatkan daftar input yang akan menghasilkan hasil hash yang diberikan. Sayangnya, dengan One-Way Hash Function, Hector harus terus mencoba sampai menemukan kecocokan.

Fungsi hash modern harus memenuhi dua kriteria: Fungsi ini satu arah, artinya fungsi tersebut mengubah input menjadi intisari, tetapi tidak mungkin untuk memulai dengan nilai intisari dan menyimpulkan masukan yang dapat menghasilkan intisari tersebut. Kedua, mereka tidak memiliki tumbukan yang jelas, artinya tidak mungkin menemukan sepasang plainteks berbeda yang menghasilkan intisari yang sama.

Catatan: Beberapa penulis menyebut properti kedua ini sebagai "collision free", tetapi itu adalah istilah yang menyesatkan. Setiap fungsi hash akan memiliki tabrakan—banyak di antaranya, karena fungsi tersebut mengambil input yang relatif besar dan menghasilkan intisari yang relatif kecil. Secara fisik tidak mungkin untuk mengurangi 512 bit menjadi 128-bit digest dan tidak memiliki collision. Intinya adalah bahwa tabrakan tidak dapat diprediksi. Kita tahu tabrakan akan terjadi; hanya tidak mungkin untuk memprediksi pasangan mana yang akan bertabrakan atau, dengan satu masukan, untuk menghitung masukan lain yang dengannya yang pertama akan bertabrakan.

Algoritma Hash MD5

MD5 (Message Digest 5) adalah Hash MD5 diperkenalkan oleh Profesor Ronald L. Rivest dan merupakan bagian dari kriptografi modern. Algoritma Hash MD5 adalah fungsi hash searah dengan nilai hash 128 bit. Hash dikatakan sebagai fungsi searah karena pesan yang dimasukkan akan diubah menjadi pesan pendek atau "intisari pesan" dan sulit untuk kembali ke pesan awal. Fungsi Hash dapat mengubah pesan input yang memiliki panjang berubah-ubah, menjadi pesan singkat yang panjangnya selalu diperbaiki.

Langkah-langkah untuk menghasilkan message digest adalah sebagai berikut:

1. Penambahan bit-bit pengganjal Pesan mula-mula ditambahkan dengan bit-bit pengganjal sehingga panjang pesan (dalam satuan bit) paralel dengan modulo 512. Setelah panjang data menjadi kelipatan 512, selanjutnya akan dikurangi 64 bit.
2. Penambahan nilai panjang pesan semula Representasi sebesar 64 bit tersebut akan ditambahkan ke pesan.
3. Inisialisasi penyangga (buffer) MD5. MD5 membutuhkan 4 buah register sebagai buffer dengan panjang masing-masing 32 bit, sehingga total panjang buffer sebesar 128 bit. Keempat register ini diberi nama A, B, C dan D yang diinisialisasi dalam nilai heksa sebagai berikut:
   A = 01 23 45 67
   B = 89 ab cd ef
   C = fe dc ba 98
   D = 76 54 32 10
4. Pengolahan pesan dalam blok bernilai 512 bit.
   Dalam langkah ini, awalnya didefinisikan 4 buah fungsi dengan inputan 3 buah word 32 bit dan menghasilkan 1 buah word 32 bit. Inisialisasi fungsi dinyatahkan dalam nama F, G, H dan I adalah sebagai berikut:
   F (X, Y, Z) = XY or (~X)Z
   G (X,Y,Z) = XZ or Y(~Z)
   H (X, Y, Z) = X xor Y xor Z
   I (X, Y, Z) = Y xor (X or (~Z))

Fungsi-fungsi tersebut masing-masing berisikan 16 kali operasi dasar terhadap inputan yang memanfaatkan suatu nilai elemen T. nilai T adalah sebuah array berisi 64 elemen yang didapat dengan perhitungan tertentu menggunakan fungsi sinus. Pesan yang diinput dibagi menjadi n buah blok berukuran 512 bit. Setiap blok tersebut diproses bersama dengan buffer sehingga menghasilkan output sebesar 128 bit. Dalam proses ini terdiri dari 4 kali proses, masing-masing proses melakukan operasi tiap-tiap fungsi 16 kali banyaknya dan memakai elemen operasi dasar T. Hasil akhir dari

algoritma MD5 ini akan menghasilkan output (message digest) dengan panjang yang tetap sebesar 32 karakter.

Algoritma Hash SHA-1

SHA dikembangkan oleh National Institute of Standards and Technology (NIST) dan dipublikasikan sebagai Federal Information Processing Standards (FIPS 180) pada tahun 1993. Secure Hash Standard (SHS) menspesifikasikan SHA-1 untuk menghitung nilai hash dari sebuah message atau file. SHA-1 memiliki panjang pesan maksimal 264 bits dan memiliki keluaran sebesar 160 bits yang dinamakan message digest atau hash code. Message digest tersebut dapat dimanfaatkan sebagai input untuk Digital Signature Algorithm (DSA), yang digunakan untuk menghasilkan signature untuk memvalidasi pesan tersebut.. Pada saat ditemukan kelemahan pada algoritma SHA-0, berbagai revisi dan perbaikan dilakukan untuk membuat suatu algoritma yang lebih baik. Hasil dari perbaikan tersebut diterbitkan pada tahun 1995 dan dijadikan acuan untuk pembuatan algoritma SHA-1. Algoritma SHA-1 merupakan revisi teknis dari algoritma SHA. Algoritma SHA-1 dapat dikatakan aman karena proses perhitungannya tidak memungkinkan untuk menemukan pesan yang sebenarnya dari message digest yang dihasilkan. Setiap perubahan yang terjadi pada pesan saat perjalanan akan menghasilkan message digest yang berbeda. Algoritma SHA-1 berbasis pada algoritma MD4 dan rancangannya sangatlah mirip terhadap algoritma tersebut.

6.5 Tanda Tangan Digital

Seperti yang tersirat dari namanya, Tanda Tangan Digital adalah jenis tanda tangan, tetapi satu-satunya perbedaan adalah ia melibatkan penggunaan pin atau algoritme matematika untuk menandatangani dan memvalidasi keaslian dokumen, file, atau perangkat lunak, bukan pena dan kertas. Tanda tangan digital digunakan untuk memastikan bahwa file yang dikirim secara digital milik sumber yang ditentukan dan mencapai penerima yang dituju dalam format aslinya tanpa gangguan apa pun.

Secara sederhana, tanda tangan digital bekerja dengan cara yang sama seperti segel amplop. Bayangkan Anda ingin mengirim dokumen yang ditandatangani secara fisik dari satu negara ke negara lain. Anda perlu mengirim dokumen melalui kurir. Proses ini melibatkan banyak dokumen dan dengan demikian membuang-buang waktu yang tak ternilai.

Sebaliknya, jika Anda baru saja menggunakan tanda tangan digital, dokumen tersebut dapat dikirim secara elektronik dalam hitungan menit. Dengan cara ini Anda dapat menghemat waktu dan juga uang. Sejumlah penelitian yang dilakukan di seluruh dunia menunjukkan bahwa menggunakan tanda tangan digital dapat menghemat waktu kerja seminggu penuh bagi profesional yang bekerja. Waktu yang dihemat dikombinasikan dengan penghematan uang yang tak terbantahkan pasti akan mendorong penerimaan tanda tangan digital yang cepat di seluruh dunia.

Cara umum menghitung tanda tangan digital adalah dengan enkripsi kunci publik; penanda tangan menghitung nilai tanda tangan dengan menggunakan kunci pribadi, dan orang lain dapat menggunakan kunci publik untuk memverifikasi bahwa tanda tangan berasal dari kunci pribadi yang sesuai.

Tanda tangan digital harus memenuhi dua persyaratan dan idealnya akan memenuhi dua lagi:

- tidak dapat ditempa (wajib). Tidak ada orang lain selain penandatangan yang dapat menghasilkan tanda tangan tanpa kunci pribadi penandatangan.
- otentik (wajib). Penerima dapat menentukan bahwa tanda tangan itu benar-benar berasal dari penandatangan.
- tidak dapat diubah (diharapkan). Tidak ada penandatangan, penerima, atau pencegat apa pun yang dapat mengubah tanda tangan tanpa terbukti adanya gangguan.
- tidak dapat digunakan kembali (diharapkan). Setiap upaya untuk menggunakan kembali tanda tangan sebelumnya akan dideteksi oleh penerima.

Untuk mendukung tanda tangan digital, kita membutuhkan algoritma kunci publik yang kuat. Algoritma RSA yang dijelaskan sebelumnya dalam bab ini baik untuk tanda tangan digital, tetapi itu bukan satu-satunya kemungkinan.

6.5.1 Kriptografi Kurva Eliptik

Kurva eliptik dapat dipakai dalam kesepakatan kunci, tanda tangan digital, pembangkit bilangan acak semu, dan lain-lain. Secara tak langsung, kurva eliptik dapat dipakai untuk enkripsi dengan menggabungkan kesepakatan kunci dan skema enkripsi simetris. Kurva eliptik juga dipakai dalam beberapa algoritme faktorisasi prima berdasarkan kurva eliptik yang memiliki kegunaan dalam kriptografi, misalnya faktorisasi kurva eliptik Lenstra.

Elliptic Curve Cryptography (ECC) ditemukan pada tahun 1985 oleh Victor Miller (IBM) dan Neil Koblitz (University of Washington) sebagai mekanisme alternatif untuk mengimplementasikan kriptografi kunci publik. Tidak seperti RSA, ECC didasarkan pada logaritma di bidang terbatas; keuntungan dari ECC adalah bahwa keamanan yang setara dapat dimiliki dengan panjang kunci yang lebih pendek daripada RSA.

Masalah dengan RSA sejak awal adalah bahwa pengembangnya mematenkan algoritme. Dengan demikian, pengguna algoritme mungkin diharuskan membayar biaya lisensi. Pengembang algoritma ECC menempatkannya di domain publik, sehingga menghindari lisensi dan biaya. (Pengembang lain telah mematenkan teknologi yang menggabungkan ECC, jadi tidak semua yang melibatkan ECC bebas paten.)

Matematika di balik kriptografi kurva eliptik cukup canggih, lebih dari yang dapat kami sajikan di sini. Kurva eliptik adalah (x,y) koordinat titik-titik yang memenuhi persamaan seperti $y^2 = x^3 + ax + b$ untuk konstanta a dan b. Nick Sullivan menunjukkan bahwa setiap garis lurus nonvertikal melewati paling banyak tiga titik pada kurva. Dan, jika diberikan dua titik, P dan Q, kita dapat menemukan titik ketiga R yang dilalui garis PQR. Ketika $P = (x_P, y_P)$ *dan* $Q = (x_Q, y_Q)$,

$$P + Q = R$$

dimana

$$s = (y_P - y_Q) / (x_P - x_Q)$$

$$x_R = s^2 - x_P - x_Q$$

dan

$$yR = -y_P + s(x_P - x_R)$$

Perhatikan bahwa s adalah kemiringan garis yang melalui P dan Q. Dengan demikian, diberikan P dan Q kita dapat menemukan titik ketiga pada garis PQR secara aljabar. Itu berarti kita juga dapat menemukan titik T berikutnya di QRT, lalu V di RTV, dan seterusnya. Ada juga rumus yang memungkinkan kita memulai dengan satu poin dan mendapatkan yang kedua: satu poin bisa membuat kita mendapatkan yang kedua, dan dua poin membuat kita mendapatkan yang ketiga.

Kriptosistem kurva eliptik menambahkan satu putaran lagi, yang seharusnya familiar dari matematika algoritma RSA: Operasi ECC dilakukan dalam grup berhingga, bilangan bulat mod p untuk beberapa p prima. Jadi, meskipun nilai x dan y meningkat tanpa terikat dalam persamaan dasar ECC, membatasi aritmatika ke bidang terbatas inilah yang membuat masalah kriptografi sulit untuk dibalik. Diberikan titik awal P dan titik akhir Z dan membatasi hasil untuk berada di medan berhingga, pertanyaannya adalah berapa banyak langkah yang diperlukan untuk berpindah dari P ke Z. Secara lebih formal, cari nilai k untuk Pk = Z. Dalam dengan kata lain, cari k, logaritma basis P dari Z. Ternyata cara tercepat yang diketahui untuk menjawab pertanyaan itu adalah dengan memulai dengan P dan menghasilkan semua titik tengah sampai Anda memperoleh Z.

Kriptografi kurva eliptik jarang digunakan sendiri untuk enkripsi kunci publik. Namun, sering digunakan sebagai komponen dalam tanda tangan digital. Pada tahun 2005 NSA mempresentasikan strategi dan rekomendasinya untuk mengamankan komunikasi sensitif dan tidak rahasia pemerintah AS. Strategi tersebut mencakup serangkaian algoritme kriptografi canggih yang direkomendasikan yang dikenal sebagai Suite B. Protokol yang termasuk dalam Suite B adalah Elliptic Curve Diffie-Hellman (ECDH) dan Elliptic Curve Menezes-Qu-Vanstone (ECMQV) untuk pertukaran kunci dan persetujuan; Elliptic Curve Digital Signature Algorithm (ECDSA) untuk tanda tangan digital; Advanced Encryption Standard (AES) untuk enkripsi simetris; dan Secure Hashing Algorithm (SHA). Apa yang menarik bagi NSA tentang ECC adalah keamanan, efisiensi, dan skalabilitasnya yang kuat di atas algoritme kriptografi kunci publik.

6.5.2 Algoritma Tanda Tangan Digital El Gamal

Algoritma kunci publik lainnya dirancang pada tahun 1984 oleh Taher El Gamal. Meskipun algoritme ini tidak banyak digunakan secara langsung, ini sangat penting dalam Standar Tanda Tangan Digital AS (DSS). Algoritma ini bergantung pada kesulitan menghitung logaritma diskrit atas bidang yang terbatas. Karena didasarkan pada aritmatika di bidang terbatas, seperti RSA, ia memiliki beberapa kesamaan dengan RSA.

Dalam algoritma El Gamal, untuk membangkitkan pasangan kunci, pertama-tama pilihlah p prima dan dua bilangan bulat, a dan x, sedemikian hingga a < p dan x < p dan hitung y = ax mod p. Prima p harus dipilih sehingga (p - 1) memiliki faktor prima yang besar, q. Kunci pribadi adalah x dan kunci publik adalah y, bersama dengan parameter p dan a.

Untuk menandatangani pesan m, pilih bilangan bulat acak k, 0 < k < p - 1, yang belum pernah digunakan sebelumnya dan yang relatif prima untuk (p - 1), dan hitung

$$r = ak \bmod p$$

dan

$$s = k–1 (m – xr) \bmod (p – 1)$$

di mana k-1 adalah invers perkalian dari k mod (p - 1), sehingga k * k-1 = 1 mod (p - 1). Tanda tangan pesan kemudian r dan s. Penerima dapat menggunakan kunci publik y untuk menghitung tahun rs mod p dan menentukan bahwa itu setara dengan am mod p. Untuk mengalahkan enkripsi ini dan menyimpulkan nilai x dan k yang diberikan r, s, dan m, penyusup harus menemukan cara menghitung logaritma diskrit untuk menyelesaikan y = ax dan r = ak.

Algoritma Tanda Tangan Digital

Digital Signature Algorithm (DSA) (juga disebut Digital Signature Standard atau DSS) [NIS94] adalah algoritme El Gamal dengan beberapa batasan. Pertama, ukuran p secara khusus ditetapkan pada 2511 < p < 2512 (sehingga p panjangnya kira-kira 170 digit desimal). Kedua, q, faktor prima yang besar dari (p - 1) dipilih sehingga 2159 < q < 2160. Algoritme secara eksplisit menggunakan H(m), nilai hash, alih-alih teks pesan lengkap m. Akhirnya, perhitungan r dan s diambil mod q. Salah satu interpretasinya adalah bahwa perubahan ini membuat algoritme mudah digunakan untuk orang yang tidak ingin atau perlu memahami matematika yang mendasarinya. Namun, perubahan tersebut juga melemahkan potensi kekuatan enkripsi dengan mengurangi ketidakpastian bagi penyerang.

Standar Tanda Tangan Digital AS

Setelah merancang algoritme tanda tangan digital DSA, pemerintah AS menetapkan standar untuk penggunaan algoritme itu untuk membuat tanda tangan digital. Namun, pemerintah agak terlambat untuk permainan: Industri swasta telah berkumpul di sekitar pendekatan tanda tangan digital berdasarkan enkripsi RSA, dan komite standar untuk komunitas perbankan telah menetapkan metode ketiga menggunakan ECC. Dengan demikian, Publikasi FIPS 186-3, Digital Signature Standard [NIS09] mencakup dan menyetujui penggunaan ketiga metode tersebut oleh pemerintah. Standar baru, versi 186-4, diterbitkan pada Juli 2013.

Pengaturan Tanda Tangan Digital di Indonesia

Mengenai keberadaan “tanda tangan digital/elektronik” di Indonesia, menurut Pasal 1 angka 12 Undang-Undang Nomor 19 Tahun 2016 tentang Perubahan atas Undang-Undang Nomor 11 Tahun 2008 tentang Informasi dan Transaksi Elektronik (“UU 19/2016”) istilah tersebut didefinisikan sebagai berikut:

Tanda Tangan Elektronik adalah tanda tangan yang terdiri atas Informasi Elektronik yang dilekatkan, terasosiasi atau terkait dengan Informasi Elektronik lainnya yang digunakan sebagai alat verifikasi dan autentikasi. Tanda tangan elektronik meliputi:

1. Tanda tangan elektronik tersertifikasi, yang harus memenuhi persyaratan:
   a. memenuhi keabsahan kekuatan hukum dan akibat hukum tanda tangan elektronik sebagaimana dimaksud dalam Pasal 59 ayat (3) PP PSTE;

b. menggunakan sertifikat elektronik yang dibuat oleh jasa penyelenggara sertifikasi elektronik Indonesia; dan
c. dibuat dengan menggunakan perangkat pembuat tanda tangan elektronik tersertifikasi. .

2. Tanda tangan elektronik tidak tersertifikasi, yang dibuat tanpa menggunakan jasa penyelenggara sertifikasi elektronik.

Tanda tangan elektronik berfungsi sebagai alat autentikasi dan verifikasi atas:
a. identitas penanda tangan; dan
b. keutuhan dan keautentikan informasi elektronik.

Persetujuan penanda tangan terhadap informasi elektronik yang akan ditandatangani dengan tanda tangan elektronik harus menggunakan mekanisme afirmasi dan/atau mekanisme lain yang memperlihatkan maksud dan tujuan penanda tangan untuk terikat dalam suatu transaksi elektronik

Jadi tanda tangan elektronik tersebut lazimnya dilakukan pada transaksi elektronik. Transaksi elektronik adalah perbuatan hukum yang dilakukan dengan menggunakan komputer, jaringan komputer, dan/atau media elektronik lainnya.

Mengenai keabsahannya, tanda tangan elektronik, memiliki kekuatan hukum dan akibat hukum yang sah selama memenuhi persyaratan sebagai berikut:

a. data pembuatan tanda tangan elektronik terkait hanya kepada penanda tangan;
b. data pembuatan tanda tangan elektronik pada saat proses penandatanganan elektronik hanya berada dalam kuasa penanda tangan;
c. segala perubahan terhadap tanda tangan elektronik yang terjadi setelah waktu penandatanganan dapat diketahui;
d. segala perubahan terhadap informasi elektronik yang terkait dengan tanda tangan elektronik tersebut setelah waktu penandatanganan dapat diketahui;
e. terdapat cara tertentu yang dipakai untuk mengidentifikasi siapa penanda tangannya; dan
f. terdapat cara tertentu untuk menunjukkan bahwa penanda tangan telah memberikan persetujuan terhadap informasi elektronik yang terkait.

Persyaratan tersebut merupakan persyaratan minimum yang harus dipenuhi dalam setiap tanda tangan elektronik. Ketentuan ini membuka kesempatan seluas-luasnya kepada siapa pun untuk mengembangkan metode, teknik, atau proses pembuatan tanda tangan elektronik.

UU ITE dan perubahannya sendiri telah memberikan pengakuan secara tegas bahwa meskipun hanya merupakan suatu kode, tanda tangan elektronik memiliki kedudukan yang sama dengan tanda tangan manual pada umumnya yang memiliki kekuatan hukum dan akibat hukum.
Jadi berdasarkan penjelasan di atas, suatu tanda tangan elektronik dapat dikatakan sah apabila memenuhi ketentuan sebagaimana dijelaskan dalam Pasal 11 UU ITE dan Pasal 59 PP PSTE.
Sumber : https://www.hukumonline.com

6.6 Kriptografi Kuantum

Sekarang kita beralih ke bentuk kriptografi yang muncul berdasarkan kemajuan dalam fisika. Beberapa analis mengatakan kriptografi ini memiliki potensi untuk merevolusi enkripsi dan kriptanalisis, meskipun yang lain skeptis itu akan menjadi praktis segera.

Kita telah melihat bagaimana para peneliti mengandalkan aspek matematika untuk menghasilkan masalah yang sulit dan untuk merancang algoritma. Pada bagian ini, kita melihat pandangan alternatif tentang bagaimana kriptografi dapat dilakukan di masa depan. Pendekatan yang kami uraikan saat ini tidak ada di pasar, juga tidak mungkin dalam beberapa tahun ke depan. Tapi itu menggambarkan perlunya pemikiran kreatif dalam menciptakan teknik enkripsi baru. Meskipun ilmu di balik pendekatan ini sulit, pendekatan itu sendiri sebenarnya cukup sederhana.

Pendekatan baru, kriptografi kuantum, adalah varian dari ide di balik papan satu kali. Ingat dari awal bab ini bahwa one-time pad adalah satu-satunya skema enkripsi yang terbukti tidak dapat dipecahkan. Papan satu kali membutuhkan dua salinan dari serangkaian panjang angka yang tidak dapat diprediksi, masing-masing satu salinan untuk pengirim dan penerima. Pengirim menggabungkan angka dengan unit plaintext untuk menghasilkan ciphertext. Jika angka benar-benar tidak dapat diprediksi (yaitu, mereka sama sekali tidak memiliki pola yang terlihat), penyerang tidak dapat memisahkan angka dari ciphertext.

Kesulitan dengan pendekatan ini adalah bahwa hanya ada sedikit sumber string angka acak yang dapat dibagikan. Ada banyak fenomena alam yang dapat menghasilkan serangkaian angka yang tidak dapat diprediksi, tetapi kemudian kita menghadapi masalah mengkomunikasikan string itu ke penerima sedemikian rupa sehingga pencegat tidak dapat memperolehnya. Kriptografi kuantum membahas kedua masalah, menghasilkan dan mengkomunikasikan angka. Kasus 6-1 menjelaskan betapa pentingnya kriptografi kuantum.

Studi Kasus 6-1 : Memperkenalkan Kriptografi Kuantum: Apa Kapan dan Bagaimana

Pada saat kemajuan teknologi telah menciptakan suatu keadaan proliferasi data yang hampir konstan, kebutuhan akan transmisi informasi sensitif yang aman tidak pernah lebih penting. Para pengambil keputusan TI terbukti jelas telah mengenali tantangan ini, dengan penelitian terbaru dari Toshiba mengungkapkan bahwa, untuk lebih dari setengah (52%) bisnis di Eropa, keamanan data adalah prioritas investasi tiga besar untuk tahun mendatang. Namun terlepas dari ini, bisnis sering ketinggalan dalam membuat mereka sadar dan berada di depan tren dan perkembangan keamanan cyber.

Masuklah kriptografi kuantum, yang, dengan memanfaatkan prinsip-prinsip fisika kuantum, memiliki kemampuan untuk mengantarkan era baru komunikasi online yang aman. Tetapi apa sebenarnya kriptografi kuantum, masalah-masalah apa saja yang dipecahkannya, dan bagaimana hal itu dapat mengisi celah dalam pertahanan online

agar memungkinkan bisnis untuk tetap selangkah lebih maju dari setiap ancaman rumit, sekarang dan di masa depan?

Bagaimana cara kerja kriptografi kuantum?

"Sederhananya, kriptografi kuantum menyediakan cara yang aman untuk menghasilkan dan mendistribusikan kunci rahasia antara dua pihak pada jaringan optik," kata Dr. Andrew Shields, Asisten Direktur Pelaksana di Laboratorium Penelitian Toshiba di Cambridge. “Dengan memanfaatkan kondisi inheren dari partikel yang tidak bisa dipastikan, seperti elektron atau foton, kriptografi kuantum dapat digunakan untuk menghasilkan angka acak yang diperlukan bagi aplikasi kriptografi. Lebih jauh lagi, dengan mengirim aliran foton tunggal yang disandikan melalui jaringan komunikasi optik, maka dimungkinkan untuk berbagi kunci digital rahasia yang dapat digunakan dalam mengenkripsi atau mengotentikasi informasi."

Mengapa kita membutuhkannya?

Saat ini secara luas dianggap bahwa enkripsi kunci publik adalah bagian penting dari keamanan data, tetapi hal tersebut ditantang oleh strategi serangan baru dan munculnya komputer kuantum yang pada akhirnya akan membuat kebanyakan enkripsi saat ini menjadi tidak aman. Tantangan keamanan hari ini dan kekhawatiran akan keamanan di masa depan mendorong penerapan solusi dan layanan kriptografi kuantum yang andal yang memungkinkan adanya keamanan data yang lebih baik. Hasilnya, pasar kriptografi kuantum global diperkirakan akan tumbuh dari USD 285,7 Juta pada 2017 menjadi USD 943,7 Juta pada tahun 2022, 27% dari CAGR.

Namun kedatangan kriptografi kuantum tidak segera dapat dipetik hasilnya atau diakui secara luas dalam bidang yang relevan. Hal ini tidak mengurangi nilai esensial dan tak tertandingi pada saat kita memasuki zaman kuantum, jadi seberapa jauhkah kita dengan kriptografi kuantum, dan apa yang perlu dikembangkan untuk mencapai hal tersebut?

Kapan akan menjadi arus utama?

Meskipun belum tersedia secara komersial, para ilmuwan sekarang berada pada tahap mampu untuk mempergunakan teknologi tersebut dan menunjukkan manfaatnya. Toshiba's Cambridge Research Laboratory baru-baru ini menerbitkan sebuah makalah yang menjelaskan terobosan yang dibuat menggunakan protokol yang dikenal sebagai Twin-Field QKD, memperluas jangkauan QKD hingga lebih dari 500 kilometer dari serat telekomunikasi standar.

“Ini membuka potensi komunikasi yang aman antara kota-kota seperti London, Paris, Dublin, Manchester dan Amsterdam. Selain itu, sejumlah proyek kolaboratif besar seperti proyek Innovate UK EQUIP dan program Horizon 2020 milik Komisi UE juga bekerja untuk mengembangkan teknologi tersebut dan menjadikan QKD alat yang dapat diakses dan bernilai bagi perusahaan, ”kata Dr. Shields.

Di luar keinginan para pembuat keputusan TI dalam menjaga data mereka tetap aman saat dalam perjalanan dari A ke B, demikian juga perubahan dari undangundang

dalam menjadikan penyimpanan identitas seseorang dengan aman menjadi hukum, dengan Peraturan Perlindungan Data Umum (GDPR) sebuah contoh utama dari ini. Ditambah dengan berkembang dan terversifikasinya ancaman cyber serta kesulitan yang terkait dengan keamanan data, serta perlunya standar baru dalam melindungi informasi tersebut, QKD dapat menjadi alat penting dalam memastikan data disimpan dengan aman dan terjaga.

Merencanakan masa depan

Kriptografi kuantum memiliki potensi besar untuk menjadi teknologi kunci dalam melindungi infrastruktur komunikasi dari serangan dunia maya dan menempatkan bisnis di garis depan dalam hal melindungi informasi operasi yang penting. Tidak seperti solusi keamanan lain yang ada, kriptografi kuantum aman dari semua kemajuan di masa depan dalam matematika dan komputasi, bahkan dari kemampuan menghitung angka dari komputer kuantum. Standarisasi protokol QKD tetap penting bagi kriptografi kuantum komersial. Kebutuhan dalam memungkinkan interoperabilitas teknologi, mengembangkan pasar komponen serta proses dan teknologi, akan mengurangi biaya pembuatan dan penyebaran QKD dan melihat dimulainya masa depan yang lebih aman bagi kita semua. *Sumber https://m.merdekanews.co/ Di akses Juli.2021*

6.6.1 Fisika Kuantum

Tidak seperti pendekatan kriptografi lainnya, kriptografi kuantum didasarkan pada fisika, bukan matematika. Ini menggunakan apa yang kita ketahui tentang perilaku partikel cahaya. Partikel cahaya dikenal sebagai foton. Mereka melakukan perjalanan melalui ruang, bergetar ke segala arah; kami mengatakan mereka memiliki orientasi arah dari getaran utama mereka. Meskipun foton dapat memiliki orientasi arah dari 0° hingga 360°, untuk tujuan penjelasan ini, kita dapat mengasumsikan hanya ada empat orientasi arah (dengan membulatkan setiap orientasi aktual ke 90° terdekat). Kita dapat menyatakan empat orientasi ini dengan empat simbol, ↔, ↕ , ⤢ , dan ⤡ . Hal ini dimungkinkan untuk membedakan antara sebuah tanda ↔ dan foton ↕ dengan kepastian yang tinggi. Namun, foton ⤢ dan foton ⤡ terkadang muncul sebagai ↔ atau ↕. Demikian pula, dimungkinkan untuk membedakan antara foton ⤢ dan ⤡, tetapi terkadang ↔ dan ↕ akan dikenali sebagai foton ⤢ atau ⤡. Untungnya, kekurangan tersebut sebenarnya memberikan beberapa kebingungan dari algoritma kriptografi.

Filter polarisasi adalah perangkat atau prosedur yang menerima foton apa pun sebagai input tetapi hanya menghasilkan jenis foton tertentu sebagai output. Ada dua jenis filter foton: **+** dan **×**. Sebuah filter + dengan benar membedakan antara foton ↔ dan ↕ tetapi memiliki peluang 50 persen untuk juga menghitung sebuah ⤢ atau ⤡ sebagai ↔ atau sebaliknya sebagai foton ↕, sebuah filter × membedakan antara ⤢ dan ⤡ tetapi juga dapat menerima setengah dari foton ↔ dan foton ↕. Pikirkan sebuah filter + sebagai celah horizontal sempit di mana foton ↔ dapat meluncur dengan mudah, tetapi sebuah foton ↕ akan selalu terhalang. Kadang-kadang (mungkin separuh waktu), sebuah foton ⤢ atau foton ⤡ bergetar dengan cara menyelinap melalui celah juga. Sebuah filter × dianalogikan seperti celah vertikal.

6.6.2 Penerimaan Foton

Kriptografi kuantum beroperasi dengan mengirimkan aliran foton dari pengirim ke penerima. Pengirim menggunakan salah satu filter polarisasi untuk mengontrol jenis foton yang dikirim. Penerima menggunakan salah satu filter dan mencatat orientasi foton yang diterima. Tidak masalah jika penerima memilih filter yang sama dengan yang dilakukan pengirim; yang penting adalah apakah si penerima kebetulan memilih tipe yang sama seperti yang dilakukan si pengirim.

Properti terpenting dari kriptografi kuantum adalah tidak ada yang bisa menguping komunikasi tanpa memengaruhi komunikasi. Dengan sedikit pengkodean deteksi kesalahan sederhana, pengirim dan penerima dapat dengan mudah menentukan keberadaan penyadap. Prinsip ketidakpastian Heisenberg mengatakan bahwa kita tidak dapat mengetahui kecepatan dan lokasi partikel pada waktu tertentu; setelah kita mengukur kecepatan, lokasi telah berubah, dan setelah kita mengukur lokasi, kecepatan telah berubah. Karena prinsip ini, ketika kita mengukur properti apa pun dari partikel, itu memengaruhi properti lainnya. Jadi, misalnya, mengukur orientasi foton mempengaruhi foton. Filter celah horizontal memblokir semua dan setengah foton, sehingga mempengaruhi aliran foton yang masuk. Pengirim tahu apa yang dikirim, penerima tahu apa yang diterima, tetapi penyadap akan mengubah aliran foton secara dramatis sehingga pengirim dan penerima dapat dengan mudah menentukan seseorang sedang mendengarkan.

Mari kita lihat bagaimana pendekatan yang tidak biasa ini dapat digunakan untuk kriptografi.

6.6.3 Kriptografi Dengan Foton

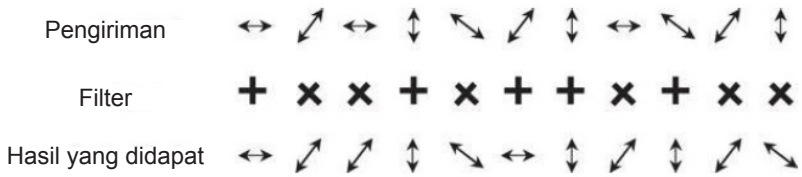
Algoritma kriptografi tidak efisien, karena lebih dari dua kali bit yang ditransmisikan tidak digunakan dalam kriptografi. Bit yang ditransmisikan adalah foton yang, untungnya, cukup tersedia.

Misalkan pengirim, Doni, menghasilkan serangkaian foton, mengingat orientasi mereka. Doni dan penerimanya, Elsa, menyebut ↔ atau ⤢ 0 dan ↕ atau ⤡ 1. Rangkaian seperti itu ditunjukkan pada Gambar 12-9.



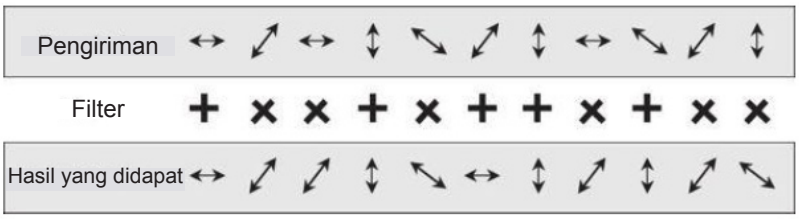
Gambar 6-9 Transmisi Foton

Sekarang, Elsa menggunakan salah satu filter polarisasinya, + dan × secara acak, merekam hasilnya. Ingatlah bahwa filter + akan secara akurat membedakan antara sebuah foton ↔ dan ↕, tetapi terkadang juga mendeklarasikan foton ⤢ atau ⤡ sebagai ↔. Jadi Elsa tidak tahu apakah hasil yang dia ukur adalah yang dikirim Doni. Pilihan filter Elsa, dan hasil yang diperolehnya, ditunjukkan pada Gambar 6-10.



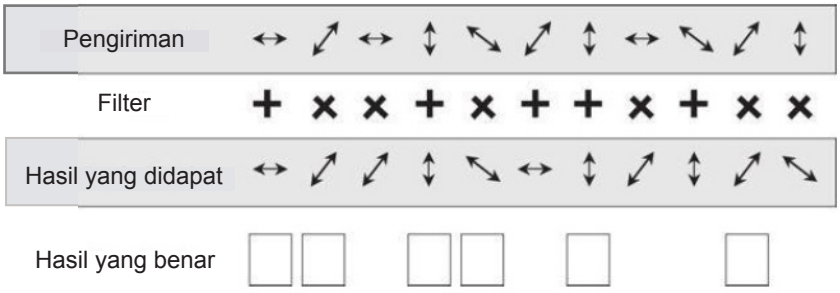
Gambar 6-10 Hasil Diinterpretasikan melalui Filter

Beberapa dari hasil tersebut benar dan beberapa salah, tergantung pada filter yang dipilih Elsa. Sekarang Elsa mengirimkan kepada Doni jenis filter yang dia gunakan, seperti yang ditunjukkan pada Gambar 6-11.

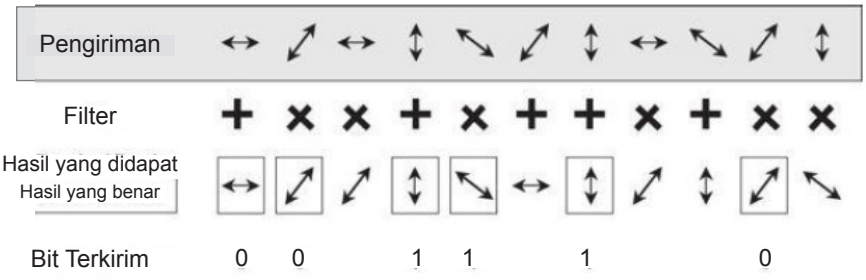


Gambar 6-11 Filter Digunakan

Doni memberitahu Elsa filter mana yang benar yang dia gunakan, seperti yang ditunjukkan pada Gambar 6-12, dari mana Elsa dapat menentukan mana dari hasil yang diperoleh yang benar, seperti yang ditunjukkan pada Gambar 6-13. Dalam contoh ini, Elsa kebetulan memilih filter yang tepat enam kali dari sepuluh, sedikit lebih tinggi dari yang diharapkan, sehingga enam dari sepuluh foton yang ditransmisikan diterima dengan benar. Mengingat bahwa ↔ atau ⤢ berarti 0 dan ↕ atau ⤡ berarti 1, Elsa dapat mengubah foton-foton tersebut menjadi bit, seperti yang ditunjukkan pada gambar. Secara umum, hanya setengah dari foton yang ditransmisikan yang akan diterima dengan benar, sehingga hanya setengah dari bandwidth saluran komunikasi ini yang membawa data yang berarti.



Gambar 6-12 Filter yang Benar



Gambar 6-13 Hasil yang Benar

Perhatikan bahwa Elsa dapat memberi tahu Doni filter mana yang dia gunakan dan Doni dapat memberi tahu Elsa filter mana yang akan memberikan hasil yang benar tanpa mengungkapkan apa pun tentang bit aktual yang dikirimkan. Dengan cara ini, Doni dan Elsa dapat berbicara tentang transmisi mereka tanpa seorang penyadap mengetahui apa yang sebenarnya mereka bagikan.

6.6.4 Penerapan

Teori kriptografi kuantum solid, tetapi beberapa kesulitan teknis masih harus diselesaikan sebelum skema tersebut dapat dipraktikkan. Untuk menerapkan kriptografi kuantum, kita membutuhkan sumber foton secara acak tetapi berorientasi terdeteksi (untuk pengirim) dan sarana untuk menyaring foton yang diterima dengan andal. Pistol foton dapat menembakkan foton sesuai permintaan. Beberapa tim peneliti yang berbeda bekerja untuk mengembangkan senjata foton untuk kriptografi, tetapi sejauh ini tidak ada yang berhasil. Teknologi terbaik saat ini melibatkan laser berdenyut, tetapi di sini juga ada masalah. Kadang-kadang laser memancarkan tidak hanya satu tetapi dua foton, yang mengganggu pola penerimaan dan transmisi. Namun, dengan kode koreksi kesalahan pada aliran bit, relatif mudah untuk mendeteksi dan memperbaiki beberapa bit yang salah.

Di pihak penerima juga, ada masalah. Satu perangkat tunduk pada kegagalan bencana di mana ia memancarkan lonjakan arus. Meskipun lonjakan ini mudah dideteksi, namun perangkat harus disetel ulang, yang membutuhkan waktu.

Implementasi eksperimental kriptografi kuantum masih di laboratorium. Badan Evaluasi dan Penelitian Pertahanan Inggris di Malvern, Inggris, mendemonstrasikan komunikasi yang berhasil melalui atmosfer pada jarak 2 km, dan Laboratorium Nasional Los Alamos AS sedang menguji perangkat portabel yang dapat beroperasi lebih dari 45 km pada malam yang cerah. Institut Nasional untuk Standar dan Teknologi AS telah mendemonstrasikan sistem kriptografi kuantum yang beroperasi lebih dari satu kilometer di atas serat kaca dengan kecepatan empat megabit per detik. BBN Communications dan Universitas Harvard telah mengaktifkan jaringan bersama yang diamankan dengan enkripsi kuantum. Jaringan ini memiliki enam server dan mencakup jarak sepuluh kilometer. Komunikasi yang andal hingga 20 kilometer telah dicapai, dan beberapa ilmuwan berharap dapat menempuh jarak 50 kilometer dengan andal dalam waktu dekat.

Hasil ini menunjukkan kemajuan yang signifikan saat kriptografi kuantum bergerak dari bangku penelitian ke laboratorium prototipe. Meskipun masih belum siap untuk adopsi publik secara luas, kriptografi kuantum menjadi kemungkinan nyata untuk penggunaan komersial dalam dekade berikutnya.

6.7 Kesimpulan

Seperti yang kami katakan di awal bab ini, kriptografi adalah bidang studi yang sangat terspesialisasi. Bahkan dalam bab detail ini, kami hanya menyajikan lapisan teratas dari bidang yang kaya ini.

Berbeda dengan bab-bab lain dalam buku ini, bab ini tidak menyertakan latihan. Masalah dalam pemecahan kode dengan cepat menjadi sulit. Latihan yang melibatkan perhitungan tangan dari salah satu sandi dalam bab ini melelahkan dan membosankan; nilai pendidikan dari masalah tersebut masih bisa diperdebatkan. Dan pertanyaan yang meminta Anda untuk menemukan kekurangan dalam satu algoritme atau menemukan algoritme baru untuk melakukan sesuatu adalah subjek makalah pascasarjana, bukan latihan dalam kursus. Kami tahu pembaca kami dapat menjawab pertanyaan tentang menghafal, dan apa pun selain itu terlalu banyak pekerjaan.

Pembaca yang ingin mempelajari lebih lanjut tentang kriptografi harus mempelajari buku-buku yang lebih maju tentang topik tersebut, mengingat banyak yang membutuhkan latar belakang matematika yang kuat. Dan untuk sejarah bidang yang sangat menarik ini, kami merekomendasikan ensiklopedis David Kahn "*The Code Breakers*".

Daftar Pustaka

Ahmad, Yani. “Jurus Ampuh Membasmi Virus Komputer”. Ruang Kata. Jakarta.2009.

Agrawal, R., and Srikant, R. “Privacy-Preserving Data Mining.” Proc ACM SIGMOD Conf on Management of Data, May 2000.

Akerlof, George A. 1970. “The Market for ‘Lemons’: Quality Uncertainty and the Market Mechanism.” The Quarterly Journal of Economics 84 (3): 488–500.

Albrecht, M., et al. “Plaintext Recovery Attacks Against SSH.” Proc 2009 IEEE Symp Security and Privacy, 2009, p16–26

Amin, Ziad. 2012. Analisis Vulnerabilitas Host Pada Keamanan Jaringan Komputer di PT. Sumeks Tivi Palembang (PALTV) Menggunakan Router Berbasis UNIX. Jurnal Teknologi dan Informatika. Vol. 2, No. 3. Anjik, Sukmaaji. “Jaringan Komputer”. Andi. Yogyakarta. 2008

Anderson, Jonathan, Joseph Bonneau, and Frank Stajano. 2010. “Inglorious Installers: Security in the Application Marketplace.” Paper presented at the Ninth Workshop on the Economics of Information Security (WEIS 2010), June 7-8, Harvard University. http://www.econinfosec.org/archive/weis2010/papers/session3 /weis2010_anderson_j.pdf.

Bonneau, Joseph, Cormac Herley, Paul C Van Oorschot, and Frank Stajano. 2012. “The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes.” In IEEE Symposium on Security and Privacy (SP) 2012, 553–67.doi:10.1109/SP.2012.44

Burleson, W., et al. “Design Challenges for Secure Implantable Medical Devices.” Procl EEE/ACM Design Automation Conf,2012.

Cheswick, W. “An Evening with Berferd, in Which a Cracker Is Lured, Endured, and Studied.” Proc Winter USENIX Conf, Jun 1990.

Comer, Douglas E. (2008). Computer Network and Internet 5th Edition. United States of America: IGI Global

Computer Security Institute. (2009). Computer Crime and Security Survey. New York

Ditkaminfo, Ditjen Aptika, KEMKOMINFO. (2011). Panduan Keamanan Webserver. Jakarta. http://publikasi.kominfo.go.id/handle/54323613/120

Fadia, Ankit. 2006. An Unofficial Guide to Ethical Hacking. India: Thomson Learning

Kurniawan, Yusuf. Kriptografi Keamanan Internet dan Jaringan Komunikasi, Bandung, Informatika, 2005.

Pressman, R. S. (2012). Rekayasa Perangkat Lunak. Yogyakarta: Penerbit Andi.

Rafiudin, Rahmat. Menguasai Security Unix. Jakarta, Elex Media Komputindo, 2007

Rafiudin, Rahmat. (2009). Investigasi Sumber-Sumber Kejahatan Internet. Yogyakarta: Andi.

Ramesh, S. (2014). How to Hack : Hacking Secrets Exposed.howtohack.gohacking.com.

Rani, P. U., Rao, D. Vara P. (2011). A Novel Implementation of ARM based Design Firewall to prevent SYN Flood Attack. International Journal of Computer Trends and Technology. Volume 2, Issue 2. 63-70

Richardson, R. (2008). CSI Computer Crime & Security Survey 2008.

Rohman, Abdul. (2010). Membangun DNS Server Dan Web Server Dengan Debian Linux. 1-24

Scarfone, K. A. and Hoffman, P. (2009). Guidelines on Firewalls and Firewall Policy. United State of America: National Institute of Standards and Technology. 1-48

Sidik, B. (2004). Unix dan Linux : Panduang Bekerja Dalam Lingkungan Unix dan Linux. Bandung: Informatika.

Simarmata, Janner. Pengamanan Sistem Komputer. Yogyakarta, Andi, 2006.

Sopandi, Dede. Instalasi dan Konfigurasi Jaringan Komputer. Bandung, Informatika, 2006

Sutedjo Dharma Oetomo, Budi. Konsep dan Perancangan Jaringan Komputer. Yogyakarta, Andi, 2003\

Tanenbaum, Andrew S., and Wetherall, D. J. (2011). Computer Networks 5th Edition. United States of America: Pearson Education

Truelove , James. LAN Writing An Illustrated Networking Cabling Guide, McGraw-Hill, 2000

Weippl, Edgar R. (2005). Security in E-Learning-An Abstraction Based Approach. United States of America: Springer

W, T. W., & Sanjaya, A. (2008). Jurnal Artificial. ICT Research Center UNAS. Studi Keamanan Komputer, 1-8.