MODUL PERKULIAHAN

EDP Audit

Enkripsi dan Kriptografi

(*Encryption and Cryptography*)

Abstract

Modul ini berisi tentang pembahagasn sarana utama atas perlindungan aset informasi. Jika diterapkan dengan benar, enkripsi akan menggagalkan hampir setiap serangan singkat dari upaya yang disponsori secara nasional. Enkripsi dapat digunakan untuk melindungi aset informasi, apakah disimpan pada tape atau disk, atau saat transit pada link komunikasi

Kompetensi

Mahasiswa mampu pememahami dan mengenali tetantang enkripsi dan kriptografi dan bagaimana penerapannya dalam kegiatan pengamanan sebuah informasi dan data.

Pengantar

Enkripsi adalah sarana utama atas perlindungan aset informasi. Jika diterapkan dengan benar, enkripsi akan menggagalkan hampir setiap serangan singkat dari upaya yang disponsori secara nasional. Enkripsi dapat digunakan untuk melindungi aset informasi, apakah disimpan pada tape atau disk, atau saat transit pada link komunikasi.[1]

Definisi dan Gambaran

Sebelum tahun 1990-an, pemerintah nasional, kontraktor pemerintah, dan sistem perbankan swasta merupakan pengguna utama dari teknologi enkripsi. Dengan perkembangan internet dan perdagangan elektronik, namun, kebutuhan untuk pertukaran yang aman dari informasi elektronik kini juga menjadi suatu yang sangat penting bagi entitas komersial dan masyarakat konsumen secara umum. Tampaknya ada pemahaman global bahwa kriptografi adalah cara terkuat untuk mengamankan informasi elektronik terhadap pencurian atau kompromi. Namun, kriptografi dapat menjadi sekutu dan musuh dari amannya pertukaran informasi elektronik. Di satu sisi, teknologi enkripsi dapat melindungi informasi dari tampilan atau serangan yang tidak sah. Di sisi lain, orang-orang yang tidak jujur atau licik dapat menggunakan teknik kriptanalisis untuk membocorkan, mengubah, mencuri, mengalihkan, atau sebaliknya mengganggu pertukaran informasi elektronik. Pembahasan berikut ini memberikan serangkaian referensi dan kutipan yang membantu dalam perspektif kebutuhan untuk pengembangan teknik enkripsi yang kuat.

Pada tahun 1997, Ian Goldberg, seorang mahasiswa pascasarjana Universitas California-Berkeley, menghubungkan 250 komputer yang menganggur satu sama lain dengan cara yang memungkinkan dia untuk menguji 100 miliar kemungkinan kunci per jam. Dengan menggunakan metode ini, ia mampu memecahkan 40-bit algoritma enkripsi RSA Data Security, Inc dalam tiga setengah jam.[2]

Sistem komputer Departemen Pertahanan (DOD) di Amerika Serikat mungkin telah mengalami sebanyak 250.000 serangan hacker pada tahun 1995, menurut laporan dari Kantor Akuntansi Umum Amerika Serikat (GAO). Serangan tersebut sering berhasil, memberikan akses orang tak dikenal dan tidak sah ke informasi yang sangat sensitif, dan mereka dua kali lipan jumlahnya setiap tahun karena lebih mudah dan lebih luasnya penggunaan internet dan meningkatnya kecanggihan hacker komputer.

Menurut GAO, DOD tidak memiliki kebijakan yang seragam untuk menilai risiko, melindungi sistem, menanggapi insiden, atau menilai kerusakan. Selain itu, pelatihan pengguna dan administrator sistem dan jaringan serampangan dan kendala dengan sumber daya yang terbatas. Solusi teknis akan membantu, tapi keberhasilan mereka tergantung pada apakah DOD menerapkannya bersamaan dengan kebijakan dan langkah-langkah personil yang lebih baik.[3]

Hacker komputer Belanda mencuri rahasia militer Amerika Serikat selama Perang Teluk Persia dan menawarkannya ke Irak. Rahasia tersebut bisa mengubah jalannya perang. Tapi Irak diduga tidak pernah menggunakan informasi tersebut, takut sebuah tipuan.[4]

Internet telah membuat kemungkinan untuk mengumpulkan sumber-sumber komputasi besar dalam memecahkan sebuah kunci. Pada tahun 1994, sebuah kunci RSA 129-digit rusak melalui usaha gabungan dari 1.600 komputer di seluruh dunia. Serangan tersebut, yang dikoordinasikan melalui email dan terlibat pencarian faktor prima dari angka 129 digit, menghabiskan 5.000 MIPS dalam sebulan selama selang waktu delapan bulan dari waktu sesungguhnya.[5] (MIPS adalah singkatan dari instruksi mesin per detik).

Bersembunyi di balik keyboard anonim, sekelompok hacker berjuang selama dua minggu untuk menerobos militer Amerika Serikat dan jaringan komputer sipil. Mereka berhasil melampaui impian terliar mereka. . . Orang jahat tersebut [adalah] tim khusus keamanan nasional Amerika Serikat yang diam-diam menguji kerentanan sistem komputer bangsanya menggunakan perangkat lunak yang ditemukan di Internet. . . Hacker [tersebut] memperoleh akses ke sistem komputer di seluruh negeri. . . , termasuk Komando Pasifik Amerika Serikat di Hawaii. [Mereka juga] memperoleh akses ke sistem jaringan tenaga listrik Amerika Serikat dimana mereka dapat menyabotase untuk menenggelamkan bangsanya ke dalam kegelapan.[6]

Pada bulan Agustus 1999, tim ilmuwan internasional di Belanda mampu menentukan faktor prima dari angka 512-bit yang membuat model kunci dalam algoritma kriptografi RSA-155 yang terkenal yang digunakan secara luas dalam perangkat keras dan perangkat lunak untuk melindungi lalu lintas data elektronik (misalnya, versi internasional dari lapisan socket yang aman [SSL]). RSA-155 dirancang oleh tiga ilmuwan (Ronald Rivest, Adi Shamir, Leonard Adelman) di Institut Teknologi Massachusetts pada pertengahan tahun 1970-an. RSA-155 memiliki dua bagian, yaitu langkah penyaringan dan langkah pengurangan matriks. Untuk langkah penyaringan, sekitar 300 komputer SGI Sun cepat dan komputer pribadi (PC) Pentium dijalankan secara paralel terutama pada malam dan akhir pekan dan menggunakan sekitar 8.000 MIPS per tahun. Untuk langkah pengurangan matriks, menggunakan sebuah superkomputer Cray C916 di Pusat Komputer Akademi Amsterdam SARA. Sejumlah upaya tersebut memakan waktu sekitar tujuh bulan. Namun, para ilmuwan mengatakan bahwa penggunaan upaya pengolahan yang terdistribusi

melalui internet dengan ribuan peserta, kemungkinan akan mengurangi waktu anjak satu minggu. Hal ini menyebabkan kriptografer Bruce Schneier terkenal di dunia dalam merekomendasikan penggunaan kunci 2.048-bit.[7]

Tetapi bahkan kunci besar memiliki kejatuhan mereka. Kunci enkripsi khusus terdiri dari 40 sampai 2.048 bit data acak, yang harus disimpan pada hard drive PC dimana semuanya diajukan dengan cara yang sangat logis, teratur. Menurut Adi Shamir (asisten perancang RSA) dan Nicko van Someron, potongan keacakan menonjol, membuatnya mudah untuk program jahat dalam menemukan mereka.[8]

Enkripsi ini bisa dibilang aspek yang paling penting dari keamanan informasi. Ini adalah komponen utama dalam infrastruktur keamanan informasi secara keseluruhan dari setiap proses elektronik. Hampir semua pertukaran elektronik atas data yang signifikan atau yang penting menerapkan penggunaan beberapa bentuk enkripsi. Enkripsi sangat penting untuk pertukaran informasi yang berkaitan dengan masalah-masalah keamanan nasional, untuk transaksi keuangan elektronik dalam semua sistem perbankan besar, untuk perdagangan elektronik di kalangan pedagang dan konsumen, untuk pertukaran data elektronik (EDI) antara perusahaan dan pelanggan mereka, dan untuk keamanan sandi dan informasi rahasia lainnya yang berada di hampir semua sistem komputasi.

Kontrol kriptografi yang tepat dapat membantu menjamin kerahasiaan, integritas, keaslian, dan tanpa penolakan atas pesan elektronik yang dikirim atau diangkut antara atau di tengah berbagai sistem komputasi. Kebijakan, prosedur, keamanan fisik atas perangkat, kontrol keamanan logis, dan kriptografi semua memainkan peran penting dalam lingkungan keamanan sistem informasi (SI) secara keseluruhan. Tanpa kontrol kriptografi yang efektif, kontrol SI yang lain hanya mendukung infrastruktur yang lemah. Kontrol tanpa kriptografi jauh lebih rentan terhadap pengelakan karena bergantung pada pendidikan manusia dan kemampuan manusia dalam melakukannya. Di zaman modern, kriptografi, sementara mengandalkan manusia dalam penyusunan dan aspek-aspek tertentu dari kontrol, pada dasarnya satu set kontrol terkomputerisasi, sehingga memberikan potensi kecepatan yang secara signifikan lebih besar dan keandalan dari kontrol berdasarkan manusia. Oleh karena itu, jika dirancang dan dilaksanakan dengan baik, kontrol kriptografi hanya dapat dirusak oleh komputer lain. Untungnya, manusia harus mengarahkan komputer untuk memecahkan enkripsi algoritma dan kontrol kriptografi lainnya. Pikiran dari komputer cerdas secara mandiri menentukan kapan, bagaimana, dan mana kontrol

kriptografi yang memecahkan dan mengungkapkan kepada dunia jauh lebih menarik daripada pengetahuan ketika upaya hacking teridentifikasi, entah dimana di dunia setidaknya ada satu manusia yang melakukan tindakan tersebut. Mungkin waktunya akan tiba ketika komputer harus dihadapi sebagai musuh langsung.

Karena pentingnya kriptografi dalam membantu mengamankan informasi elektronik di hampir semua sistem komputer, pemahaman dasar dari konsep ini adalah penting bagi auditor SI dalam melakukan pekerjaannya secara efektif. Sisa dari bab ini memberikan sejumlah informasi yang cukup dalam memahami dan menilai kecukupan pengendalian kriptografi secara efektif yang mungkin ditemui auditor.

TERMINOLOGI

Istilah *enkripsi, kriptografi, kriptanalisis,* dan *kriptologi* sering digunakan secara bergantian. Namun, perbedaan dalam hal jaminan ini yaitu definisinya sehingga dapat digunakan dalam konteks yang tepat dalam pembahasan yang sudah kompleks dan kadang-kadang membingungkan.

- *Enkripsi* adalah tindakan atau proses menerjemahkan pesan ke dalam bentuk tersembunyi dengan menggunakan formula rahasia, atau algoritma.
- *Dekripsi* adalah tindakan atau proses menerjemahkan pesan tersembunyi ke dalam bentuk aslinya, yang dapat dibaca. Mengenkripsi dan mendekripsi identik dengan istilah penulisan dan pengartian dalam kode.
- *Analgoritma* adalah prosedur langkah demi langkah untuk memecahkan masalah langkah dalam jumlah terbatas. Sebagaimana diterapkan pada enkripsi, algoritma merupakan formula rahasia yang digunakan untuk mengenkripsi dan mendekripsi pesan. Setiap kali rumus digunakan untuk mengenkripsi pesan, menghitung kunci rahasia acak yang unik, yang harus digunakan untuk mendekripsi pesan.
- *Kriptografi* adalah seni atau ilmu mengenkripsi dan mengartikan pesan dengan menggunakan kunci rahasia atau kode. Beberapa penggunaan awal kriptografi dapat ditelusuri kembali ke peradaban awal seperti Mesir pada sekitar 2000 SM. Kekaisaran Romawi juga menggunakan kriptografi sekitar 2000 tahun kemudian. Sejak saat itu, kebutuhan dan penggunaan kriptografi didokumentasikan sepanjang sejarah. Dengan munculnya komputer dan kebutuhan untuk komunikasi elektronik yang aman, penggunaan kriptografi telah menyebar luas di seluruh dunia pada angka yang semakin meningkat.

- *Kriptoanalisis* adalah seni atau ilmu mengartikan pesan terenkripsi tanpa manfaat kunci atau kode rahasia.
- *Kriptologi* adalah studi ilmiah baik dari kriptografi maupun kriptanalisis.

TUJUAN DARI KONTROL KRIPTOGRAFI

Tujuan dari kontrol kriptografi adalah menjamin kerahasiaan, integritas, dan keaslian informasi elektronik yang dikirim secara memadai, sekaligus memberikan tanpa penolakan oleh pengirim. Enkripsi, ditambah dengan *hashing* dan tanda tangan digital, menjadi solusi yang paling diterima secara umum untuk memastikan transmisi informasi elektronik yang cukup aman, terutama dengan kebutuhan untuk transaksi perdagangan elektronik. Enkripsi, *hashing*, dan tanda tangan digital masing-masing dapat dianggap sebagai salah satu dari tiga kaki yang mendukung pesan elektronik yang aman (lihat Tampilan 11.1). Jika salah satu kaki gagal, pesan tersebut tidak lagi sepenuhnya aman.

Enkripsi membantu menjamin kerahasiaan informasi yang sedang dikirim. *Kerahasiaan* dicapai bila hanya penerima yang dituju atas informasi yang dikirimkan dapat membacanya. Enkripsi juga digunakan untuk melindungi data yang tersimpan pada media elektronik seperti perangkat penyimpanan disk, pita magnetik, dan disket.

Hashing membantu memastikan integritas pesan. *Integritas* dicapai bila informasi yang dikirim belum diubah, informasi lain belum ditambahkan dalam transmisi, dan informasi belum dihapus dari transmisi.

Tanda tangan digital membantu memastikan keaslian transmisi elektronik dan membantu memastikan tanpa penyanggahan dari transmisi tersebut oleh pembuatnya. *Keaslian* tercapai bila penerima pesan cukup yakin bahwa pesan tersebut berasal dari entitas yang tampaknya memiliki sumber dan bukan dari beberapa entitas lain yang tidak diketahui. *Tanpa penolakan* dicapai ketika pengirim pesan tidak dapat menyangkal fakta bahwa ia mengirimkannya. Konsep enkripsi, *hashing*, dan tanda tangan digital akan dibahas dalam bagian berikut.

ENKRIPSI

Dalam dunia komputasi, algoritma enkripsi dapat diklasifikasikan ke dalam dua kategori : simetris atau asimetris. Algoritma simetris menggunakan kunci yang sama untuk mengenkripsi dan mendekripsi pesan. Mungkin algoritma simetris yang paling terkenal dan diimplementasikan secara luas adalah Algoritma Enkripsi Data (DEA), yang diadopsi sebagai Standar Pengolahan Informasi Federal (FIPS) untuk informasi sensitif tapi tidak rahasia oleh pemerintah AS pada tahun 1977. Standar ini dikenal dengan Standar Enkripsi Data (DES). DES dikembangkan oleh IBM di bawah kontrak dengan National Institute of Standards and Technology (NIST), yang sebelumnya dikenal dengan National Bureau of Standards. DES menggunakan panjang kunci 56-bit. Penggunaan DES oleh instansi pemerintah telah menyebabkan penerimaan umum untuk enkripsi komersial. Sebagai contoh, DES saat ini digunakan oleh banyak lembaga keuangan dan jasa peralihan anjungan tunai mandiri (ATM) untuk membantu memastikan transaksi ATM yang aman. Selain itu, Fedwire, sistem transfer wire United State Federal Reserve Bank, menggunakan DES untuk transaksi antara lembaga keuangan.

Kemajuan teknologi telah mengikis kekuatan masa depan dan efektivitas DES. Pada tanggal 17 Juni 1997, algoritma enkripsi DES dirusak oleh Rocke Verser, sebuah Loveland, Colorado, programmer. Dia menyatakan, "Kami telah menunjukkan bahwa DES dapat retak, dan itu tidak sulit untuk melakukannya. Berarti kita perlu membuat tampilan yang sangat serius dalam bagaimana data dienkripsi dan disimpan serta disahkan".[9] Verser menciptakan program *kekuatan brutal* yang cukup fleksibel untuk diunduh melalui Internet dan dijalankan di Unix, Windows, Macintosh, dan OS/2-berbasis komputer. Program ini dirancang untuk menguji semua kemungkinan kunci matematis untuk pesan yang disandikan DES RSA. Sebuah kunci 56-bit memiliki kemungkinan lebih dari 72 kuadriliun kunci (tepatnya 72,057,594,037,972,936). Verser memperkerjakan asisten perusahaan, perorangan, dan ilmuwan di seluruh dunia dengan menawarkan membagi $ 10.000 hadiah 60/40 dengan operator komputer yang akhirnya menentukan kunci kemenangan tersebut. Tim Verser, yang akhirnya berkembang menjadi jaringan dari puluhan ribu komputer relawan, memulai upaya pertekannya pada bulan Februari 1997. Memanfaatkan sumber daya pemrosesan komputer yang menganggur dari jaringan di seluruh dunia, tim Verser berada pada waktu pengujian hampir 7 miliar kunci per detik dan 601 triliyun kunci per

hari! Komputer yang menang, desktop Pentium 90-MHz dengan RAM 16 MB, yang dioperasikan oleh Michael Sanders di Salt Lake City berbasis iNetZ Corporation. Kunci tersebut diidentifikasi setelah pengujian sekitar 18 kuadriliun kunci, atau sekitar 25 persen dari kemungkinan kunci. Pesan yang menang terbaca, "kriptografi yang kuat membuat dunia menjadi tempat yang lebih aman."[10]

Peretakan DES tersebut sebagai tanggapan atas seluruh dunia "RSA Secret-Key Challenge" yang disponsori oleh RSA Data Security, Inc, anak perusahaan yang sepenuhnya dimiliki oleh Security Dynamics, Inc. Pada Konferensi Keamanan Data Januari 1997-nya, RSA menawarkan jumlah berkisar $ 1.000 sampai $ 10.000 untuk melanggar berbagai panjang sementara kunci RC5 dari ukuran maksimum yang berbeda dan menawarkan $ 10.000 untuk melanggar panjang tetap kunci 56-bit algoritma DES. RSA didirikan pada tahun 1982 dan berkantor pusat di Redwood City, California, dinamai oleh pendirinya: Ronald Rivest, Adi Shamir, dan Leonard Adelman.

RSA mengumumkan "Tantangan DES II" pada 13 Januari 1998, pada Konferensi Keamanan Data-nya di San Francisco. Tujuan dari tantangan ini adalah menemukan kunci DES rahasia yang digunakan untuk mengenkripsi pesan dalam waktu kurang dari yang dibutuhkan tim Rocke Verser saat memenangkan Tantangan RSA awal. Tim yang menang, yang terdiri dari programmer dan para peminat dikenal dengan *Distributed.Net*, memecahkan tantangan hanya dalam 39 hari. Tim *Distributed.Net* mengkoordinasikan upayanya dari 22.000 peserta di seluruh dunia, yang menghubungkan lebih dari 50.000 unit pengolah pusat (CPU). Pesan yang menang terbaca, "Banyak tangan membuat pekerjaan menjadi ringan." "Tim mencari lebih dari 61 kuadriliun kunci pada tingkat puncak 26 triliun kunci per detik. Kunci yang menang ditemukan oleh mesin Amerika Serikat yang didukung oleh CPU Alpha setelah mencari 85 persen dari total solusi yang memungkinkan."[11]

Pada tanggal 17 Juli 1998, Electronic Frontier Foundation (EFF) melaporkan bahwa satu komputer telah digunakan untuk mengalahkan algoritma DES 56-bit. Proyek dengan biaya sekitar $ 220.000, menggunakan komputer bernama "Deep Crack" untuk memecahkan pesan enkripsi DES dalam 56 jam. Menggunakan *kekuatan brutal* untuk menguji sekitar 18 kuadriliun kemungkinan kunci. Deep Crack memiliki 36.864 mikroprosesor, yang masing-masing dapat menguji 2,5 juta kemungkinan kunci per detik. Karena paling banyak ada 72 kuadriliun kemungkinan kunci, Deep Crack bisa memecahkan pesan terenkripsi DES dalam waktu kurang dari 9 hari dan 1 jam. Akibatnya, setiap lembaga keuangan yang memiliki

kartu kredit, debit, atau ATM yang dijamin dengan pengganti kerugian nilai verifikasi kartu, kode verifikasi kartu, atau standar enkripsi data harus segera menilai keamanan kriptografinya.[12]

Karena peningkatan kecepatan pemrosesan atas komputer dan biaya yang lebih rendah, DES mencapai akhir masa pakainya. Bahkan, DES saat ini dapat dikalahkan dengan pengetahuan dan peralatan yang tepat. Kemudahan dengan algoritma simetris yang dapat dikalahkan pada dasarnya fungsi dari kecepatan komputer yang digunakan, panjang kunci, dan sumber daya keuangan yang tersedia untuk hacker. Komputer yang lebih cepat dapat menguji lebih banyak kemungkinan dalam jangka waktu tertentu. Mengenai panjang kunci, setiap bit tambahan ditambahkan dengan panjang dua kali lipat dari jumlah kemungkinan kombinasi.[13] Karena komputer yang cepat biasanya lebih mahal daripada yang lambat, jumlah kas yang tersedia untuk menerapkan penggunaan komputer cepat merupakan kendala bagi hacker. Satu artikel menjelaskan hubungan ini dengan efektif :

> Dengan pertimbangan teknologi saat ini, sekitar 90 juta kombinasi kunci DES atau kombinasi 5 juta RC4 dapat diproses per detik. Biaya perangkat keras komputer untuk mencapai hal ini adalah sekitar $ 50.000 - $ 75.000. Dengan kata lain, sekitar $ 50.000, dengan pertimbangan teknologi saat ini, hanya akan memakan waktu sebentar atau sekali untuk memecahkan enkripsi terikat dengan panjang kunci 26 bit. Ini akan memakan waktu sekitar satu jam untuk memecahkan panjang kunci 38 bit. Sebuah kunci 40 bit bisa rusak dalam waktu sekitar 4 jam,[14] kunci 48 bit dalam waktu sekitar 1 bulan, dan kunci 56 bit dalam waktu 30 tahun atau lebih. Naikkan harga sekitar $ 1 juta dan DES dapat rusak dalam waktu sekitar 10 hari. Keamanan terkait dengan algoritma enkripsi 128 bit sangat aman, dengan pertimbangan keadaan teknologi saat ini dan kondisi yang diharapkan dari teknologi untuk 30 tahun ke depan.[15] Kutipan ini mengasumsikan bahwa hanya ada satu dari beberapa komputer yang digunakan. RSA Secret-Key Challenges menunjukkan bahwa beberapa komputer dengan internet yang bekerja bersama-sama dapat mengurangi cakrawala waktu ini secara eksponensial.

Pada tahun 1996, sebagai akibat dari semua ini dan kemajuan teknologi lainnya yang mengancam keamanan DES, NIST memulai proses pemilihan penggantian algoritma, yang dikenal dengan Standar Enkripsi Tambahan (AES). Tujuan dari NIST adalah untuk mengganti DES dengan algoritma lain yang memiliki ukuran blok 128 bit dan ukuran kunci
128, 192, atau 256 bit.

Pada bulan Juni 1998, total dari 15 kandidat untuk AES diserahkan ke NIST saat babak 1 proses seleksi. Kode sumber dan dokumentasi dari semua kandidat ditinjau secara terbuka oleh komunitas kriptografi pada umumnya untuk keamanan, efisiensi, dan keacakan. Pada bulan Maret 1999, para kandidat tunduk terhadap pengawasan lebih lanjut di antara rekan-rekan konferensi AES kedua yang diselenggarakan di Roma, Italia. Revisi dan penyempurnaan algoritma kandidat diizinkan selama babak 1. Babak 1 mencapai puncaknya pada bulan Agustus 1999 dengan NIST menyebutkan lima finalis AES:

1. **MARS**, yang dikembangkan oleh IBM, adalah kunci bersama blok sandi simetrik, yang mendukung blok 128 bit dan ukuran kunci variabel. MARS menawarkan keamanan yang lebih baik tiga kali lipat dari DES saat dijalankan lebih cepat secara signifikan dibandingkan DES tunggal. Kombinasi keamanan yang tinggi, kecepatan tinggi, dan fleksibilitas membuat MARS sempurna untuk kebutuhan enkripsi dari dunia informasi dengan baik ke abad berikutnya.
2. **RC6**, oleh Ron Rivest yang bekerjasama dengan Laboratorium RSA, merupakan perbaikan evolusioner dari RC5 dan membuat penggunaan penting atas rotasi data yang bergantung. RC6 menawarkan keamanan yang baik dan kinerja yang baik.
3. **Rijndael**, oleh Joan Daemen dan Vincent Rijmen Belgia, memiliki blok variabel dan panjang kunci 128, 192, atau 256 bit. Kedua blok dan panjang kunci dapat diperpanjang dengan sangat mudah untuk kelipatan 32 bit. Rijndael dapat diimplementasikan dengan sangat efisien pada berbagai prosesor dan hardware.
4. **Serpent**, oleh Ross Anderson (UK), Eli Biham (Israel), dan Lars Knudsen (Norwegia), yang merupakan sandi blok 128 bit. Ini lebih cepat dari DES dan mendukung implementasi potongan bit yang sangat efisien.
5. **Twofish**, oleh Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, dan Niels Ferguson, menggunakan blok 128 bit dan panjang variabel kunci 128, 192, atau 256 bit. Memberikan fitur pengaturan kuci yang efisien pada mikroprosesor besar, kartu cerdas, dan perangkat keras.[16]

Babak 2 berakhir tak lama setelah konferensi AES ketiga, yang diselenggarakan di kota New York pada bulan April 2000. Pada Oktober 2000, NIST mengumumkan bahwa Rijndael telah dipilih untuk AES baru yang diusulkan. Menurut NIST, Rijndael menunjukkan kombinasi terbaik dari keamanan, kinerja, efisiensi, kemudahan implementasi dan fleksibilitas. Rijndael lebih fleksibel dan dapat diimplementasikan secara efisien pada berbagai platform dengan menggunakan operasi yang sangat sederhana.[17] Pemilihan yang

diusulkan atas Rijndael sebagai AES diumumkan dalam *Federal Register* pada tanggal 28 Februari 2001, dan menjadi sasaran periode komentar publik 90-hari. Akhirnya, pada tanggal 6 Desember 2001, *Federal Register*, Rijndael diumumkan oleh NIST sebagai Standar Pengolahan Informasi Federal (FIPS) 197, juga dikenal sebagai AES. Dan efektif pada tanggal 26 Mei 2002.[18]

Ada kemungkinan DES akan tetap berada pada standar pemerintah untuk aplikasi yang kurang sensitif, dengan AES ditetapkan sebagai standar ketika sensitivitas dari data yang terlindungi lebih tinggi. Jika kekuatan pemrosesan komputer terus meningkat pada tingkat Hukum Moore (yaitu, dua kali lipat setiap 18 bulan), AES mungkin akan menggantikan DES di hampir semua aplikasi komersial. Meskipun AES tidak akan menjadi obat mujarab bagi keamanan yang kami inginkan. Menggunakan teknik yang disebut "serangan sisi saluran," bahkan pesan yang terenkripsi AES berisiko. Serangan ini menganalisis hal-hal seperti jumlah waktu yang operasi kriptografi habiskan, konsumsi daya, emisi radiasi, dan analisis kesalahan untuk membantu menentukan kunci rahasia.[19] Tidak akan pernah ada hal tersebut seperti keamanan total.

Algoritma asimetris memerlukan penggunaan kunci yang berbeda tetapi secara matematis terkait dalam mengenkripsi dan mendekripsi pesan. Kunci ini biasanya disebut sebagai kunci publik dan swasta. Kunci publik disediakan secara terbuka kepada publik sehingga entitas dengan pembuat kunci yang berkomunikasi dapat mengirim informasi elektronik. Kunci pribadi dirahasiakan oleh pembuat kunci. Setelah pesan dienkripsi dengan salah satu kunci, hanya kunci lain yang dapat mendekripsinya. Selain itu, kepemilikan salah satu kunci tidak memungkinkan pemegangnya untuk menentukan kunci lainnya. Biasanya, pengirim pesan menggunakan kunci publik penerima untuk mengenkripsi pesan. Setelah diterima, penerima menggunakan kunci pribadinya untuk mendekripsi pesan. RSA adalah pengembang algoritma asimetris yang terkenal.

Algoritma asimetris menggunakan panjang kunci yang lebih panjang dari algoritma simetris. Untuk mengalahkan algoritma asimetris, seseorang harus menentukan kesesuaian kunci rahasia dari kunci publik. Dalam kasus RSA, setara dengan anjak bilangan bulat besar yang memiliki dua faktor perdana.[20] Berbagai pendekatan matematika diterapkan dalam kriptografi lainnya. Kutipan berikut memberikan perspektif yang baik dari keamanan relatif algoritma asimetris:

Untuk kriptosistem RSA (asimetris), modulus 256 bit mudah diperhitungkan oleh pengguna komputer dengan rata-rata pengalaman dan sumber daya. Kunci dengan 384 bit dapat dipecah oleh kelompok riset universitas atau perusahaan; kunci 512 bit berada dalam jangkauan pemerintah utama. Kunci dengan 768 bit mungkin tidak aman dalam jangka panjang. Kunci dengan 1.024 bit dan lebih harus aman untuk beberapa tahun kecuali kemajuan algoritmik besar yang dibuat dalam anjak, kunci 2.048 bit dianggap oleh kebanyakan agar aman selama beberapa dekade.[21]

Algoritma enkripsi simetris atau asimetris bisa dikalahkan setidaknya dengan dua cara. Pertama, algoritma itu sendiri mungkin lemah, atau secara matematis dapat diprediksi. Sebagai contoh, pada tahun 1995, dua mahasiswa lulusan tahun pertama di Universitas California-Berkeley, David Wagner dan Ian Goldberg (ya, Ian Goldberg yang sama yang disebut pada awal bab ini) menemukan metode untuk memecahkan skema enkripsi kunci publik yang dikerahkan dalam perangkat lunak browser Netscape Navigator World Wide Web yang populer. Untuk setiap transaksi yang dienkripsi, perangkat lunak memerlukan kunci baru. Untuk membuat kunci, dibutuhkan angka awal, dimana dengan menggunakan waktu dan tanggal transaksi serta informasi tertentu mengenai sistem komputer pengguna. Semua informasi ini dapat diperoleh oleh pemecah kode, yang terus menghadapi tugas yang banyak mengurangi pemecahan kode. Bahkan, menurut laporan, "Goldberg dan Wagner bisa memecahkan kode Netscape dalam waktu kurang dari satu menit menggunakan komputer yang sederhana."[22] Demikian pula, Paul Kocher mengidentifikasi fakta bahwa kunci untuk beberapa sistem enkripsi dapat diprediksi dengan mencatat waktu yang terlewati algoritma yang diperlukan untuk mendekripsi pesan.[23]

Serangan *kekuatan brutal* juga dapat digunakan untuk mengalahkan algoritma enkripsi. Serangan *kekuatan brutal* menerapkan penggunaan satu atau lebih komputer untuk menguji semua kemungkinan kunci secara matematis sampai yang benar teridentifikasi. Semakin panjang kunci, menjadi semakin sulit untuk mengalahkan algoritma enkripsi dalam hal waktu dan uang. Namun, kunci yang lebih panjang memiliki kelemahan operasional. Semakin panjang kuncinya, semakin banyak memakan waktu dan mahal bagi penerima yang dituju untuk mendekripsi informasi. Karena banyak kriptografi asimetris memiliki kunci yang lebih panjang dari kebanyakan kunci kriptografi simetris, bisa banyak pesanan dari ukurannya yang lebih lambat daripada pasangan simetrisnya. Bahkan, satu kelompok penulis baru-baru ini melaporkan bahwa beberapa panjang kriptografi kunci pribadi (simetris) sekitar 100 kali lebih cepat daripada beberapa kriptografi kunci publik

(asimetris).[24] Akibatnya, kriptografi asimetris kurang praktis untuk mengenkripsi transmisi bervolume tinggi, *real-time*, atau informasi yang besar. Misalnya, kebanyakan jaringan ATM menggunakan sistem enkripsi simetris seperti DES. Beberapa ATM hanya mengenkripsi nomor identifikasi pribadi (PIN) karena transmisi antara ATM dan komputer pusat di lembaga keuangan pemegang kartu. ATM lain mengenkripsi seluruh pesan transaksi dan PIN. Kecepatan pemrosesan komputer telah cukup maju dimana banyak ATM baru yang mendukung enkripsi DES tiga kali lipat tanpa mempengaruhi kecepatan transaksi secra signifikan.

Enkripsi asimetris tampaknya menjadi metode yang diterima secara umum dalam menjamin kerahasiaan sebagian besar transaksi perdagangan elektronik. Tampilan 11.2 menunjukkan bagaimana enkripsi asimetris dapat digunakan dalam memberikan kerahasiaan pesan. Pengirim mengenkripsi pesan menggunakan kunci publik penerima dan kemudian mengirimkan pesan terenkripsi ke penerima. Penerima mendekripsi pesan menggunakan kunci pribadinya. Kerahasiaan tercapai karena hanya penerima yang mengetahui kunci pribadinya dan satu-satunya yang dapat mendekripsikan pesan pengirim. Ketika enkripsi tidak dapat menjamin kerahasiaan yang lengkap, dengan kunci yang cukup panjang, algoritma enkripsi dapat memberikan tingkat yang wajar atas jaminan kerahasiaan. Tingkat kewajaran diperlukan untuk aplikasi tertentu yang akan bergantung pada pentingnya dan/atau nilai dari informasi yang dilindungi.

Namun, prosedur ini tidak memberikan jaminan bahwa pesan belum diubah atau belum dikirim oleh seorang penipu. Bagian berikutnya dibuat atas contoh dalam Tampilan 11.2 dengan memasukkan konsep *hashing* untuk membantu memastikan integritas dari pesan yang dikirim.

HASHING

Tujuan utama dari *hashing* adalah untuk membantu memastikan bahwa informasi elektronik yang dikirim ke penerima belum diubah, informasi lain belum ditambahkan ke transmisi, dan informasi belum dihapus dari transmisi. Integritas pesan seperti ini dapat dicapai melalui pemanfaatan *fungsi hash satu arah*. Sebuah fungsi *hash* satu arah merupakan rumus matematika yang menggunakan pesan elektronik sebagai masukannya dan membuat blok data yang disebut *inti pesan*. Ketika pesan elektronik dan kunci kriptografi diproses

melalui fungsi *hash* satu arah, blok yang dihasilkan dari data tersebut disebut dengan *kode otentikasi pesan* (MAC).

Dua fungsi *hashing* satu arah yang umum adalah Message Digest 5 (MD-5) dan Secure Hash Algorithm 1 (SHA-1). MD-5 tidak dianggap aman seperti SHA-1. SHA-1 saat ini adalah Standar Pengolahan Informasi Federal (FIPS) pemerintah Amerika Serikat dan standar dari Institut Standar Nasional Amerika (ANSI).

Fungsi *hash* satu arah harus dirancang agar hanya dapat digunakan untuk menghitung inti pesan atau MAC dalam satu arah. Dengan kata lain, seseorang tidak harus dapat menentukan informasi asli dari inti pesan atau MAC yang terkait.

Karakteristik lain yang diinginkan dari fungsi *hash* satu arah adalah tidak harus menghasilkan inti pesan atau MAC yang sama untuk set data yang berbeda. Kepastian tersebut dicapai dengan merancang fungsi *hash* agar membuat inti pesan atau MAC yang panjang. Semakin panjang inti pesan atau MAC, semakin kecil risiko yang ada atas "bentrokan *hash*" dari dua set sumber data yang berbeda.

Pengirim melakukan hal berikut ini :

Mengenkripsi pesan menggunakan kunci publik penerima → Pesan terenkripsi

Mengirim pesan terenkripsi → Penerima

Penerima melakukan hal berikut ini :

Mendekripsikan pesan yang terenkripsi menggunakan kunci pribadi penerima → Pesan

Tampilan 11.3 menggambarkan bagaimana *hashing* dapat diterapkan dalam hubungannya dengan enkripsi asimetris (kunci publik) untuk mencapai kerahasiaan dan integritas pesan. Pengirim merujuk pesan ke fungsi *hashing* satu arah untuk membuat inti pesan. Pesan dan inti pesan dienkripsi, menggunakan kunci publik penerima, dan kemudian file data yang terenkripsi ditransmisikan ke penerima. Penerima mendekripsikan pesan dan menambahkan inti pesan menggunakan kunci pribadinya, merujuk pesan yang didekripsi ke algoritma *hashing* satu arah yang sama yang digunakan oleh pengirim pesan, dan membandingkan inti pesan yang dihasilkan dengan yang diterima dari pengirim. Jika

integritas pesan utuh, pengirim inti pesan akan setuju dengan inti yang dikomputasi oleh penerima. Seperti dalam Tampilan 11.2, kerahasiaan dicapai karena hanya penerima yang mengetahui kunci pribadinya dan karena itu satu-satunya yang dapat mendekripsi pesan pengirim. Selain itu, integritas tercapai karena fungsi *hashing* satu arah yang sama yang digunakan oleh pengirim untuk membuat inti pesan dapat menciptakan inti pesan identik. Bagian berikutnya dibuat atas contoh dalam Tampilan 11.3 dengan memasukkan konsep tanda tangan digital dan sertifikat digital untuk membantu memastikan keaslian pesan yang dikirim.

TANDA TANGAN DIGITAL DAN SERTIFIKAT DIGITAL

Tanda tangan digital dan sertifikat digital digunakan untuk memberikan jaminan kepada penerima pesan bahwa pesan tersebut asli dan tidak dapat ditolak oleh pengirimnya. Untuk menandatangani pesan secara digital, pengirim menunjukkan pesan ke fungsi *hashing* satu arah. Inti pesan yang dihasilkan dienkripsi, menggunakan kunci pribadi pengirim, sehingga hasilnya dalam *tanda tangan digital*. Tanda tangan digital ditambahkan pesan yang telah dienkripsi dengan kunci publik penerima.

Pengirim melakukan hal berikut ini :

Hash pesan menggunakan fungsi *hash* satu arah → inti

Mengenkripsi pesan dan inti menggunakan kunci publik penerima → Pesan dan inti ternekripsi

Mengirim pesan dan inti yang dienkripsi → Penerima

Penerima melakukan hal berikut ini :

Mengartikan/mendekripsikan pesan dan inti menggunakan kunci pribadi penerima → Pesan dan inti

Hash pesan menggunakan fungsi *hash* satu arah seperti pengirim → inti

Membandingkan inti → Jika sama, maka pesan asli. Jika berbeda, maka tolak transaksi.

Sebelum menerima pesan dari pengirim, penerima harus memperoleh *sertifikat digital* secara independen atas pengirimnya. Sertifikat digital diterbitkan oleh *otoritas sertifikat* (CA) terpercaya. Sertifikat digital mengidentifikasikan pengirimnya dan berisi kunci publik pengirim serta tanda tangan digital dari CA terpercaya. (Lihat paragraf terakhir dalam

bagian ini untuk rincian lebih lanjut tentang bagaimana pengirim memperoleh sertifikat digital).

Pada saat menerima pesan, penerima mendekripsi pesan dengan kunci pribadinya. Seperti dalam Tampilan 11.2 dan 11.3, penerima sekarang terjamin kerahasiaan pesannya. Selanjutnya, penerima merujuk pesan yang didekripsi dengan fungsi *hashing* satu arah yang sama yang digunakan oleh pengirim untuk menghasilkan inti pesan. Kemudian penerima mendekripsi tanda tangan digital menggunakan kunci publik pengirim yang terdapat dalam sertifikat digital yang diperoleh dari CA terpercaya, dengan demikian inti pesan terungkapkan. Penerima membandingkan inti pesan dalam tanda tangan digital dengan memperhitungkan ulang inti pesan untuk memastikan bahwa belum ada perubahan apapun (integritas). Karena kunci publik pengirim yang terdapat dalam sertifikat digital berhasil mendekripsi tanda tangan digital, penerima dapat meyakini keaslian pesannya. Selain itu, karena sertifikat digital diperoleh secara independen dari CA terpercaya, pengirim tidak dapat menolak pesan tersebut. Tampilan 11.4 merangkum semua kontrol ini.

Tampilan 11.4 Enkripsi Asimetris dengan *Hashing* dan Tanda Tangan Digital

Pengirim melakukan hal berikut ini :

Hash pesan menggunakan fungsi *hash* satu arah → inti

Mengenkripsi inti dengan kunci pribadi pengirim → Tanda tangan digital

Mengenkripsi pesan menggunakan kunci publik penerima → Pesan terenkripsi

Mengirim pesan yang terenkripsi dan tanda tangan digital → Penerima

Penerima melakukan hal berikut ini :

Mendekripsikan pesan menggunakan kunci pribadi pengirim → Pesan

Hash pesan menggunakan fungsi *hash* satu arah seperti pengirim → inti

Mendapatkan sertifikat digital pengirim secara independen → Nama dan kunci publik pengirim

Mendekripsikan tanda tangan digital menggunakan kunci publik pengirim → inti dan kunci publik pengirim

Membandingkan inti → Jika sama, berarti pesan asli; jika berbeda, tolak.

Membandingkan kunci publik pengirim dari tanda tangan digital dan sertifikat digital → Jika sama, maka pengirim asli; jika berbeda, tolak, kemungkinan penipu.

Untuk memperoleh sertifikat digital, pemohon harus memanfaatkan jasa otoritas sertifikat. *Otoritas sertifikat* dibentuk untuk membantu memastikan bahwa pemegang kunci publik tahu siapa yang membuat pesan menggunakan kunci pribadinya. Otoritas sertifikat adalah organisasi yang menyatakan keaslian kunci publik, mengidentifikasi pembuat kunci publik/pribadi, dan mendistribusikan kunci publik. Otoritas sertifikat yang ada yaitu lembaga keuangan, penjual keamanan produk, dan lembaga pemerintah.[25] Beberapa negara bagian, seperti Utah dan Washington, telah menyusun undang-undang untuk upaya memastikan bahwa otoritas sertifikat memenuhi standar tertentu sebelum mereka berlisensi dan dengan demikian secara hukum "dipercaya". Tanpa CA, pemegang kunci publik mungkin berpikir mereka mengirim dan menerima pesan yang dienkripsi dengan satu pihak ketika mereka benar-benar bertukar pesan dengan pihak yang tidak dikenal dan berpotensi berbahaya. Bahkan dengan CA, pemegang kunci publik yang naif dapat mengenkripsi dan mengirim pesan, dengan menggunakan kunci publik dari pemegang kunci pribadi yang berbahaya, jika mereka mendapatkan sertifikat digital dari pengirim daripada CA terpercaya. CA terpercaya menawarkan jaminan keaslian yang jauh lebih besar karena sekali CA mengidentifikasi pemohon dengan memuaskan, dapat membuatkannya sertifikat digital, mengenkripsi sertifikat menggunakan kunci sertifikasi pribadi, dan mengirimkan sertifikat kepada pemilik dan dengan pihak-pihak lain dimana pemilik ingin bertukar pesan. Pemilik sertifikat digital dan penerima sertifikat selanjutnya dapat mendekripsi menggunakan kunci verifikasi publik CA dan kemudian menggunakan kunci publik pengirim untuk mendekripsi tanda tangan digital pengirim yang telah ditambahkan ke pesan pengirim.

MANAJEMEN KUNCI

Manajemen kunci enkripsi simetris praktis untuk jumlah yang relatif terbatas dari pasangan komunikasi yang ingin bertukar informasi. Sebagai contoh, sebagian besar jaringan ATM yang menggunakan enkripsi simetris cukup dapat mengelola kunci karena sejumlah ATM dengan komputer utamanya yang harus berkomunikasi relatif kecil (misalnya, beberapa ribu). Kunci enkripsi biasanya diberikan dan dikendalikan oleh badan pusat seperti penjual layanan peralihan jaringan.

Sayangnya, dengan perdagangan elektronik, manajemen kunci menjadi tantangan yang jauh lebih besar. Mempertimbangkan bahwa setiap individu di dunia yang memanfaatkan internet bisa menjadi pelanggan potensial dari setiap bisnis di dunia, pada masing-masing bisnis perlu berkomunikasi secara aman dengan setiap individu. Juga, setiap bisnis di dunia bisa menjadi pelanggan dari hampir setiap bisnis lain di dunia. Akibatnya, jumlah potensial pasangan komunikasi mengejutkan. Pada konferensi yang saya hadiri, salah satu pembicara memberikan rumus matematika untuk menentukan jumlah kunci unik yang diperlukan dalam menjamin kerahasiaan antara semua pasangan komunikasi.[26] Rumusnya adalah :

$$K = \frac{n^2 - n}{2}$$

Dalam rumus ini, K adalah jumlah kunci unik dan n adalah jumlah entitas yang berkomunikasi. Dengan menggunakan rumus ini, tabel yang dapat dibuat untuk menggambarkan seberapa cepat jumlah kunci unik yang dibutuhkan meningkat seiring dengan jumlah entitas yang berkomunikasi meningkat :

n	*Formula*	*K*
10	$\frac{10^2 - 10}{2}$	= 45
100	$\frac{100^2 - 100}{2}$	= 4.950
1.000	$\frac{1.000^2 - 1.000}{2}$	= 499.500
10.000	$\frac{10.000^2 - 10.000}{2}$	= 49.999.500

Seperti yang dapat dilihat, dengan 10.000 entitas yang berkomunikasi saja, jumlah kunci unik yang dibutuhkan hampir 50 juta. Bayangkan berapa banyak kunci yang akan diperlukan jika jumlah entitas yang berkomunikasi sampai 1 miliar. Pada kenyataannya, potensi jumlah entitas yang berkomunikasi di dunia adalah beberapa miliar. Manajemen kunci terpusat dengan banyak entitas yang berkomunikasi ini jelas akan menjadi tugas yang tak terkendali. Karena kriptografi kunci publik tidak memerlukan manajemen kunci

terpusat, jauh lebih praktis untuk perdagangan elektronik daripada kriptografi kunci pribadi. Meskipun kriptografi kunci publik tidak memerlukan penggunaan otoritas sertifikat terpercaya untuk mengeluarkan sertifikat digital, manajemen kunci sebenarnya dilakukan oleh pemegang kunci pribadi, sehingga menghapus beban manajemen kunci terpusat.

ASPEK POLITIK DARI KRIPTOGRAFI

Di Amerika Serikat, perusahaan yang mengembangkan produk-produk enkripsi dilarang mengekspor perangkat lunak enkripsi simetris yang lebih besar dari 40 bit. Pembatasan itu bergeser ke 56 bit untuk beberapa perusahaan pada tahun 1997.[27] Pada Juni 1997, Netscape dan Microsoft diberikan pengecualian terhadap pembatasan ekspor Amerika Serikat, dimana mereka diizinkan untuk menjual perangkat lunak yang dilindungi oleh teknologi enkripsi 128-bit untuk lembaga perbankan asalkan perangkat lunak tersebut hanya digunakan untuk transaksi keuangan.[28]

Pada tanggal 17 Juli 2000, pemerintahan Clinton mengumumkan akan melonggarkan kontrol terhadap ekspor perangkat lunak enkripsi. Perusahaan-perusahaan AS tidak perlu lagi izin untuk mengekspor produk enkripsi kepada setiap pengguna akhir di 15 negara Uni Eropa, Australia, Norwegia, Republik Ceko, Hungaria, Polandia, Jepang, Selandia Baru, dan Swiss.[29]

Masalah pembatasan ekspor produk enkripsi berasal dari kenyataan bahwa penjahat, termasuk mata-mata dari negara lain, bisa menggunakan kriptografi untuk menutupi kegiatan mereka. Oleh karena itu, pemerintah AS percaya bahwa hal tersebut harus dapat mengakses program pemulihan kunci untuk tujuan penegakan hukum dan perlindungan kepentingan keamanan nasional.

Sementara alasan ini tampaknya tepat, lawan, termasuk RSA Data Security, Inc, berpendapat bahwa pengguna produk kriptografi memiliki hak privasi. Mereka percaya bahwa badan-badan penegak hukum AS mungkin menyalahgunakan wewenang mereka dalam mendapatkan perangkat lunak pemulihan kunci dan kemudian melanggar hak-hak privasi dari berbagai individu dan perusahaan.

Masalah lain dengan pembatasan ekspor adalah mencegah perkembangan pasar terbuka dari produk kriptografi oleh perusahaan-perusahaan AS. Akibatnya, perusahaan-perusahaan AS [illegible] dalam kerugian yang kompetitif dengan beberapa rekan-rekannya di

Eropa dan benua lainnya yang mampu memasarkan perangkat lunak enkripsi kunci simetris 128-bit. Lebih jauh lagi, produk enkripsi "bit tinggi" dapat dibeli untuk digunakan oleh penjahat dan mata-mata di Amerika Serikat, sehingga mengalahkan kontrol yang dimaksudkan oleh pembatasan ekspor AS.

Negara-negara lain juga membatasi impor dan penggunaan produk kriptografi. Beberapa negara-negara ini memiliki masalah kriminal yang sama dengan pemerintah AS. Namun, negara-negara lain seperti China membatasi impor dan penggunaan produk kriptografi agar mereka dapat mempertahankan kemampuannya dalam menyensor informasi elektronik yang masuk. Pemerintah ini menggunakan kedok bahwa mereka melindungi rakyat dan tanah air mereka. Pada kenyataannya, mereka benar-benar menyensor beberapa transmisi elektronik yang masuk.

Keterlibatan pemerintah dunia membawa penerangan terhadap pentingnya kriptografi dalam perlindungan informasi yang penting untuk keamanan nasional. Kutipan berikut, dari website Badan Keamanan Nasional AS (NSA), menjelaskan bagaimana kriptografi telah menjadi perhatian selama beberapa dekade :

> Pada tahun 1972, Presiden AS mendirikan Pusat Pelayanan Keamanan (CSS) untuk memberikan upaya kriptologi yang lebih menyatu dengan Departemen Pertahanan. Sebagai Kepala CSS, Direktur NSA melakukan kontrol atas kegiatan intelijen sinyal dari layanan militer AS. Pada tahun 1984, di bawah arahan Presiden, misi dari Badan Keamanan Nasional (NSA) diperluas mencakup keamanan sistem informasi untuk sistem keamanan nasional. NSA memiliki dua misi, yaitu untuk membantu sistem kode desain yang akan melindungi integritas sistem informasi Amerika Serikat, dan untuk mencari kelemahan dalam kode lawan. NSA mempekerjakan pembuat kode dan pemecah kode utama negara, dan merupakan salah satu perusahaan terbesar dari matematikawan di AS dan mungkin dunia.[30]

Jelas, kriptografi memainkan peran penting dalam semua aspek kehidupan kita. Hal ini mempengaruhi kita sebagai individu mengenai privasi pribadi dan keamanan atas transaksi keuangan kita dan informasi lainnya. Hal ini juga mempengaruhi aktivitas kompetitif dan pribadi atas bisnis dan organisasi yang mempekerjakan kita dan yang kita miliki. Akhirnya, hal itu mempengaruhi keberadaan masa depan negara-negara di mana kita hidup. Kontrol kriptografi bermanfaat bagi kita dalam tiga bidang, tetapi kriptografi juga dapat digunakan secara jahat.

Beberapa buku telah ditulis mengenai masalah penggunaan sembarangan atas informasi. Salah satu buku yang lebih dikenal ditulis oleh Winn Schwartau. Schwartau mengelompokkan peperangan informasi menjadi tiga "kelas". Kelas 1 adalah perang

informasi pribadi, yaitu studi mengenai semua sumber informasi tentang masing-masing kita sebagai individu. Kelas 2 adalah perang informasi perusahaan, terkait dengan studi mengenai informasi yang mempengaruhi bisnis, perdagangan, atau kepentingan ekonomi. Kelas 3 adalah perang informasi global, meliputi studi mengenai informasi yang terkait dengan kepentingan nasonal.[31] Masing-masing kelas dari perang informasi ini mempengaruhi kita semua. Kontrol kriptografi memainkan peran penting dalam memerangi perang informasi. Bila digunakan dalam hubungannya dengan kebijakan dan prosedur yang efektif, dan kontrol keamanan fisik dan logis, kontrol kriptografi dapat menciptakan lingkungan SI yang tangguh dan aman terhadap serangan berbahaya. Untuk tetap efektif, kontrol keamanan SI ini harus terus berevolusi dan beradaptasi karena penyerang menjadi lebih inovatif dan agresif.

Studi kasus 11.1 dan 11.2 menggambarkan berbagai masalah yang berhubungan dengan enkripsi.

Studi Kasus 11.1

Algoritma Sandi yang Lemah

Selama peninjauan sebelum implementasi aplikasi berbasis jaringan yang baru dipasang, saya menilai apakah file sandinya cukup aman. Awalnya, saya meminta agar administrator keamanan sistem membuat ID pengguna audit pengujian untuk saya. Saya memasukkan sandi yang berisi berbagai karakter, termasuk dua angka satu. Selanjutnya saya menemukan bahwa saya bisa melihat file sandi dengan menggunakan program pemanfaatan umum. Setelah penglihatan awal, ID pengguna dapat dibaca, tetapi sandi muncul bergegas dalam keadaan tampak seperti urutan bermacam-macam simbol tipografi.

Namun, dengan sandi yang saya pilih untuk ID pengguna audit pengujian saya, saya melihat bahwa sandi terenkripsi dua simbol "%" dalam posisi yang sama seperti dua angka satu di sandi saya. Saya menyimpulkan bahwa algoritma enkripsi sandi sudah bisa ditebak. Saya kemudian mencatat karakter dari sisa kata sandi saya dengan karakter enkripsi yang terkait. Selanjutnya saya mengubah kata sandi saya, menggunakan karakter lain, dan mencatat karakter terenkripsi yang sesuai. Menggunakan prosedur ini, saya bisa memecahkan algoritma enkripsi dasar yang digunakan oleh aplikasi. Jika saya ingin, saya bisa menerjemahkan sandi administrator keamanan sistem dan mengambil alih sistem.

Masalah ini dicatat dalam laporan audit, namun, tidak ada rekomendasi untuk memerlukan algoritma yang lebih aman yang disampaikan karena aplikasi dianggap memiliki risiko yang relatif rendah. Namun, direkomendasikan agar kelemahan keamanan dikomunikasikan kepada penjual sehingga algoritma enkripsi sandi lebih aman untuk penerbitan aplikasi pada versi mendatang dapat dikembangkan.

Contoh ini menunjukkan bagaimana kita tidak boleh berasumsi bahwa algoritma enkripsi yang digunakan dalam aplikasi modern bahkan cukup aman atau canggih. Auditor sistem informasi harus meminta penjual dan programer jenis enkripsi apa yang digunakan oleh aplikasi. Auditor juga harus berusaha untuk memecahkan file sandi terenkripsi dan desain pengujian lainnya untuk menilai efektivitas pengendalian kriptografi. (Catatan: Pengujian ini harus dilakukan dengan pengetahuan terlebih dahulu dari manajemen auditee.) Mereka juga harus bertanya apakah aplikasi telah diklasifikasikan sesuai dengan Kriteria Umum (CC) atau standar yang setara. Jika klasifikasi CC telah ditetapkan, tingkat klasifikasi harus dinilai kecukupannya. Lihat Lampiran B untuk rincian lebih lanjut mengenai CC.

Studi Kasus 11.2

Kontrol Enkripsi Fedwire

Studi kasus ini memberikan informasi mengenai beberapa kontrol enkripsi yang digunakan oleh Federal Reserve Bank (FRB) Amerika Serikat dengan sistem transfer wire nasionalnya, Fedwire. Hal ini didasarkan pada pengalaman saya di lembaga keuangan yang menggunakan aplikasi berbasis mikro Fedline II untuk berinteraksi dengan Fedwire dengan tujuan mengirim dan menerima transfer wire dan transaksi kliring otomatis (ACH) dan melakukan komunikasi elektronik lainnya dengan FRB seperti pemesanan uang tunai.

Sebelum berinteraksi dengan Fedwire, setiap mikro Fedline II yang telah diinstal harus dilengkapi dengan papan enkripsi yang disediakan oleh FRB. Papan enkripsi berisi algoritma enkripsi DES. Papan diproduksi dengan baterai lithium, yang memiliki daya simpan 5 sampai 10 tahun. Sebagaimana kecepatan komputer yang terus meningkat, papan baru harus disediakan secara berkala kepada lembaga keuangan untuk memastikan operasi yang efisien. Dalam versi yang lebih lama dari papan enkripsi, baterai bisa diganti tanpa mempengaruhi fungsinya. Namun, untuk membantu mengurangi risiko gangguan dan keretakan proses enkripsi, papan baru dirancang dengan baterai yang tak dapat tergantikan.

Jika ada upaya untuk mengganti baterai, data yang diperlukan dalam melakukan enkripsi akan dihapus dari *firmware* dan tidak dapat dipulihkan kembali, sehingga menyebabkan papan enkripsi tidak berguna. Jenis papan enkripsi dapat ditentukan dengan membaca nomor seri di tepi vertikal papan enkripsi, yang dapat dilihat dari bagian belakang komputer mikro tanpa harus membongkarnya.

Instalasi papan enkripsi dengan mudah tidak mengaktifkannya. Untuk mengaktifkan papan, pemasang harus mendapatkan dan memasukkan informasi sinkronisasi dari Departemen Keamanan Data FRB. Hal ini biasanya dilakukan melalui telepon selama instalasi. Prosedur ini mencegah pihak yang tidak sah mencuri papan enkripsi dan menyalin perangkat lunak Fedline II, dan kemudian melakukan penipuan lembaga keuangan.

FRB memberikan garis bantuan, dimana lembaga keuangan dapat dihubungi mengenai instalasi dan penggunaan Fedline II dan Fedwire. Namun, rincian spesifik untuk proses enkripsi Federal Reserve yang lebih dalam jelas disimpan dengan sangat rahasia.

Daftar Pustaka

1. Winn Schwartau, *Information Warfare* (New York: Thunder's Mouth Press, 1996): 683.
2. "Student Breaks Encryption Code," *KIRO Radio News Fax* (January 30, 1997): 1.
3. "Hackers Attack Department of Defense," *Internal Auditor* (October 1996): 10.
4. "Hackers Stole Gulf War Secrets," *KIRO Radio News Fax*, (March 25, 1997): front page.
5. Dr. Dorothy E. Denning, *Manager's Guide to Cyberspace Attacks and Countermeasures* (San Francisco: Computer Security Institute, 1997), 12–13.
6. Laura Myers, "Hired Hackers Breach Nation's Computer System," *Seattle Post-Intelligencer*, April 17, 1998: A3.
7. "Code That Safeguards Internet Transactions Broken," *NewsEdge Corporation Newsbyte* (August 30, 1999); Bruce Schneier, "The 1999 Crypto Year-in-Review," *Information Security* (December 1999): 23.
8. "Lengthy Keys Are Easier to Detect," *SC Magazine* (May 1999): 17.
9. Don Clark, "Group Cracks Financial-Data Encryption Code," *Wall Street Journal* (June 19, 1997): A3.
10. From RSA's website: *www.rsa.com* (July 11, 1997).
11. "RSA's Secret-Key Challenge Solved Again," *Secure Computing* (April 1998): 14.
12. Tom Trusty, "Beware the Deep Crack Threat," *Bank Fraud* (Chicago: Bank Administration Institute, September 1998): 2; Bruce Schneier, "The 1998 Crypto Year-in-Review," *Information Security* (January 1999): 21.
13. Denning, *Manager's Guide to Cyperspace Attacks and Countermeasures,* 12.
14. In fact, Ian Goldberg did it in 3.5 hours, as mentioned at the beginning of this chapter.

15. Michael R. Anderson, *Internet Security – Firewalls & Encryption, The Cyber Cop's Perspective* (1996): 2.
16. *Http://csrc.nist.gov/encryption/aes/* (October 13, 1999).
17. "Advanced Encryption Standard Announced," *Security Wire Digest* (October 5, 2000).
18. *Http://csrc.nist.gov/encryption/aes/* (July 11, 2002).
19. Bruce Schneier, "When in Rome. . . ," *Information Security* (May 1999): 22.
20. Bank for International Settlements, *Security of Electronic Money* (August 1996): 63.
21. *Id.*
22. "Cypherpunks Unveil Netscape Flaws," *Infosecurity News* (November/December 1995): 13.
23. Denning, *Manager's Guide to Cyperspace Attacks and Countermeasures,* 13.
24. Alexander Kogan, Ephraim F. Sudit, and Miklos A. Vasarhelyi, "Implications of Internet Technology: On-Line Auditing and Cryptography," *Information Systems Audit & Control Journal* (Volume III, 1996): 46.
25. "Certificate-Authority Services Emerge," *Infosecurity News* (May 1997): 14.
26. Jon C. Graff, "Session 202: Internet Encryption," Information Systems Audit & Control Association's Computer Audit, Control, and Security (CACS) Conference (May 1997).
27. "Export Granted For 56-Bit Encryption," *Infosecurity News* (May 1997): 14.
28. John Markoff, "Netscape and Microsoft Are Cleared on Exports," *New York Times* (June 25, 1997): C8.
29. "Government Loosens Controls on Export of Encryption Software," *Seattle Times Wire Services* (July 18, 2000).
30. National Security Administration website: *www.nsa.gov* (February 27, 1997).
31. Schwartau, *Information Warfare,* 9–10.