3
Sistemas Criptográficos

3.1
Introdução

À medida que a Internet se populariza, a quantidade de documentos e transações eletrônicas aumenta. A necessidade de segurança eletrônica é uma realidade, e a Criptografia é o suporte básico. São parâmetros de segurança:

•***Confidencialidade***: garantir a confidencialidade da mensagem em relação a pessoas não autorizadas;

•***Integridade***: garantir que a mensagem não foi modificada;

•***Autenticidade***: garantir a origem da mensagem; e

•***Não-repúdio***: garantir que uma pessoa não possa negar que enviou a mensagem.

3.2
Conceitos iniciais

3.2.1
Criptologia

Nesta seção, detalhamos alguns conceitos iniciais sobre esta ciência.

A palavra criptografia é composta pelos termos gregos: kryptos = secreto e grafo = escrita.

Consiste na ciência e na arte de se comunicar secretamente.

Um ***sistema criptográfico*** possui 5 componentes (31):

1.Um espaço de textos em claro (T);

2.Um espaço de textos cifrados (ou criptograma, C);

3.Um espaço de chaves (K);

4.Uma família de transformações de cifragem E_e: T→C, e ∈ K;

5.Uma família de transformações de decifragem D_d: C→T, d ∈ K.

E para as chaves e,d ∈ K, D_d é a inversa de E_e, logo para t ∈ T temos D_d (E_e (t)) = t.

A seguir pontuamos alguns conceitos que estão formalizados em (32):

•***Criptografar***: converter um texto em claro em um texto cifrado (crip-

tografado) através de uma operação inversível dependente de uma chave secreta $C = E_e(T)$.

•***Decriptografar***: reconstituir o texto em claro original através da operação inversa sobre o texto cifrado com o conhecimento da chave secreta $T = D_d(C)$.

•***Criptoanálise***: análise da criptografia utilizada com o objetivo de se detectar fragilidades e a decriptografia do texto cifrado sem o conhecimento da chave.

•***Criptologia***: ciência que estuda a criptografia e a criptoanálise.

•***Sistema criptográfico***: é conjunto formado por um algoritmo, a coleção de textos em claro, textos cifrados e chaves.

•***Chave secreta***: informação que só o remetente e o destinatário possuem. É utilizada para criptografar e decriptografar a mensagem. Nem sempre a chave do remetente é igual a do destinatário, mas estão sempre relacionadas.

•***Canal***: meio pelo qual a mensagem trafega, pode ser seguro ou não.

•***Sistemas criptográficos simétricos***: é fácil deduzir mutuamente o par de chaves (e,d), uma em função da outra. Muitas vezes: e=d.

•***Sistemas criptográficos assimétricos ou de chave pública***: é computacionalmente difícil deduzir mutuamente o par de chaves (e, d), uma em função da outra, onde e = chave pública, e d = chave privativa (ou privada, ou secreta).

3.2.2
Segurança de um Sistema Criptográfico

Há dois enfoques na discussão de segurança em sistemas criptográficos: sistemas ***computacionalmente seguros*** e sistemas com ***segurança perfeita***.

Um sistema é ***computacionalmente seguro*** se o melhor algoritmo para quebrá-lo necessita de N operações, onde N é um número muito grande. O problema é que não existe uma demonstração de que algum sistema criptográfico em uso é seguro com esta definição. Na prática, um sistema está computacionalmente seguro se o melhor método conhecido para quebrar o sistema requer uma quantia muito grande de tempo de processamento (mas isto é muito diferente de uma prova de segurança). Um outro enfoque é prover evidência de segurança computacional comparando a segurança do sistema criptográfico com problemas garantidamente difíceis (redução matemática). É importante observar que isto é apenas uma prova de segurança relativa por redução a algum outro problema, não uma prova absoluta de segurança.

Um sistema criptográfico tem ***segurança incondicional*** ou ***segurança perfeita*** quando não pode ser quebrado, até mesmo com recursos

computacionais infinitos.

A segurança incondicional de um sistema criptográfico não pode ser estudada obviamente do ponto de vista da complexidade computacional, mesmo porque é admitido que o tempo de computação pode ser infinito. O instrumento apropriado para estudar segurança incondicional é a teoria das probabilidades.

Shannon (30) definiu um modelo matemático preciso para identificar um sistema criptográfico seguro. O objetivo de um criptoanalista é determinar a chave K, o texto original, ou ambos. Porém, ele pode ficar satisfeito com alguma informação probabilística sobre o texto original: se é um arquivo de som, ou um texto em alemão, ou dados de planilha eletrônica.

Um sistema criptográfico tem segurança perfeita quando o texto cifrado não fornece nenhuma informação sobre o texto original. Em termos probabilísticos isto pode ser expresso como

$$P(x|y) = p(x)$$

onde $p(x)$ é a probabilidade da mensagem original ser x e $p(x|y)$ é a probabilidade condicional da mensagem original ser x dado que a mensagem cifrada é y.

Shannon demonstrou que para um sistema ter segurança perfeita é necessário que o número de possíveis chaves seja pelo menos tão grande quanto o número de possíveis mensagens. Portanto, a chave deve ser pelo menos do mesmo tamanho que a mensagem, e nenhuma chave pode ser usada de novo.

Um sistema conhecido de segurança perfeita é o ***One-time Pad***, que foi descrito primeiro por Gilbert Vernam em 1917 para uso em cifragem e decifragem automática de mensagens de telégrafo. É interessante observar que, por muitos anos, o ***One-time Pad*** era considerado um sistema inquebrável. Todavia, não havia prova disto até Shannon desenvolver o conceito de segredo perfeito, mais de 30 anos depois. O sistema também é atraente por causa da facilidade de cifragem e decifragem

Infelizmente, existem grandes desvantagens na adoção de um sistema seguro como ***One-time Pad***. Primeiro, devido ao fato de que o número de chaves que deve ser enviado por meio seguro é pelo menos tão grande quanto o número de textos em claro. Segundo, porque é necessário que o número de bits da chave seja igual ao número de bits da mensagem. Este não seria um problema principal se a mesma chave pudesse ser usada para codificar mensagens diferentes. Porém, a segurança perfeita depende do fato que cada chave é usada para uma só cifragem. Assim, uma nova chave deve ser gerada para cada mensagem transmitida. Se esta exigência não for satisfeita, o sistema

pode ser atacado. Por exemplo, se a mensagem original for conhecida, basta realizar um XOR entre a mensagem original e o texto cifrado para desvendar a chave.

Um algoritmo de cifragem é quebrável quando é computacionalmente viável testar todas as chaves possíveis para decriptografar uma determinada mensagem. Este tipo de ataque é chamado de força bruta ou busca exaustiva e seu êxito depende do tamanho da chave.

Uma outra técnica que pode ser utiliza com sucesso é o chamado ataque estatístico. É baseado na constatação que, em vários textos de um mesmo idioma, a freqüência de cada letra é mais ou menos fixa. O inimigo usa este conhecimento para comparar a freqüência de cada letra da mensagem cifrada com as freqüências no idioma suposto. Se o ciframento foi feito através de uma simples substituição, a quebra é imediata.

Duas técnicas básicas para a diminuição das redundâncias em uma mensagem são, de acordo com Shannon (30), confusão e difusão. A confusão oculta a relação entre o texto original e o texto cifrado. Isso dificulta o estudo do texto cifrado na procura por redundâncias e padrões estatísticos. A substituição é um método de confusão. A difusão suaviza a redundância do texto original de forma que as redundâncias fiquem espalhadas no texto cifrado. Assim, pode-se dizer que uma boa cifra de substituição acrescenta confusão à informação, enquanto uma boa cifra de transposição acrescenta difusão.

3.2.3 Tipos de Inimigo

Basicamente, temos dois tipos de inimigos:

1. ***Inimigo Passivo***: intercepta a mensagem e tenta ganhar conhecimentos através dela, mas não interfere no processo de comunicação. Possui basicamente dois objetivos:

•Análise do conteúdo: tenta descobrir o conteúdo (ou parte) de uma mensagem cifrada.

•Análise de tráfego: tenta descobrir a freqüência com que as mensagens passam pelo canal, além do destinatário e remetente.

2. ***Inimigo Ativo***: interfere na comunicação de diversas formas:

•Interrupção: o inimigo interrompe as comunicações como um todo. Uma mensagem não consegue chegar no destinatário;

•Modificação: o inimigo intercepta e altera a mensagem;

•Fabricação: um inimigo gera mensagens falsas e insere no canal;

•Impersonificação (disfarce): o inimigo está tentando se passar por uma outra pessoa, falsificando mensagens ou tentando conseguir acesso a um sistema;

•Repetição (replay): o inimigo possui uma mensagem previamente interceptada e tenta usá-la novamente;
•Negação de serviço (DoS): o inimigo atrapalha o funcionamento do sistema.

3.2.4 Tipos de Ataque

Os tipos de ataques levam em consideração o conhecimento adquirido pelo inimigo, ou seja, se ele possui:
•Somente o texto cifrado;
•Pares de textos em claro e cifrado (T,C), mas não pode escolher;
•Pares escolhidos de textos em claro de entrada, gerando textos cifrados na saída;
•Ou, se escolhe os textos cifrados e obtém os textos em claro equivalentes.

3.2.5 Níveis de Risco

O nível de risco de uma aplicação está diretamente relacionado às perdas financeiras, ou mesmo humanas, que uma falha na segurança pode causar:

•***Baixo risco***: se os dados foram descobertos ou sabotados, o remetente poderá contornar o fato com relativa inconveniência. Geralmente são informações que não tem a atenção do atacante.

•***Médio risco***: algum grau de perda é esperado e tolerado, mas sérias perdas podem prejudicar a empresa. Exemplo: site de vendas na Web com transações comerciais. Devem se proteger de ataques, pois detém uma pequena atenção de hackers.

•***Alto risco***: uma falha na segurança causa danos financeiros ou embaraços. Exemplo: invasão de bancos, alteração de informações de Instituições importantes do país. A segurança deve ser levada a sério, pois são o alvo natural e contínuo de ataques.

•***Aplicação crítica***: Falhas podem causar perda de vidas humanas. Aplicações militares. Exemplo: automação de fábricas e setor de transporte. A segurança deve ter uma importância no projeto inicial e revisões periódicas.

3.2.6 Vulnerabilidades

Para se ter um sistema seguro devemos nos preocupar com os seguintes aspectos passíveis de vulnerabilidades:
•algoritmos e protocolos;
•pessoas responsáveis pelo sistema;

•número de pessoas que conhecem a chave do sistema;
•local de armazenamento da chave;
•pessoas que se relacionam com as pessoas responsáveis pelo sistema.

3.3 Sistemas Criptográficos Clássicos

A seguir, a descrição de alguns sistemas criptográficos clássicos (3).

3.3.1 Monoalfabéticos

(a) Aditivos

O exemplo mais antigo de sistemas aditivos de que se tem notícia é o conhecido como CIFRA DE CÉSAR, usado pelo general e estadista romano Júlio César em sua campanha na Gália (49 - 44 A.C.) e na correspondência com seus amigos. Nele, cada letra do texto-claro é substituída pela letra correspondente a três posições adiante no alfabeto.
O método do sistema aditivo consiste em adicionar ao valor que representa a posição ocupada pela letra no alfabeto o valor do desvio (chave), indicando a posição da letra-cifra.
Atribui-se a ALBERTI, secretário da Cúria Romana, a invenção de um dispositivo simples, na idade média (476 a 1453) conhecido como DISCO DE CIFRAR. Considerando um alfabeto com 26 letras o desvio pode variar de 0 até 25. Isto funciona como a operação de ADIÇÃO em ARITMÉTICA MODULAR e pode ser representado, matematicamente, como:

$$C_i = m_i + d \pmod{26}$$

O total de chaves possíveis é 26, considerado também o desvio=0 que reproduz o texto em claro. Obviamente, o sistema é totalmente inseguro.

(b) Multiplicativos

Existe, por analogia com o anterior, o sistema multiplicativo, no qual a operação de soma é substituída pela multiplicação:

$$C_i = m_i * d \pmod{26}$$

Entretanto, algumas chaves não são aproveitáveis (0, 2, 4., ...). Caso fossem aproveitadas, a volta do criptograma para o texto em claro (decriptografia) causaria ambigüidades. Por exemplo, com d = 2 a cifra B representa

o A ou o N ? Observa-se que somente os números primos com o módulo podem ser aproveitados como chaves (propriedade da aritmética modular).

Então, só são aproveitadas as chaves 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23 e 25. O total de chaves possíveis é 12, considerada também a multiplicação por 1 que reproduz o texto em claro. O sistema, bastante inseguro como o anterior, foi usado em composição com o aditivo para gerar o conhecido como afim.

(c) Afins

Conforme foi dito, combina os dois sistemas anteriores, e pode ser representado por:

$$C_i = (\propto * m_i) + \beta$$

Sua notação é freqüentemente representada por [$\propto$, β], a chave do sistema. O sistema aditivo é caso particular do afim quando $\propto = 1$ enquanto o multiplicativo o é para $\beta = 0$

O total de chaves possíveis é 312 (12 * 26) considerada, também aqui a chave [1, 0] que reproduz o texto em claro.

Nos três sistemas monoalfabéticos vistos até agora, o conhecimento da correspondência criptograma-texto em claro para uma única letra denuncia a correspondência de todo o alfabeto. Essa fraqueza de correspondência pode ser contornada pela utilização dos alfabetos desordenados.

Esses se enquadram nos sistemas monoalfabéticos genéricos, dos quais os anteriores são casos particulares.

(d) Genéricos

Neste sistema o alfabeto-cifra pode estar em qualquer ordem. O total de chaves possíveis é de 26! ($\approx$ 4 * 10 26). Considerando um processador que leve 1 microssegundo para testar cada chave, a pesquisa exaustiva de todas as chaves consumiria 1,7 * 10^{14} anos ($\approx$ 50.000 vezes a idade estimada da Terra).

Encarado sob o ponto de vista do número de chaves possíveis, este sistema poderia ser considerado absolutamente seguro, como no início de sua utilização.

Em relação aos anteriores, quando o alfabeto-cifra é desordenado, não proporciona o conhecimento da correspondência das demais letras ao se descobrir uma delas.

3.3.2 Polialfabéticos

É uma combinação ordenada de diversos sistemas monoalfabéticos. Um exemplo é o Sistema de Vigenère:

•a chave é um conjunto de p letras: (L_1, L_2, L_3, ..., L_p);

•a mensagem deve ser dividido em blocos de p letras: (A_1, A_2, A_3, ..., A_p);

•texto cifrado C é obtido a partir da fórmula:

$$C = ((A_1 + L_1) \bmod 26, (A_2 + L_2) \bmod 26, \ldots, (A_p + L_p) \bmod 26)$$

3.3.3 Permutação

Seja um vetor de n elementos: V=(v_1, v_2, ..., v_n). Uma Permutação é uma operação definida por um vetor P=(p_1, p_2, ..., p_m).

P deve ter todos os números entre 1 e m, sem repetição de nenhum, isto faz com que P tenha uma permutação inversa. A mensagem é dividida em blocos de m letras e a chave deve ser aplicada para permutar as letras.

3.3.4 Esteganografia

Esteganografia é a arte de esconder mensagens secretas em um meio de maneira que as mesmas passem desapercebidas. Exemplos:

•raspar a cabeça de um mensageiro e escrever a mensagem no seu couro cabeludo. Esperar o cabelo crescer e enviar o mensageiro;

•escrever uma carta com tinta invisível;

•substituir os bits menos significativos de uma imagem ou som pelos bits da mensagem que se quer esconder.

3.3.5 Rotores

Seja $M_e(.)$ um Sistema Monoalfabético:

$$M_e(x) = (x+e) \bmod 26$$
$$M_e{}^{-1}(x) = (x-e) \bmod 26$$

Seja $\pi_e(.)$ um Sistema de Permutação. Um rotor é uma função R(.) do tipo:

$$R_{s;k}(x) = M_k{}^{-1} [\ \pi_s{}^{-1} (M_k(x))\ ]$$
$$R_{s;k}{}^{-1}(x) = M_k{}^{-1}[\ \pi_s (M_k(x))\ ]$$

É uma composição de funções implementada com discos (rotores). A máquina Enigma era um sistema de 3 ou 4 discos, e cada Divisão de Exército nazista possuía uma combinação de discos a qual era atribuída uma cor.

3.4 Substituição Homofônica

A história da criptografia demonstra que um tipo de ataque muito utilizado é o ataque estatístico, onde um conhecimento das particularidades de uma determinada língua é usado heuristicamente para a quebra dos códigos. Uma solução é a técnica de substituição homofônica.

A substituição homofônica consegue suavizar a distribuição de probabilidades (ou frequências) dos símbolos de um texto em claro. Desta forma, a tarefa do criptoanalista é dificultada. A substituição homofônica convencional associa um símbolo a um conjunto de símbolos substitutos possíveis, chamados homofônicos, tais que a distribuição destes novos símbolos seja equiprovável. Assim, cada símbolo do alfabeto original é desdobrado em outros equiprováveis.

Em 1988, Günther (5) introduziu uma generalização da substituição homofônica, uma nova forma de escolha dos homofônicos é especificada. Um ano mais tarde, Massey et al. (11) aplicaram conceitos de Teoria da Informação à proposta de Günther e formularam a Substituição Homofônica de Tamanho Variável. Este novo tipo de substituição homofônica é uma generalização da convencional, onde tem-se homofônicos representados por códigos de comprimentos diferentes, e ainda, cada símbolo do alfabeto original pode ser substituído por homofônicos não equiprováveis, ou seja, os homofônicos poderão ter probabilidades distintas. Este é o objetivo da substituição homofônica: converter a seqüência de texto em claro em uma seqüência completamente aleatória.

Figura 3.1: Um esquema geral para substituição homofônica.

A figura 3.1 representa um esquema de substituição genérico para representar tanto a substituição homofônica convencional como a substituição homofônica de tamanho de variável. O canal homofônico é sem memória. O alfabeto de entrada $\{ u_1, u_2, \ldots, u_L \}$ coincide com o grupo de símbolos possíveis de U. O alfabeto de saída $\{ v_1, v_2, v_3, \ldots \}$ também é finito ou infinitamente contável, e com probabilidades de transição $P(V=v_j \mid U=u_i)$. Para cada j

existe exatamente um i tal que P(V=v_j | U=u_i) $\neq$ 0. Cada v_j com P(V=v_j | U=u_i) > 0 será um homofônico para u_i.

Quando o canal homofônico da figura 3.1 é determinístico, todas as probabilidades de transição diferentes de zero são iguais a 1 (podemos também dizer que V=U). Neste caso, então a figura 5.1 representa uma codificação de fonte usual (ou até mesmo uma compressão de dados).

Quando o canal homofônico é não determinístico mas a codificação binária livre de prefixo é simples, ou seja, todos os códigos têm o mesmo comprimento m, então a figura 3.1 representa substituição homofônica convencional.

Quando o canal homofônico é não determinístico e a codificação binária livre de prefixo possui comprimento variável, então a figura 3.1 representa substituição homofônica de tamanho de variável, como definida por Günther (5).

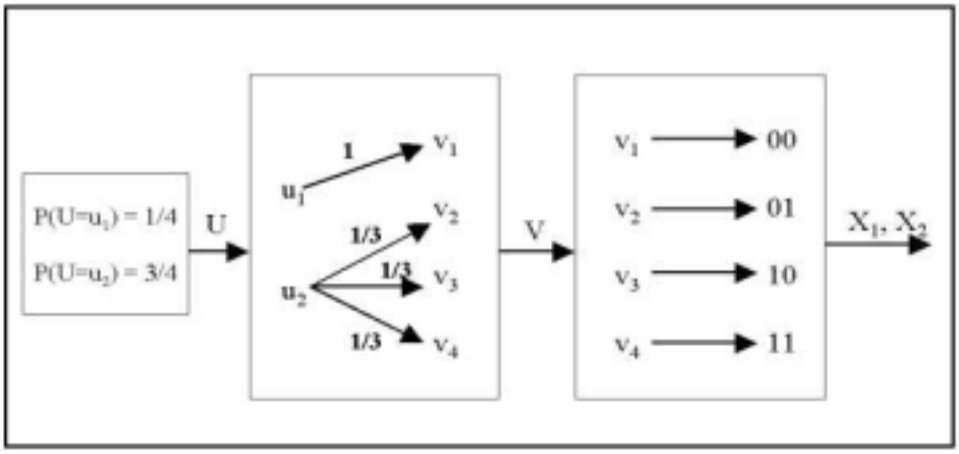

Figura 3.2: Substituição homofônica convencional.

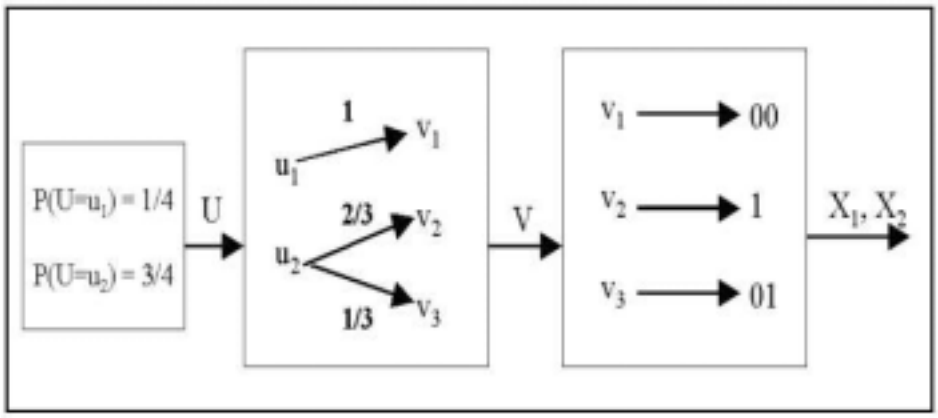

Figura 3.3: Substituição homofônica de tamanho variável.

As figuras 3.2 e 3.3 mostram dois exemplos do esquema geral de substituição homofônica ilustrado na figura 3.1, ambos para a mesma mensagem de fonte binária (isto é, L = 2). A substituição de tamanho de variável tem um comprimento médio do código E[W] = 3/2 e a substituição convencional tem E[W] = 2. Esta redução dos símbolos codificados é uma vantagem da substituição homofônica de tamanho de variável que a torna perfeita.

Uma substituição homofônica é ***perfeita*** se a seqüência D-ária codificada como X_1, X_2,... é completamente aleatória. Para o caso sem memória (fonte e canal), considerado na figura 3.1, isto é equivalente à condição que o código X_1, X_2,... , X_w para V = V_1 é completamente aleatório.

Uma substituição homofônica D-ária é ótima para uma dada mensagem fonte, se e somente se é perfeita e minimiza o comprimento médio do código E[W].

Para simplificar os conceitos, foi utilizado nesta seção D = 2, ou seja, representação binária. Massey et al. (11) apresentam a generalização correspondente.

3.4.1 Distribuição diádica

Uma distribuição diádica é uma distribuição na qual cada probabilidade é uma potência negativa de 2. Por exemplo, (2^{-2} , 2^{-2} , 2^{-2} , 2^{-3} , 2^{-4} , 2^{-4}) é uma distribuição diádica. Em uma distribuição diádica, para cada símbolo u_i de U, as probabilidades dos homofônicos de u_i formam uma decomposição de P(U = u_i) em uma soma dos números referentes as potências inteiras negativas de 2. Por exemplo, para P(U = u_2) = $\frac{3}{4}$ a decomposição pode ser $\frac{1}{4} + \frac{1}{4} + \frac{1}{4}$ ou $\frac{1}{2} + \frac{1}{4}$.

Milidiu et al. (18) mostram que uma codificação canônica de Huffman para uma distribuição diádica gera um fluxo de codificação aleatório. Esse fato é utilizado para a aleatorização de um texto cifrado resultante da codificação (18).

Massey et al. (11) mostram que se o canal é ótimo então a decomposição de P (U = u_i) para qualquer u_i deve consistir em diferentes potências negativas de 2. Logo, para o exemplo P(U = u_2) = $\frac{3}{4}$ acima, a decomposição correta é $\frac{1}{2} + \frac{1}{4}$. Se P(U = u_i) = $\frac{A}{B.n}$ para alguns números inteiros A e B, então a probabilidade pode ser decomposta em uma soma finita de distintas potências negativas de 2. Caso contrário, então teremos uma soma infinita de distintas potências negativas de 2.

Demonstram também um limite superior justo de H(V) para um canal homofônico ótimo:

$$H(U) \leq H(V) = E(W) < H(U) + 2.$$

Esta conclusão mostra que um canal homofônico ótimo nunca aumenta a entropia de sua entrada U por mais de dois bits, independente do valor de H(U).

3.5 Sistemas Criptográficos Simétricos

Os sistemas criptográficos simétricos podem ser implementados com cifras:

•de cadeia ou de fluxo (stream): cifram um bit por vez à medida que o texto em claro vai chegando;

•de bloco: operam sobre blocos (grupos de bytes) do texto em claro sobre o qual a mesma chave é aplicada.

Os algoritmos são geralmente rápidos, e as chaves pequenas (56, 128, 256 bits). Necessitam que emissor e receptor tenham conhecimento do par de chaves (e,d) através de um canal seguro. O algoritmo mais usado atualmente ainda é o DES (Data Encryption Standard). Existe ainda uma tentativa de se aumentar a segurança do DES chamado de DES triplo, ou Triple-DES ou 3DES (29). O padrão atual é o AES (Advanced Encryption Standard).

Outros sistemas criptográficos simétricos são (29):

•variantes do DES;

•Lucifer;

•RC2, RC4, RC5, RC6;

•IDEA;

•Skipjack;

•Blowfish, Twofish;

•Safer.

3.6 Sistemas Criptográficos de Chave Pública

Em um sistema criptográfico de chave pública, temos:

•Chave que cifra = chave pública

•Chave que decifra = chave privativa

Qualquer um pode mandar uma mensagem cifrada com a chave pública, mas apenas quem possui a chave privativa vai decifrar. E o conhecimento da chave pública não revela conhecimento sobre a chave privativa.

Protocolo de comunicação para a troca de mensagens com sistemas criptográficos de chave pública:

1.A possui uma chave pública P_a e uma chave privativa R_a;

2.B possui uma chave pública P_b e uma chave privativa R_b;
3.A pega a chave pública P_b de B e manda $E_{Pb}(x)$;
4.B decifra: $D_{Rb}[E_{Pb}(x)] = x$.

O algoritmo mais utilizado é o RSA.

Outros sistemas criptográficos de chave pública são (29):

•Knapsack;
•Rabin;
•Schnorr;
•ElGamal;
•Curvas Elípticas.

3.7 Geração de chaves

O processo de geração de chaves deve ser o mais randômico possível. O atacante não pode reproduzir o processo de geração, pois isso fragiliza o sistema.

Geração de uma semente randômica para um algoritmo de números aleatórios:

•Por hardware: circuitos eletrônicos sensíveis a eventos físicos naturais aleatórios que se convertem numa sequência imprevisível de bits;
•Interação com o usuário: velocidade de digitação, o usuário informa um número ou palavra qualquer, movimenta o mouse, etc.;
•Semente interna: relógio do sistema, número de arquivos no disco rígido, espaço livre de memória, ou então uma combinação destes.

Duas importantes características dos números aleatórios:

•Distribuição uniforme, isto é, deve existir uma chance igual para cada número a ser gerado;
•A sequência de geração não deve se repetir.

No algoritmo RSA, após a geração é feito um teste de aleatoriedade para determinar se o número é primo. É a forma encontrada para o não conhecimento de um algoritmo que gere números primos grandes.

3.8 Funções de Resumo da Mensagem

São funções hash one-way (29) que geram um "código" ou "resumo" da mensagem de tal modo que:

•seja difícil gerar a mensagem original a partir do "resumo";
•seja difícil encontrar uma outra mensagem com o mesmo "resumo".

Exemplos:
•MD4 (Message Digest), de 1990;
•MD5, de 1992;
•MD2, de 1992;
•SHA-1 (Secure Hash Algorithm), de 1992;

3.9 Assinaturas Digitais

Conceitos:

(i) Assinatura digital: o emissor assina o hash da mensagem cifrando com sua chave privativa.

(ii) Autenticação: o receptor calcula o hash da mensagem e compara com o hash da mensagem decifrado usando-se a chave pública do emissor. A figura 3.4 mostra esse esquema de autenticação. Autenticação é o processo de se verificar a autenticidade de uma dada entidade, de tal forma que se possa ter certeza que a entidade é realmente quem ela diz ser. Permite também garantir o não-repúdio.

(iii) Certificado Digital: possui uma chave pública que será disponibilizada para alguém que quer enviar uma mensagem mas quer ter a certeza da autenticidade do dono da chave pública. Uma Autoridade Certificadora (AC) assina o certificado, garantindo assim a associação entre a chave pública e o dono. O padrão mais utilizado hoje é o X.509, que contém vários dados, por exemplo:
•chave pública;
•versão;
•Número de série;
•Data de criação;
•Data de expiração;
•Algoritmos utilizados (ex: RSA, MD5);
•Dados sobre o dono (nome, e-mail, empresa, etc.);
•Dados sobre a AC;
•Assinatura da AC (ex: Verisign, Thawte).

Neste capítulo, uma coleção de técnicas criptográficas foi descrita. A criptografia é o suporte básico para se prover a segurança das transações eletrônicas, e um bom conhecimento destas técnicas é essencial para a implementação de novas aplicações.

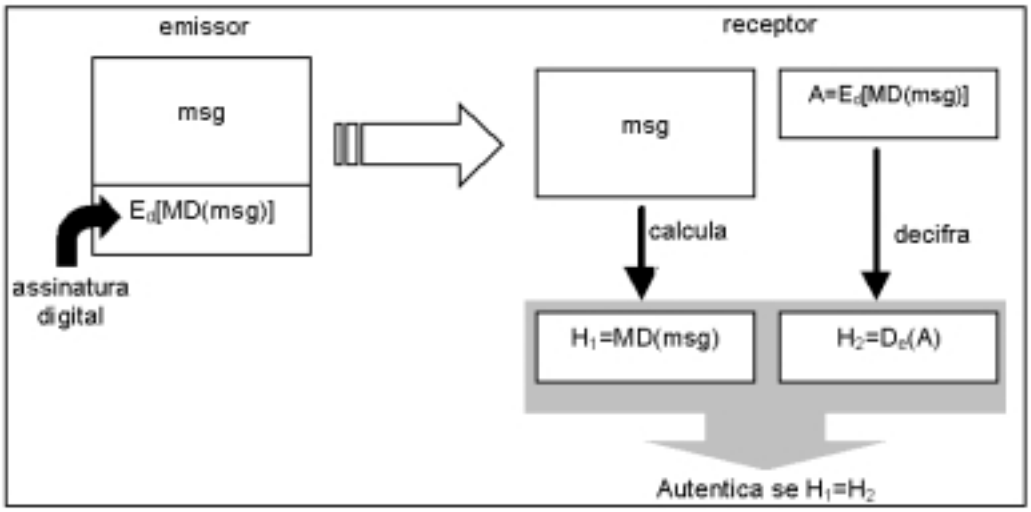

Figura 3.4: Esquema de autenticação digital.