

Segurança em Sistemas Computacionais

Prof. Dr. Carlos André Batista de Carvalho



1

Fundamentos de Segurança

- Definição
 - “Segurança de computadores: A **proteção** oferecida a um sistema de informação automatizado para atingir os **objetivos** apropriados de preservação da **integridade**, **disponibilidade** e **confidencialidade** de **ativos** de sistemas de informação (incluindo hardware, software, firmware, **informações/dados** e telecomunicações).”
 - Outros objetivos de segurança: autenticidade e responsabilidade
 - **Cenários** de segurança
 - Segurança de redes, sistemas operacionais, softwares, nuvem
 - Ambientes não computacionais (ex. residencial, trabalho)
- Motivação
 - Existência de **Ameaças (fonte de ameaças)**
 - Possibilidade de **violação** de algum objetivo de segurança
 - Exemplo: vazamento de dados

2

2

Fundamentos de Segurança

- Objetivos ou Propriedades
 - Confidencialidade
 - Limita o acesso apenas a usuários **autorizados**
 - Integridade
 - Proteção contra modificação (e destruição) **não autorizada**
 - Disponibilidade
 - Permite o acesso sempre que necessário a entidades **autorizadas**
 - Autenticidade
 - Identificação das **entidades** envolvidas (elas são **autorizadas**)
 - Verifica se **origem dos dados** é de uma entidade **autorizada**
 - Responsabilidade (*accountability*)
 - Capacidade de atribuir ações unicamente a determinada entidade
 - Auditoria, **Irretratabilidade** (Não Repúdio), Análise Forense

3

3

Fundamentos de Segurança

- Uma violação de segurança atinge um desses princípios
 - É possível mensurar o **impacto** (dano) em um **ativo** resultante de uma violação
 - Violação **intencionais**
 - **Ataque**
 - **Tentativa** deliberada de provocar uma violação
 - Exemplo: A captura de tráfego afeta a confidencialidade
 - Superfície de ataque: “ponto de exposição”
 - **Atacante / Adversário**
 - Violação accidental
 - Fonte de ameaça

4

4

2

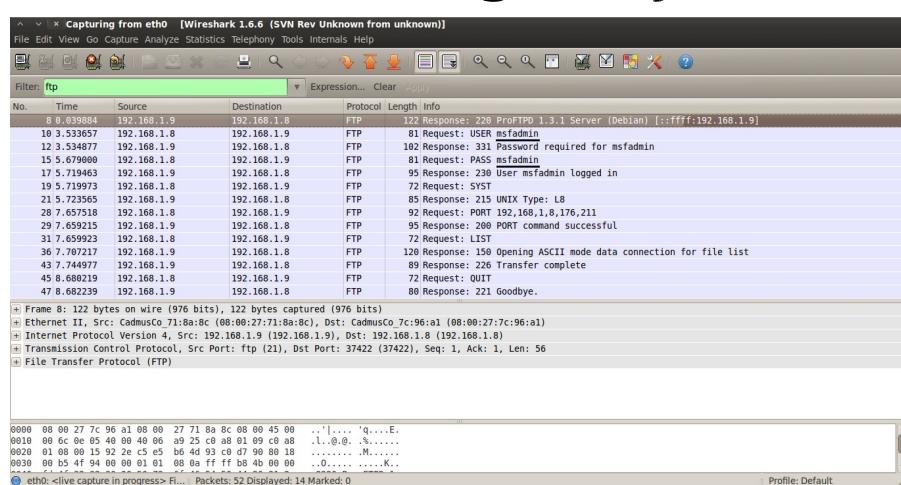
Fundamentos de Segurança

- Uma violação pode ocorrer quando existe alguma **vulnerabilidade**
 - Uma **fraqueza/falha** que pode ser **explorada** em um ataque
 - Por meio de ferramentas chamadas **exploits**
 - Exemplo: O Wireshark pode ser utilizado para capturar pacotes em uma rede de computadores
 - Quando os pacotes são enviados sem criptografia
- Os ataques nem sempre são isolados
 - Várias etapas
 - **Engenharia Social**
 - **Políticas de segurança**

5

5

Fundamentos de Segurança

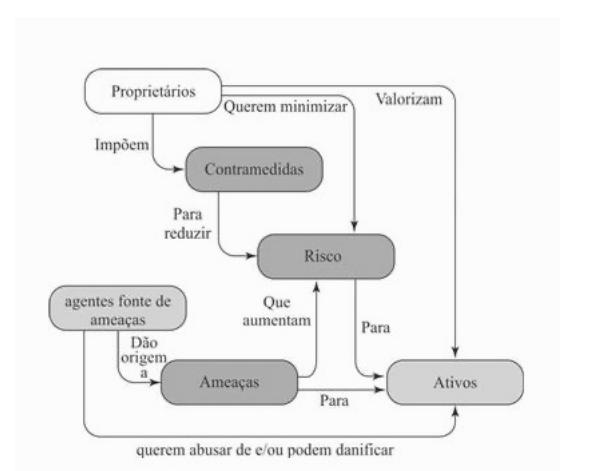


6

6

Fundamentos de Segurança

- Controle de segurança (**contramedida**)
 - “É qualquer processo (ou dispositivo incorporado a tal processo) projetado para **detectar**, **impedir** ou **permitir** a recuperação de um ataque a segurança”
 - Exemplo: Protocolos de redes (TLS/SSL)
- Conceitos e relacionamentos
 - **Risco** (Expectativa da perda de segurança)
 - **Probabilidade vs Impacto**
- Segurança é um processo
 - As soluções não são definitivas e 100% seguras
 - **Plan-Do-Check-Act** (ISO 27000)



7

7

Fundamentos de Segurança

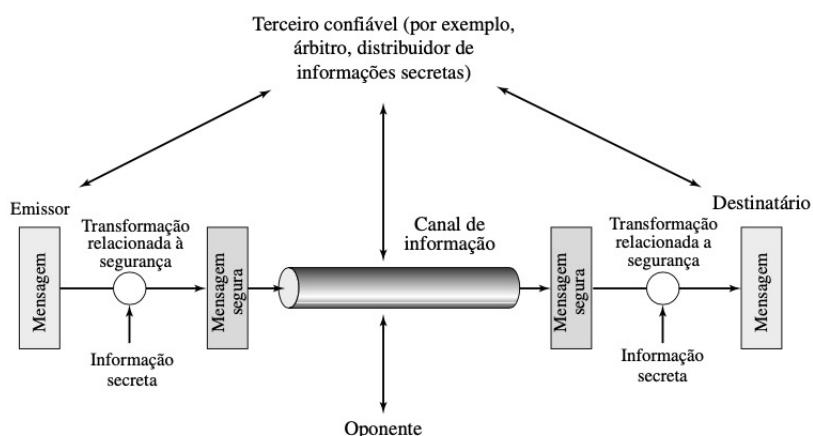
- Desafios
 - Segurança não é simples
 - Identificar potenciais fraquezas e detalhes sobre os ataques
 - É fácil esquecer de algum detalhe
 - Projeto vs Implementação
 - Para o atacante, basta encontrar uma brecha
 - Conjunto de mecanismos utilizados em conjunto
 - **Particularidades dos ambientes de cada sistema**
 - É necessário investimento
 - Segurança não pode ser visto como um custo
 - Preocupação constante (projeto e monitoramento)
 - Segurança vs Usabilidade/Eficiência

8

8

Fundamentos de Segurança

- Modelos de Segurança
 - Comunicação



9

9

Fundamentos de Segurança

- Modelos de Segurança
 - Controle de Acesso

Oponente

- humano (por exemplo, hacker)
- software (por exemplo, vírus, verme)



Canal de acesso



Função de portaria

Sistema de informações

Recursos de computação (processador, memória, E/S)
Dados
Processos
Software
Controles de segurança internos

10

10

Fundamentos de Segurança

- Ameaças
 - Revelação não autorizada
 - Fraude
 - Usurpação
 - Disrupção
- Ataques
 - Passivos
 - Leitura não autorizada
 - Análise de tráfego
 - Ativos
 - Interrupção/fabricação/modificação de mensagens
 - Negação de serviço
 - Replay/delay
 - Internos ou Externos

11

11

Ferramentas Criptográficas

- Confidencialidade e cifras simétricas
- Autenticidade e funções *Hash*
- Cifras assimétricas
- Assinatura digital e gerenciamento de chaves
- Números aleatórios e pseudoaleatórios
- Proteção de dados armazenados

12

12

Criptografia

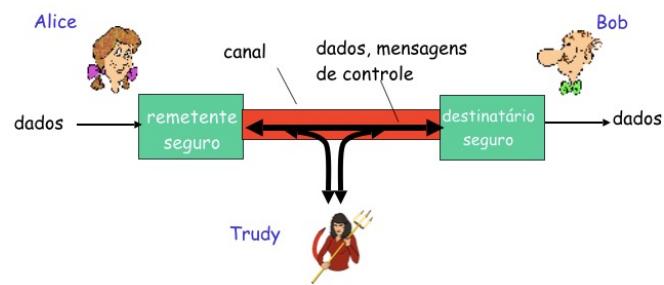
- Definição: escrita oculta ou secreta
 - “[de cript(o) + grafia.] S.f. 1. Arte de escrever em cifra ou em código. 2. Conjunto de técnicas que permitem criptografar informações (como mensagens escritas, dados armazenados ou transmitidos pelo computador, etc.).” (Dicionário Aurélio)
 - Investigação de métodos e técnicas que podem ser usadas para esconder o conteúdo de uma mensagem (texto em claro), produzindo um criptograma (texto incompreensível para um leitor desautorizado)
- Cifragem
 - Entrada: texto em claro + chave de cifragem
 - Saída: criptograma
 - Processo reverso: decifragem

13

13

Criptografia

- Amigos e inimigos: Alice, Bob, Trudy
 - Bem conhecidos no mundo da segurança em rede
 - Bob, Alice (amigos!) querem se comunicar “com segurança”
 - Trudy (intrusa) pode interceptar, excluir, acrescentar mensagens



14

14

Criptografia

- Técnicas
 - Esteganografia
 - Criptografia clássica
 - Transposição
 - Substituição
 - Criptografia moderna
 - Cifras simétricas
 - Gerenciamento de chaves
 - Cifras assimétricas
 - Funções hash

15

15

Criptografia

Criptoanálise

- Computacionalmente Seguro
- Força Bruta
 - Testar todas as chaves possíveis até obter um **texto legível**

Tamanho da chave (bits)	Número de chaves possíveis	Tempo requerido em 1 decifração/μs	Tempo requerido em 10^6 decifrações/μs
32	$2^{32} = 4,3 \times 10^9$	$2^{31} \mu\text{s} \times 35,8 \text{ minutos}$	2,15 milissegundos
56	$2^{56} = 7,2 \times 10^{16}$	$2^{55} \mu\text{s} = 1.142 \text{ anos}$	10,01 horas
128	$2^{128} = 3,4 \times 10^{38}$	$2^{127} \mu\text{s} = 5,4 \times 10^{24} \text{ anos}$	$5,4 \times 10^{18} \text{ anos}$
168	$2^{168} = 3,7 \times 10^{50}$	$2^{167} \mu\text{s} = 5,9 \times 10^{36} \text{ anos}$	$5,9 \times 10^{30} \text{ anos}$
26 caracteres (permutação)	$26! = 4 \times 10^{26}$	$2 \times 10^{26} \mu\text{s} = 6,4 \times 10^{12} \text{ anos}$	$6,4 \times 10^6 \text{ anos}$

16

16

Criptografia

Criptoanálise

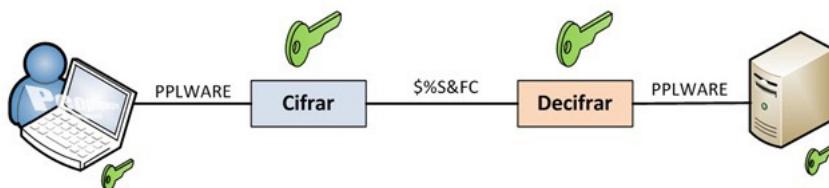
- Reduzir o esforço da força bruta
 - Ataques de dicionário
 - Técnicas de criptoanálise
 - Depende da natureza do algoritmo
 - Utiliza amostras de criptogramas e até mesmo textos em claros
- O sucesso da criptoanálise compromete mensagens antigas e novas
 - Chaves de sessão
- O foco são as fraquezas inerentes aos algoritmos
 - Existem outros aspectos que podem resultar em falhas de segurança
 - Como os algoritmos são utilizados?
 - Gerenciamento e distribuição de chaves?

17

17

Cifras Simétricas

- Esquema de criptografia
 - A chave de cifragem é igual a chave de decifragem (distribuição de chaves)



- Os algoritmos são eficientes (rápidos)
- Algoritmos de conhecimento público
 - Segurança reside no sigilo da chave

18

18

Cifras Simétricas

Cifras de Blocos

- Conversão de um texto em claro (bloco) de m bits em um criptograma de mesmo tamanho.
 - Tamanho do bloco é fixo
 - Tamanho da chave pode ser diferente do tamanho do bloco
- Mensagens de tamanhos variáveis
 - Divisão do texto em claro em blocos antes da cifragem
 - É necessário um número inteiro de blocos
 - Enchimento (*Padding*)
 - Modos de operação

19

19

Cifras Simétricas

Cifras de Blocos

- Enchimento (RFC 1321)
 - Inserir bit *1* no final
 - Inserir zero ou mais *0s* até completar um bloco
 - Operação inversa
 - Retirar os *0s* menos significativos
 - Retirar o *1*
 - Exemplos (blocos de tamanho 4)

Original	0101010111	10010110	1101001
Padding	010101011110	100101101000	11010011

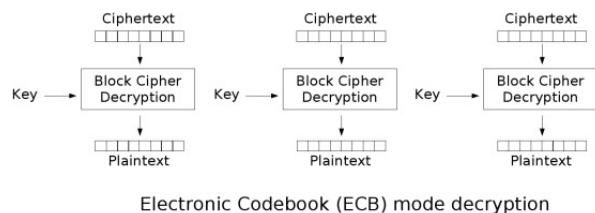
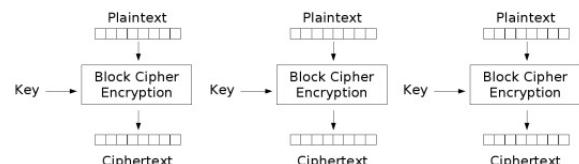
20

20

Cifras Simétricas

Cifras de Blocos

- Modos de operação
 - ECB (*electronic codebook*)
 - Divisão em um número inteiro de blocos
 - Blocos cifrados de forma independente
 - Concatenação



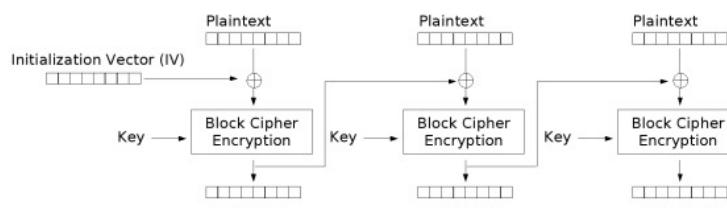
21

21

Cifras Simétricas

Cifras de Blocos

- Modos de operação
 - CBC (*Cipher Block Chaining*)
 - A saída da cifragem de um bloco é usada na cifragem do bloco seguinte
 - Cifragem: $c_i = e_k(p_i \oplus c_{i-1})$
 - c_0 : Vetor de inicialização



Cipher Block Chaining (CBC) mode encryption

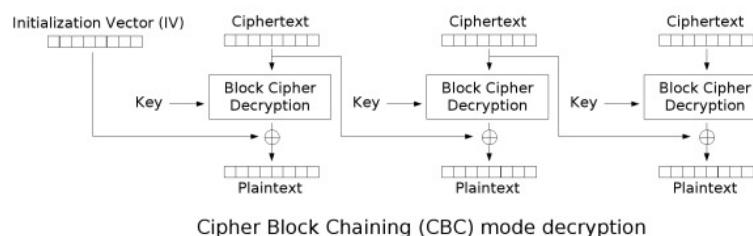
22

22

Cifras Simétricas

Cifras de Blocos

- Modos de operação
 - CBC (*Cipher Block Chaining*)
 - Decifragem: $p_i = d_k(c_i) \oplus c_{i-1}$
 - c_0 : Vetor de inicialização



23

23

Cifras Simétricas

Cifras de Blocos

- Princípios de projeto das cifras
 - Aleatoriedade
 - Sem correlação entre a entrada e a saída
 - Substituição, Transposição e Operação com a chave
 - Execução em rodadas
 - Efeito avalanche
- Exemplos

	DES	Triple DES	AES
Tamanho do bloco de texto às claras (bits)	64	64	128
Tamanho do bloco de texto cifrado (bits)	64	64	128
Tamanho da chave (bits)	56	112 ou 168	128, 192 ou 256

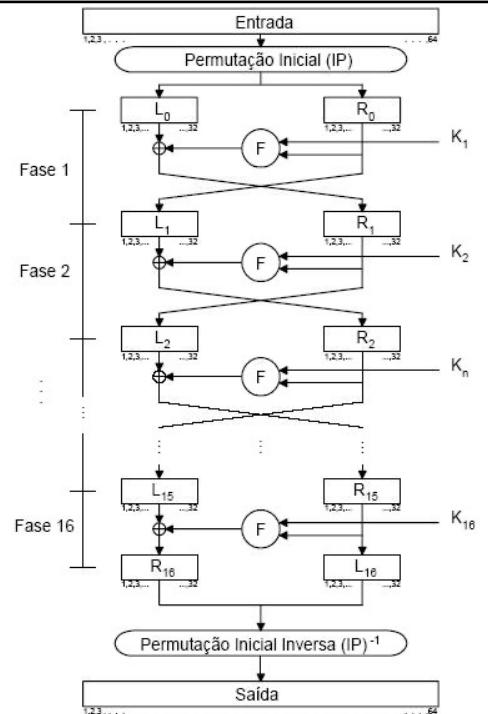
24

24

Cifras Simétricas

Cifras de Blocos

- DES (*Data Encryption Standard*)
 - FIPS 46
 - Aprovado em 1976 pelo NBS (Atual NIST)
 - Padrão mundial por mais de 20 anos
 - Teve de ser substituído devido ao tamanho da chave
 - Baseado no Lucifer (IBM)
 - Estrutura de Feistel
 - <http://www.cs.bham.ac.uk/research/projects/le/msys/DES/DESPage.jsp>



25

Cifras Simétricas

Cifras de Blocos

- DES (*Data Encryption Standard*)
 - Permutação inicial
 - Divisão em L_0 e R_0
 - 16 rodadas da Estrutura de Feistel
 - $f(R_{i-1}, k_i) = P(S(E(R_{i-1}) \oplus k_i))$
 - E(X): função de expansão (entrada: 32 bits, saída: 48 bits)
 - S(X): função de substituição (8 caixas-S)
 - caixa-S: entrada de 6 bits e saída de 4 bits
 - P(X): função de permutação (32 bits)
 - Gerador de 16 sub-chaves k_i de 48 bits
 - Concatenação: $R_{16} | L_{16}$
 - Permutação inicial inversa

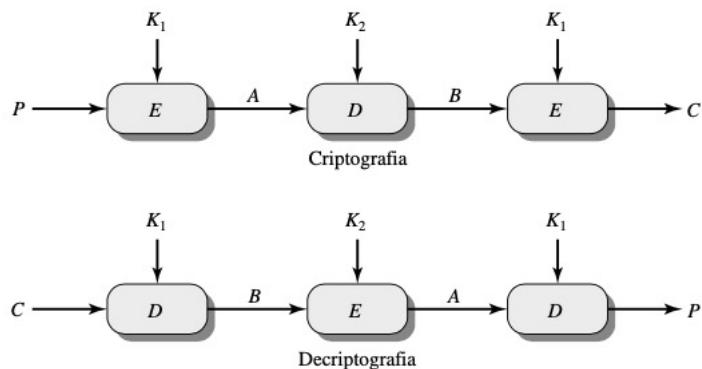
26

26

Cifras Simétricas

Cifras de Blocos

- Triplo DES
 - FIPS 46-3
 - 2 ou 3 Chaves
 - Compatibilidade com DES
 - Lento



27

27

Cifras Simétricas

Cifras de Blocos

- AES (*Advanced Encryption Standard*)
 - Blocos de 128 bits
 - Chaves de 128, 192 e 256 bits
 - 10, 12 ou 14 rodadas
 - Competição 1998
 - 15 candidatos
 - Finalistas (1999): MARS, RC6, Serpent, Twofish e Rijndael
 - 2001: O algoritmo de Rijndael foi eleito como AES (FIPS 197)
 - Belgas: Vincent Rijmen e Joan Daemen
 - Critérios: Segurança, desempenho, facilidade de implementação e flexibilidade

■ <https://www.youtube.com/watch?v=mlzxpkdXP58&feature=youtu.be>

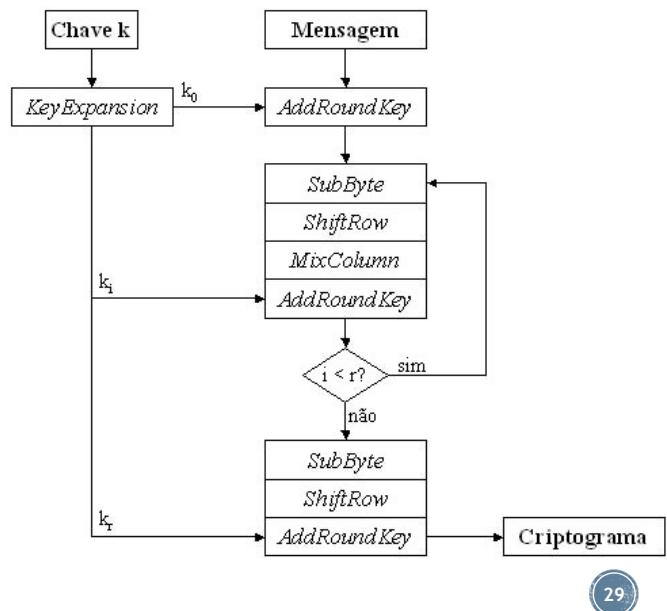
28

28

Cifras Simétricas

Cifras de Blocos

- AES (*Advanced Encryption Standard*)
 - Cifragem diferente das decifragem
 - Uso de funções inversas
 - Decifragem na ordem inversa
 - Cada rodada composta por 4 operações
 - Função *KeyExpansion*
 - Gerar as $r+1$ sub-chaves

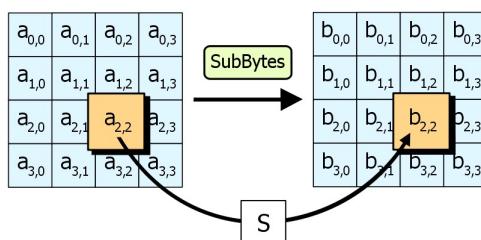


29

29

A Cifra AES

- Operações
 - *SubByte*: transformação não-linear (substituição de 1 byte)
 - Os 4 bits da esquerda determinam a linha e os 4 bits da direita a coluna
 - Usa uma tabela 16x16 com todas as 256 permutações de 8 bits possíveis
 - Caixa-S construída usando transformações em GF(2⁸)
 - Resistente aos ataques conhecidos



05 - Advanced Encryption Standard

30

A Cifra AES

- SubByte: Caixa-S

		y																
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
x		0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
x		1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
x		2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
x		3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
x		4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
x		5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
x		6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
x		7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
x		8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
x		9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
x		A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
x		B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
x		C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
x		D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
x		E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
x		F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

05 - Advanced Encryption Standard

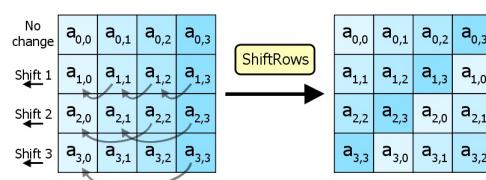


31

A Cifra AES

- Operações

- ShiftRow: permutação dos 16 bytes de entrada
 - Permutação de bytes entre as colunas



- Como é a operação inversa?

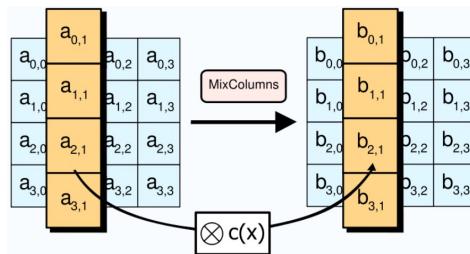
05 - Advanced Encryption Standard



32

A Cifra AES

- Operações
 - *MixColumn*: transformação linear (multiplicação de matrizes)
 - Embaralhar os bytes em de uma mesma coluna



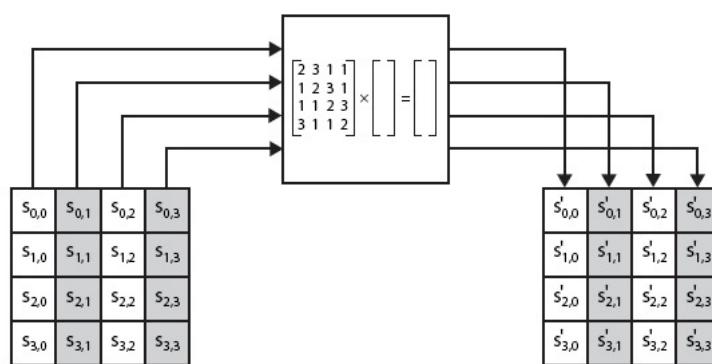
05 - Advanced Encryption Standard



33

A Cifra AES

- *MixColumn*
 - Módulo $m(x) = x^8 + x^4 + x^3 + x + 1$



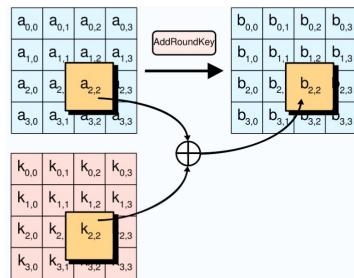
05 - Advanced Encryption Standard



34

A Cifra AES

- Operações
 - *AddRoundKey*: soma módulo dois (XOR) com a chave da rodada



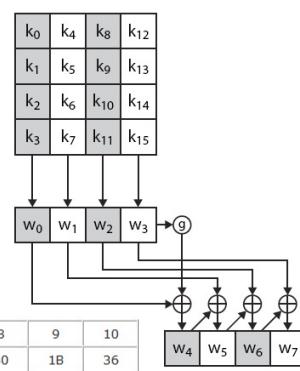
05 - Advanced Encryption Standard



35

A Cifra AES

- Expansão da chave
 - Produção de um vetor com 44/52/60 palavras
 - Chave: 4 palavras iniciais
 - $W_i = W_{i-1} \oplus W_{i-4}$
 - Função g
 - Rotação da palavra (para cima)
 - Substituição de cada byte
 - XOR com Rcon[j]
 - $Rcon[j] = (RC[j], 0, 0, 0)$

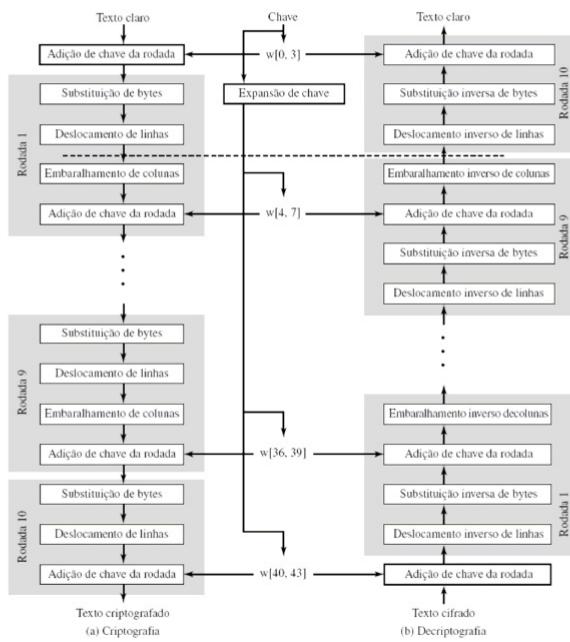


05 - Advanced Encryption Standard



36

A Cifra AES



05 - Advanced Encryption Standard

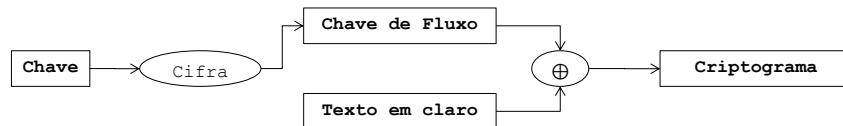


37

Cifras Simétricas

Cifras de Fluxo

- Características
 - Mais eficientes do que as cifras de bloco
 - Geração da chave de cifragem (*stream key*) do tamanho da mensagem (em bytes)
 - Gerador pseudo-aleatório de uma sequência de bytes baseado na chave secreta
 - Geração continua de bytes
 - O algoritmo de decifragem é igual ao de cifragem
 - O reuso de chaves não é recomendado



38

38

19

Cifras Simétricas

Cifras de Fluxo

- RC4 (Rivest Cipher 4)
 - RSA Data Security, Inc. (1987)
 - Chave de 1 a 256 bytes
 - Tamanho sugerido de pelo menos 16 bytes (128 bits)
 - Período da cifra maior que 10^{100}
 - KSA (*Key-Scheduling Algorithm*)
 - Gera, com base na chave, um vetor com todas as permutações possíveis de um byte
 - PRGA (*Pseudo-Random Generation Algorithm*)
 - Seleciona um byte e atualiza o vetor
 - Usado no WEP (*Wired Equivalent Privacy*)
 - <https://www.youtube.com/watch?v=KM-xXYZXElk>
- A5 (Usado no GSM)

39

39

Cifras Simétricas

Cifras de Fluxo

- Manipulação de um vetor S de 256 bytes
 - Com todas as permutações de 8 bits possíveis
- Algoritmos
 - KSA
 - Inicialização de S com os valores de 0 a 255 em ordem crescente
 - Inicialização de um vetor temporário T de 256 bytes baseado na chave
 - A chave é repetida, periodicamente, caso seja menor que 256 bytes
 - Permutação inicial de S


```
/* Permutação inicial de S */
j = 0;
for i = 0 to 255 do
    j = (j + S[i] + T[i]) mod 256;
    Swap (S[i], S[j]);
```

40

40

20

Cifras Simétricas

Cifras de Fluxo

- Algoritmos

- PRGA

- Seleciona o byte de saída e atualiza o vetor S
 - A chave não é mais utilizada

```
/* Geração de fluxo */
i, j = 0;
while (true)
    i = (i + 1) mod 256;
    j = (j + S[i]) mod 256;
    Swap (S[i], S[j]);
    t = (S[i] + S[j]) mod 256;
    k = S[t];
```

41

41

Cifras Simétricas

- Problema da distribuição de chaves

- Chave de Seção
 - Confiança em uma terceira entidade
 - Viabilidade?

- Algoritmos de criptografia não são utilizados apenas para prover **confidencialidade**

42

42

Cifras Simétricas

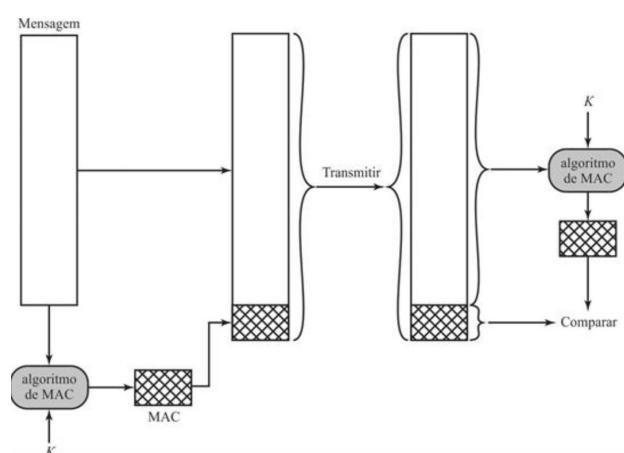
- Cifras simétricas oferecem
 - Prevenir violações de confidencialidade
 - Autenticação de mensagens
 - Detectar violações de integridade e autenticidade
 - O destinatário consegue reconhecer a mensagem decifrada como válida
 - A decifragem de um criptograma alterado é ilegível
 - Apenas o uso de criptografia é insuficiente para garantir se uma mensagem é autêntica
 - Ataque de reprodução (*replay*)
 - Inclusão de um *timestamp* / número de sequência
 - ECB - Alteração na ordem de blocos
 - A confidencialidade pode não ser um requisito necessário
 - Dados de portais da transparéncia

43

43

Autenticação de mensagem

- Código de Autenticação de Mensagem
 - Integridade e Autenticidade
 - Usado mesmo quando a confidencialidade é necessária
 - Algoritmo
 - Usa uma chave secreta e compartilhada
 - Não prover a irretratabilidade
 - Calcular o MAC
 - Tamanho fixo
 - Independente do tamanho da entrada
 - Verificar o MAC recebido
 - Pode utilizar *funções hash*

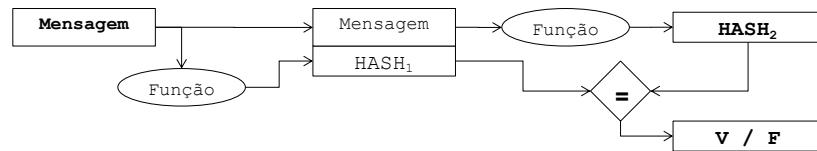


44

44

Função Hash

- Função de mão única
 - Sem inversa
 - Eficiência
 - Saída de tamanho fixo, independente do tamanho da entrada
 - **Não utiliza chave**
- Cálculo e verificação de um *hash* (resumo)



45

45

Função Hash

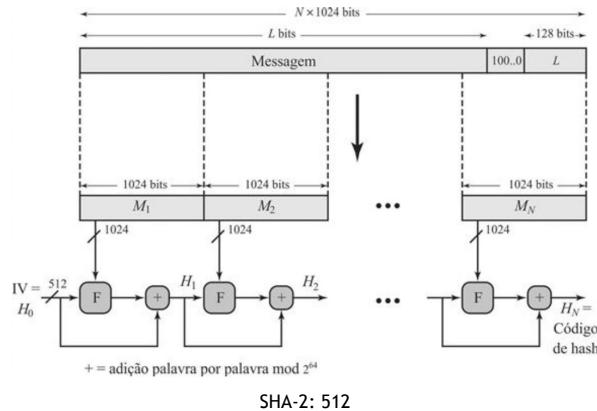
- Requisitos
 - Baixa colisão → poucas mensagens geram o mesmo *hash*
 - Segurança
 - *Resistência à pre-imagem*: Dado h , é difícil encontrar x tal $H(x) = h$
 - *Colisão fraca (segunda pre-imagem)*: Dado x , é difícil encontrar y tal $H(x) = H(y)$
 - *Colisão forte*: Dificuldade de encontrar duas mensagens quaisquer com o mesmo *hash*
 - Mudança de um bit altera completamente o código *hash*
- Criptoanálise
 - Força Bruta
 - Ataque do Aniversário
- *Funções Hashs* vs *Checksums*

46

46

Função Hash

- **Secure Hash Algorithm**
 - Primeiras versões insecuras
 - **SHA-2**
 - 224, 256, 384 ou 512 bits
 - **SHA-3**
 - Concurso finalizado em 2012
 - Tamanhos: 224, 256, 384 ou 512 bits



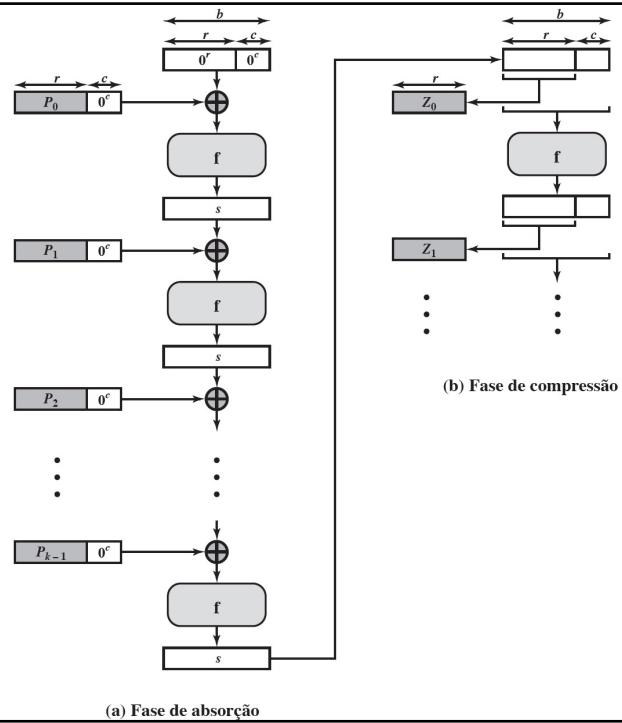
▪ <https://youtu.be/FZeLKrQTZtE?list=PLvvVTirhNtUfuW5hzimbe2cvKPnouhGj>

47

47

Função Hash

- **SHA-3**
 - Paradigma esponja
 - f - 24 rodadas
 - Permutações e substituições



48

48

Função Hash

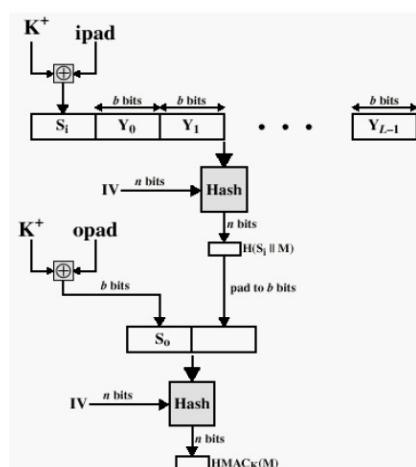
- Aplicações
 - Autenticidade de Mensagens
 - Criptografar o HASH
 - HMAC
 - Senhas
 - Integridade de arquivos

49

49

Autenticação de mensagem

- CMAC
 - Usado no WPA
- HMAC: Keyed-Hash Message Authentication Code
 - RFC 2104 e FIPS 198
 - Usados em protocolos como o IPSec e SSL



50

50

Autenticação de entidades

- Baseada no que o usuário sabe
 - Senhas
 - Problemas: senhas fracas, divulgação, captura, advinhação...
- Baseada no que o usuário tem
 - Cartões, tokens
 - Problemas: custo, roubo...
- Baseada nas características do usuário
 - Biometria
 - Problemas: algoritmos de identificação (mudanças do usuário)
- Combinação de técnicas
 - Autenticação em dois fatores

51

51

Problema da distribuição de chaves

- Artigo: “*New Directions in Cryptography*” (1976)
- Protocolo de Diffie-Hellman
 - Usado na prática com números grandes
 - Alice e Bob precisam está conversando
 - Espera para troca de valores
- Conceito de chave pública (cifra assimétrica)
 - Menos eficientes
 - Como usar na prática?
 - Baseadas em funções matemáticas reversíveis sob circunstâncias especiais
 - Geração de um par de chaves
 - Irretratabilidade e Assinatura digital

52

52

Protocolo de Diffie-Hellman

- Baseado em aritmética modular
 - Função de mão única (sem inversa)
 - $Y^x \pmod{P}$ - Exemplo: $3^x \pmod{5} = 1$
- Algoritmo
 - Alice e Bob entram em acordo com valores de Y e P
 - Podem ser divulgados
 - Cada um escolhe um valor de x , secreto
 - Para diferenciar, Alice escolhe um valor A e Bob um valor B
 - Alice e Bob executam a função para o valor escolhido
 - E enviam seus resultados (α e β) pelo meio inseguro
 - Alice calcula $\beta^A \pmod{P}$ e Bob $\alpha^B \pmod{P}$ calcula
 - Resultados são iguais (chave)

53

53

Protocolo de Diffie-Hellman

- Exemplo

Fase	Alice	Bob
1. Escolha de Y e P	$Y = 7$ e $P = 11$	
2. Escolha de x (secreto)	$A = 3$	$B = 6$
3. Calcular $Y^x \pmod{P}$	$\alpha = 7^3 \pmod{11} = 2$	$\beta = 7^6 \pmod{11} = 4$
4. Trocar α e β		
5. Calcular a chave	$\beta^A \pmod{P} = \alpha^B \pmod{P}$	
	$4^3 \pmod{11} = 9$	$2^6 \pmod{11} = 9$

54

54

Cifras Assimétricas

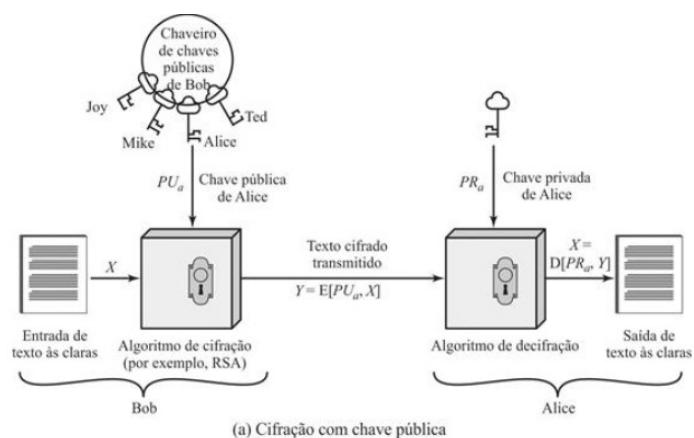
- Princípios de uma cifra de chave pública
 - Cada usuário possui um par de chaves
 - Algoritmo eficiente de geração das chaves
 - A chave privada é mantida em segredo
 - A chave pública é divulgada
 - Repertório com chaves públicas (cartório)
 - Chave privada dificilmente obtida a partir da chave pública
 - Criptoanálise
 - Cifragem eficiente com uma das chaves e decifragem eficiente com a outra
 - Confidencialidade
 - Irretratabilidade (autenticidade e integridade)
 - Cifras de blocos
 - Tamanho de chave maior que nas cifras simétricas

55

55

Cifras Assimétricas

- Confidencialidade

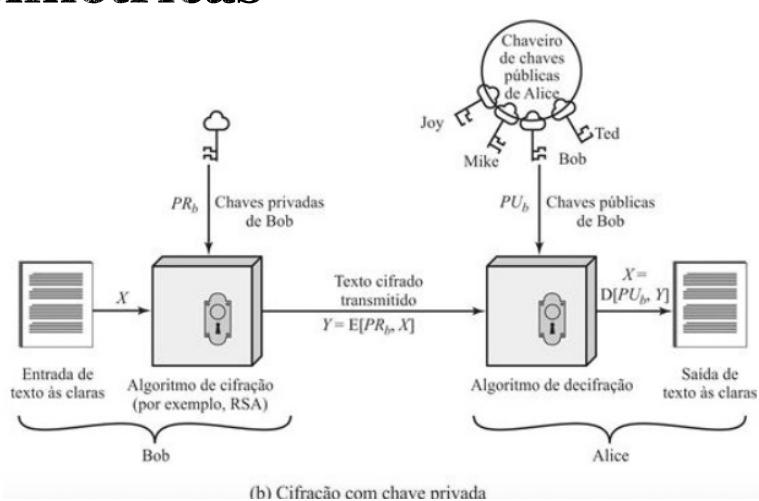


56

56

Cifras Assimétricas

- Irretratabilidade
- Integridade
- Autenticidade



57

57

Cifras Assimétricas

- RSA
 - Geração do par de chaves
 - Escolhe dois números primos p e q (mantidos em segredo)
 - $N = p * q$
 - Teste de primalidade
 - Chave pública (e, N)
 - Escolhe um número e , primo relativo a $(p-1) * (q-1)$
 - Chave privada (d, N)
 - $e * d = 1 \text{ mod } (p-1)*(q-1)$
 - Utiliza-se algoritmo de Euclides Estendido
 - <https://www.youtube.com/watch?v=ygo3Xa06tZc>

58

58

Cifras Assimétricas

- Exemplo RSA
 - Geração do par de chaves
 - $p = 17$ e $q = 11$
 - $N = p * q = 187$
 - $e = 7$
 - $\text{MDC}(160, 7) = 1$
 - $e * d = 1 \pmod{(p-1)(q-1)}$
 - $7 * d = 1 \pmod{160}$
 - $d = 23$
 - $K_p = \{7, 187\}$
 - $K_s = \{23, 187\}$

59

59

Finding Inverses

```

EXTENDED_EUCLID( $m, b$ )
1. ( $A_1, A_2, A_3 = (1, 0, m)$  ;
    $(B_1, B_2, B_3 = (0, 1, b)$ )
2. if  $B_3 = 0$ 
   return  $A_3 = \text{gcd}(m, b)$ ; no inverse
3. if  $B_3 = 1$ 
   return  $B_3 = \text{gcd}(m, b)$ ;  $B_2 = b^{-1} \pmod{m}$ 
4.  $Q = A_3 \text{ div } B_3$ 
5. ( $T_1, T_2, T_3 = (A_1 - Q B_1, A_2 - Q B_2, A_3 - Q B_3)$ 
6. ( $A_1, A_2, A_3 = (B_1, B_2, B_3)$ 
7. ( $B_1, B_2, B_3 = (T_1, T_2, T_3)$ 
8. goto 2

```

59

60

Cifras Assimétricas

- RSA

- Cifragem e Decifragem

- Conversão da mensagem em números/blocos (menores que N)
 - $C = M^e \pmod{N}$
 - $M = C^d \pmod{N}$

61

61

Cifras Assimétricas

- Exemplo RSA

- Cifragem

- $K_p = \{7, 187\}$ e $K_s = \{23, 187\}$
 - $M = 88$ (caractere 'X' em ASCII)
 - $C = M^e \pmod{N}$
 - $C = 88^7 \pmod{187} = 88^1 * 88^2 * 88^4 \pmod{187}$
 - $C = 88 * 77 * 132 \pmod{187} = 11$

- Decifragem

- $M = C^d \pmod{N}$
 - $M = 11^{23} \pmod{187} = 11^1 * 11^2 * 11^4 * 11^{16} \pmod{187}$
 - $M = 11 * 121 * 55 * 154 \pmod{187} = 88$

62

62

Cifras Assimétricas

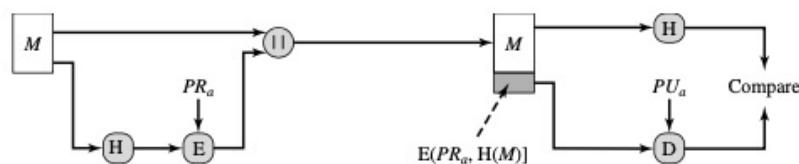
- RSA
 - É possível obter a chave privada a partir da chave pública?
 - A segurança reside na dificuldade de fatorar número grandes
 - Limite de segurança
 - Tempo para fatorar N
 - Chaves de pelo menos 1024 bits
 - Segurança equivalente a uma cifra de bloco de 80
- Outros algoritmos
 - El-Gamal
 - Intratabilidade do problema do logaritmo discreto
 - $Y = A^x \pmod{B}$ → calcular x , conhecendo Y, A e B
 - Algoritmos baseados em curvas elípticas

63

63

Assinatura digital

- O uso de cifras assimétricas para prover integridade, autenticidade e irretratabilidade é ineficiente
 - Cifragem usando chave pública é custosa
 - Se for necessário prover confidencialidade a cifragem deve ser dupla
- Solução:
 - Função Hash + Cifra Assimétrica



64

64

Assinatura digital

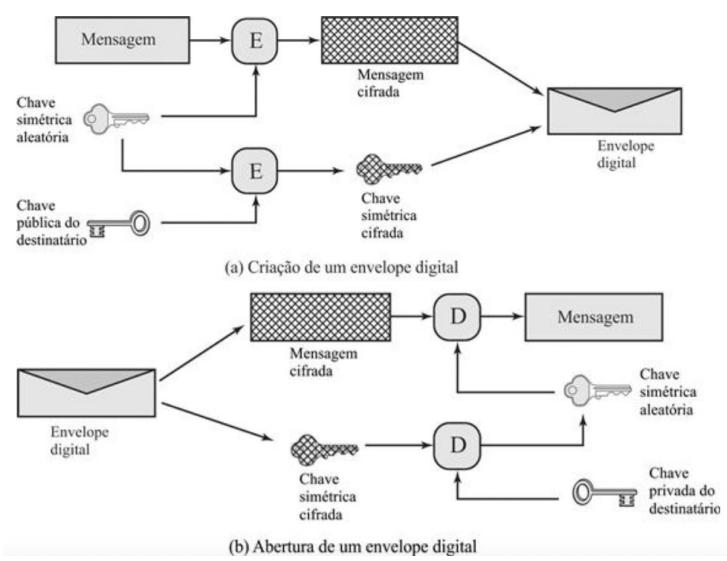
- Solução: algoritmo exclusivo para assinatura digital
 - DSS (*Digital Signature Standard*)
 - FIPS 186-3
 - Algoritmo
 - Cálculo do *hash* utilizando o SHA
 - Assinatura realizada com o DSA (*Digital Signature Algorithm*)
 - Algoritmo usado apenas para assinatura
 - Não pode ser usado em criptografia ou troca de chaves
 - DSA
 - Geração de chaves
 - Assinatura
 - Verificação

65

65

Gerenciamento de chaves

- Envelope digital
 - Apenas confidencialidade
- Como oferecer também a irretratabilidade?



66

Mecanismos vs Serviços de Segurança

	Cifra Simétrica	Cifra Assimétrica	Diffie Hellman	MAC	Assinatura digital
Confidencialidade	✓	✓			
Integridade	✓	✓		✓	✓
Autenticidade	✓	✓		✓	✓
Irretratabilidade		✓			✓
Troca de chaves	✓	✓	✓		

Protocolos de Segurança:

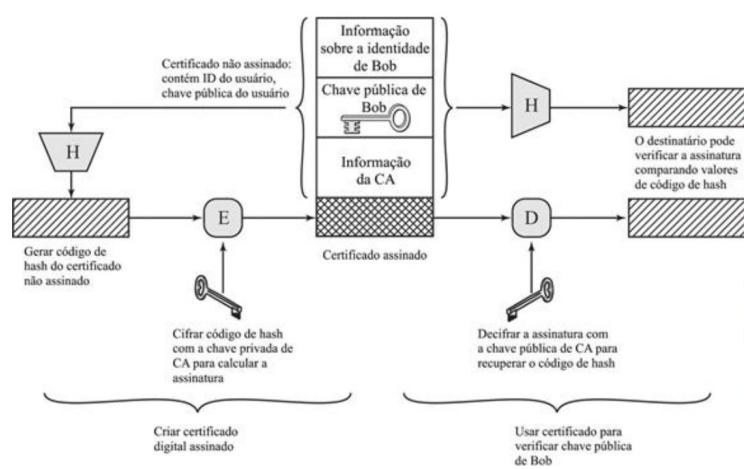
Combinações de algoritmos para prover os serviços desejados

67

67

Certificação digital

- E se alguém fingir ser outra pessoa?
- Certificados X.509
 - Quais informações?
- Autoridade Certificadora
 - Cartório Digital
 - Terceiro confiável
- <https://badssl.com/>



68

68

Certificação digital

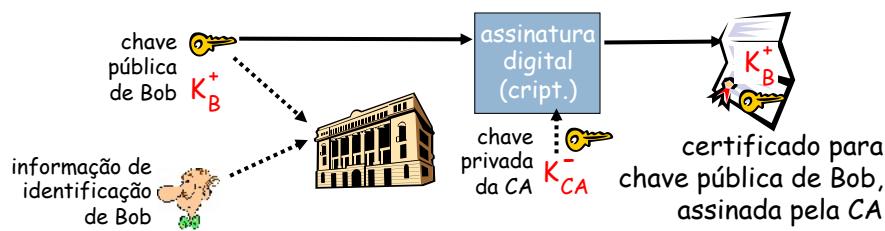
- Elementos de um certificado X.509
 - Versão
 - Número do certificado
 - Único para uma determinada AC
 - Tipo do Algoritmo
 - Algoritmo e função *hash* usada pela AC para assinar o certificado
 - Nome do titular
 - Nome do AC
 - Período de validade
 - Informações de chave pública da entidade (chave pública e algoritmo)
 - Assinatura da AC
 - Extensões

69

69

Autoridades de certificação

- **autoridade de certificação (CA):** vincula chave pública à entidade particular, E.
- E (pessoa, roteador) registra sua chave pública com CA.
 - E fornece “prova de identidade” à CA.
 - CA cria certificado vinculando E à sua chave pública.
 - certificado contendo chave pública de E assinada digitalmente pela CA - CA diz “esta é a chave pública de E”



69

70

Certificação digital

- Infraestrutura de chave pública (ICP)
 - Definição (RFC 3647)
 - “Uma ICP pode ser definida como um conjunto de hardware, software, pessoas, políticas e procedimentos necessários à criação, gerenciamento, armazenamento, distribuição e revogação de certificados, baseados em criptografia de chave pública.”
 - ICP-Brasil
 - Infra-estrutura hierárquica
- Outra solução
 - Cadeia de confiança (PGP)

71

71

Números aleatórios

- Aplicações
 - As saídas dos algoritmos simétricos devem ser aleatórias
 - Geração de chaves
 - Simétricas temporárias
 - Cifras assimétricas
 - Impedir ataques de repetição
- Geradores de números pseudoaleatórios
 - Sequência de números a partir de uma semente
 - Resultados iguais para uma mesma semente
 - Aleatoriedade e Imprevisível
 - A frequência de cada número deve ser aproximadamente a mesma
 - Não é possível definir o próximo elemento a partir de todos os anteriores
 - Testes de aleatoriedade

73

73

Proteção de Dados Armazenados

- O foco é normalmente com a proteção de dados em trânsito
- Controle de acesso dos sistemas operacionais é limitado
- Sistemas de Backups
- Proteção de chaves armazenadas
- Exemplos de aplicações
 - PGP
 - TrueCrypt

74

74

Trabalho de implementação

- Desenvolver um programa assinar e verificar assinaturas digitais
 - Usuário fornece o texto (ou arquivo) e a chave
 - Geração das chaves assimétricas
 - Função Hash
 - Linguagem livre
 - Uso de bibliotecas (Bouncy Castle, PyCrypto, etc)
- <https://www.devgan.com/cryptotools/cryptography-tools>

75

75