



Camada de Apresentação do RM-OSI

Andressa Silva – dressyh@live.com

Matheus Cunha Reis – matheuscunhareis30@gmail.com

Matheus dos Santos Mendes – matheusmendes1@hotmail.com

Ronistone Gonçalves Júnior – ronistonejunior@gmail.com



Sumário

1. Introdução
2. Problemas de Design da Camada de Apresentação
3. Abstract Syntax Notation 1 (ASN.1)
4. Técnicas de Compressão de Dados
5. Criptografia
6. Exemplos



Introdução

- Camada sem função muito bem definida.
- Camada onde conversões eram feitas para permitir a comunicação entre máquinas ASCII e máquinas EBCDIC.
- Permite que programas orientados visualmente fossem utilizados em diversos terminais.
- Representação, conversão, criptografia e compressão de dados.
- “Camada de Representação”

Introdução

7-Aplicação

Interfaces com aplicativos

6-Apresentação

Formatos / Criptografia

5-Sessão

Controle de Sessões entre Aplicativos

4-Transporte

Conexão entre hosts / Portas

3-Rede

Endereço lógico / Roteadores

2-Enlace de Dados

Endereço físico / Pontes e Switches

1-Física

Hardware / Sinal elétrico / bits



Introdução



- Camadas inferiores lidam com a movimentação ordenada de bits.
- Foco em preservar o significado das informações independente das configurações das máquinas envolvidas.
- São feitos acordos e conversões para que assegurar que diferentes máquinas possam se “entender”.
- Responsabilidade de codificar estruturas de dados complexas desde o formato interno usado no equipamento transmissor para um fluxo de bits adequado à transmissão e depois decodificá-los na representação exigida no destino.



8.1 Problemas de Design da Camada de Apresentação

➤ A camada possui 4 funções principais :

1. Permitir ao usuário uma forma de executar primitivas do serviço de sessão.
2. Fornecer meios de especificar estruturas de dados complexas.
3. Gerenciar o conjunto de estruturas de dados atualmente exigido.
4. Converter dados entre os formatos interno e externo.



8.1.1 Representação de Dados

- Diferentes computadores possuem diferentes representações de dados.
- Código de Caracteres EBCDIC e ASCII
- Aritmética de complemento 16-32 bits e aritmética de complemento 60 bits
- Chips Intel numeram seus bytes da direita pra esquerda, já alguns chips motorola numeram na ordem inversa
- Fabricantes seguem seus padrões para evitar incompatibilidade entre seus próprios produtos.



8.1.1 Representação de Dados

- Exemplo de problema na representação de Dados :
- Considere duas máquinas : máquina 1 e máquina 2.
- Uma delas utiliza complemento de um, e a outra, complemento de dois.
- Máquina 1 transmite um array de inteiros de 16 bits, bit a bit.
- Mesmo que todos os dados sejam recebidos sem erro é possível que resultem em valores diferentes.
- Por exemplo no padrão de bits FFF0 será impresso como -15 na máquina de complemento de um e com -16 na máquina de complemento de dois.



8.1.2 Compressão de Dados

- Na maioria dos casos, o custo de usar uma rede depende da quantidade de dados enviados.
- Intuitivamente, pode-se deduzir que quanto maior o volume de tráfego de dados, maior será o custo.
- A importância da compressão de dados é justamente reduzir tal custo, diminuindo o volume de dados antes de enviá-los.
- A Compressão de Dados está intimamente ligada a sua representação.
- É utilizada também para economizar tempo e espaço em disco.



8.1.3 Segurança e Privacidade de Rede

- Antes do advento da rede, manter a segurança dos arquivos era fácil, visto que o mainframe em que os dados eram armazenados era guardado por seguranças.
- Com a expansão da rede, veio a expansão dos problemas de segurança.
- Alguns serviços de segurança podem ser imaginados :
 1. Proteger os dados de leitura por pessoas não autorizadas.
 2. Evitar que pessoas não autorizadas insiram ou excluam mensagens.
 3. Verificar o transmissor de cada mensagem.
 4. Tornar possível aos usuários enviar eletronicamente documentos assinados.



8.1.3 Segurança e Privacidade de Rede

- A Criptografia é uma alternativa para lidar com os problemas de segurança na rede.
- Sua localização é um tanto controversa no modelo OSI, visto que ela pode ser realizada em qualquer camada.
- As três camadas mais adequadas para se operar são : Camada Física, de Transporte e Apresentação.
- Física: Adicionar uma unidade criptográfica entre o computador e o meio físico. Todo bit sai criptografado, todo bit que entra é descriptografado. Esse modelo é chamado de criptografia de enlace, simples mas inflexível
- Transporte: Se existisse a criptografia ficaria disponível para todas as sessões
- Apresentação: O custo de se criptografar qualquer dado só é pago nos lugares necessários, fazendo dela a melhor escolha.

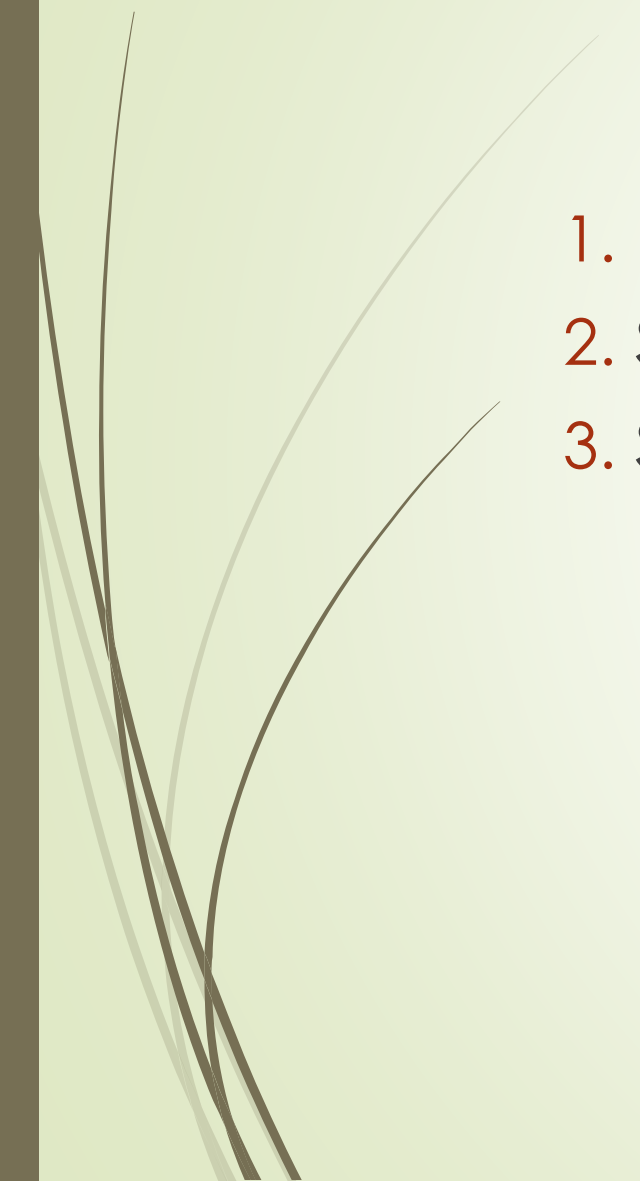


8.1.4 Primitivas do Serviço de Apresentação no OSI

- As primitivas de Serviço são idênticas às da camada de Sessão.
- Além destas, outras primitivas são usadas com a função de permitir ao usuário incluir quaisquer estruturas de dados complexas necessárias à aplicação utilizada.
- As estruturas de dados necessárias a uma aplicação podem ser coletadas em grupos, chamados de contextos.
- Gerenciar o processo de negociação em que os usuários concordam sobre qual estrutura de dados correspondem a cada contexto.



8.2 Abstract Syntax Notation 1 (ASN.1)

1. Estruturas de Dados
 2. Sintaxe Abstrata
 3. Sintaxe de Transferência
- 

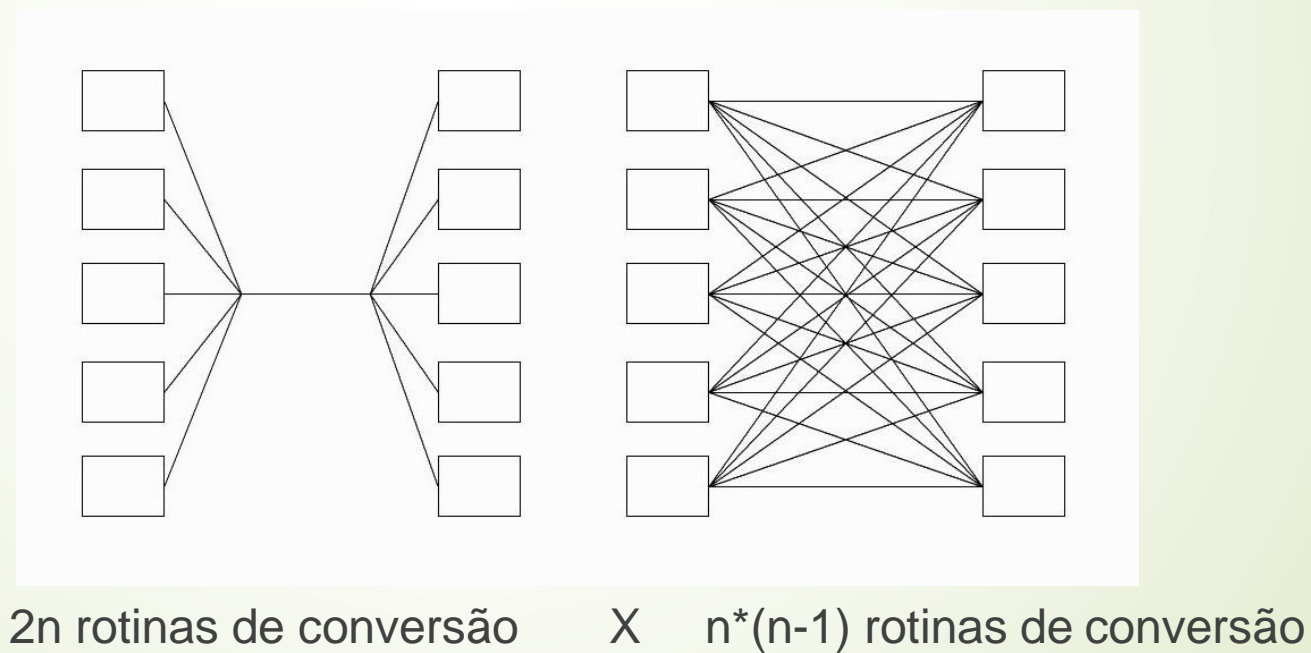


8.2.1 Estruturas de Dados

- Aplicações trocam entre si estruturas de dados complexas.
- Cada aplicação possui um conjunto de estruturas relevantes à sua operação e que podem ser transmitidas pela rede.
- São transmitidas como UDPAs(Unidade de Dados do Protocolo de Aplicações).
- Definir todos os tipos de estruturas de dados e juntá-los em um módulo.
- Quando for transmitida a estrutura ela pode ser passada juntamente com seu nome em ASN.1 para a camada de apresentação.
- A camada de apresentação receptora utiliza a identidade ASN.1 da estrutura de dados para decodificá-la.

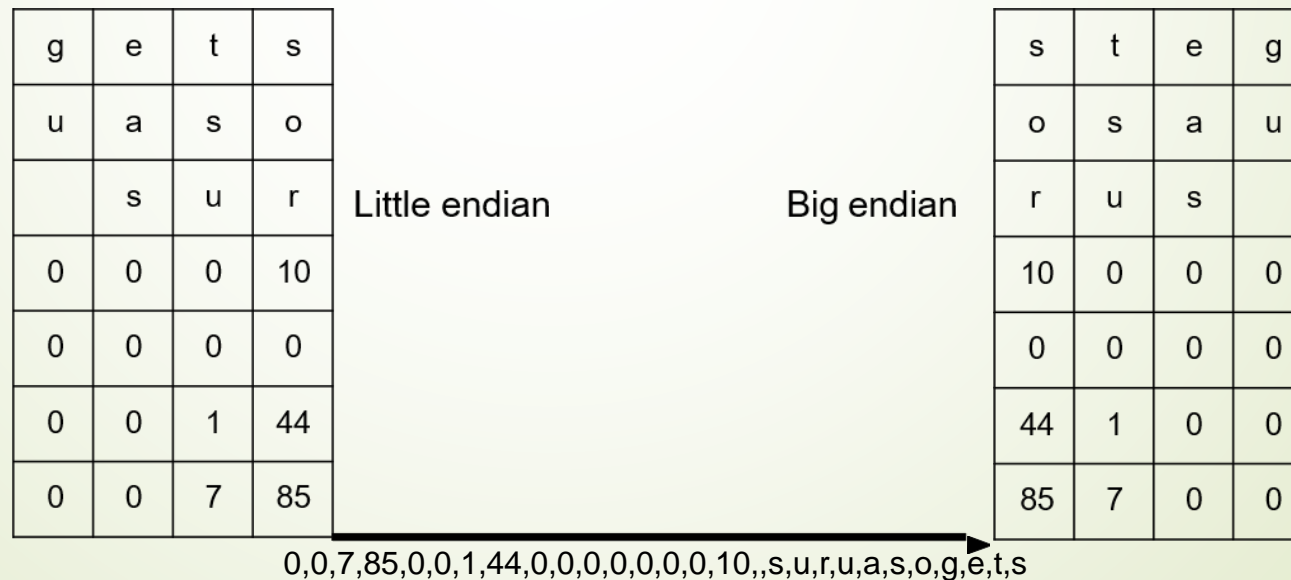
8.2.1 Estruturas de Dados

- Outra abordagem seria exigir que cada máquina esteja ciente do formato utilizado pelas outras máquinas da rede.



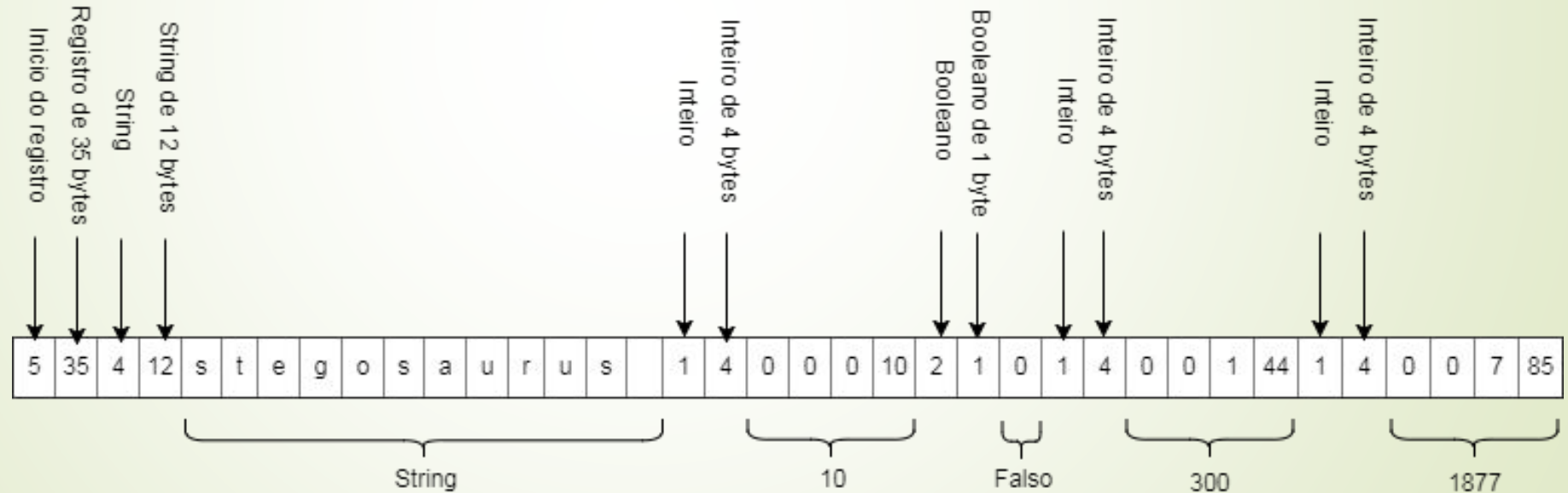
8.2.1 Estruturas de Dados

- Um problema muito complicado que a ASN.1 tem que solucionar está relacionado a ordenação dos bytes em computadores diferentes
 - Little endian - bytes numerados de forma que o byte 0 seja o de mais baixa ordem
 - Big endian - bytes numerados de forma que o byte 0 seja o de mais alta ordem



8.2.1 Estruturas de Dados

- Uma forma mais eficiente de resolver o problema é fazer com que cada tipo de dados seja auto-identificado na linha.



8.2.2 Sintaxe Abstrata

- Notação da ASN.1 usada para a definição de tipos de dados. Concebidas pela ISO destinadas exclusivamente a leitura por pessoas.
- Os tipos de dados são divididos em tipos primitivos e tipos construídos.

Tipo primitivo	Significado
INTEGER	Inteiro de tamanho arbitrário
BOOLEAN	TRUE ou FALSE
BIT STRING	Lista de 0 ou mais bits
OCTET STRING	Lista de 0 ou mais bytes
ANY	União de todos os tipos
NULL	Absolutamente nenhum tipo
OBJECT IDENTIFIER	Nome de objeto

8.2.2 Sintaxe Abstrata

- Tipos construídos são obtidos através de uma combinação por meio de construtores.

Construtor	Significado
SEQUENCE	Lista ordenada de diversos tipos
SEQUENCE OF	Lista ordenada de um único tipo
SET	Conjunto desordenado de diversos tipos
SET OF	Conjunto desordenado de um único tipo
CHOICE	Qualquer tipo individual tirado de uma determinada lista

- Existem ainda tipos predefinidos como exemplo oito tipos que são subconjuntos de um OCTET STRING, tais como o *NumericString* e *PrintableString*.



8.2.2 Sintaxe Abstrata

- Padrões internacionais costumam definir tipos de dados complexos com campos opcionais e esses campos não precisam ser transmitidos.
- É utilizado no ASN.1 o conceito de tagging onde qualquer tipo ou campo de dados pode possuir uma tag.
- São permitidos quatro tipos: UNIVERSAL, APPLICATION, PRIVATE e específica ao contexto.
- Cada tag consiste de um inteiro precedido por uma palavra reservada.
- A propósito da marcação está relacionada com as regras de codificação e com a sintaxe de transferência.
- Com o uso de tag não é necessário a transmissão do tipo, é utilizado a palavra IMPLICIT para realizar essa supressão.

8.2.2 Sintaxe Abstrata

Exemplo de sintaxe abstrata:

```
type dinossauro = record
  nome : array [1..12] of character;
  tamanho : integer;
  carnivoro : boolean;
  ossos : integer;
  descoberta: integer
end;
```

Sintaxe abstrata em Pascal

```
Dinossauro ::= SEQUENCE {
  nome      OCTET STRING, --12 caracteres
  tamanho   INTEGER,
  carnivoro BOOLEAN,
  ossos     INTEGER,
  descoberta INTEGER
}
```

Sintaxe abstrata em ASN.1

```
Dinossauro ::= [PRIVATE 6] SEQUENCE {
  nome [0]      IMPLICIT OCTET STRING, --12 caracteres
  tamanho [1]   IMPLICIT INTEGER,
  carnivoro [2] IMPLICIT BOOLEAN,
  ossos [3]     IMPLICIT INTEGER,
  descoberta [4] IMPLICIT INTEGER OPTIONAL
}
```

Sintaxe abstrata em ASN.1 com tags

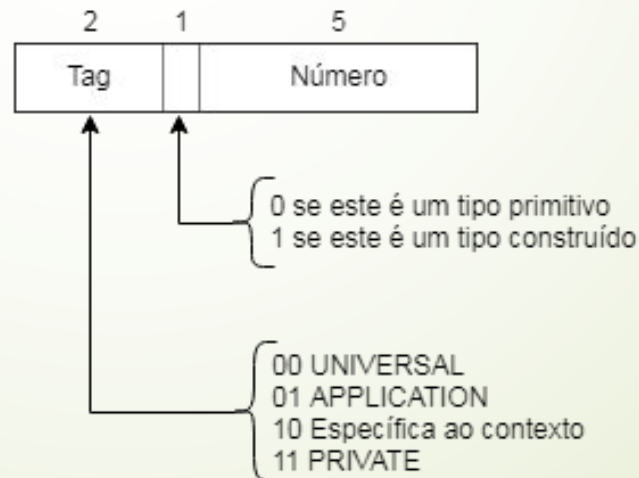


8.2.3 Sintaxe de Transferência

- É o formato do fluxo de bits.
- O ASN.1 garante precisa garantir representar as estruturas de dados na linha de transmissão de forma não ambigua.
- Cada valor transmitido consiste potencialmente em quatros campos:
 1. O identificador.
 2. O tamanho do campo de dados, em bytes.
 3. O campo de dados.
 4. O flag de fim de conteúdo.
- O primeiro campo identifica o item que o segue e possui três subcampos.

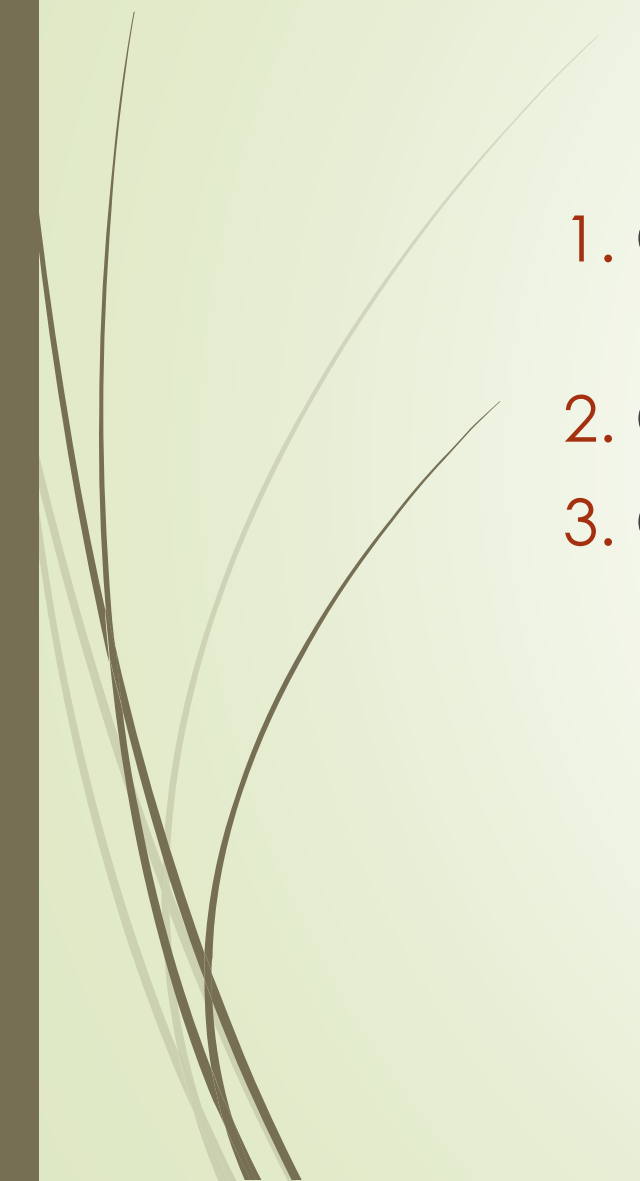
8.2.3 Sintaxe de Transferência

- Os dados podem ser passados desde a camada de aplicação até a camada de apresentação em unidades pequenas.
- Por conta de espaço de buffer limitado a camada de apresentação pode ter que iniciar a transmissão sem todo o campo.
- Quando isso acontece é utilizado a flag de fim de conteúdo.
- A codificação do campo de dados depende do tipo de dados presentes.






8.3 Compressão de Dados

1. Codificação de um Conjunto Finitos de Símbolos Igualmente Prováveis
 2. Codificação Dependente da Frequência
 3. Codificação Dependente do Contexto
- 



8.3 Compressão de Dados

- Dados enviados através de um canal podem ser vistos como uma sequência de símbolos
 - Esses conjuntos de símbolos podem ser representados através de bits, dígitos decimais, letras, palavras selecionadas, países, etc.
- 



8.3.1 Codificação de um Conjunto Finito de Símbolos Igualmente Prováveis

- Em muitas aplicações, as mensagens são derivadas de um conjunto finito e são expressas em ASCII
- Como por exemplo:
- Em um projeto de automação de bibliotecas. Os títulos na coleção da biblioteca poderiam ser considerados de forma conveniente como um conjunto (finito) de símbolos.
- Um livro típico em cerca de 20 caracteres em seu título. Expresso em ASCII, tal título de livro requer 140 bits.



8.3.1 Codificação de um Conjunto Finito de Símbolos Igualmente Prováveis

- Possuindo a lista dos livros numerada, é possível reduzir tais números de bits de 140 para 26.
- Nos casos em que é feita uma referência ocasional a um item que não esteja na lista numerada, o nome pode ser detalhado por completo, usando uma convenção de escape.



8.3.2 Codificação Dependente de Frequência

- Em textos, alguns símbolos ocorrem mais frequentemente do que outros.
- Símbolos comuns são atribuídos à codigos curtos
- Símbolos raros são atribuídos à códigos longos
- Não há nenhuma maneira de se chegar à codificação mínima teórica com símbolos codificados de forma independente, porque muitos deles requerem um número fracionário de bits.



8.3.2 Codificação de Huffman

- Método de compressão que usa as probabilidades de ocorrência dos símbolos no conjunto de dados a ser comprimido.
- Usado para determinar códigos de tamanho variável para cada símbolo.
- Para atribuir aos caracteres mais frequentes os códigos binários de menor comprimento, constrói-se uma árvore binária baseada nas probabilidades de ocorrência de cada símbolo.
- Chega a uma aproximação razoável ao limite teórico.
- A codificação de Huffman é amplamente utilizada em aplicações de compressão, que vão de GZIP, PKZIP, BZIP2 a formatos de imagens como JPEG e PNG.

8.3.2 Codificação de Huffman

- Algoritmo para construção da árvore

```
enquanto tamanho(alfabeto) > 1:  
    S0 := retira_menor_probabilidade(alfabeto)  
    S1 := retira_menor_probabilidade(alfabeto)  
    X := novo_nó  
    X.filho0 := S0  
    X.filho1 := S1  
    X.probabilidade := S0.probabilidade + S1.probabilidade  
    insere(alfabeto, X)  
fim enquanto  
  
X = retira_menor_símbolo(alfabeto) # nesse ponto só existe um símbolo.  
  
para cada folha em folhas(X):  
    código[folha] := percorre_da_raiz_até_a_folha(folha)  
fim para
```

8.3.2 Codificação de Huffman

➤ Cadeia em caracter: AAAAAABBBBBBCCCCDDDEEF

➤ Cadeia em bits:

00000000000000000000001001001001001010010010010011011011100100101



8.3.2 Codificação de Huffman

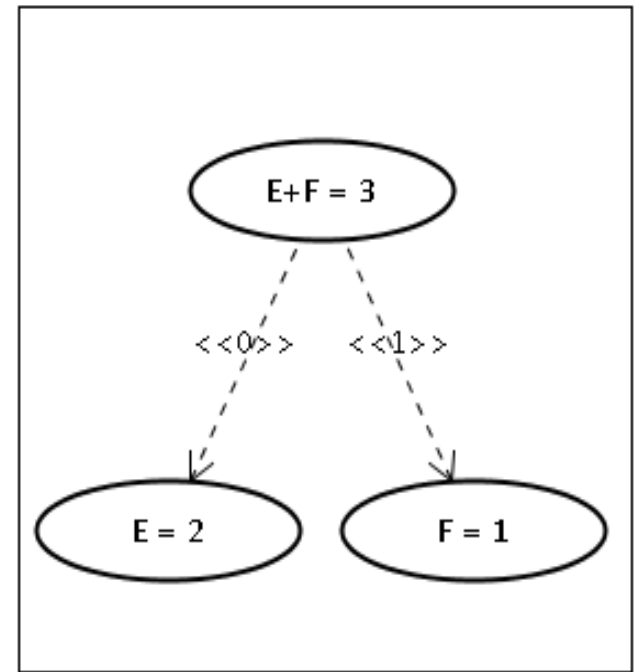
➤ Cadeia: AAAAAABBBBBBCCCCDDDEEF

A = 6

B = 5

C = 4

D = 3



8.3.2 Codificação de Huffman

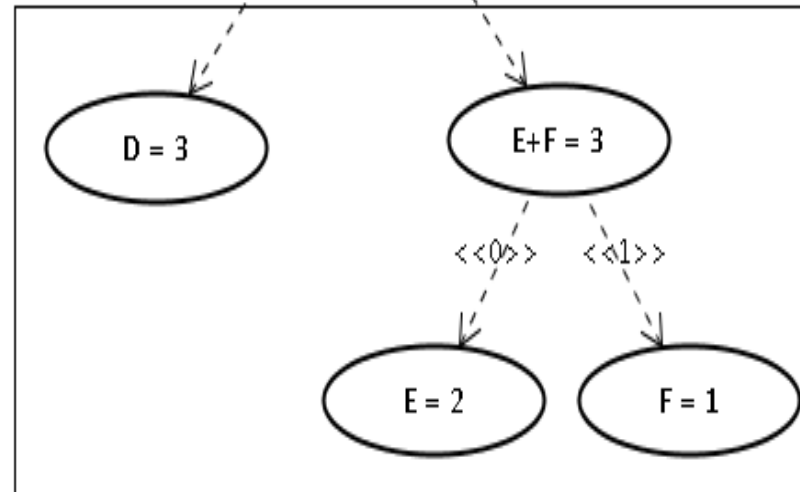
➤ Cadeia: AAAAAABBBBBBCCCCDDDEEF

A = 6

D+E+F = 6

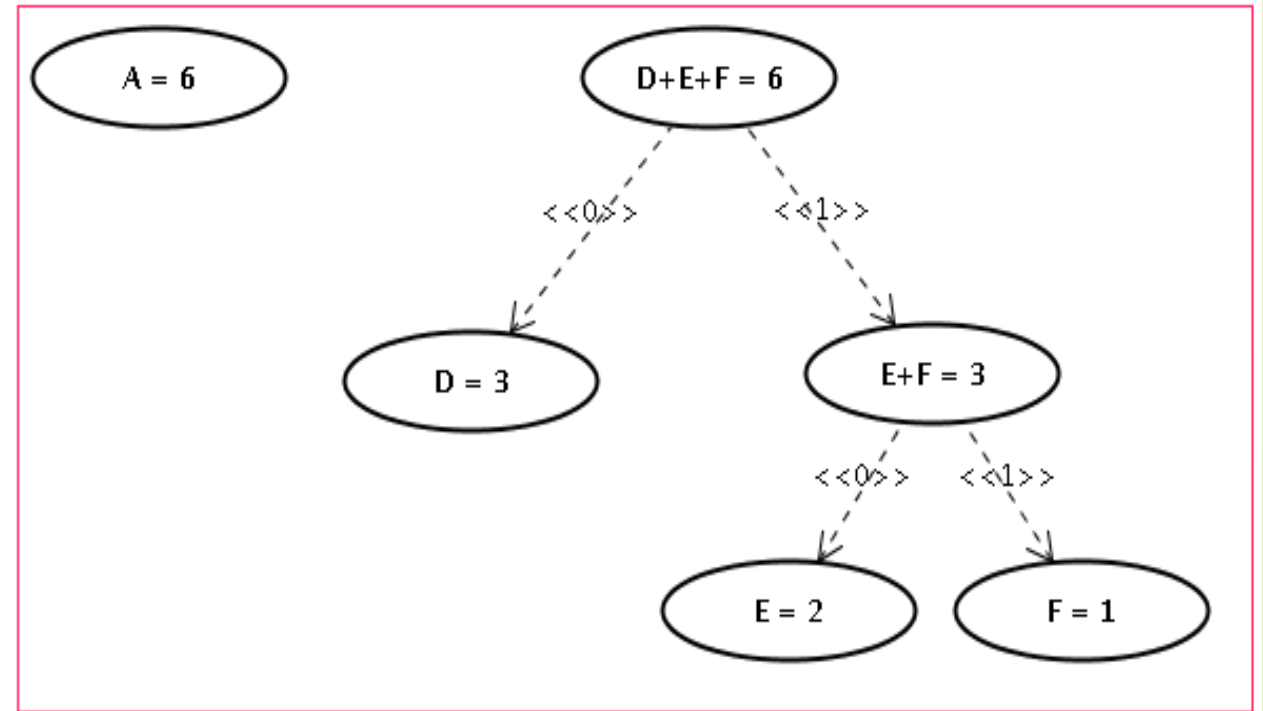
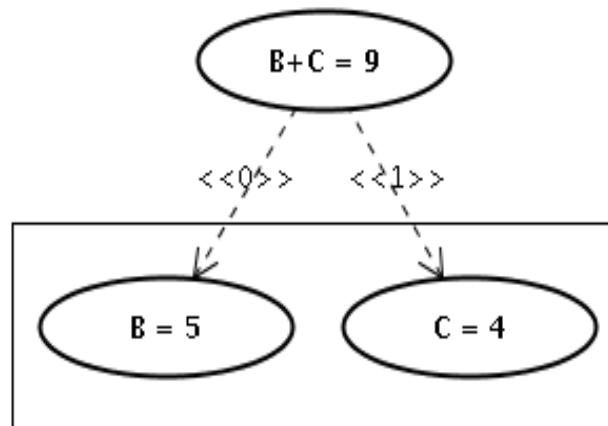
B = 5

C = 4



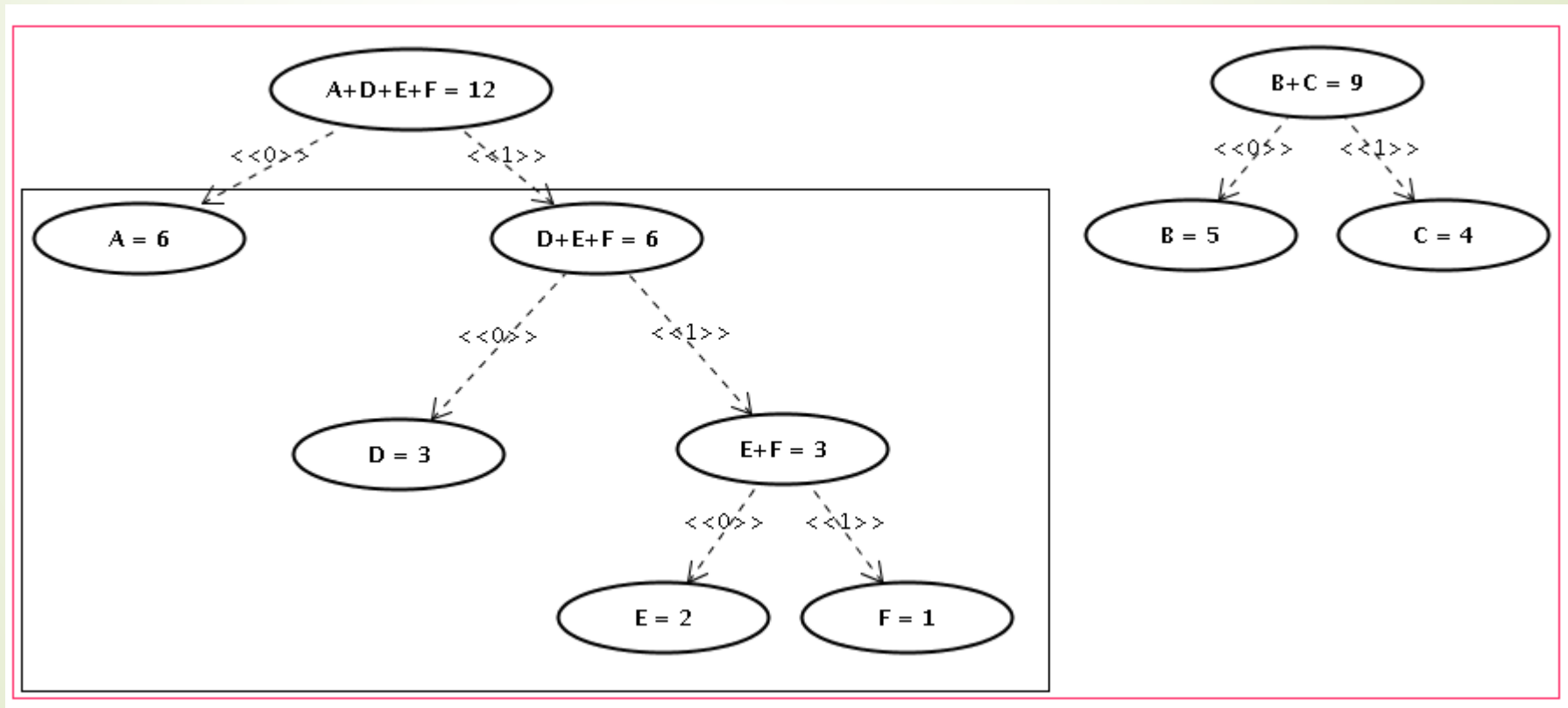
8.3.2 Codificação de Huffman

➤ Cadeia: AAAAAABBBBBBCCCCDDDEEF



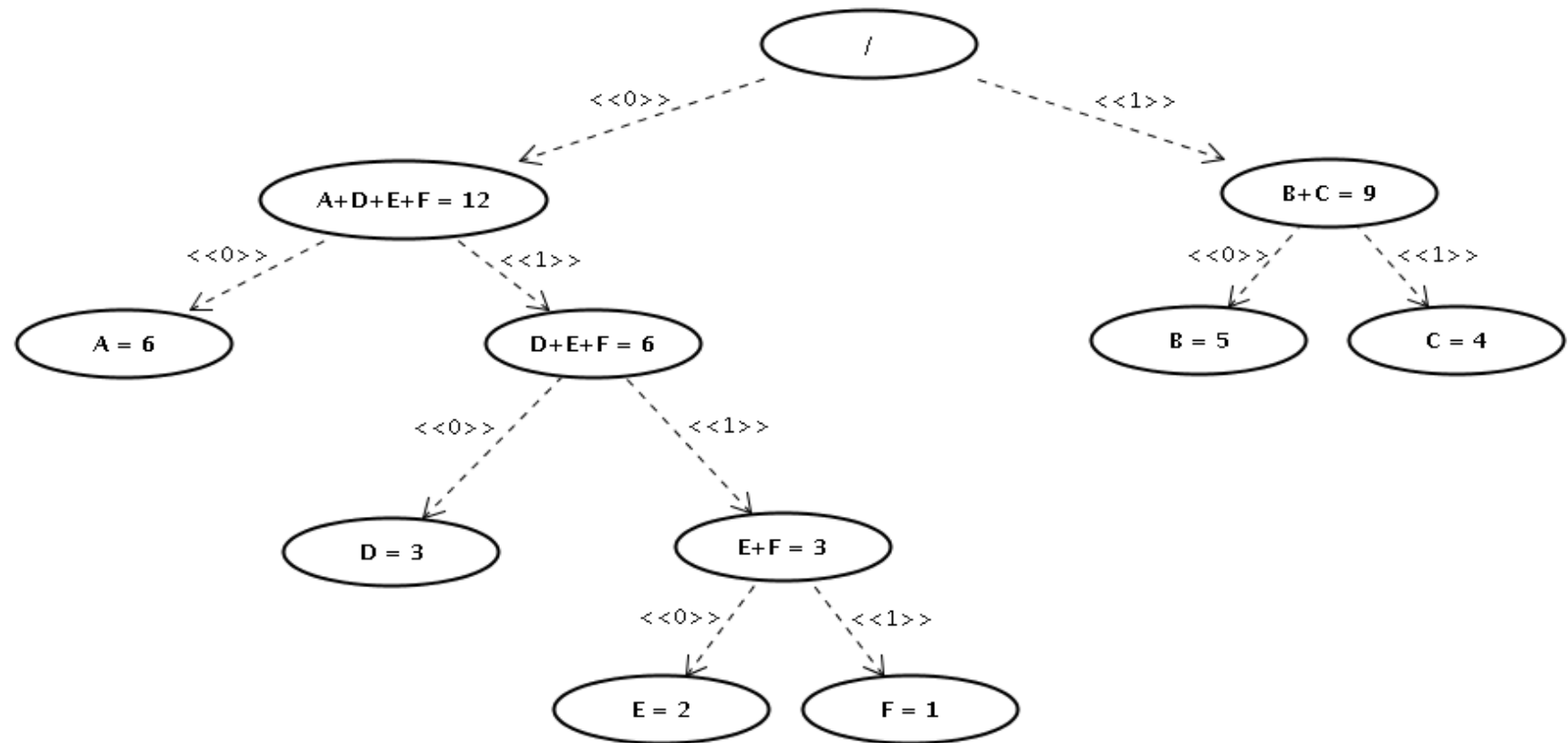
8.3.2 Codificação de Huffman

► Cadeia: AAAAAABBBBBBCCCCDDDEEF



8.3.2 Codificação de Huffman

➤ Cadeia: AAAAAABBBBBBCCCCDDDEEF



8.3.2 Codificação de Huffman

- Cadeia: AAAAAABBBBBBCCCCDDDEEF
- Cadeia original (63 bits):
00000000000000000000001001001001001010010010010011011011100100101
- Cadeia comprimida (51 bits):
000000000000101010101011111111010010010011001100111
- Economia de 12 bits = 20%

8.3.2 Codificação de Huffman

- Método de Codificação Run-length (ou RLE): uma técnica para comprimir cadeias de caracteres onde existem seqüências longas de caracteres repetidos.
- Para a compressão de textos o método de RLE não é muito eficiente.
- Exemplo: cadeia de bits
- Um símbolo k indica quantos bits 0 entre bits 1 consecutivos

0001000001000000100010000001

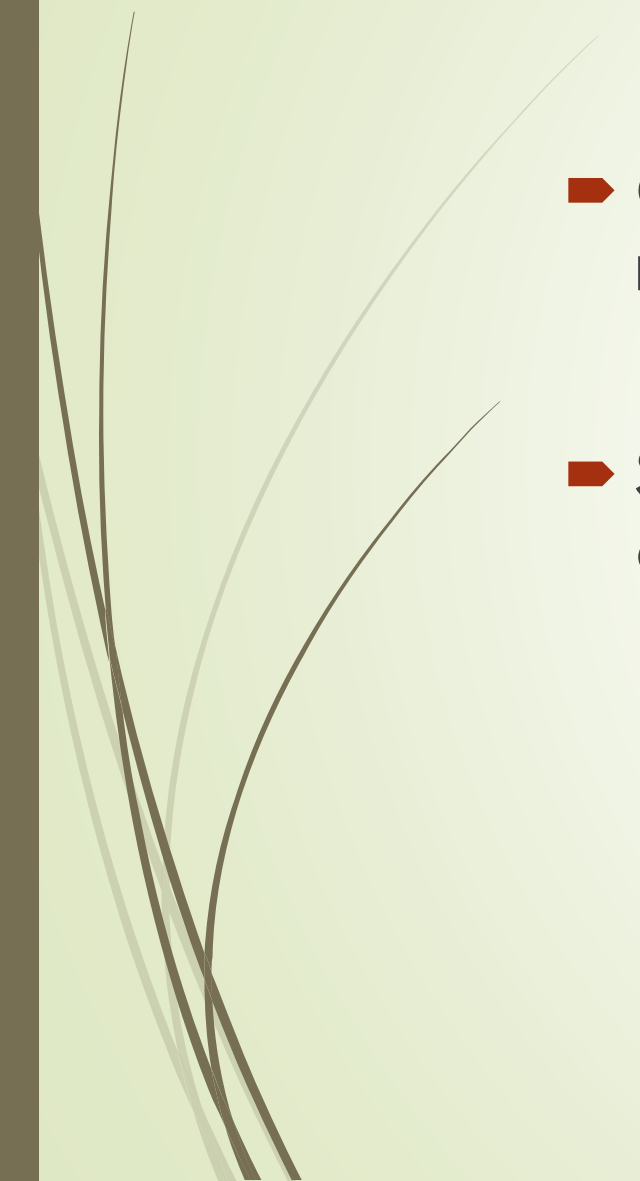
3 5 6 3 6

001 101 110 001 110

Economia de 46,5%



8.3.2 Codificação de Huffman

- Outro método é comprimir sequências de símbolos repetidos em uma contagem mais o símbolo.
 - Sequências de brancos, linefeeds e zeros iniciais são os candidatos mais prováveis a sofrerem compressão.
- 



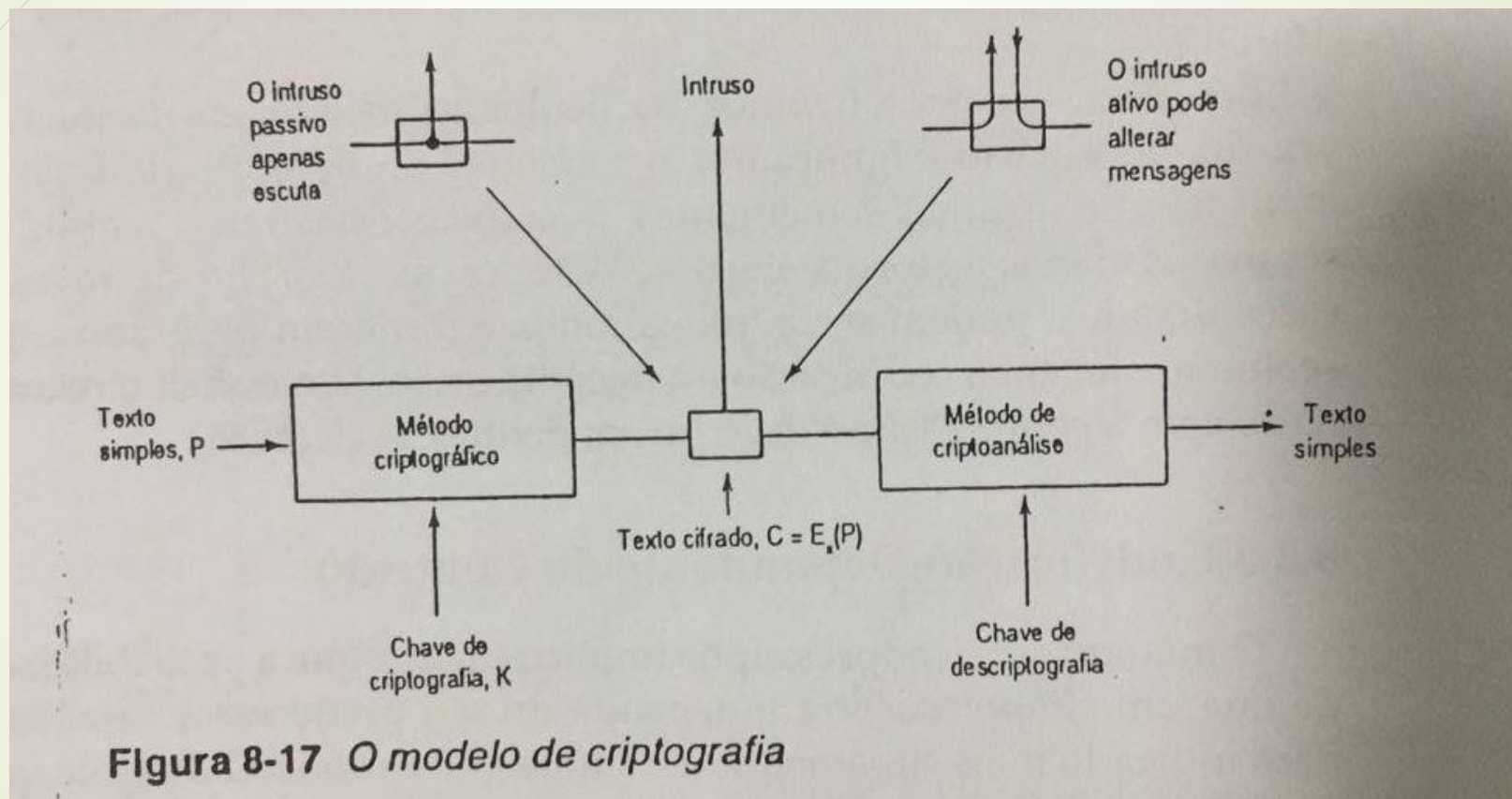
8.4 Criptografia

1. Criptografia Tradicional
2. O Padrão de Criptografia de Dados
3. O Problema da Distribuição de Chaves
4. Criptografia com Chave Pública
5. Autenticação e Assinaturas Digitais

8.4.1 Criptografia Tradicional


- Quatro grupos usaram e contribuíram para a arte da criptografia: militares, corpo diplomático, memorialistas e os amantes.
- Antes do uso dos computadores, os militares possuíam duas principais preocupações quanto a criptografia:
 - A criptografia era restrita a capacidade do encarregado de codificar/decodificar as mensagens com poucos equipamentos e em um curto espaço de tempo.
 - Existia uma grande dificuldade de mudar de um método criptográfico para outro, visto que isso acarretaria no treinamento de um grande número de pessoas.
- A preocupação de um funcionário relacionado a codificação ser capturado pelo inimigo fez com que fosse essencial ter uma maneira de trocar o método criptográfico de modo instantâneo.

8.4.1 Criptografia Tradicional





8.4.1 Criptografia Tradicional

- No processo de criptografia as mensagens, denominadas textos simples, são criptografadas por uma função parametrizada através de uma chave. Após isso, elas são transmitidas por um canal até o receptor.
 - No receptor, utilizando a chave, a mensagem criptografada é decodificada e se transforma novamente no texto simples.
- 



8.4.1 Criptografia Tradicional

- A importância da criptografia se dá principalmente na proteção da mensagem, quando essa é enviada através do meio de transmissão. Isso, porque podem haver 'intrusos' na rede que consigam ter acesso a ela.
- O intruso não é o destinatário da mensagem, logo ele não possui a chave necessária para decodificá-la e portanto não consegue entender diretamente o que está sendo transmitido.



8.4.1 Criptografia Tradicional

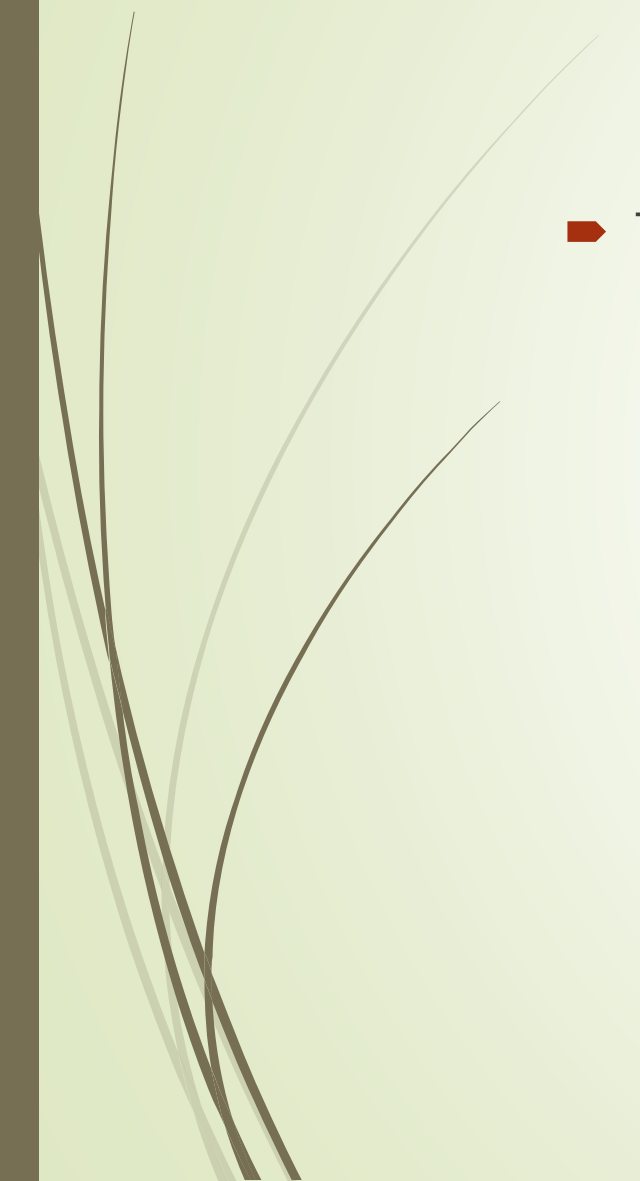
- Podemos classificar os intrusos da rede em dois tipos:
- Intruso Passivo:
 - Tem acesso a todo o conteúdo transmitido, portanto possui com exatidão o texto cifrado que foi transmitido. Mas é incapaz de adicionar ou modificar dados na mensagem.
- Intruso Ativo:
 - Além de conseguir acesso a todo conteúdo transmitido, como o intruso passivo. Ele pode injetar no canal suas próprias mensagens, modificando o texto cifrado original antes que esse chegue ao receptor.



8.4.1 Criptografia Tradicional



Termos:

- Criptografia: A arte de desenvolver cifras
 - Criptoanálise: A arte de decompor cifras
 - Criptologia: O conjunto de criptoanálise e criptografia
- 



8.4.1 Criptografia Tradicional

- O problema da criptoanálise apresenta três variações:
 - **Somente texto cifrado:** O criptoanalista só possui texto cifrado e nenhum texto simples.
 - **Texto simples conhecido:** O criptoanalista possui o texto cifrado e o texto simples correspondente.
 - **Texto simples escolhido:** O criptoanalista pode criptografar peças de texto simples da sua escolha.
- Inicialmente, pensa-se que se uma cifra restringe a abordagem de somente texto cifrado ela é segura, mas veremos posteriormente que isso não acontece.



8.4.1 Criptografia Tradicional

- Os métodos de criptografia têm sido divididos historicamente em duas categorias:
 - **Cifras de Substituição(incluindo códigos):** Cada letra ou grupo de letras é substituído por outra letra ou grupo de letras.
 - **Cifras de Transposição:** Não há substituição de letras. A criptografia acontece pela reorganização das letras da própria mensagem

8.4.1 Criptografia Tradicional

- **Cifra de César** : Um exemplo de cifra de substituição, em que a criptografia de uma determinada letra é dada por um deslocamento de 3 casas após essa letra.

Texto simples	a	b	c	d	e	f	g	h	i	j	k	l	m
Cifra	D	E	F	G	H	I	J	K	L	M	N	O	P
Texto simples	n	o	p	q	r	s	t	u	v	w	x	y	z
Cifra	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

8.4.1 Criptografia Tradicional

- Existe uma generalização da cifra de César, em que o alfabeto cifrado é movido k letras, ao invés de 3.
- Logo, k torna-se a chave da codificação, necessária no momento de decodificação da mensagem.
- Abaixo temos um exemplo do alfabeto cifrado quando $k = 7$.

Texto simples	a	b	c	d	e	f	g	h	i	j	k	l	m
Cifra	H	I	J	K	L	M	N	O	P	Q	R	S	T
Texto simples	n	o	p	q	r	s	t	u	v	w	x	y	z
Cifra	U	V	W	X	Y	Z	A	B	C	D	E	F	G



8.4.1 Criptografia Tradicional

- Existe uma generalização da cifra de César, em que o alfabeto cifrado é movido k letras, ao invés de 3.
- Logo, k torna-se a chave da codificação, necessária no momento de decodificação da mensagem.
- Abaixo temos um exemplo do alfabeto cifrado quando $k = 7$.



8.4.1 Criptografia Tradicional

- Desvantagens da Cifra de César:
 - A criptografia através de Cifra de César, mesmo que com o k variável, é facilmente quebrada por força bruta.
 - Simula um k e faz-se a decodificação, até encontrar o verdadeiro k utilizado e consequentemente a mensagem

8.4.1 Criptografia Tradicional

- **Substituição Monoalfabética:** Cada símbolo do texto simples tem um mapeamento sobre outro símbolo. A chave, no caso das letras, é a string de 26 letras correspondente ao alfabeto criptografado.

Texto simples	a	b	c	d	e	f	g	h	i	j	k	l	m
Cifra	Q	W	E	R	T	Y	U	I	O	P	A	S	D
Texto simples	n	o	p	q	r	s	t	u	v	w	x	y	z
Cifra	F	G	H	J	K	L	Z	X	C	V	B	N	M

- Note que não há nenhum tipo de lógica na sequência de letras da chave.

8.4.1 Criptografia Tradicional

- A criptografia por meio de substituição monoalfabética pode ser utilizada com $26! = 4 \times 10^{26}$ diferentes chaves possíveis. Logo, o criptoanalista não consegue decifrar o código por força bruta, como acontecia com a cifra de César.
- Apesar disso, existe outra maneira utilizada para conseguir descobrir a chave desse tipo de criptografia. Esse método tira proveito das propriedades estatísticas de linguagens naturais.

8.4.1 Criptografia Tradicional

- Como encontrar a chave de uma substituição monoalfabética?
- O primeiro método é utilizando propriedades estatísticas da linguagem. Sabe-se que a letra **e** é a mais utilizada na língua inglesa, seguida por t, o, a, n, i, etc. Portanto, na mensagem criptografada, pode-se contar a frequência relativa de cada caractere e assim atribuir como descryptografia para o caractere mais utilizado: a letra **e**, para o segundo mais utilizado: a letra t, e assim por diante
- Pode ser utilizado também diagramas ou trigramas mais utilizados na linguagem da mensagem, como: th, in, ing, the, no caso do inglês



8.4.1 Criptografia Tradicional

- Outra abordagem possível para conseguir a chave de uma substituição monoalfabética é supor frases ou palavras que provavelmente se encontram na mensagem.
- Por exemplo, considerando um texto de uma firma de contabilidade, pressupõe que exista a palavra *financeiro* em algum lugar do texto criptografado. Como *financeiro* tem a letra *i*, *n* e *e* repetidas com espaçamentos de 4, 1 e 3 letras, respectivamente. Inicia-se a busca pela chave através da pesquisa desse padrão no texto criptografado.



8.4.1 Criptografia Tradicional

- Cifra de Vigenère : Uma cifra polialfabética, onde é utilizada uma matriz quadrada contendo 26 alfabetos de César. Assim cada letra no texto simples é ligada a uma letra-chave que identifica qual alfabeto foi usado para aquela codificação.
- Uma cifra polialfabética pode ser construída mais eficiente se os alfabetos fossem arbitrários, ao invés de alfabetos de César.

8.4.1 Criptografia Tradicional

CIFRA DE VIGENÈRE																										
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

- Exemplo palavra-chave: COOKIE
- Texto Simples: ataque

Letra-chave	C	O	O	K	I	E
Texto Simples	a	t	a	q	u	e
Texto Criptografado	C	H	O	E	C	I



8.4.1 Criptografia Tradicional

➤ Vantagem:

- Letras iguais no texto simples terão diferentes representações no texto cifrado, dificultando a descoberta da chave.

➤ Desvantagens:

- Se não forem utilizados alfabetos de César, os alfabetos utilizados também deverão fazer parte da chave, pois eles serão necessários para a decodificação da mensagem.



8.4.1 Criptografia Tradicional

- Apesar dessa vantagem da cifra polialfabética, elas também podem ser decompostas. O necessário para isso é que o criptoanalista adivinhe o tamanho da chave. Conseguindo, assim, diferenciar os alfabetos que foram utilizados.
- Feito isso, pode se utilizar do método de distribuição de frequência em cada alfabeto.



8.4.1 Criptografia Tradicional

- **Chave de vez única:** Tipo de criptografia por cifra polialfabética, onde se constrói uma palavra-chave maior do que o texto simples, sendo assim o ataque visto anteriormente se torna inútil.
- Entretanto, esse método possui diversas desvantagens:
 - A chave não pode ser memorizada, pois muda a cada mensagem.
 - A quantidade de dados a serem transmitidos de uma única vez é limitada ao tamanho da chave.

8.4.1 Criptografia Tradicional

➤ Cifra Inviolável:

- Gerar uma sequência de bits aleatórios que será utilizada como chave.
- Converta o texto simples para uma string de bits utilizando ASCII.
- Por fim, compute o OU-exclusivo dessas duas strings.

- O OU-exclusivo é utilizado para que a probabilidade do bit dar 1 seja a mesma do bit dar 0

XOR	0	1
0	0	1
1	1	0

- Esse método não é utilizado pelas mesmas desvantagens listadas para a cifra de chave de vez única.

8.4.1 Criptografia Tradicional

➤ Cifra Inviolável:

- Gerar uma sequência de bits aleatórios que será utilizada como chave.
- Converta o texto simples para uma string de bits utilizando ASCII.
- Por fim, compute o OU-exclusivo dessas duas strings.

- O OU-exclusivo é utilizado para que a probabilidade do bit dar 1 seja a mesma do bit dar 0

XOR	0	1
0	0	1
1	1	0

- Esse método não é utilizado pelas mesmas desvantagens listadas para a cifra de chave de vez única.

8.4.1 Criptografia Tradicional

➤ Cifra Inviolável:

- Gerar uma sequência de bits aleatórios que será utilizada como chave.
- Converta o texto simples para uma string de bits utilizando ASCII.
- Por fim, compute o OU-exclusivo dessas duas strings.

- O OU-exclusivo é utilizado para que a probabilidade do bit dar 1 seja a mesma do bit dar 0

XOR	0	1
0	0	1
1	1	0

- Esse método não é utilizado pelas mesmas desvantagens listadas para a cifra de chave de vez única.

8.4.1 Criptografia Tradicional

- **Códigos:** Um código criptografa uma única unidade linguística de tamanho variável, geralmente uma palavra ou frase isolada. Enquanto a cifra criptografa uma unidade de tamanho fixo de texto simples em cada operação.
- Os códigos são subdivididos em:
 - Código de uma parte: O texto simples e os símbolos no código estão dispostos na mesma ordem. A codificação e a decodificação podem usar o mesmo livro de código.
 - Código de duas partes: O texto simples e os símbolos estão codificados sem uma ordem específica. Requer que tanto o transmissor quanto o receptor transportem duas vezes mais bagagem. Mais difícil de ser quebrado que o código de uma parte.
- A cifragem de uma mensagem codificada tem o nome de *supercifragem*.



8.4.1 Criptografia Tradicional

- **Cifras de Transposição:** Não há substituição de letras. A criptografia acontece pela reordenação das letras da própria mensagem.
- A cifra de transposição mais comum é a transposição colunar.

8.4.1 Criptografia Tradicional

- **Transposição Colunar:** A cifra é chaveada por uma palavra ou frase que não contém letras repetidas.
- O texto simples é escrito na horizontal. Já o texto criptografado é lido por colunas, começando com a coluna que tem a letra-chave mais baixa.

<u>M</u>	<u>E</u>	<u>G</u>	<u>A</u>	<u>B</u>	<u>U</u>	<u>C</u>	<u>K</u>
<u>7</u>	<u>4</u>	<u>5</u>	<u>1</u>	<u>2</u>	<u>8</u>	<u>3</u>	<u>6</u>
p	l	e	a	s	e	t	r
a	n	s	f	e	r	o	n
e	m	i	l	l	i	o	n
d	o	l	l	a	r	s	t
o	m	y	s	w	i	s	s
b	a	n	k	a	c	c	o
u	n	t	s	i	x	t	w
o	t	w	o	a	b	c	d

8.4.1 Criptografia Tradicional

- A decomposição da cifra de transposição colunar é feita de maneira análoga a cifra de Vigenère.
- É necessário fazer uma estimativa do número de colunas. Desse modo, utiliza-se uma palavra ou frase provável de ser utilizada na mensagem para calcular o tamanho a chave.
- Após isso é feita a ordenação das colunas. Quando o tamanho da chave, k , é pequeno, todas as $k(k-1)$ pares de ordenação podem ser analisados, e a decisão ser tomada com base na frequência relativa de digramas da linguagem utilizada. O par com a melhor correspondência é mantido vizinho e esse processo é repetido.



8.4.2 O Padrão de Criptografia de Dados

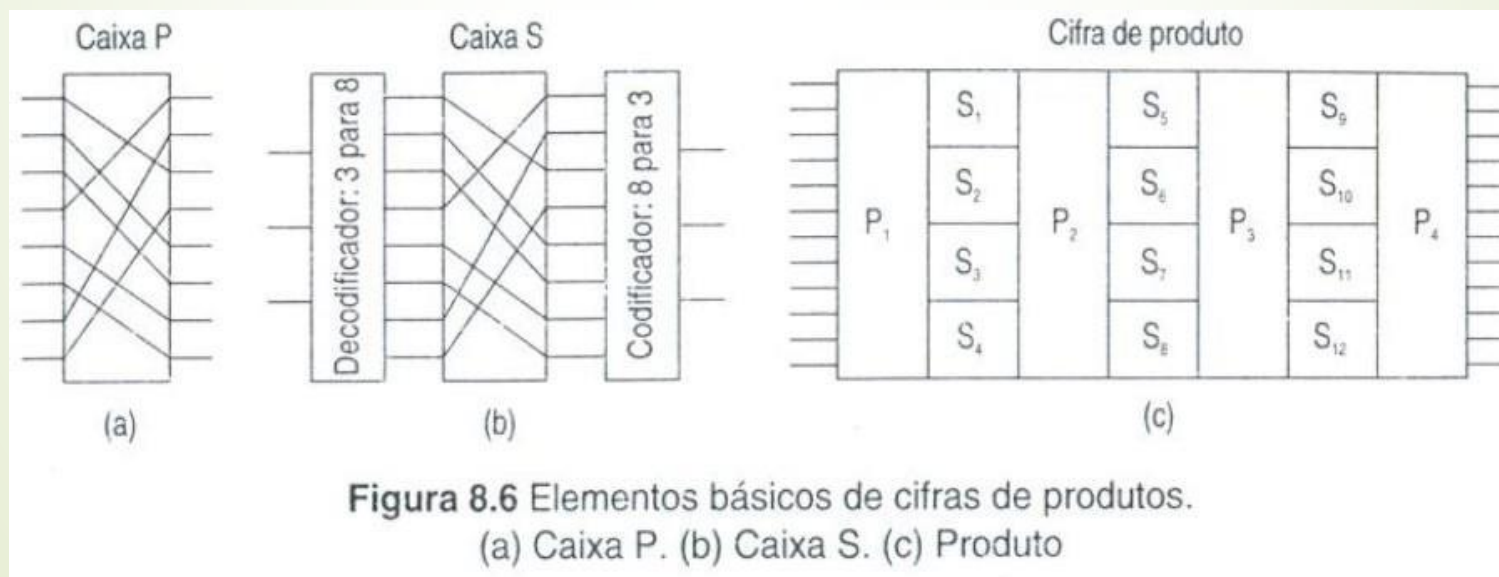
► **Criptografia Moderna**

- Mesmas ideias básicas da tradicional(transposição, substituição) mas com ênfase diferente.
- Em vez de usar algoritmos simples e chaves longas atualmente o objetivo é tornar os algoritmos bem mais complexos

8.4.2 O Padrão de Criptografia de Dados

► Criptografia Moderna

- As transposições e substituições podem ser implementadas com circuitos simples.





8.4.2 O Padrão de Criptografia de Dados

➤ DES

- Em 1977, o governo dos EUA adotou uma cifra produto desenvolvida pela IBM como seu padrão oficial para informações não-classificadas. Essa adoção estimulou vários fabricantes a implementarem o algoritmo em hardware.
- O algoritmo ficou conhecido como Padrão de Criptografia de Dados(Data Encryption Standard - National Bureau of Standards).



8.4.2 O Padrão de Criptografia de Dados

➤ DES

- Texto simples é criptografado em blocos de 64 bits, formando 64 bits de texto cifrado.
- Chave de 56 bits
- Possui 19 estágios distintos

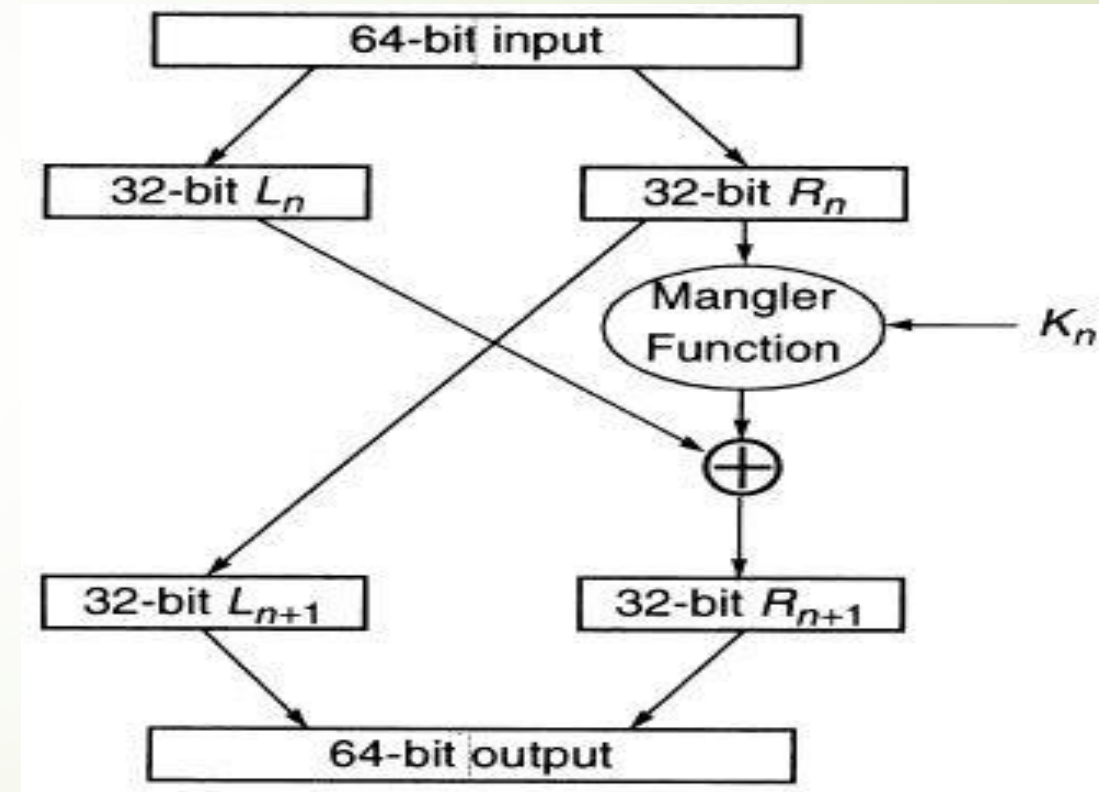
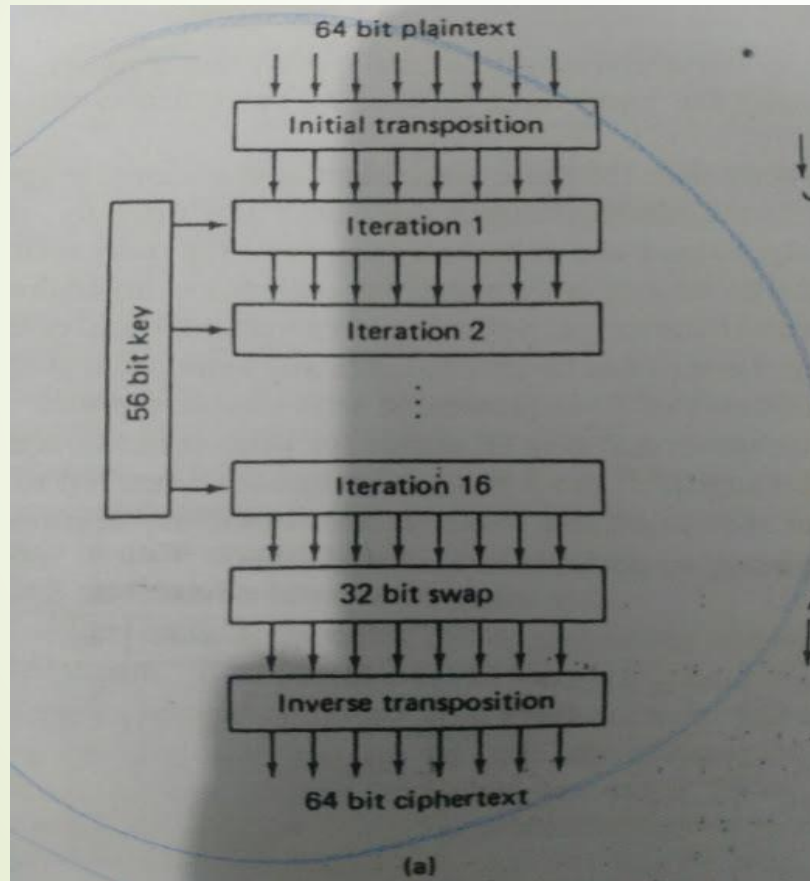


8.4.2 O Padrão de Criptografia de Dados

➤ DES

- Primeiro estágio: Transposição sobre o texto de 64bits independente da chave.
- Último estágio: inversão da primeira transposição.
- Penúltimo estágio: Troca os 32 bits mais à esquerda com os 32 bits mais à direita.
- Os 16 estágios restantes o texto é dividido em 2 entradas de 32 bits que resultam em 2 saídas de 32 bits. A saída da esquerda é uma cópia da entrada da direita. A saída da direita é o resultado do XOR da entrada da esquerda e uma função da entrada da direita da chave para esse estágio

8.4.2 O Padrão de Criptografia de Dados



8.4.2 O Padrão de Criptografia de Dados

► DES

► Função possui quatro etapas:

- Um número E de 48 bits é construído pela expansão dos 32 bits de R_{i-1} de acordo com uma regra de transposição e duplicação.
 - E e K_i sofrem juntos uma operação XOR
 - Saída do XOR é particionada em oito grupos de 6 bits, e cada grupo é levado a uma caixa-s que resultam em oito números de 4 bits.
 - Por último esses 32 bits passam por uma caixa-P.
- Antes de cada uma das 16 iterações a chave é particionada em duas unidades e rotacionada de acordo com a iteração. Isso faz com que a chave mude a cada iteração.



8.4.2 O Padrão de Criptografia de Dados

- É possível reforçar uma cifra inserindo caracteres aleatórios no texto, ou mensagens falsas entre as reais.
- O lixo adicionado é evidentemente um desperdício de memória, porém é justificável dependendo do tipo de conexão.



8.4.2 O Padrão de Criptografia de Dados

➤ **Controvérsia DES**

- O DES foi cercado de controvérsia desde sua concepção.
- Chave muito pequena. (Original 128 bits)
- Motivos e princípios mantidos em segredo.
- Muitos apontam que os reais motivos são o desejo do governo de ter uma cifra que seja forte o suficiente para barrar os outros e não ele mesmo.



8.4.3 O Problema da Distribuição de Chaves

- Um problema do DES é que exige que o receptor, ao decriptar uma mensagem, utilize a mesma chave que o transmissor usou para encriptar-la. Como distribuir chaves?
- Tradicionalmente, pares de chaves idênticas eram inventadas numa central e transmitidas aos destinos por mensageiros pessoais
- Para instituições que necessitam de alterações de chave diárias, como bancos, esse método de distribuição é bastante insatisfatório
- Uma solução conhecida é uma hierarquia de chaves



8.4.3 O Problema da Distribuição de Chaves

► Hierarquia de chaves

- Cada organização escolhe uma chave principal aleatoriamente e a envia para cada um de seus escritórios via correio pessoal
- Seus escritórios são agrupados em regiões, com o escritório central de cada região escolhendo uma chave regional
- As chaves regionais são criptografadas usando-se a chave principal, e depois distribuídas pela rede
- Quando dois escritórios quaisquer desejam se comunicar, um deles escolhe uma chave de sessão e a envia ao outro, criptografada pela chave regional
- Uma alternativa sendo um gerenciador escolher a chave de sessão e a enviar as duas partes



8.4.3 O Problema da Distribuição de Chaves

► Hierarquia de chaves

- O princípio implícito desse projeto é que essas chaves são usadas tão raramente que nenhum intruso terá a capacidade de juntar texto cifrado suficiente para decifrá-los
- O texto simples das mensagens consiste de números aleatórios de 56 bits, o que torna muito difícil sua criptoanálise.



8.4.3 O Problema da Distribuição de Chaves

- **Hierarquia de chaves**

- **Problemas:**

- Uma nova chave principal deve ser gerada e transportada fisicamente a todos os escritórios sempre que se imagina que a chave principal atual está comprometida.
- Não há como pessoas pertencentes a diferentes organizações estabelecerem comunicação de forma segura a não ser que seja realizado um encontro para haver concordância com relação a uma chave.



8.4.3 O Problema da Distribuição de Chaves

- **Quebra-cabeças (Método de Merkel)**
- Supõe que duas partes que nunca se comunicaram e que desejam estabelecer comunicação de forma segura devem utilizar o canal entre as duas para estabelecer a chave.
- Parte do princípio de que há um intruso que pode copiar tudo que é enviado no canal.
- Um Quebra-Cabeça é um criptograma feito para ser violado.



8.4.3 O Problema da Distribuição de Chaves

► Proteção da Chave

- Também é igualmente importante ocultar a chave de si mesma.
- Uma corporação não pode delegar autoridade ilimitada (na forma de chave) a qualquer funcionário.

8.4.3 O Problema da Distribuição de Chaves

► Proteção da Chave

► Algoritmo de Shamir

- Suponha que uma companhia deseja que certas mensagens sejam enviadas apenas por um grupo selecionado de pessoas. Entregar a cada uma alguns bits não funciona, pois algumas selecionados ao acaso não garantem a chave completa
- É empregado um polinômio de grau 3 - $p(x): ax^3+bx^2+cx+d$, em que a , b , c são escolhidos ao acaso e d é a chave
- Um polinômio de grau 3 pode ser determinado com 4 pontos, então são dados alguns pontos $(x.p(x))$ para os funcionários dependendo da sua função e qualquer conjunto de 4 desses podem achar o polinômio e deduzir os coeficientes e, conseqüentemente a chave

8.4.4 Criptografia com Chave Pública

- Método de Diffie e Hellman para estabelecer uma comunicação segura entre pessoas que nunca se comunicaram antes.
- É utilizado um algoritmo E para encriptação e um algoritmo D para deciptação.
- Há três requisitos para que os algoritmos funcionem corretamente:
 - $D(E(P)) = P$
 - É extremamente difícil deduzir D a partir de E
 - E não pode ser violado por uma abordagem de texto simples escolhido.

8.4.4 Criptografia com Chave Pública

- Supondo algoritmos E e D que cumpram os requisitos anteriores. Qualquer pessoa que desejasse receber informações criptografadas poderia simular um E e D próprio. Assim, o algoritmo de E poderia ser disponibilizado em modo público para qualquer pessoa que quisesse lhe mandar uma mensagem, daí o nome chave pública.
- Como D não é dedutível de E, a única pessoa capaz de decodificar a mensagem é o destinatário, que possui o algoritmo D.

8.4.4 Criptografia com Chave Pública

- Algoritmo MIT:
 - Escolha dois números primos grandes, p e q , cada um deles maior que 10100 .
 - Calcule $n = p \cdot q$ e $z = (p-1) \cdot (q-1)$
 - Escolha um número relativamente primo a z e chame-o de d .
 - Encontre e tal que $e \cdot d \equiv 1 \pmod{z}$
- Divida o texto simples em blocos de tamanho P , tal que $0 \leq P < n$. Para encriptar a mensagem P , calcule $C = (P^e) \pmod{n}$. Para decriptar C , calcule $P = (C^d) \pmod{n}$.
- Portanto, para encriptação é necessário (e, n) para decriptação é necessário (d, n) . Logo a chave pública é constituída por (e, n) e a chave secreta consiste em d

8.4.4 Criptografia com Chave Pública

- Temos como exemplo do MIT, quando pegamos números pequenos, sendo $p=3$, $q=11$. Então, $n=33$ e $z=20$. Um valor adequado para d é 7.
- Logo, resolvendo a equação $7^e \equiv 1 \pmod{20}$, temos que $e = 3$. O texto criptografado é dado por $P^3 \pmod{33}$, e o texto decodificado é dado por $C^7 \pmod{33}$.

Texto Simples (P)		Texto Cifrado (C)		Após a decriptação		
<u>Simbólico</u>	<u>Númérico</u>	P^2	$P^2 \pmod{33}$	C^7	$C^7 \pmod{33}$	<u>Simbólico</u>
S	19	6859	28	13492928512	19	S
U	21	9261	21	1801088541	21	U
Z	26	17576	20	1280000000	26	Z
A	01	1	1	1	1	A
N	14	2744	5	78125	14	N
N	14	2744	5	78125	14	N
E	05	125	26	8031810176	5	E
		Computação do emissor		Computação do receptor		

Figura 8-25 Um exemplo do algoritmo do MIT

8.4.4 Criptografia com Chave Pública

- A segurança do método MIT se baseia na dificuldade de fatorar números extensos.
- Contudo, deve-se assinalar que ninguém provou a inexistência de um artifício que permitisse violar a cifra sem fatorar n . Em compensação, ninguém demonstrou a existência de tal artifício.



8.4.5 Autenticação e Assinaturas Digitais

- No mundo real a diferença entre original e cópia é de extrema importância, diversos documentos só são validados por assinaturas.
- Para substituir assinaturas físicas e permitir que ocorram diversas transações é necessário ter meios tão satisfatórios quanto às assinaturas.
- É necessário garantir que:
 - Quem recebe pode verificar a identidade de quem enviou.
 - Quem enviou não pode negar a mensagem futuramente.



8.4.5 Autenticação e Assinaturas Digitais

► Autenticação

- Em sistemas orientados por conexão a autenticação pode ser feita quando se inicia a sessão.
- Tradicionalmente o usuário prova sua identidade através de uma senha.
- Esse método mantém uma lista de senhas internamente, o que é potencialmente um problema de segurança.
- A criptografia de chave pública fornece uma autenticação de forma segura nesse caso.



8.4.5 Autenticação e Assinaturas Digitais

Exemplo:

- Um cliente abre conta num banco, escolhe uma chave pública e também uma privada, fornecendo só a pública ao banco.
- Quando o cliente estabelece uma sessão com o banco, o banco criptografa um número aleatório com a chave pública do cliente e verifica se este consegue descriptografar.
- Somente o cliente é capaz de descriptografar a mensagem enviada pelo banco, pois só ele possui a chave privada, assim a sessão é confirmada e iniciada.

8.4.5 Autenticação e Assinaturas Digitais

➤ Assinaturas digitais com criptografia de chave pública:

- A criptografia de chave pública pode dar uma importante contribuição para solucionar o problema de que usuários desonestos reneguem suas mensagens anteriores.
- Para que a criptografia de chave pública funcione é necessário que os algoritmos de encriptação e deciptação tenham as propriedades
- $E(D(P)) = P$ e $D(E(P)) = P$

8.4.5 Autenticação e Assinaturas Digitais

- **A** pode enviar uma mensagem de texto simples assinada **P** para **B** pela transmissão

$$E_b(D_a(P))$$

- **A** conhece sua chave de deciptação (D_a), bem como a chave pública de **B** (E_b).
- Quando **B** recebe a mensagem, ele a transforma com o uso de sua chave prlvativa da maneira usual, formando $D_a(P)$, e então a armazena.

8.4.5 Autenticação e Assinaturas Digitais

- Caso A negue posteriormente ter enviado a mensagem B pode provar que recebeu pois terá a mensagem P igual a de $D_a(P)$ se simplesmente usar a chave pública de A.
- A crítica feita a esse método de assinatura é que ele associa duas funções distintas a autenticação e o segredo, por muitas vezes é necessário apenas a autenticação e não o segredo.
- Como a chave pública é lenta com frequência se torna desejável a capacidade de enviar documentos simples assinados

8.4.5 Autenticação e Assinaturas Digitais

- Outro método é o da **soma de verificação de mão única**, denominado CK.
- A partir da mensagem P é simples calcular CK, porém o inverso é praticamente impossível.
- A soma de verificação de mão única deve ser muito menor que a mensagem, por exemplo 256 bits.
- Para assinar uma mensagem P **A** primeiramente calcula CK(P) e depois aplica sua chave primitiva a ele. Formando assim

$Da(CK(P))$

8.4.5 Autenticação e Assinaturas Digitais

- **A** transmite para **B** o par $P, Da(CK(P))$.
- Quando **B** recebe aplica a chave pública de **A**, assim a mensagem criptografada fica $Ea(Da(CK(P))) = CK(P)$.
- Para verificar a autenticidade **B** aplica a soma verificação de mão única a P e verifica se encontra $CK(P)$, caso não encontre a mensagem é inválida.
- A vantagem deste método é que somente a soma de verificação tem de passar pelo processo custoso de encriptação por chave pública, independente do quão longa seja a mensagem.

8.4.5 Autenticação e Assinaturas Digitais

- Se possuímos uma autoridade central que conheça tudo tanto o segredo quanto as assinaturas digitais podem ser obtidas usando criptografia convencional.
- Para que isso aconteça é necessário que cada usuário escolha uma chave secreta e a envie a unidade central (podemos chamá-la de grande irmão já que ela sabe sobre tudo).
- Nesse cenário somente o grande irmão e A conhecem chave secreta de A.
- Quando A quer se comunicar com B ele precisa pedir uma chave de sessão para o grande irmão, este lhe enviará duas cópias. Uma encriptada K_a e uma K_b .
- A então deve enviar K_b para B junto com as instruções de descriptação.

8.4.5 Autenticação e Assinaturas Digitais

- Além disto o grande irmão pode fornecer também um serviço de assinatura, para isso é necessário uma chave especial X mantida em segredo para todos.
- Para usar este serviço de assinatura B deveria insistir que os passos a seguir fossem usados para cada mensagem P enviada para ele
- A envia $K_a(P)$ para o grande irmão.
- Grande irmão descripta $K_a(P)$, obtém P e então constrói uma nova mensagem formada pelo nome e endereço de A concatenados com a data D e a mensagem original e então criptografa com X
- Forma-se a mensagem $X(A+D+P)$.

8.4.5 Autenticação e Assinaturas Digitais

- **A** envia $X(\mathbf{A}+D+P)$ para **B**.
- **B** envia $X(\mathbf{A}+D+P)$ para o grande irmão pedindo $K_b(\mathbf{A}+D+P)$ como resultado.
- **B** descripta $K_b(\mathbf{A}+D+P)$ e então obtém-se **A**, **D** e **P**.
- Se **A** afirmar que não enviou **P** para **B** basta que **B** mostre $X(\mathbf{A}+D+P)$ e pedir para o grande irmão descriptografar, assim obtendo $\mathbf{A} + D + P$ pode-se saber se **A** mentiu ou se não.

8.4.5 Autenticação e Assinaturas Digitais

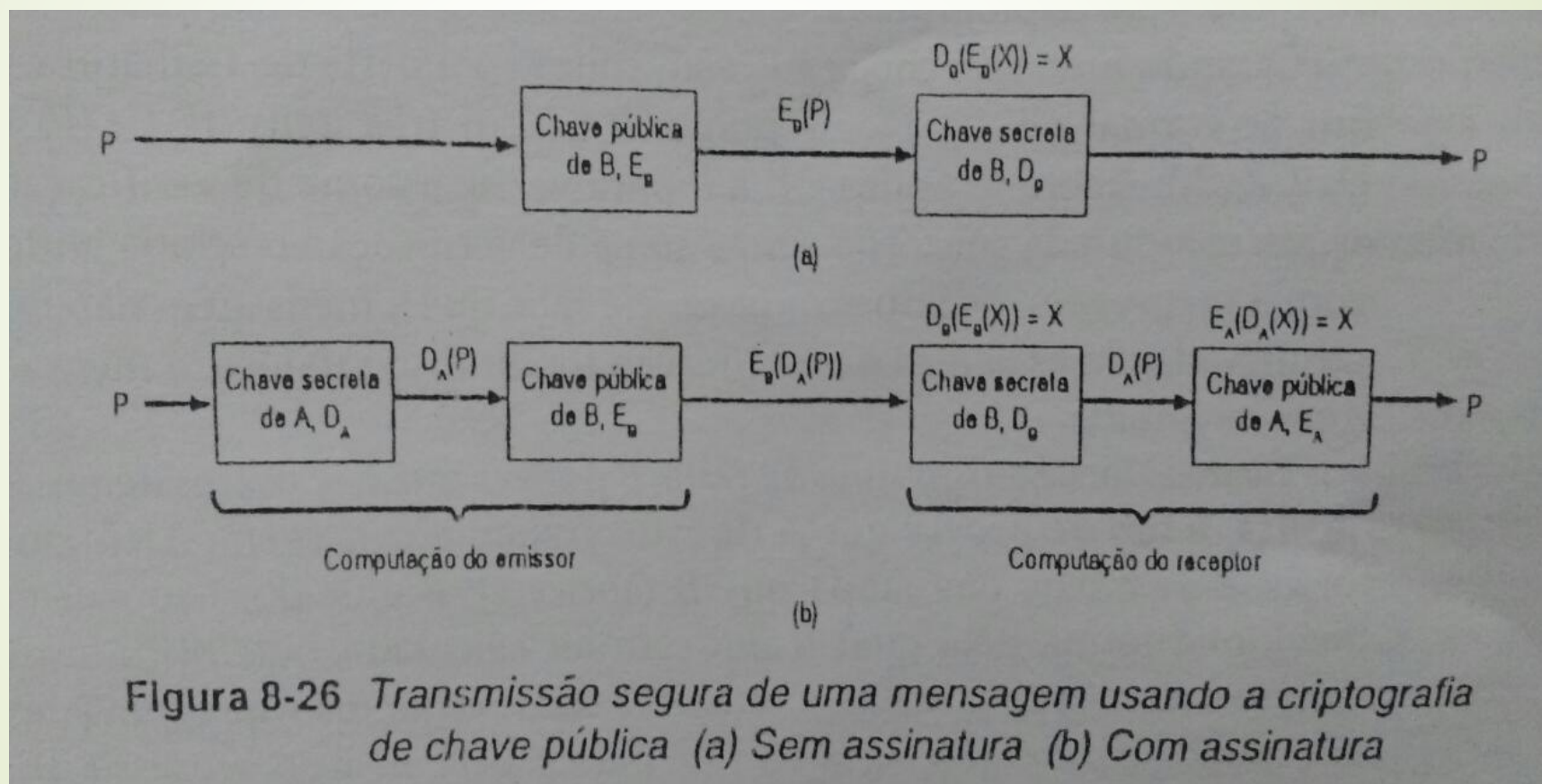
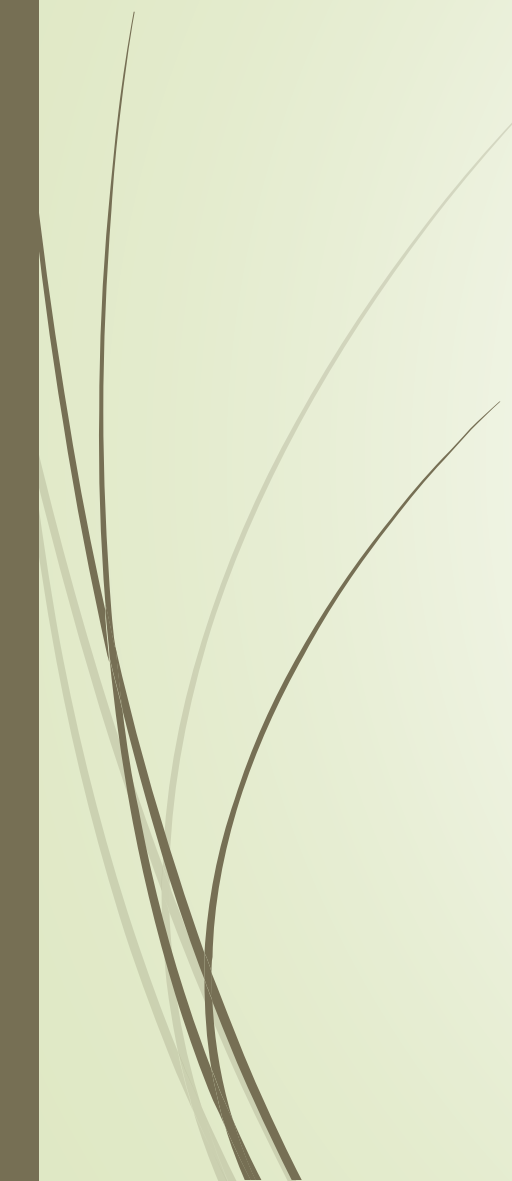


Figura 8-26 Transmissão segura de uma mensagem usando a criptografia de chave pública (a) Sem assinatura (b) Com assinatura




8.5 Exemplos da Camada de Apresentação

1. Camada de apresentação em redes publicas
 2. Camada de apresentação em ARPANET
 3. Camada de apresentação em MAP e TOP
 4. Camada de apresentação no USENET
- 



8.5 A camada de apresentação em redes públicas

- As redes públicas que implementam a camada de sessão também implementam a camada de apresentação.
- O padrão de protocolo de camada de apresentação (ISO 8823) usa ASN.1 para definir o formato de todas as PPDUs.
- O envio de PDUs é feito pela camada da sessão.



8.5 A camada de apresentação em redes públicas

- S-TOKEN-PLEASE.request é invocado quando um usuário solicita uma P-TOKEN-PLEASE.request
- As PDUs usadas pelo protocolo de apresentação se dividem em:
 - Estabelecimento de conexão
 - Liberação anormal
 - Transferência de dados
 - Gerenciamento de contexto.



8.5 A camada de apresentação no ARPANET

- O ARPANET não possui uma camada de apresentação.
- A solução utilizada é que cada aplicação defina seus próprios padrões.



8.5 A camada de apresentação em MAP e TOP

- MAP e TOP suportam as funções básicas da camada de apresentação OSI de estabelecer conexões e gerenciar múltiplos contextos.
- As primitivas associadas a alterações de contexto em limites de atividade não são suportadas porque o conceito inteiro de atividades não é suportado na camada de sessão.
- O uso da ASN.1 é obrigatório.



8.5 A camada de apresentação no USENET

- O USENET não possui uma camada de apresentação
- Cada aplicação possui um conhecimento interno dos formatos externos necessários

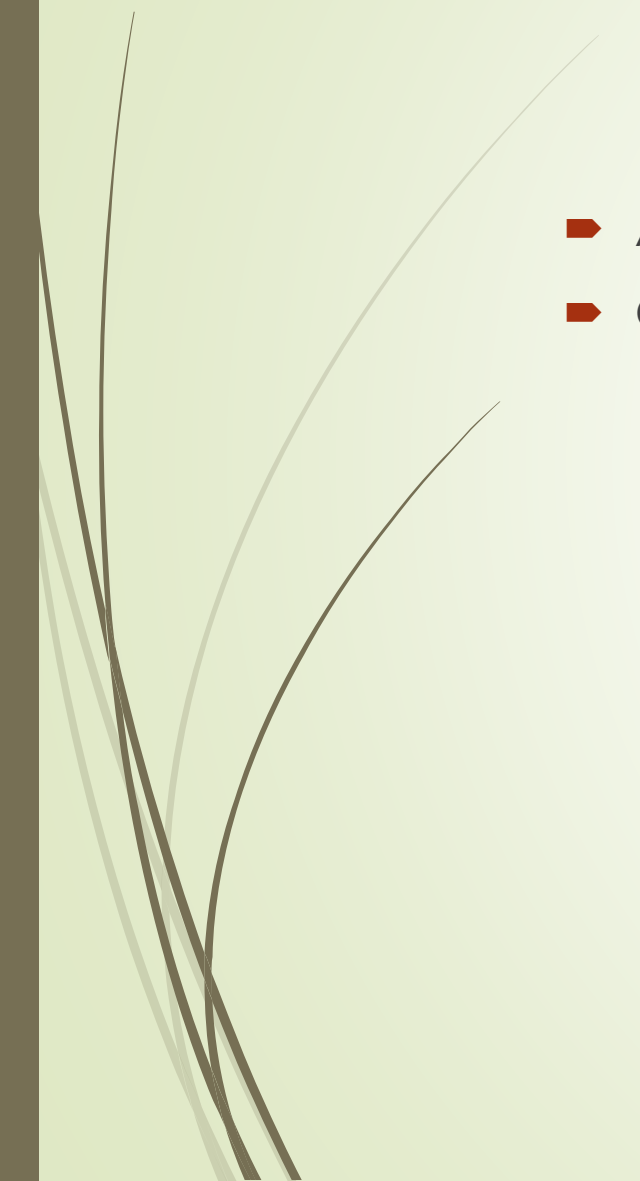


Considerações Finais

- A camada de apresentação está preocupada com:
 - Manipulação de Estruturas de Dados
 - Tipos Abstratos
 - Representações Externas no fio(syntaxes de transferência)
- Sessão estabelecida: os pares negociam um ou mais caches de contextos consistindo em alguns tipos de dados e suas syntaxes de transferência.
- Cada máquina é livre para representar as estruturas de dados internamente na forma mais conveniente.



Considerações Finais

- A notação ASN.1 pode ser usada para descrever tipos de dados e valores.
 - O ASN.1 suporta os tipos primitivos:
 - Booleanos
 - Números inteiros
 - Cadeias de bits
 - Cadeias de octetos
- 



Considerações Finais

- A compressão de dados está intimamente relacionada à sintaxe de transferência.
- Fornecem compressão de dados:
 - ASN.1
 - Codificação Huffman
 - Codificação Aritmética
 - Codificação Run-Length



Considerações Finais

- Questões relacionadas à privacidade e à segurança da rede podem ser implementadas na camada de apresentação.
- A saída da camada de apresentação pode ser criptografada antes de ser dada à camada da sessão.
- Criptografia convencional.
- Criptografia de chave pública.
- A criptografia desempenha um papel importante na autenticação e no fornecimento de assinaturas digitais.