Segurança da Informação

Fundamentos e Criptografia Simétrica

Igor Machado Coelho

10/06/2024–19/06/2024

Section 1

Módulo: Fundamentos e Criptografia Simétrica

Pré-Requisitos

São requisitos para essa aula o conhecimento de:

- Redes de Computadores (conceitos gerais)
- Módulo 1: princípios básicos
- Módulo 2: ameaças
- Módulo 3: requisitos
- Módulo 4: malware e vírus
- Módulo 5: worms
- Módulo 6: engenharia social e carga útil
- Módulo 7: contramedidas
- Módulo 8: negação de serviço

Tópicos

- Fundamentos de Criptografia
- xxxxx

Section 2

Fundamentos de Criptografia

AGENDA

- Fundamentos de Criptografia
- Criptografia Simétrica

Breve história

- Ocultação de informações para fins de segurança e inteligência é uma tarefa de milhares de anos
- Fatos marcantes: cifras de Júlio César e força-tarefa do U-boat alemão
- Avanços na cifração simétrica e introdução de cifra de chave pública na década de 1970
- Desafio constante para manter ou aumentar resistência com os avanços dos sistemas computacionais

Section 3

Confidencialidade com Criptografia Simétrica

Criptografia Simétrica

- Também conhecido como criptografia convencional, de chave secreta ou de chave única
  - Única alternativa antes da criptografia de chave pública (anos 70)
  - Alternativa ainda mais amplamente utilizada
- Componentes: texto simples, algoritmo de cifração, chave secreta, texto cifrado e algoritmo de decifração

Texto Simples ou Texto às claras

É a mensagem ou dados originais alimentados ao algoritmo como entrada.

Algoritmo de cifração

O algoritmo de cifração executa várias substituições e transformações no texto às claras.

Criptografia Simétrica: Componentes

Chave secreta

A chave secreta é também fornecida como entrada para o algoritmo de cifração. As substituições e transformações exatas realizadas pelo algoritmo dependem da chave.

Texto cifrado

É a mensagem embaralhada produzida como saída. Ele depende do texto às claras e da chave secreta. Para dada mensagem, duas chaves diferentes produzirão dois textos cifrados diferentes.

Algoritmo de decifração

É, essencialmente, o algoritmo de cifração executado ao contrário. Toma o texto cifrado e a chave secreta como entradas e produz o texto às claras original.

Ilustração do Processo de Criptografia Simétrica

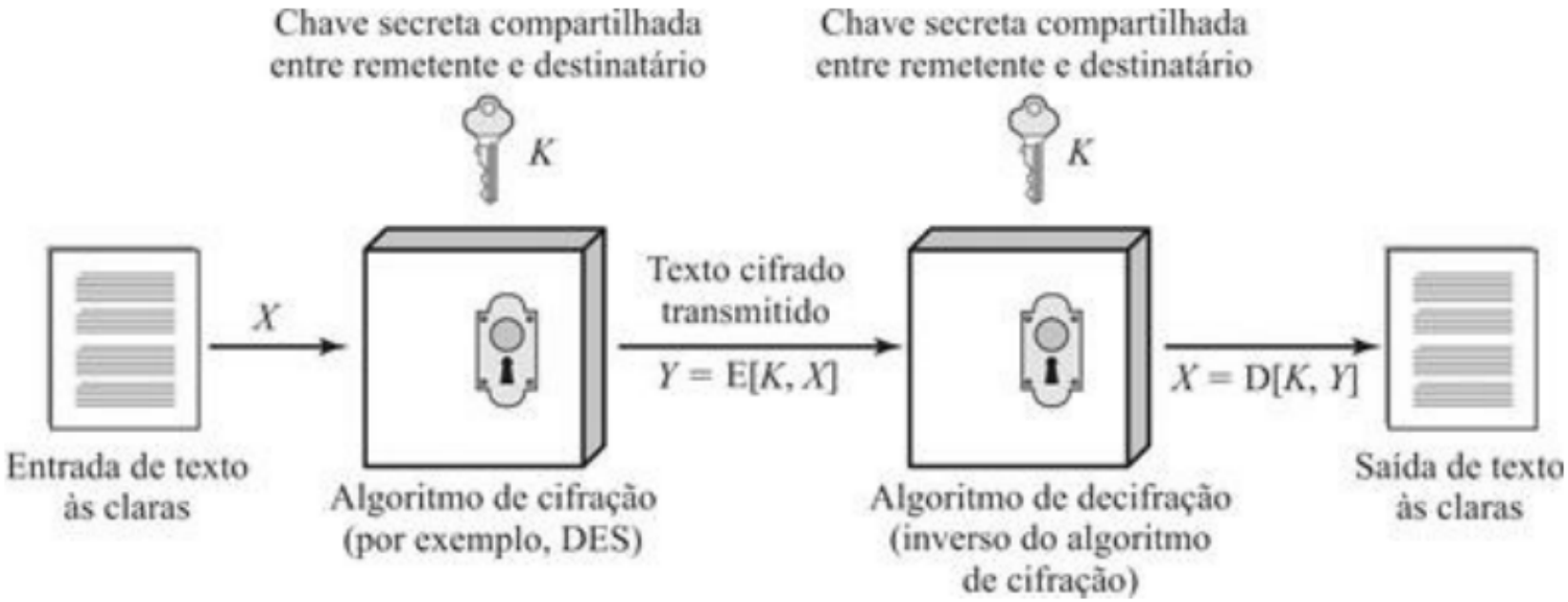


Figure 1: Criptografia Simétrica: Retirado do livro-texto

Classificações/Dimensões

Tipo de operações usadas para transformar texto às claras em texto cifrado

Dois princípios gerais: substituição, na qual cada elemento no texto às claras (bit, letra, grupo de bits ou letras) é mapeado para um outro elemento, e transposição, na qual elementos no texto às claras são rearranjados. O requisito fundamental é que nenhuma informação seja perdida (operações reversíveis). Tipicamente, múltiplos estágios.

Número de chaves

Remetente e o destinatário com mesma chave: simétrico, de chave única, de chave secreta ou de cifração convencional. Caso contrário: assimétrico, de duas chaves ou cifração de chave pública.

Modo como o texto às claras é processado

Uma cifra de bloco processa um único bloco de elementos da entrada por vez, produzindo um bloco de saída para cada bloco de entrada. Uma cifra de fluxo processa elementos de entrada continuamente, produzindo um elemento de saída por vez.

Requisitos da Criptografia Simétrica

Cifração Forte

No mínimo, gostaríamos que o algoritmo fosse tal que um oponente que conheça o algoritmo e tenha acesso a um ou mais textos cifrados não seria capaz de decifrar o texto cifrado ou adivinhar a chave. Esse requisito é usualmente enunciado de uma forma mais forte: o oponente deve ser incapaz de decifrar o texto cifrado ou descobrir a chave mesmo que esteja de posse de vários textos cifrados juntamente com o texto às claras que produziu cada texto cifrado.

Cópias da Chave Secreta

Remetente e destinatário devem obter cópias da chave secreta de maneira segura e mantê-las em segurança. Se alguém conseguir descobrir a chave e conhecer o algoritmo, toda comunicação que usar essa chave pode ser lida.

Métodos de Ataque

Existem dois métodos de ataque: **criptoanálise** ou **força-bruta**

Criptoanálise

- Ataques criptoanalíticos recorrem à natureza do algoritmo
- possivelmente algum conhecimento das características gerais do texto às claras
- possivelmente algumas amostras de pares de texto às claras e texto cifrado correspondente
- explora as características do algoritmo para tentar deduzir um texto às claras específico ou deduzir a chave que está sendo usada
- **Resultado:** efeito é catastrófico
  - todas as mensagens **futuras** e **passadas** cifradas com aquela chave são comprometidas

Ataques Criptoanalíticos (Parte 1/2)

- Apenas algoritmos fracos falham em ataque **somente texto cifrado**
  - geralmente projetados para resistir a **texto às claras conhecido**

somente texto cifrado

Menos informação, mais difícil de ocorrer.
Conhece Algoritmo de Cifração (assumimos isso em todos os casos) e também conhece o texto cifrado a ser decodificado

texto às claras conhecido

Adiciona alguns pares às claras/cifrados

texto às claras escolhido

O criptoanalista consegue escolher pares às claras/cifrados

Ataques Criptoanalíticos (Parte 2/2)

texto cifrado escolhido

O criptoanalista consegue escolher texto cifrado-alvo.

texto escolhido

Mais informação, mais fácil (embora mais raro de ocorrer).

O criptoanalista consegue escolher todo o esquema, exceto chave privada.

Ataque de Força Bruta

- O segundo método, conhecido como ataque de força bruta, é tentar todas as chaves possíveis em uma amostra de texto cifrado até obter tradução que leve a um texto às claras inteligível
- Deve ser tentado um valor proporcional ao quantitativo de todas as chaves possíveis para conseguir sucesso (tipicamente metade, em média)
- nesse nível de desempenho, uma chave de 56 bits não pode mais ser considerada segura em termos computacionais

Tempo médio requerido para busca exaustiva de chave

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} μs $\times$ 35,8 minutos	2,15 milissegundos
56	$2^{56} = 7{,}2 \times 10^{16}$	2^{55} μs = 1.142 anos	10,01 horas
128	$2^{128} = 3{,}4 \times 10^{38}$	2^{127} ms = $5{,}4 \times 10^{24}$ anos	$5{,}4 \times 10^{18}$ anos
168	$2^{168} = 3{,}7 \times 10^{50}$	2^{167} μs = $5{,}9 \times 10^{36}$ anos	$5{,}9 \times 10^{30}$ anos
26 caracteres (permutação)	$26! = 4 \times 10^{26}$	2×10^{26} μs = $6{,}4 \times 10^{12}$ anos	$6{,}4 \times 10^{6}$ anos

Figure 2: Retirado do livro-texto

ALGORITMOS COMPUTACIONALMENTE SEGUROS

- A criptografia é computacionalmente segura se:
  - Custo de quebrar cifra excede o valor da informação
  - Tempo necessário para quebrar a cifra excede a vida útil da informação
- Geralmente muito difícil estimar a quantidade de esforço necessária para quebrar
- Pode-se estimar tempo/custo de um ataque de força bruta

Algoritmos simétricos de cifração de bloco

- Os algoritmos de cifração simétricos mais comumente usados são cifras de bloco
- Uma cifra de bloco processa o texto às claras fornecido como entrada em blocos de tamanho fixo e produz um bloco de texto cifrado de tamanho igual para cada bloco de texto às claras
- O algoritmo processa cadeias mais longas de texto às claras como uma série de blocos de tamanho fixo
- Os algoritmos simétricos mais importantes, todos eles cifra de blocos, são o Data Encryption Standard (DES), o Triple DES (DES triplo) e o Advanced Encryption Standard (AES).

Comparação de três algoritmos de cifração simétricos populares

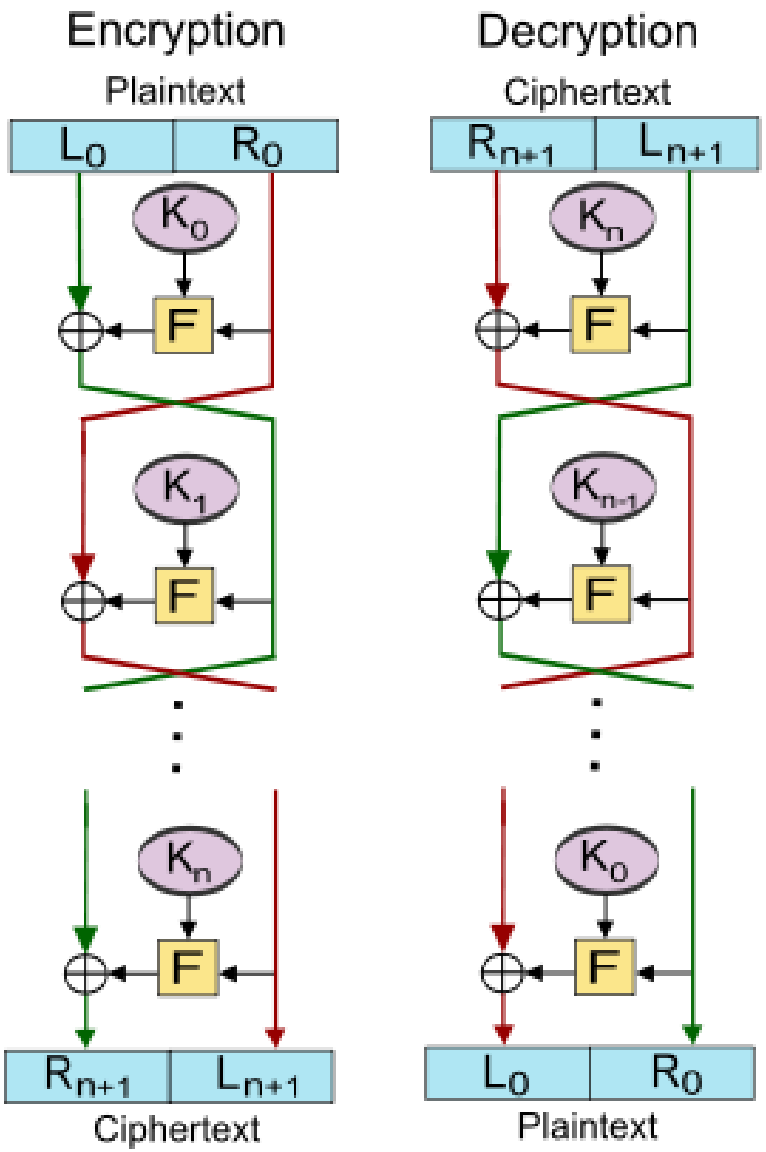
	DES	Triplo DES	AES
Tamanho do bloco de texto às claras (bits)	64	64	128
Tamanho do bloco de texto cifrado (bits)	64	64	128
Tamanho da chave (bits)	56	112 ou 168	128, 192 ou 256

DES = Data Encryption Standard
AES = Advanced Encryption Standard

Cifra de Feistel

- A Cifra de Feistel ou Rede de Feistel (também chamada de "Luby–Rackoff block cipher" que a criptoanalisaram) é utilizada em projetos bem-sucedidos de criptografia simétrica, como o DES
- Criada por Horst Feistel da IBM em 1973
- Entrada de cifração é um bloco de texto às claras de $2w$ bits de comprimento e uma chave K
- Bloco de texto às claras é dividido em duas metades, L0 e R0
- Duas metades passam por n rodadas de processamento
- Cada rodada i utiliza dados L_{i-1} e R_{i-1}
- Chaves K_i são derivadas da chave K, para cada rodada
- A cada rodada, dados dos lados esquerdo e direito (L e R) se invertem
- Decifração usa mesmo algoritmo, porém chaves K_i em ordem inversa
- Função F não precisa ser inversível, mas a rede sempre será
- Veja mais: https://en.wikipedia.org/wiki/Feistel_cipher

Ilustração da Cifra de Feistel



Parâmetros da Cifra de Feistel (Parte 1/2)

Tamanho do bloco

Blocos de tamanhos maiores significam maior segurança (se todos os outros parâmetros/aspectos forem iguais), mas velocidade de cifração/decifração reduzida. Um tamanho de bloco de 128 bits é um compromisso razoável quase universal em projetos recentes de cifra de bloco.

Tamanho da chave

Chaves de tamanhos maiores significam maior segurança, mas podem reduzir a velocidade de cifração/decifração. O comprimento de chave mais comum em algoritmos modernos é 128 bits.

Número de rodadas

A essência de uma cifra de bloco simétrica é que uma única rodada oferece segurança inadequada, mas várias rodadas oferecem segurança crescente. Um número típico é 16 rodadas.

Parâmetros da Cifra de Feistel (Parte 2/2)

Algoritmo de geração de subchaves

Maior complexidade nesse algoritmo deve resultar em maior dificuldade de criptoanálise.

Função de rodada

Novamente, maior complexidade geralmente significa maior resistência à criptoanálise.

Considerações de Projeto de uma Cifra de Feistel

Software de cifração/decifração rápida

Em muitos casos, mecanismos de cifração são embutidos em aplicações ou funções utilitárias de modo tal que não é possível a implementação em hardware. Dessa maneira, a velocidade de execução do algoritmo torna-se uma preocupação.

Facilidade de análise

Embora queiramos que o nosso algoritmo seja o mais difícil possível de criptoanalisar, há grande benefício se o algoritmo for fácil de analisar. Isto é, se o algoritmo puder ser explicado com concisão e clareza, é mais fácil analisá-lo em relação a vulnerabilidades criptoanalíticas e, por conseguinte, desenvolver um nível mais alto de garantias em relação a sua força. O DES, por exemplo, não tem funcionalidade fácil de analisar.

Data encryption standard (DES) - História

- O esquema de cifração mais amplamente usado é baseado no Data Encryption Standard (DES), adotado em 1977 pelo National Bureau of Standards (Escritório Nacional de Padrões), agora National Institute of Standards and Technology (NIST — Instituto Nacional de Padrões e Tecnologia)
- Publicado no Federal Information Processing Standard 46 (FIPS PUB 46)
- O algoritmo em si é conhecido como Data Encryption Algorithm (DEA – algoritmo de cifração de dados)
- O DES toma como entrada um bloco de texto às claras de 64 bits e uma chave de 56 bits, para produzir um bloco de texto cifrado de 64 bits.
- NIST retira em 2005 o FIPS 46-3: https://pt.wikipedia.org/wiki/Data_Encryption_Standard
- Em 2007, máquina paralela de FPGA da Universidade de Bochum e Kiel, Alemanha, viola o DES em aproximadamente seis dias e meio por um custo de $10,000 em hardware

Funcionamento do DES

- O DES funciona como uma Cifra de Feistel
- São utilizados n=16 rodadas (próximo slide)
- O bloco tem 64 bits (então meio bloco tem 32 bits)
- A chave K tem 64 bits, onde apenas 56 bits são efetivamente utilizados
- O mesmo algoritmo cifra e decifra, apenas bastando inverter a ordem das subchaves
- A função F tem quatro fases (próximos slides): Expansion (E-Expansion) (32 bits viram 48 bits), Key mixing, Substitution (S-Boxes), Permutation (P-Box) (de volta para 32 bits)
- Conceito de "confusion and diffusion" de Claude Shannon em 1940, na E-expansion e S-Boxes/P-Box dão garantia de segurança
- Key Schedule (próximos slides) divide os 56 bits da chave em duas partes de 28 bits (fase PC-1), depois rotacionados individualmente, e agregados novamente (fase PC-2) para gerar subchave de 48 bits
- Veja: https://en.wikipedia.org/wiki/Data_Encryption_Standard

Ilustração da Estrutura Feistel do DES

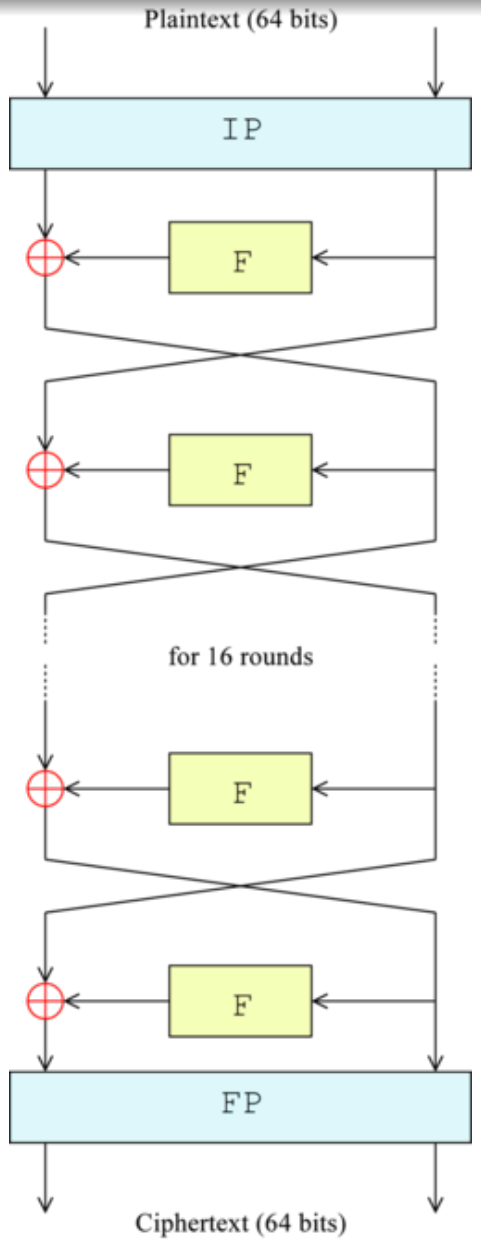


Ilustração da Função F do DES

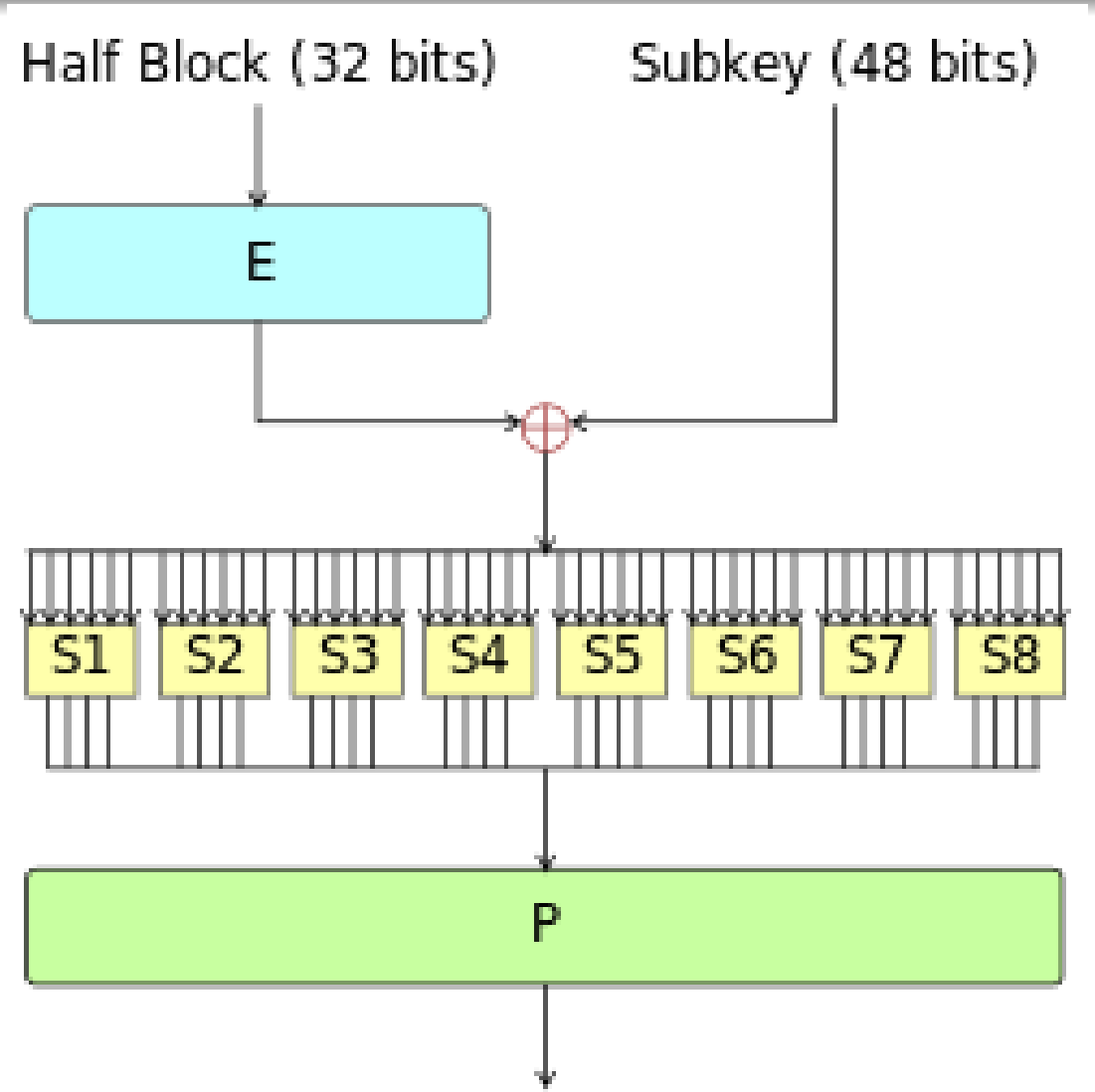


Figure 6: Wikipedia

Ilustração do Key Schedule do DES

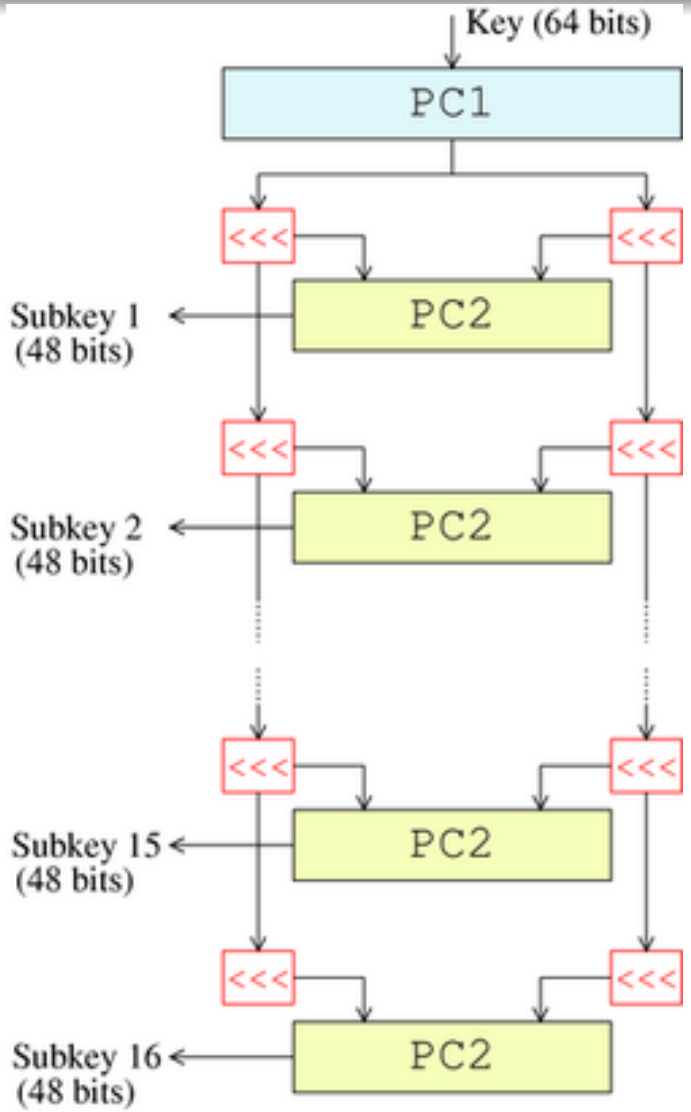


Figure 7: Wikipedia

DES - Preocupações (História)

- Preocupações com a resistência do DES caem em duas categorias: preocupações com o algoritmo em si e preocupações com a utilização de uma chave de 56 bits
- A primeira preocupação refere-se à possibilidade de uma criptoanálise pela exploração das características do algoritmo DES
  - Ao longo dos anos houve numerosas tentativas de encontrar e explorar fraquezas no algoritmo, o que transformou o DES no mais estudado algoritmo de cifração existente.
  - Apesar das numerosas abordagens, até agora ninguém relatou fraqueza fatal no DES.
- Uma preocupação mais séria é o comprimento da chave
  - Com comprimento de chave de 56 bits, há 2^{56} chaves possíveis, o que equivale a aproximadamente $7,2 \times 10^{16}$ chaves.
  - Assim, à primeira vista, um ataque de força bruta parece não ser prático
  - O DES provou-se definitivamente inseguro em julho de 1998, quando a Electronic Frontier Foundation (EFF) anunciou que tinha decifrado uma cifração DES usando uma máquina especializada denominada "decifradora DES" (DES cracker) construída por menos de USD 250k

DES - Estratégias de Ataque e Contra-Ataque

- Há mais por trás de um ataque de busca de chave do que simplesmente executar todas as chaves possíveis
- A menos que um texto às claras seja fornecido, o analista deve reconhecer o texto às claras como sendo de fato um texto às claras
- Se a mensagem for composta apenas por texto às claras em português, o resultado surgirá facilmente, se bem que a tarefa de reconhecer a língua portuguesa terá de ser automatizada
- Se a mensagem de texto foi comprimida antes da cifração, o reconhecimento será mais difícil
- Mensagem como dado mais geral (arquivo numérico), e se esse arquivo foi comprimido: ainda mais difícil de automatizar
- Assim, para suplementar a abordagem de força bruta, é preciso algum grau de conhecimento sobre o texto às claras esperado e alguns meios de distinguir automaticamente o texto às claras de um texto qualquer
- **Solução:** se a única forma de ataque que poderia ser feita a um algoritmo de cifração for a força bruta, o modo de contra-atacá-lo é óbvio: usar chaves mais longas.

Aumento exponencial do tempo

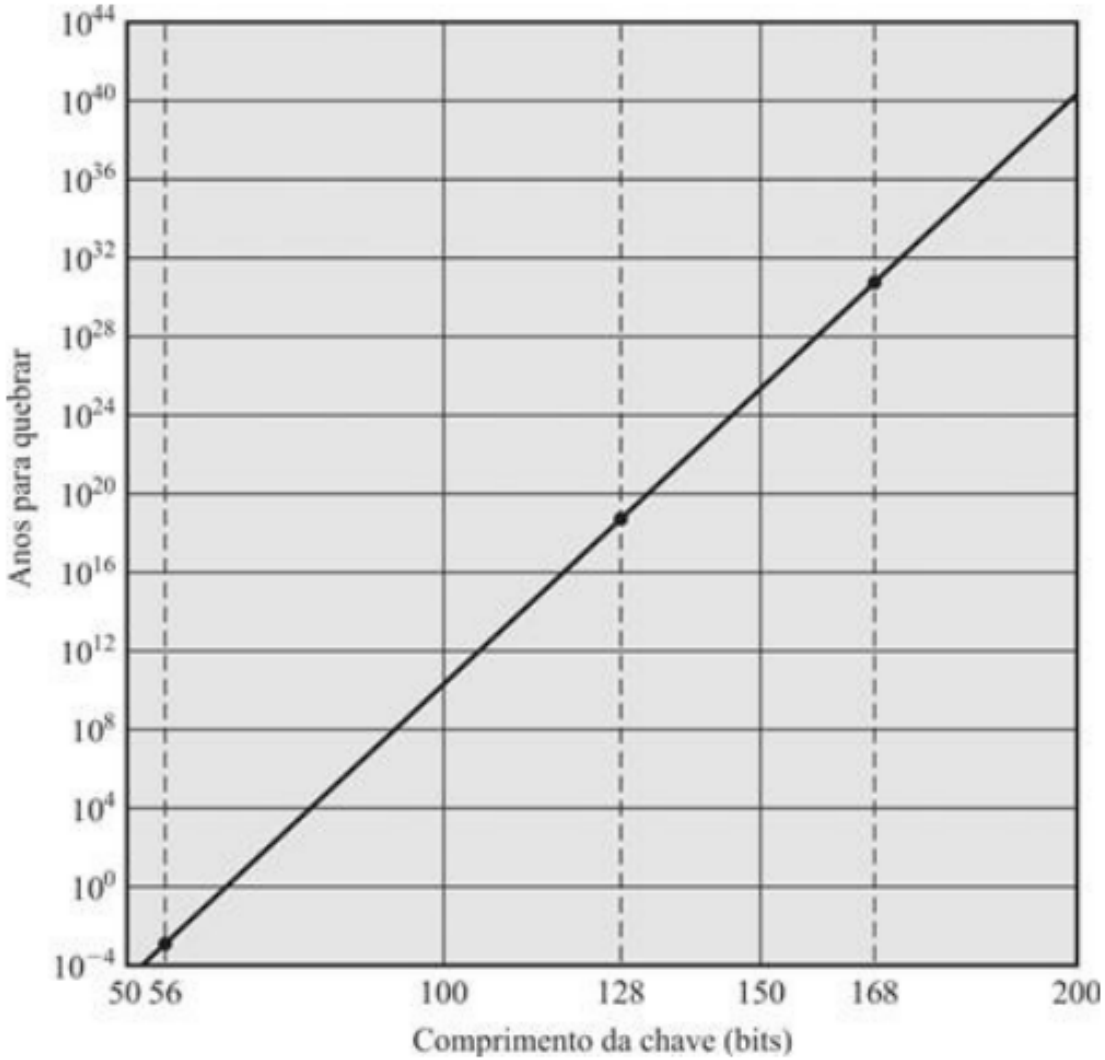


FIGURA 2.2 Tempo para quebrar um código (considerando 10^6 decifrações/μs) O gráfico considera que um algoritmo de cifração simétrico é atacado usando uma

Triplo DES - 3DES

- A vida do DES foi estendida pela utilização do triplo DES (DES triplo ou 3DES)
  - repetir o algoritmo DES básico três vezes, usando duas ou três chaves únicas, para obter um tamanho de chave de 112 ou 168 bits
- O triplo DES (3DES) foi padronizado pela primeira vez para uso em aplicações financeiras no padrão ANSI X9.17 em 1985
- O 3DES foi incorporado como parte do Data Encryption Standard em 1999, com a publicação do FIPS PUB 46- 3

3DES - Vantagens e Desvantagens

- O 3DES tem dois atrativos que garantem sua utilização ampla nos próximos anos
- A primeira é que, com o seu comprimento de chave de 168 bits, ele supera a vulnerabilidade do DES ao ataque de força bruta
- A segunda é que o algoritmo de cifração subjacente ao 3DES é o mesmo que no DES
  - Algoritmo submetido a mais escrutínio do que qualquer outro algoritmo de cifração por um período de tempo mais longo e nenhum ataque criptoanalítico efetivo baseado no algoritmo, a não ser o de força bruta, foi encontrado
- Alto nível de confiança que 3DES é muito resistente à criptoanálise
- Usa três chaves e três execuções DES: $C = E(K_3, D(K_2, E(K_1, P)))$
- Uso de decifração no segundo estágio dá compatibilidade com usuários originais de DES
- A principal desvantagem do 3DES é que o algoritmo é relativamente lento em software e tem blocos muito pequenos
- Blocos de apenas 64 bits, onde 56 bits são usados (8 para paridade)

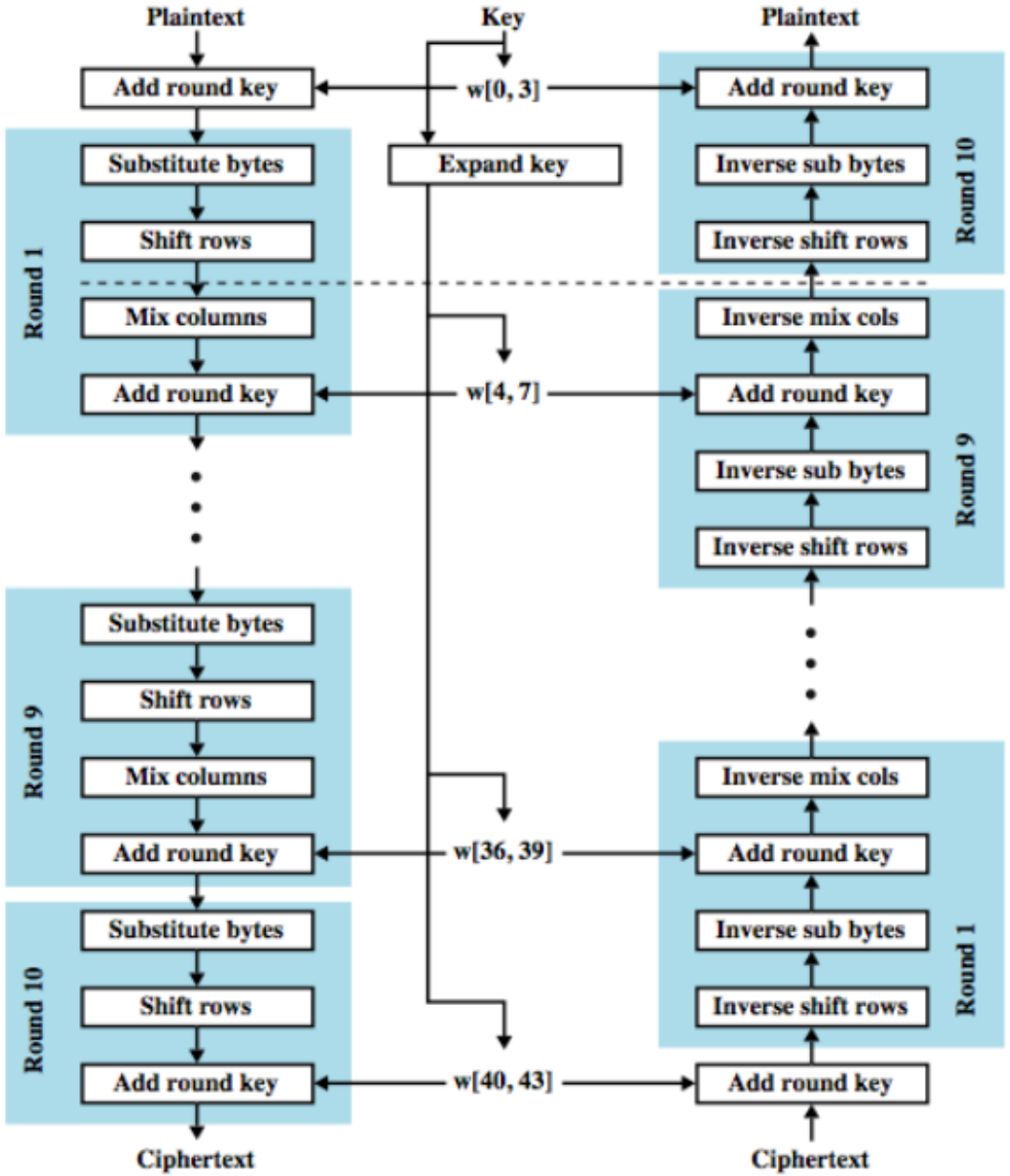
Advanced encryption standard (AES) - História

- 3DES não é um candidato razoável para utilização a longo prazo
- Como substituto, o NIST publicou em 1997 uma chamada de propostas para um novo Advanced Encryption Standard (AES)
- Cifra de bloco simétrica com comprimento de bloco de 128 bits e suporte para comprimentos de chaves de 128, 192 e 256 bits
- Critérios de avaliação incluíam: segurança, eficiência computacional, requisitos de memória, adequabilidade de hardware e software e flexibilidade
- Primeira rodada de avaliação, 15 algoritmos propostos foram aceitos
- Segunda rodada reduziu esse número a cinco algoritmos
- O NIST concluiu seu processo de avaliação e publicou um padrão final (FIPS PUB 197) em novembro de 2001 e selecionou o algoritmo de Rijndael como o algoritmo AES proposto: autores belgas Vincent Rijmen e Joan Daemen
- Agora esse algoritmo está amplamente disponível em produtos comerciais

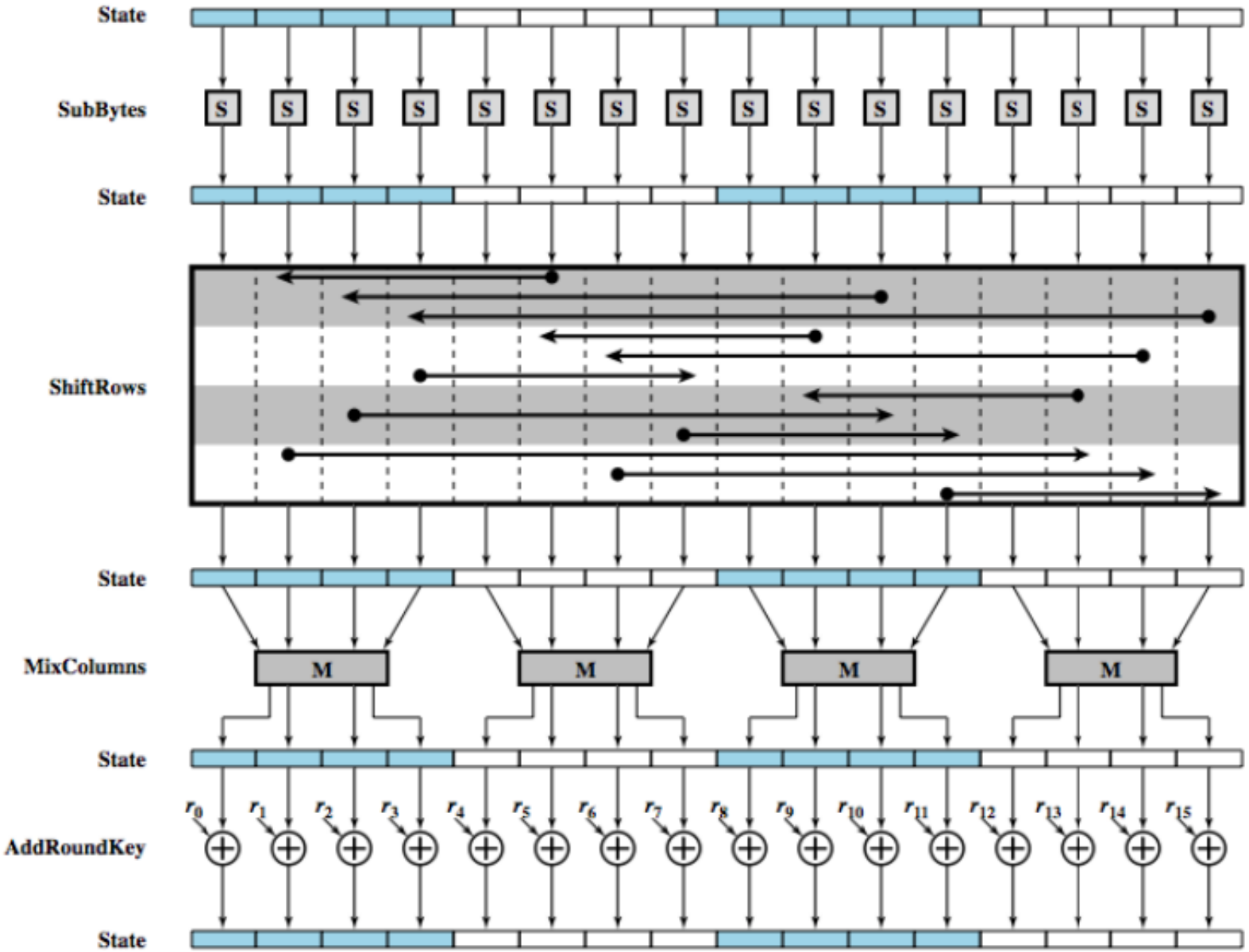
Ideia Geral do AES

- No AES, não é utilizada uma Rede de Feistel
- Implementação complexa com uso de permutações e substituições
- Entrada para algoritmos de cifração e decifração é um bloco de 128 bits
- No FIPS PUB 197, bloco representado por matriz quadrada de bytes
  - O bloco é copiado para o vetor **Estado**, que é modificado a cada estágio de cifração ou decifração
  - Após o estágio final, **Estado** é copiado para uma matriz de saída
- Chave de 128 bits representada como matriz quadrada de bytes
  - chave é expandida para um vetor de palavras de escalonamento de chave
  - cada palavra tem 4 bytes
  - escalonamento total de chaves tem 44 palavras para a chave de 128 bits
- A ordenação dos bytes dentro de uma matriz é feita por colunas
- Primeiros 4 bytes de uma entrada de texto às claras de 128 bits passada para a cifra criptográfica ocupam a primeira coluna da matriz **entrada**
  - segundos 4 bytes ocupam a segunda coluna, e assim por diante
- Primeiros 4 bytes da chave expandida, que formam uma palavra, ocupam a primeira coluna da matriz **w**

Estrutura Geral do AES (não usa Feistel)



Estrutura Geral da Rodada do AES



Section 4

Discussão

Breve discussão

Segurança para textos grandes

Como cifrar textos grandes usando criptografia simétrica de poucos bits?
Veja próximo módulo para entender melhor!

Leia mais

Livro:

- "Segurança de Computadores - Princípios e Práticas - 2012" - Stallings, William; Brown, Lawrie & Lawrie Brown & Mick Bauer & Michael Howard
  - Em Português do Brasil, CAMPUS - GRUPO ELSEVIER, 2ª Ed. 2014

Veja Capítulo 7, todas seções e finaliza o capítulo 7.

Section 5

Agradecimentos

Pessoas

Em especial, agradeço aos colegas que elaboraram bons materiais, como o prof. Raphael Machado, Kowada e Viterbo cujos conceitos formam o cerne desses slides.

Estendo os agradecimentos aos demais colegas que colaboraram com a elaboração do material do curso de Pesquisa Operacional, que abriu caminho para verificação prática dessa tecnologia de slides.

Software

Esse material de curso só é possível graças aos inúmeros projetos de código-aberto que são necessários a ele, incluindo:

- pandoc
- LaTeX
- GNU/Linux
- git
- markdown-preview-enhanced (github)
- visual studio code
- atom
- revealjs
- groomit-mpx (screen drawing tool)
- xournal (screen drawing tool)
- ...

Empresas

Agradecimento especial a empresas que suportam projetos livres envolvidos nesse curso:

- github
- gitlab
- microsoft
- google
- ...

Reprodução do material

Esses slides foram escritos utilizando pandoc, segundo o tutorial ilectures:

- https://igormcoelho.github.io/ilectures-pandoc/

Exceto expressamente mencionado (com as devidas ressalvas ao material cedido por colegas), a licença será Creative Commons.

Igor Machado Coelho

This Slide Is Intentionally Blank (for goomit-mpx)