Criptografia simétrica

Segurança de dados – MSI

Prof. Daniel Saad Nogueira Nunes

Introdução

- Um elemento importantíssimo em vários serviços e aplicações são os algoritmos criptográficos.
- A criptografia simétrica pode ser utilizada em vários contextos, mas principalmente para obtenção da **confidencialidade**.
- Funções seguras de *hashing* podem ser utilizadas para prover autenticidade.

Criptografia Simétrica

- A criptografia simétrica provê confidencialidade para dados transmitidos ou armazenados.
- Até antes da década de 70 era o único tipo de criptografia utilizada.
- Após o advento a criptografia de chave pública, ela ainda permanece como a mais amplamente utilizada.

Criptografia de Chave Simétrica

Elementos

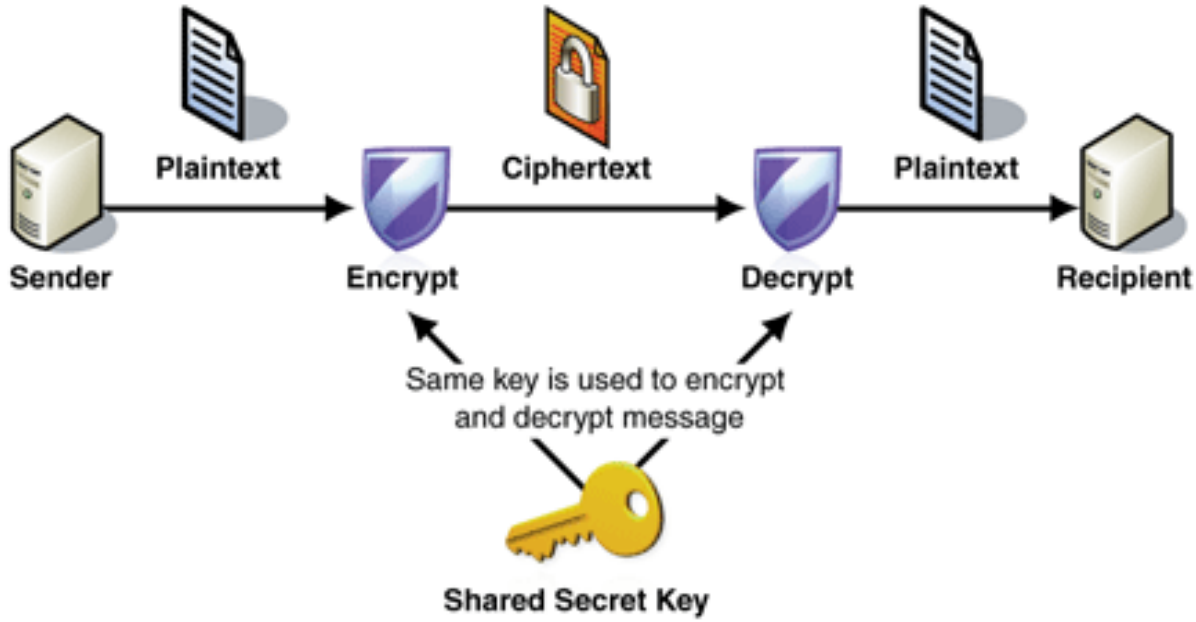
- Texto às claras: mensagem original.
- Algoritmo de cifração: responsável por executar várias substituições e transformações no texto às claras.
- Chave secreta: utilizada no algoritmo de cifração. As substituições e cifrações feitas pelo algoritmo de cifração dependem do valor da chave.

Criptografia de Chave Simétrica

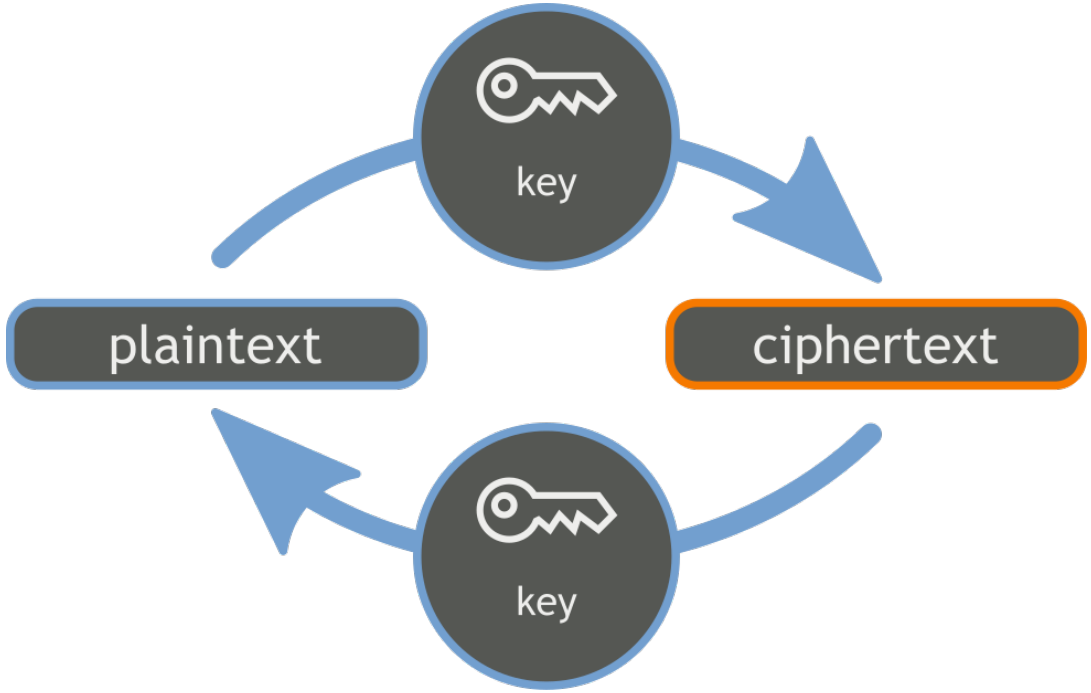
Elementos

- Texto cifrado: mensagem embaralhada produzida como saída do algoritmo de cifração. Idealmente, para dada mensagem, duas chaves diferentes produzirão dois textos cifrados diferentes.
- Algoritmo de decifração: tem como parâmetro a chave secreta e o texto cifrado e produz o texto às claras original.

Criptografia de Chave Simétrica



Criptografia de Chave Simétrica



Criptografia Simétrica

Requerimentos

- Precisamos de um algoritmo de encriptação forte.
- Devemos assumir que, mesmo que o adversário detenha a chave, e textos cifrados juntamente com cada texto às claras que produziu cada texto cifrado, seja difícil descobrir a chave utilizada.

Criptografia Simétrica

Requerimentos

- Outro problema é na obtenção da chave.
- Remetente e destinatário devem obter cópias da chave de maneira segura e mantê-las em segurança.
- Se um adversário descobre a chave, e sabe os algoritmos de cifração e decifração, toda a comunicação pode ser lida.
- Quebra da confidencialidade.

Ataques em Criptografia Simétrica

- Há duas abordagens gerais para atacar um esquema de cifração simétrica.
  - Criptoanálise.
  - Força-Bruta.

Ataques em Criptografia Simétrica

Criptoanálise

- Ataques criptoanalíticos recorrem à natureza do algoritmo.
- Podem utilizar também conhecimentos gerais das características do texto às claras e até mesmo de amostras de pares de textos às claras e textos cifrados correspondentes.
- Se o ataque for bem-sucedido na dedução da chave, o efeito é catastrófico.
- Todas as mensagens são comprometidas.

Ataques em Criptografia Simétrica

Força-Bruta

- O método de força-bruta tenta todas as chaves possíveis em uma amostra de texto cifrado até obter tradução que leve a um textos às claras inteligível.
- Em média: metade de todas as chaves possíveis deve ser tentada.
- É importante notar que o algoritmo força bruta faz mais do que varrer o espaço de chave.
- De alguma forma ele deve dizer se o texto às claras correspondente faz sentido.

Ataques em Criptografia Simétrica

Força-Bruta

- Mesmo dispondo de supercomputadores, organizados em uma arquitetura paralela, é inviável a quebra da chave via força-bruta.

Força-Bruta

Tamanho da chave	Número de Chaves	Tempo com 1 decifração/μs	Tempo com 10^6 decifrações por μs
32	$2^{32} = 4.3 \cdot 10^9$	35.8 minutos	2.15 ms
56	$2^{56} = 7.2 \cdot 10^{16}$	1142 anos	10.01 horas
128	$2^{128} = 3.4 \cdot 10^{38}$	$5.4 \cdot 10^{24}$ anos	$5.4 \cdot 10^{18}$ anos
168	$2^{168} = 3.7 \cdot 10^{50}$	$5.9 \cdot 10^{36}$ anos	$5.9 \cdot 10^{30}$ anos
26 caracteres	$26! = 4 \cdot 10^{26}$	$6.4 \cdot 10^{12}$ anos	$6.4 \cdot 10^6$ anos

Algoritmos Simétricos de Cifração de Bloco

- Os algoritmos de cifração simétricos mais comuns são cifras de blocos.
- Uma cifra de bloco divide o texto às claras em blocos de tamanho fixo.
- Blocos de texto cifrado de igual tamanho são produzidos para cada bloco de texto às claras.

Algoritmos Simétricos de Cifração de Bloco

DES

- DES: Data Encryption Standard.
- Amplamente utilizado, mas agora é considerado inseguro.
- Extremamente importante no avanço da criptografia moderna.

DES

- Divide o texto às claras em blocos de 64 bits.
- Utiliza chaves de 56 bits.
- Produz blocos de texto cifrado de 64 bits.

DES

- Preocupações com o DES dividem-se em duas categorias:
  1. Preocupação com fraquezas no algoritmo.
  2. Preocupação com a utilização de uma chave de 56 bits.

DES

- Durante vários anos, houve numerosas tentativas, através de criptoanálise, de encontrar e explorar fraquezas no DES.
- Apesar disto, não foi relatado nenhuma fraqueza fatal no DES.

DES

- Outra preocupação séria, que tornou o padrão inseguro, é a da utilização de chaves com comprimento de 56 bits.
- Equivale a $\approx 7.2 \cdot 10^{16}$.
- Olhando a tabela anterior, um ataque força bruta não pareceria viável em um computador simples.
- Mas utilizando um maior poder de processamento, a tarefa torna-se trivial.

DES

- Outra preocupação séria, que tornou o padrão inseguro, é a da utilização de chaves com comprimento de 56 bits.
- Equivale a $\approx 7.2 \cdot 10^{16}$.
- Olhando a tabela anterior, um ataque força bruta não pareceria viável em um computador simples.
- Mas utilizando um maior poder de processamento, a tarefa torna-se trivial.

DES

- Se o algoritmo de cifração não possui fraquezas e a única alternativa é a força-bruta, a contramedida é óbvia.
- Aumentar o tamanho da chave.

Triplo DES

- O Triplo DES (ou 3DES) utiliza o algoritmo DES básico três vezes.
- Utiliza duas ou três chaves únicas de modo a conseguir um tamanho de chave de 112 ou 168 bits.

Triplo DES

- Em relação ao DES simples, ele tem dois atrativos que garantem a sua utilização segura pelos próximos anos.
  1. Com o comprimento de chave de 168-bits, ele supera a vulnerabilidade do DES ao ataque de força-bruta.
  2. Como o 3DES utiliza o DES como base, sabemos que ele é resistente, uma vez que nenhum ataque criptoanalítico foi capaz de revelar fraquezas.

Triplo DES

- O 3DES no entanto tem como desvantagem a sua lentidão quando implementado em software.
- O DES original foi projetado para implementação em hardware.
- Como o 3DES requer três vezes mais cálculos que o DES ele fica mais lento quando implementado em software.
- Outra desvantagem é o tamanho de bloco utilizado. Ambos DES e 3DES, utilizam um bloco de 64 bits. Por questão de eficiência e segurança, um tamanho maior de bloco é mais desejável na prática.

Triplo DES

- Por conta destas desvantagens, o 3DES não é um candidato muito forte para permanecer por vários anos.
- Para contornar estes obstáculos, o AES foi proposto.

AES

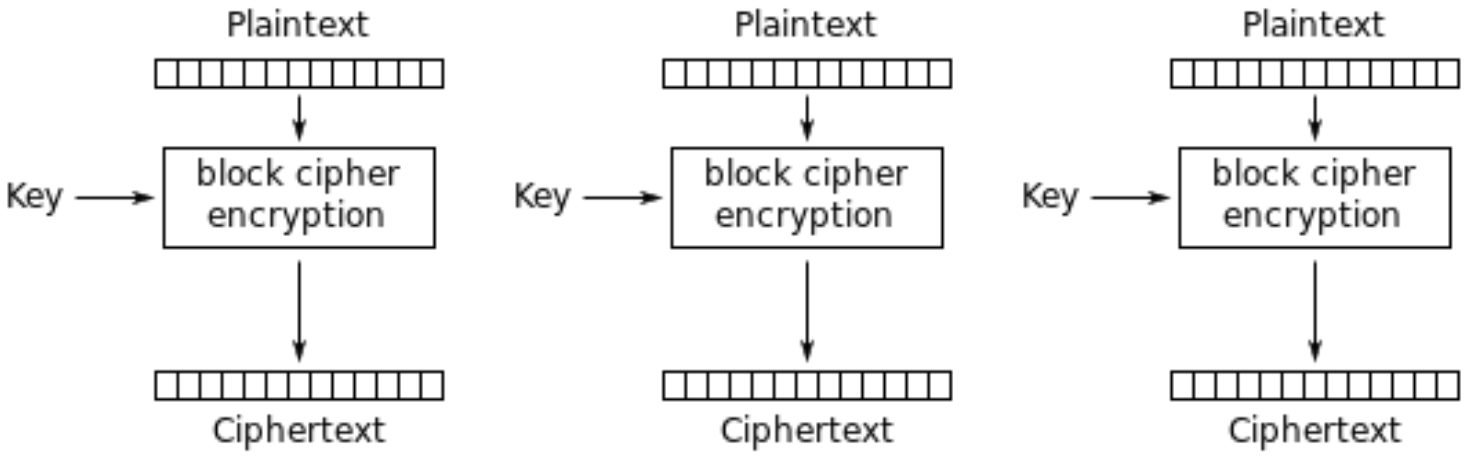
- AES: Advanced Encryption Standard.
- O NIST, verificando as desvantagens do 3DES, publicou uma chamada de propostas para um novo padrão de esquema criptográfico.
- Em uma primeira avaliação, 15 algoritmos propostos foram aceitos.
- Uma segunda rodada reduziu para 5 algoritmos.
- Enfim, o algoritmo de Rijndael foi aceito como a solução.

Criptografia Simétrica

Questões Práticas de Segurança

- A cifração simétrica é aplicada geralmete a blocos de 64 ou 128 bits.
- Mensagens de texto, e-mail e outras fontes de texto em claro são divididos em blocos deste tamanho.
- A abordagem mais simples para a cifração de múltiplos blocos é conhecida como ECB (eletronic codebook).
- ECB: divide um texto às claras de nb bits em partições $\{P_1, \ldots, P_n\}$ de b bits.
- Cada bloco deste é cifrado usando a mesma chave, produzindo então uma sequência de n BLOCOS DE b bits de texto $\{C_1, \ldots, C_n\}$.

Criptografia Simétrica



Electronic Codebook (ECB) mode encryption

Criptografia Simétrica

Questões Práticas de Segurança

- Para mensagens longas e padronizadas, o modo ECB não pode ser seguro.
- Um criptoanalista pode explorar regularidades no texto às claras para facilitar uma decifração.
- Ex: se a mensagem sempre começa com um padrão definido, o criptoanalista tem uma dica.
- Para aumentar a segurança da cifração de bloco simétrica, várias técnicas alternativas foram desenvolvidas, os **modos de operação**.
- Estes modos superam a fraqueza do ECB.

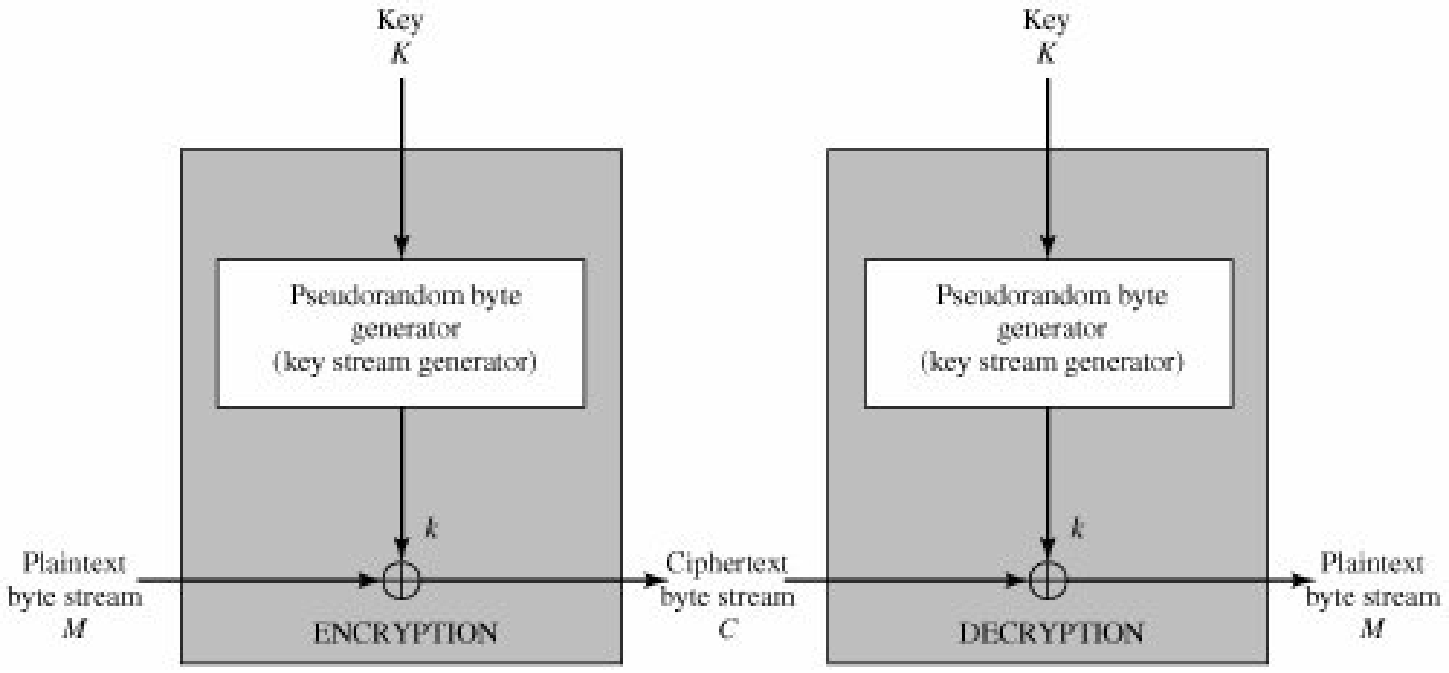
Cifras de Fluxo

- Enquanto uma cifra de fluxo processa a entrada um bloco por vez, a cifra de fluxo processa elementos sequencialmente.
- A cada elemento da entrada, um elemento da saída é produzido.
- Embora cifras de blocos sejam mais utilizadas, há aplicações em que uma cifra de fluxo é mais adequada.

Cifras de Fluxo

- Uma cifra de fluxo típica faz a cifração de texto às claras um byte por vez, embora também possa ser projetada para atuar sobre bits individuais ou sobre uma quantidade maior de bits.
- Em uma estrutura que opera sobre bytes, uma chave entra em um gerador de bits **pseudoaleatórios**, que produz um fluxo de números de 8 bits aparentemente aleatórios.
- Um fluxo pseudoaleatório é um fluxo imprevisível sem o conhecimento de entrada e tem caráter aparentemente aleatório.
- A saída do gerador, denominada de **fluxo de chave**, é combinada um byte por vez com o fluxo de texto às claras, usando $\oplus$.

Cifras de Fluxo



Cifras de Fluxo

- Com o gerador de números pseudoaleatórios adequadamente projetado, uma cifra de fluxo pode ser tão segura quanto uma cifra de bloco de comprimento de chave comparável.
- A vantagem imediata em relação à cifras de bloco é que cifras de fluxo são mais rápidas.
- A vantagem de cifras de blocos é que é possível reutilizar chaves.

Cifras de Fluxo

- Ideias para aplicações como stream de vídeos, aúdio,. . .
- Cifras de bloco são melhores alternativas quando o texto às claras está inteiramente disponível.

Autenticação

- A cifração fornece proteção contra ataques passivos (escutas).
- Um requisito diferente é proteger contra ataques ativos (falsificação de dados e transações).
- A proteção contra tais ataques é denominada autenticação de mensagem e dados.

Autenticação

- Uma mensagem, arquivo, documento ou coleção de dados é dita **autêntica** quando é genuína e veio de sua fonte alegada.
- Autenticação é um procedimento que permite que as partes comunicantes verifique se as mensagens recebidas ou armazenadas são autênticas.

Autenticação

- Também podemos desejar verificar se uma mensagem foi transmitida no momento correto (se ela não foi artificialmente atrasada ou repetida) e a sequência em relação a outras mensagens que fluem entre duas partes.
- Todas estas preocupações são agrupadas sob a categoria de integridade de dados, vista anteriormente.

Autenticação Utilizando Criptografia Simétrica

- A criptografia simétrica garante integridade dos dados?
- Ela garante autenticação?

Autenticação Utilizando Criptografia Simétrica

- Considerando que somente o remetente e destinatário compartilham chave, então somente o remetente genuíno conseguiria cifrar uma mensagem válida para o destinatário.
- Além disso, se a mensagem incluir um código de detecção de erros e um número de sequência, o destinatário terá certeza que não houve qualquer alteração e que o sequenciamento é adequado.
- Se a mensagem ainda incluir um *timestamp*, o destinatário terá certeza de que a mensagem não foi atrasada além do que é normalmente esperado.

Autenticação Utilizando Criptografia Simétrica

- No entanto, a utilização da criptografia simétrica isoladamente **NÃO** é o suficiente para garantir autenticidade.
- No modo de cifração ECB, um atacante pode reordenar os blocos cifrados, de modo que o destinatário conseguirá decifrá-los (mas na ordem incorreta).

Autenticação de Mensagem sem Cifração

- Para autenticar mensagens sem cifração, necessitamos de um mecanismo conhecido como tag de autenticação.
- Iremos explicar como autenticar mensagens sem cifração. No entanto não garantiremos confidencialidade.
- Combinando os conceitos aqui apresentados com a cifração, obteremos confidencialidade e integridade.

Autenticação de Mensagem sem Cifração

- A autenticação de mensagens muita das vezes pode ser fornecida como uma funcionalidade separada da cifração.

Autenticação de Mensagem sem Cifração

Exemplo

Cenário: Notificação de usuários de que um serviço está indisponível.

- Um sistema central recebe dos demais um texto em claro com uma tag de autenticação.
- Ele verifica a autenticidade da mensagem e propaga para os demais hospedeiros em caso da mensagem ser genuína.
- Caso a mensagem tenha sido violada, um alerta geral é enviado aos demais hospedeiros.

Autenticação de Mensagem sem Cifração

Exemplo

Cenário: execução de um programa.

- Um programa com tags de identificação pode ser utilizado sem ser decifrado.
- Economia de recursos computacionais.
- Podemos verificar a integridade do programa ao avaliar a tag.

MAC

- Uma técnica de autenticação envolve uma chave secreta para gerar um pequeno bloco de dados, o MAC ou Message Authentication Code.
- Suponha que A e B compartilham uma chave K_{AB} em comum.
- Quando A tem uma mensagem a ser enviada a B, A calcula o MAC como uma função dependente da mensagem e da chave:

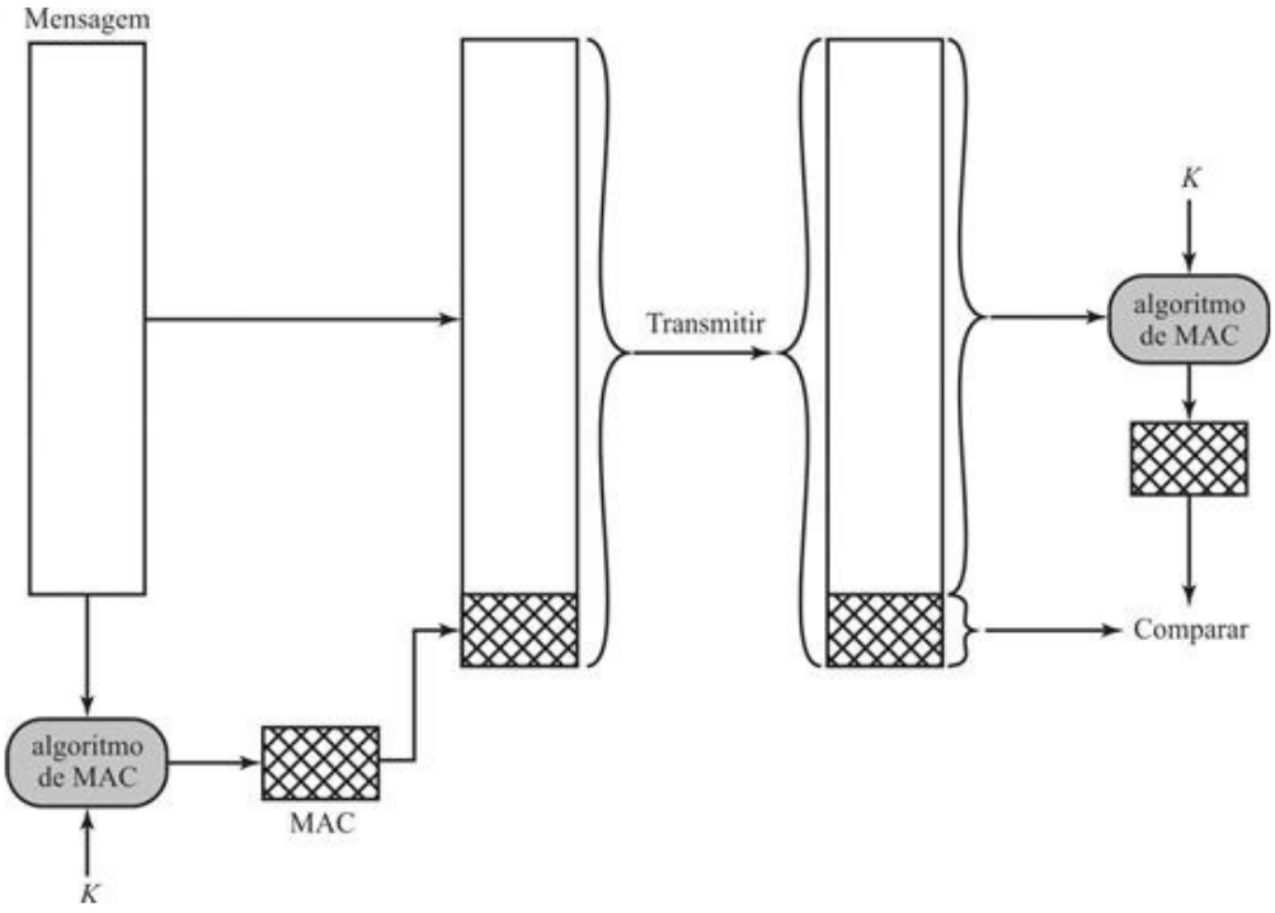
$$\mathrm{MAC}_M = f(K_{AB}, M)$$

- A mensagem anexada do código MAC_M são enviadas ao destinatário.

MAC

- B executa o mesmo cálculo na mensagem recebida usando a mesma chave secreta para gerar um novo MAC'_M.
- MAC'_M é comparado com MAC_M de modo a verificar a integridade.

MAC


Mensagem
Transmitir
K
algoritmo de MAC
Comparar
algoritmo de MAC
MAC
K

MAC

- Considerando que apenas o destinatário e o remetente conhecem o valor da chave secreta e se o código recebido corresponder ao código calculado então podemos concluir três coisas:
  1. O destinatário tem certeza de que a mensagem não foi alterada.
  2. O destinatário tem certeza de que a mensagem veio do remetente alegado.
  3. Se for feita uma inclusão de um número de sequência, o destinatário tem certeza de que a mensagem veio na ordem correta.

MAC

- Vários algoritmos poderiam ser utilizados para gerar o código.
- Uma recomendação existente aponta o DES para ser utilizado para cifrar a mensagem e alguns bits do final do texto cifrado poderiam ser utilizados como MAC.
- Tipicamente 16 ou 32 bits finais.
- NIS FIPS PUB 113.

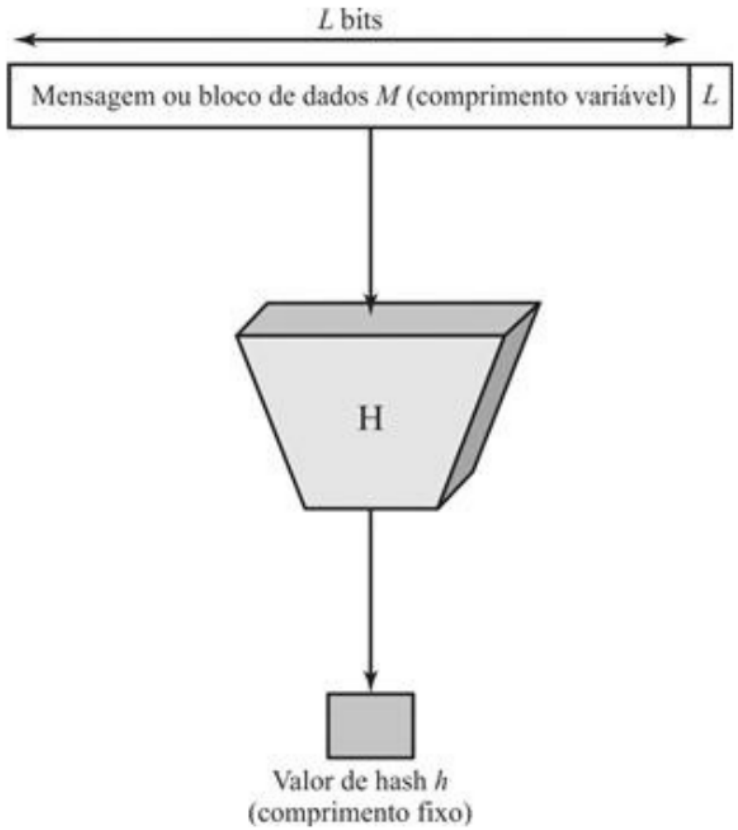
MAC

- O esquema apresentado é parecido com a cifração.
- Uma diferença é que o algoritmo de autenticação não precisa ser reversível, como deve ser para decifração.
- Por conta de suas propriedades matemáticas, a função de autenticação é menos vulnerável à quebra do que a cifração.

Função de Hash de uma Via

- Uma alternativa para o MAC é a utilização de funções hash de uma via (não inversíveis).
- Como ocorre com o MAC, uma função hash aceita uma mensagem M e produz um **resumo** criptográfico $h(M)$, de tamanho fixo.
- Normalmente a mensagem é preenchida até um múltiplo de algum comprimento fixo (Ex: 1024 bits), processo conhecido como padding.
- O padding é utilizado como medida para aumentar a dificuldade do atacante de produzir uma mensagem alternativa com o mesmo valor de hashing.

Função de Hash de uma Via



Função de Hash de uma Via

- Ao contrário do MAC, uma função de hash não precisa de uma chave secreta como entrada.
- Para autenticar a mensagem, o resumo criptográfico é enviado juntamente com esta.
- Premissa: a função hash tem que ser segura.

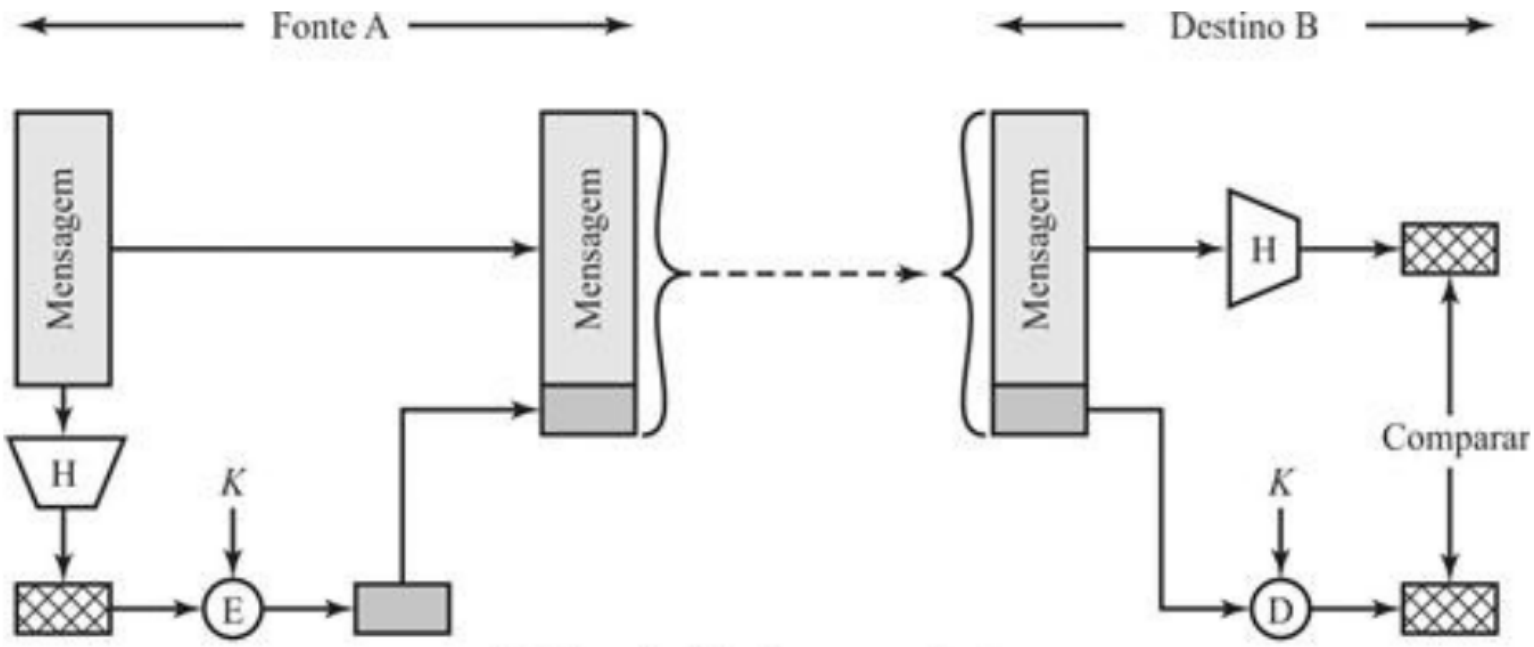
Função de Hash de uma Via

- Existem várias formas de utilizar a função hash para garantir integridade:
  1. Cifração convencional.
  2. Cifração de chave pública.
  3. Utilização de valor secreto.

Autenticação via Hashing

- Utilizando cifração simétrica, o resumo da mensagem só pode ser decifrado utilizando a chave secreta.
- Como a chave só é conhecida pelos comunicantes, a autenticação está assegurada.

Função de Hash de uma Via

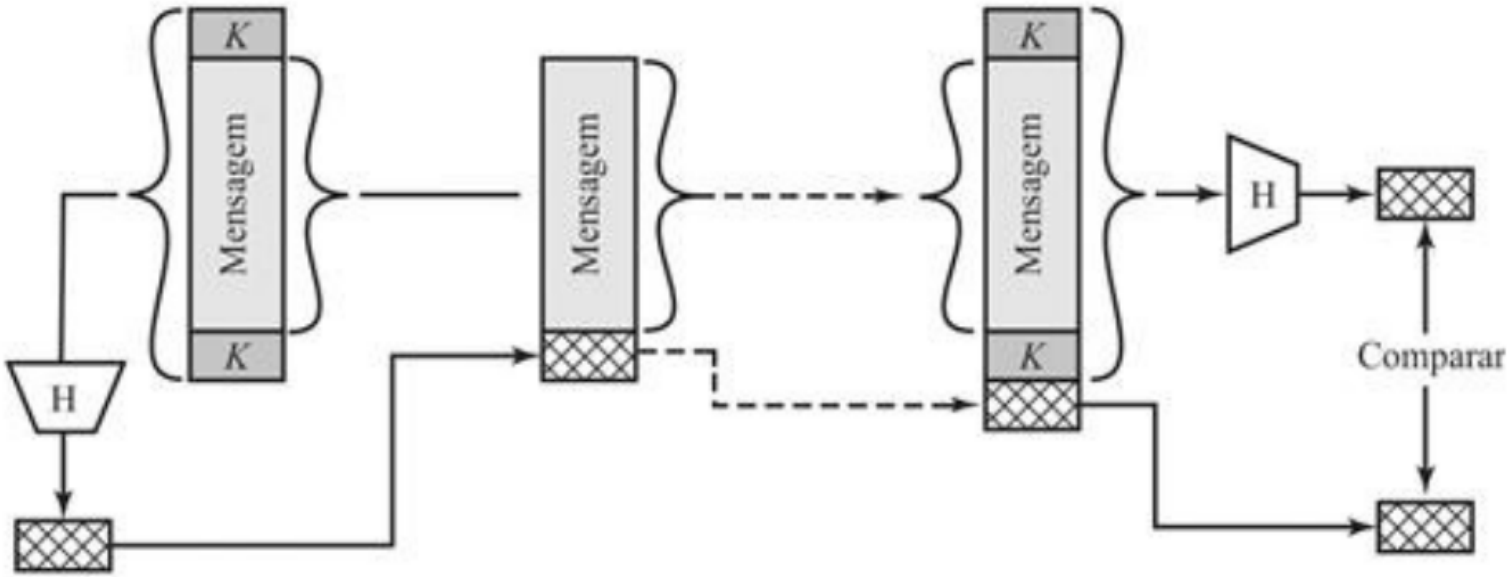


(a) Usando cifração convencional

Autenticação via Hashing

- Sem utilizar a cifração, é possível possibilitar a autenticação sem maiores custos computacionais.
- Requer um valor secreto compartilhado entre os comunicantes.

Função de Hash de uma Via



(c) Usando valor secreto

GPG

- O Gnu Privacy Guard (GPG) é uma implementação do padrão OpenPGP, para prover comunicação segura.
- Baseado em criptografia simétrica e de chave pública.
- Com ele podemos:
  - Gerenciar chaves.
  - Cifrar e decifrar documentos.
  - Assinar documentos digitalmente.

GPG

Cifração

Uso: `gpg --symmetric --cipher-algo AES256 filename`

- Esse comando requisitará uma senha (chave), que será utilizada para cifrar o documento.
- A flag `--symmetric` diz que queremos utilizar a criptografia simétrica.
- A flag `--ciper-algo` especifica o algoritmo de cifração a ser utilizado.

GPG

Decifração

Uso: `gpg -o decrypted_filename -d filename`

- Esse comando requisitará uma senha (chave), que será utilizada para decifrar o documento.

Exercício 1

Crie um arquivo texto com uma mensagem e cifre ele com GPG.
Mande o arquivo para dois amigos via e-mail. Um deverá saber o segredo (chave) utilizada na cifração, enquanto o outro não.
Peça para os dois decifrarem a mensagem e veja o que acontece.

Exercício 2

Crie um arquivo texto com uma mensagem e crie outro arquivo contendo o hash SHA256 dele.
Compacte o arquivo com `.tar.gz`
Envie o arquivo compactado para um colega que saiba o segredo.
Peça para o amigo verificar se o arquivo está íntegro.