PMR3412 - Redes Industriais - 2021

Aula 10 - Segurança: Conceitos Básicos - Integridade e Autenticação: HMAC, Assinaturas Digitais e Certificados

Prof. Dr. André Kubagawa Sato
Prof. Dr. Marcos de Sales Guerra Tsuzuki

21 de Outubro de 2021

PMR-EPUSP

Revisão

Criptografia - Princípios básicos e Confidencialidade

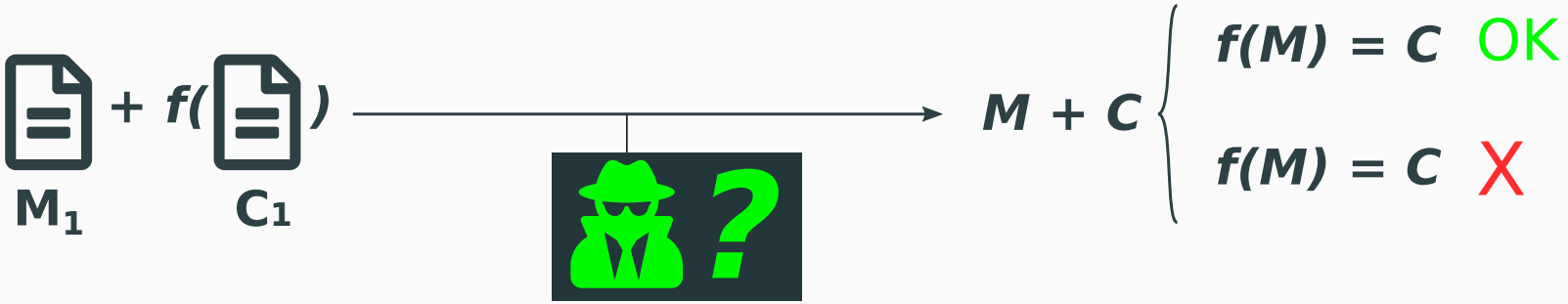
- Princípio de Kerckhoff: **um sistema criptográfico deve ser seguros mesmo se tudo é conhecido sobre ele, exceto a chave**.
- A criptografia é a principal ferramenta para providenciar proteção para informação. Ela fornece as seguintes proteções:

- Confidencialidade: encriptação (criptografia simétrica $\times$ criptografia assimétrica)

Integridade e Autenticação: HMAC e Assinaturas Digitais

HMAC - Código de Autenticação de Mensagem (MAC) Simplista

- Lembrando: mensagens encriptadas podem ser modificadas de forma maliciosa, sem a necessidade do invasor decifrá-las. Por isso necessitamos garantir integridade além da confidencialidade.
- Para tal, o código de Autenticação de Mensagem (MAC) pode ser transmitido junto com a imagem para determinar se a mensagem foi alterada.
- Sendo assim, para garantir integridade, podemos propor um MAC simplista:
  1. O remetente calcula um código $C_1 = f(M_1)$ para uma dada mensagem M_1;
  2. o remetente envia M_1 junto com C_1 para o destinatário;
  3. o destinatário recebe M e C, mas não sabe se foram modificados; e
  4. o destinatário calcula $f(M)$ e compara com C para determinar se a mensagem não foi alterada.



- Onde f pode ser uma função hash. *Qual o problema com esta estratégia?*

HMAC - Introdução

- Lição: um verdadeiro MAC requer uma chave. Assim, apenas pessoas na posse da chave podem calcular o MAC (**integridade**).
- Ademais, como apenas a pessoa com a chave pode gerar o MAC válido, temos também **autenticação**.
- Uma das soluções é o HMAC (Hash-based Message Authentication Code, RFC 2104), que é basicamente um hash que foi "*keyed*".
- Quando um algoritmo é "*keyed*", sua saída é dependente da entrada + chave. Um exemplo ingênuo: concatenar a chave (simétrica) com a mensagem:

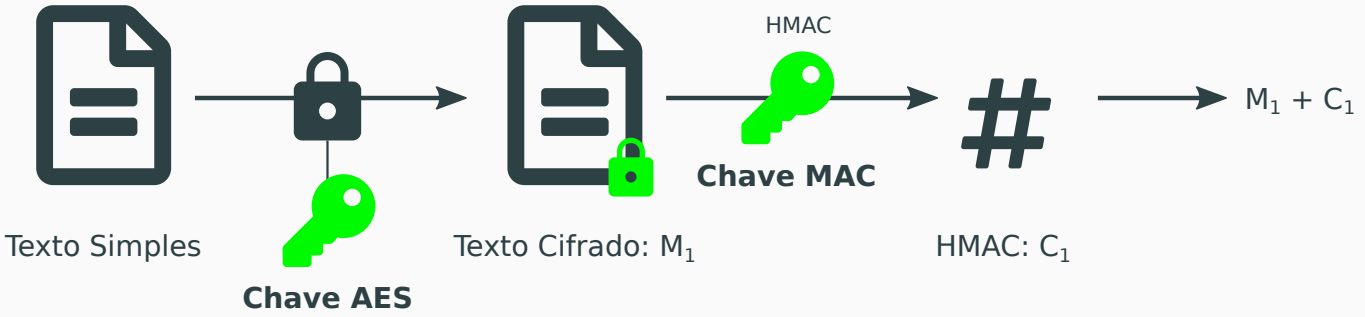
$$C = H(\text{concat}(K_{A,B}, \text{text}))$$

- Assim, no destinatário, é necessário ter K para recalcular o HMAC.
- No entanto, por segurança, o HMAC é definido no RFC 2104 como:

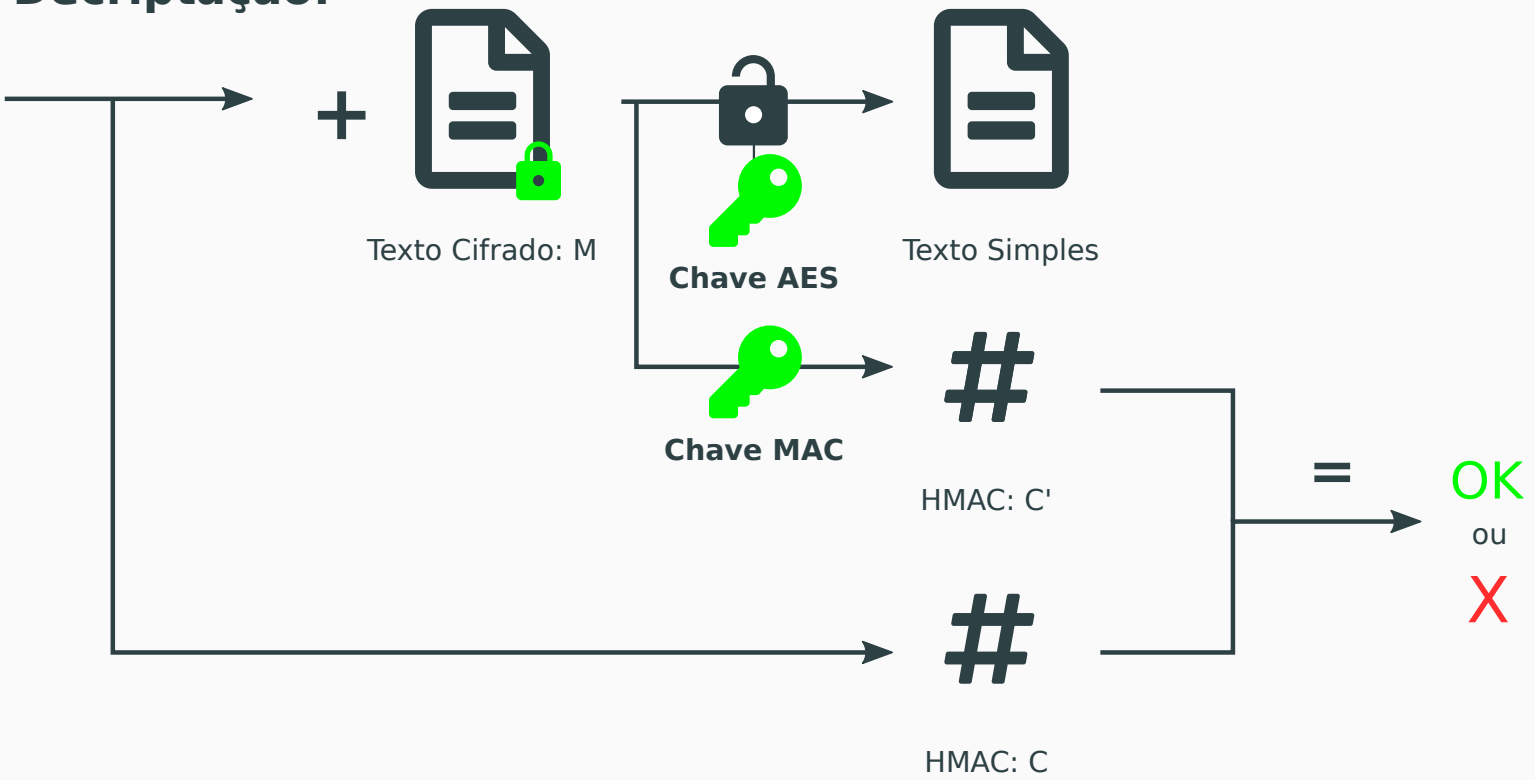
$$C = H(K_{A,B} \oplus \text{opad}, H(K_{A,B} \oplus \text{ipad}, \text{text}))$$

HMAC - Visão Geral (com *Encrypt-Then-MAC*)

Encriptação:



Decriptação:



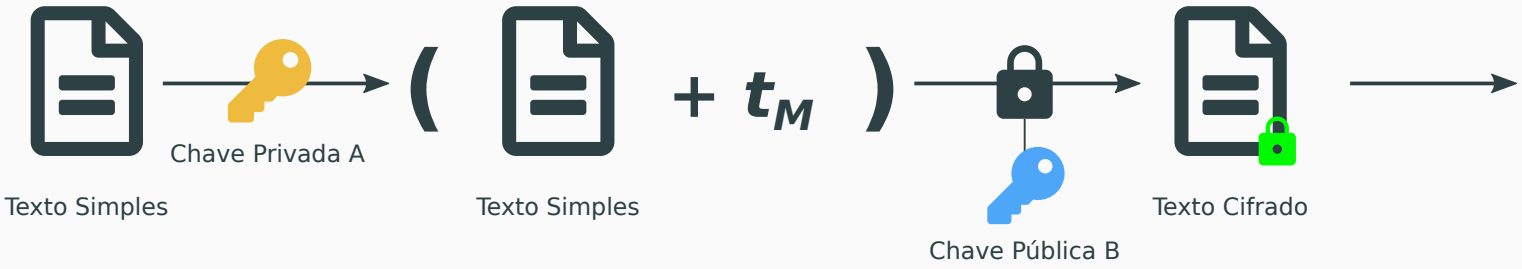
Assinatura Digital - Introdução

- Problema: MAC necessita de chaves compartilhadas.
- Solução: Assinaturas digitais se baseiam em criptografia assimétrica.
- Lembrando: no RSA, a direção de encriptação pode ser invertida; i.e., tanto a chave pública como a privada podem ser utilizadas para encriptação.
- Assim, a parte que possui a chave privada pode encriptar algo que pode somente ser decodificado com a chave pública pelo destinatário. Isso é prova que o remetente possui a chave privada (autenticação).
- Essencialmente, assinatura digital RSA (*tag*) é o hash encriptado com a chave privada, ou seja:

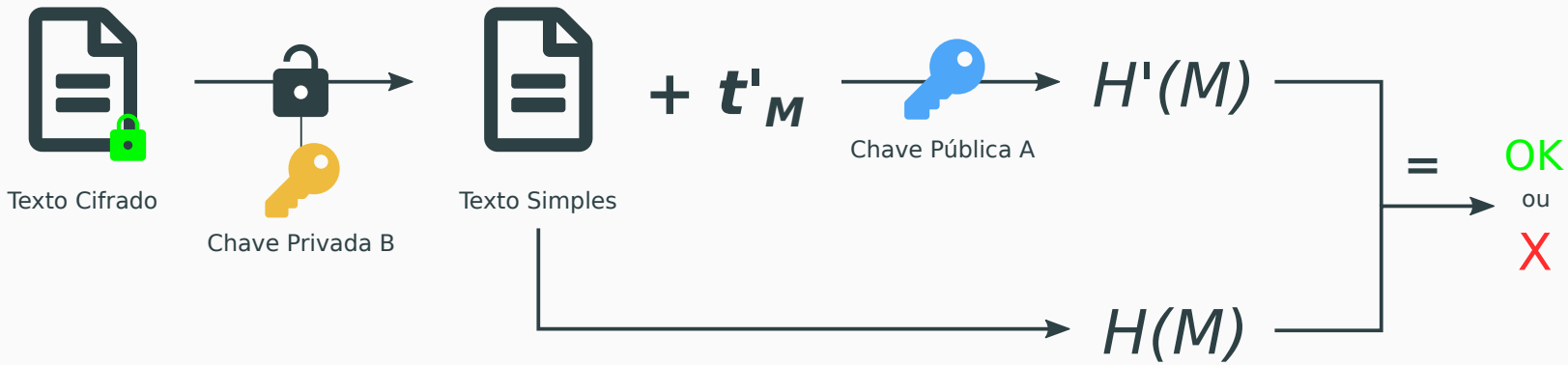
$$t_M = \{H(M)\}_{K^{-1}}$$

Assinatura Digital - Visão Geral (com *Sign-Then-Encrypt*)

Encriptação:



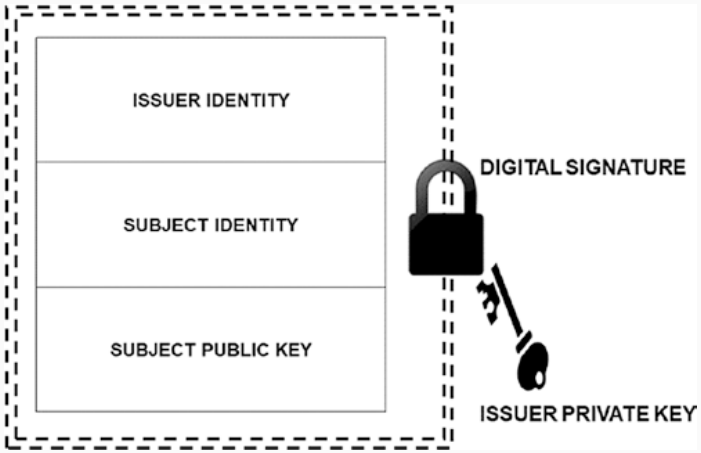
Decriptação:

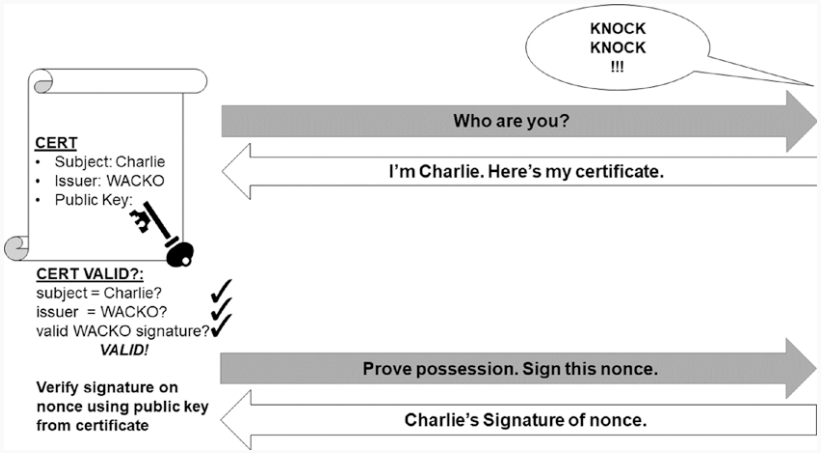


Certificados: Provando a Propriedade de Chaves Públicas

Certificados - Introdução

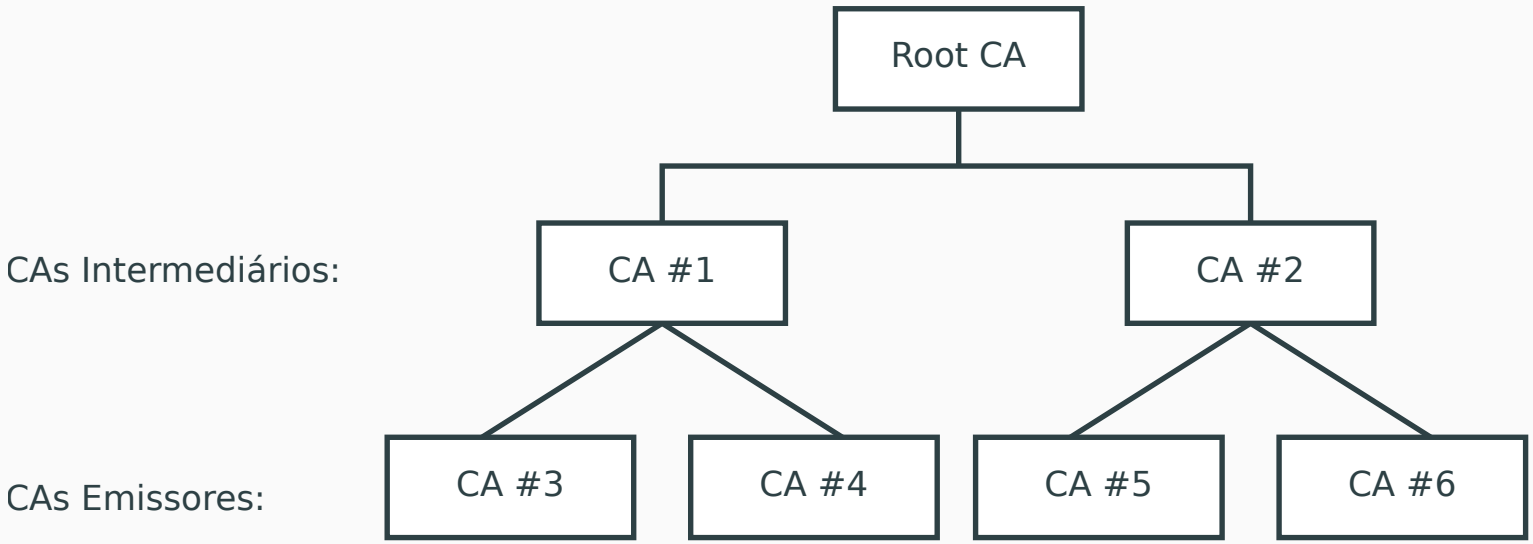
- Até agora assumimos que todas as partes interessadas possuíam as chaves públicas das demais partes. Isso pode não ser viável, no entanto.
- No início da PKI (Public Key Infrastructure) este problema era resolvido escolhendo um único “registro” confiável; este era responsável por transmitir todos os mapeamentos identidade-para-chave pública. Com o tempo, começou a acreditar que este modelo não era escalável, então surgiram os certificados.
- Certificados de chave pública são basicamente dados: geralmente inclui a chave pública, metadados do usuário e uma assinatura de um ente confiável.





Certificados - *Certificate Authorities* (CAs)

- Com o advento dos certificados, é possível existir múltiplos emissores de certificado. Estes são conhecidos como *Certificate Authority* (CA).;
- Os CAs também possuem certificados, mas quem assina eles?
  - No caso de CAs “intermediários”: são assinados por um CA de nível superior.
  - No caso de CA raiz: ele próprio assina (*self signed*).
- Como qualquer pode auto assinar um certificado, é imperativo que o(s) CA(s) raiz(es) devem ser confiáveis! Pois todos os CAs abaixo dele serão certificados por ele.

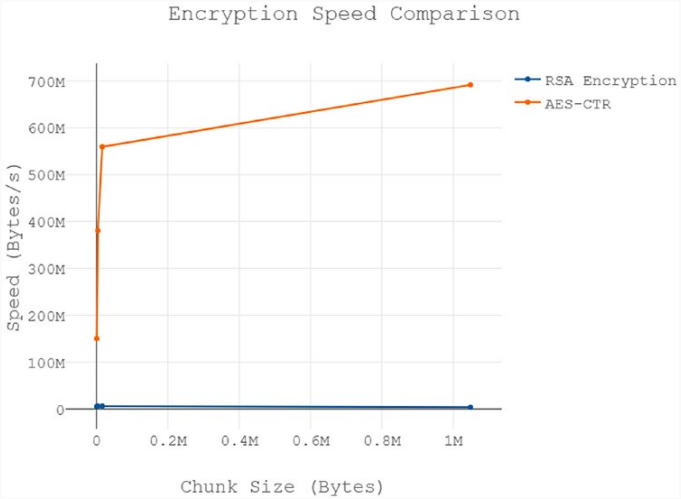


- O calcanhar de Aquiles dos certificados/chaves públicas é: como desabilitar os certificados uma vez que a chave privada correspondente é comprometida?
- Antes de responder, devemos notar que os CAs foram propostos para permitir verificação offline; em contraposição com o modelo de registros online.
- Sendo assim, a reposta é que não é possível fazer a revogação de um certificado em tempo real com este sistema.
- Duas alternativas são: *Online Certificate Status Protocol* (OCSP), que checa o status na hora, e os *Certificate Revocation Lists* (CRLs), que são publicados de tempo em tempo.
- Esta é mais uma razão para ter bastante zelo na proteção de chaves privadas. Outro cuidado pode ser manter a expiração de certificados relativamente curta.

Combinando Algoritmos Simétricos e Assimétricos: Troca de Chaves

Combinando Algoritmos Simétricos e Assimétricos - Introdução

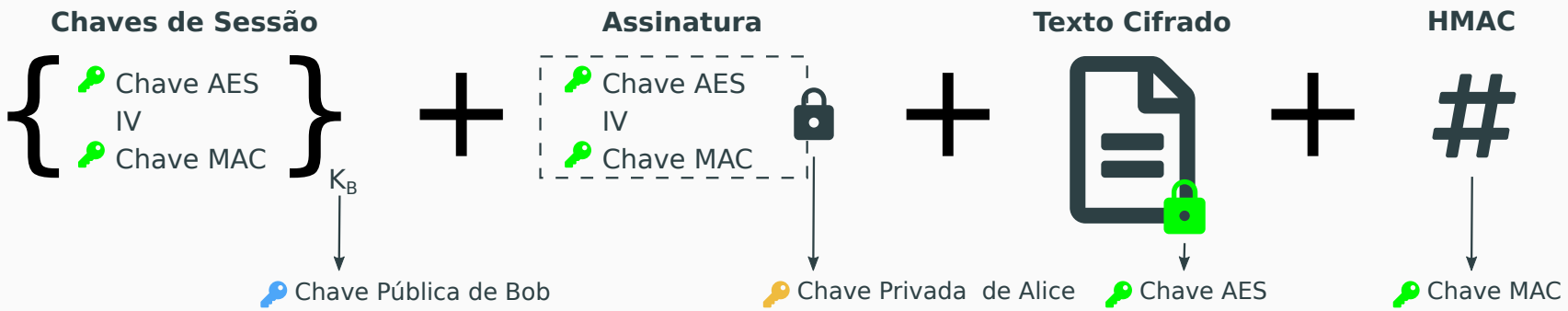
- Apesar do RSA ser capaz de encriptação, não é eficiente para realizar a comunicação completa. Assim, pode utilizada para estabelecer uma sessão, enquanto que a comunicação nesta sessão é feita por criptografia simétrica (+ eficiente).
- A chave da sessão pode ser descartada ao fim da comunicação. Este tipo de arranjo é bastante eficiente pois se aproveita das seguintes características:
  - **Criptografia simétrica**: mais eficiente, fácil para gerar as chaves, não existe chave privada, maior risco se o uso da chave for prolongado.
  - **Criptografia assimétrica**: boa para identificação de longo prazo, garante prova de identidades via certificados.



Combinando Algoritmos Simétricos e Assimétricos - Troca de Chaves com RSA

- Assumindo que as chaves públicas e certificados já estão em posse das pessoas envolvidas: Alice e Bob.
- Algoritmo simplificado: uma transmissão de Alice para Bob pode ser um stream de bytes concatenadas contendo:

Dado	Chave	Encriptado?
Chave AES, IV e chave MAC	Chave pública de Bob	Sim
Assinatura de Alice da chave AES , IV, chave MAC	Chave privada de Alice	Não
Mensagem	Chave AES	Sim
HMAC	Chave HMAC	Não



Referências

Referências - Aula 10

- Capítulos 4, 5 e 6 do livro “Practical Cryptography in Python: Learning Correct Cryptography by Example” deSeth James Nielson e Christopher K. Monson.

The End!