Sumário

- Autenticação.
- Canais seguros.
- Aplicações de segurança.

Funções usadas para autenticação

- Cifra: *o texto cifrado serve como autenticador da mensagem.*
- Soma de controlo ("checksum") criptográfica: *uma função pública e uma chave secreta que criam um valor de comprimento fixo que autentica a mensagem.*
- Função de hash: *uma função pública que cria um valor de comprimento fixo que autentica a mensagem.*

Autenticação com cifra

- Ao executar a operação de decifrar com sucesso, autentica-se a mensagem.
- Problema:

  Como verificar se operação de decifrar teve sucesso?

  É necessário que a mensagem tenha alguma estrutura reconhecível.
- Se for usada chave simétrica, não impede a falsificação nem a repudiação.

Soma de controlo criptográfica

- Gerar um valor de tamanho fixo que depende do texto original, de modo que cada texto tenha uma soma de controlo diferente, com recurso a uma chave secreta.

  Diferente de CRC, pois podem haver diferentes mensagens com o mesmo CRC.
- Tornar muito difícil arranjar outro texto com a mesma soma de controlo.

  Mesmo (sobretudo) se o texto for parecido.
- Não é necessário ser reversível.

  Basta realizar a mesma operação e comparar os resultados.

Vantagens do uso da soma de controlo

- Sempre que a confidencialidade não é importante mas a autenticidade é.

  Por exemplo, verificar se um determinado executável foi produzido por um compilador de confiança.

- Permite separar os aspectos de autenticação dos aspectos de confidencialidade.

Funções de hash

- Permitem obter uma soma de controlo que não depende de uma chave secreta.
- Pode ser usado para acrescentar à mensagem a redundância necessária para a reconhecer como válida.

  A mensagem mais a soma de controlo são cifradas e decifradas em conjunto.

- Pode ser usado para garantir a integridade do texto, cifrando apenas a soma de controlo.
- Mais rápido que a cifra.

Algumas funções de hash conhecidas

- MD5 (Message Digest).

  Desenvolvido por Ron Rivest a partir de uma versão anterior (MD4) pelo mesmo autor.

- Secure Hash Algorithm.

  Desenvolvido pelo National Institute of Standards and Technology e também baseado no MD4.

Canais seguros

- Troca de informação entre dois participantes de modo a que:

  O destinatário saiba quem enviou.

  Só o destinatário consiga ler.

  O destinatário recebe o que foi realmente enviado.

Canais seguros

- Com criptografia assimétrica:

  Dois pares de chaves, uma pública e outra privada.

- Com criptografia simétrica:

  Os dois participantes devem conhecer préviamente a chave.

Canais seguros com criptografia simétrica

- Os dois participantes usam sempre a mesma chave conhecida à partida.
- Vários problemas:

  Ou se mantêm muitas chaves (díficil memorizar).

  Ou se usa a mesma com todos (mais fácil ser descoberta).

  Quanto mais tempo se usa uma chave mais fácil se torna de descobrir.

Canais seguros com criptografia simétrica

- Usar uma chave diferente, negociada para cada sessão.
- Como negociar uma chave secreta?

  Usando criptografia assimétrica.

  Usando um terceiro elemento, de confiança, que conhece as chaves secretas de cada participante.

Kerberos (versão simplificada)

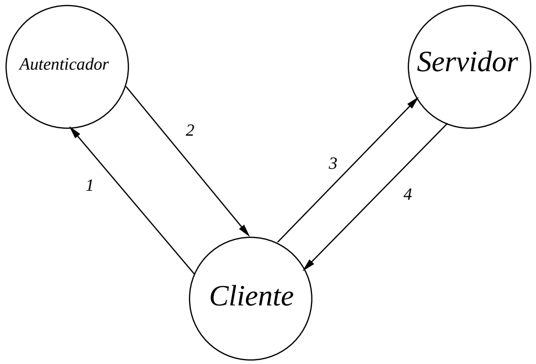
- Servidor de autenticação que permite estabelecer canais seguros baseado em criptografia simétrica.
- Distingue três entidades.

  Utilizador (tem chave secreta pessoal K_U).

  Servidor (tem chave secreta K_S).

  Servidor de autenticação (conhece ambas as chaves secretas).

Kerberos (ilustração)



Kerberos: funcionamento

- 1. User → Kerberos: U, server.
- 2. Kerberos → User:

  $T_{(C,\mathsf{server})} = E_{K_{\mathsf{server}}}[C, server, K_{(C,\mathsf{server})}, time, life]$.

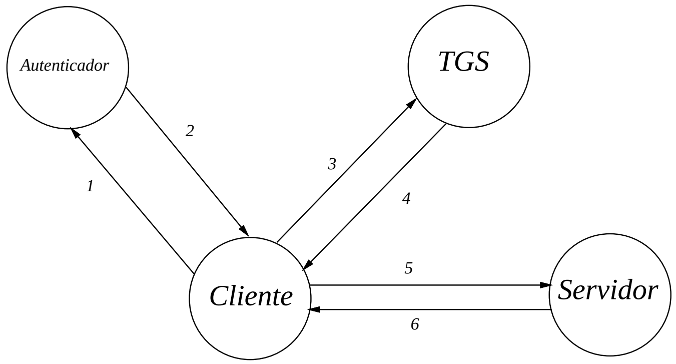
  $E_{K_U}[T_{(C,\mathsf{server})}, K_{(C,\mathsf{server})}, \mathsf{server}, time, life]$.
- 3. Client → Server: $E_{K_{(C,\mathsf{server})}}[C, time] + T_{(C,\mathsf{server})}$.
- 4. Server → Client: $E_{K_{(C,\mathsf{server})}}[time + 1]$.

Kerberos (versão completa)

- Existe um segundo servidor, designado por "ticket granting service" (TGS) que sabe as chaves secretas dos servidores mas não dos utilizadores.
- O servidor de autenticação só sabe as chaves secretas dos utilizadores e do TGS

  É necessário mais uma interacção.

Kerberos completo (ilustração)



Canais com cripto mista

- Usa-se a criptografia assimétrica para negociar uma chave secreta (simétrica) para a sessão:

  $A \rightarrow B : E_{K_{pb}}[I_a, A].$

  $B \rightarrow A : E_{K_{pa}}[I_a, I_b].$

  $B, A : K_{ab} = hash(I_a, I_b).$

Geração de números aleatórios

- As chaves simétricas temporárias devem ser geradas aleatóriamente.

  De modo a não ser possivel adivinhar uma chave com base na anteriormente gerada.

- É necessário utilizar geradores de números aleatórios dedicados.

  Exemplo: ANSI X9.17 Pseudo Number Generator baseado no algoritmo DES.

Aplicações de segurança: PGP

- PGP: Pretty Good Privacy.

  Pacote de software que permite executar diversas operações de segurança.

  Cifra de mensagens e ficheiros, canais seguros, autenticação, etc.

- Software livre mas com uma versão suportada comercialmente.
- Exemplo de como combinar diversas funçoes e técnicas de segurança.

  Criptografia simétrica, criptografia assimétrica, função de hash, codificação e compressão.

PGP: algoritmos utilizados

- RSA: *para suportar criptografia assimétrica.*

  Três comprimentos de chaves: casual (384 dígitos binários), comercial (512 dígitos binários) e militar (1024 dígitos binários).

- IDEA: *para suportar criptografia simétrica (chaves de 128 dígitos binários).*
- MD5: *para autenticar mensagens.*
- ZIP: *para comprimir mensagens.*
- Radix 64 conversion: *para compatibilidade com o e-mail.*

PGP: cifra de mensagens

- Algoritmos utilizados: IDEA e RSA.
- A mensagem é cifrada usando o IDEA com uma chave simétrica de utilização exclusiva.
- A chave simétrica utilizada no passo anterior é cifrada com a chave pública do destinatário usando o RSA e enviada conjuntamente com o texto cifrado.

PGP: assinatura

- Algoritmos utilizados: RSA e MD5.
- Uma assinatura é gerada usando o MD5.
- A assinatura é cifrada com a chave secreta do remetente usando o RSA e enviada conjuntamente com o texto.

PGP: compressão e conversão

- Compressão: uma mensagem pode ser comprimida para transmissão ou armazenamento usando o ZIP.

  Poupa espaço e largura de banda e diminui a redundância do texto.
- Conversão: uma mensagem pode convertida de modo a serem usados apenas caracteres de 7 dígitos para compatibilidade com o e-mail.
- Segmentação: também é possível segmentar (e reunir) uma mensagem se o tamanho exceder limites impostos pelo sistema.

PGP: gestão de chaves

- Permite gerar pares de chaves assimétricas.
- Permite armazenar as chaves secretas pessoais de modo seguro.
- Permite armazenar chaves públicas.
- Gera automáticamente chaves simétricas (usando uma função aleatória).

PGP: chaves pessoais

- Cada utilizador pode ter vários pares de chaves assimétricas.

  As chaves do mesmo utilizador distinguem-se por um identificador (simplesmente os últimos 64 dígitos binários da chave pública).

- Para cada chave assimétrica armazena-se a seguinte informação (Private Key Ring):

  Data da criação, identificador de chave, chave pública, chave secreta (cifrada), identificador do utilizador.

- A chave secreta do utilizador é cifrada (IDEA) com recurso a uma simétrica obtida através de uma palavra chave que o utilizador deve memorizar.

PGP: chaves públicas

- Cada utilizador pode armazenar as chaves públicas de outros utilizadores.
- Associada a cada chave pública armazena-se a seguinte informação (Public Key Ring):

  Data de criação, identificador da chave, chave pública, e identificador do utilizador.

  Mais informação de confiança.

PGP: utilização

- Envio de mensagem assinada e cifrada.

  pgp -es ficheiro id_destinatario .

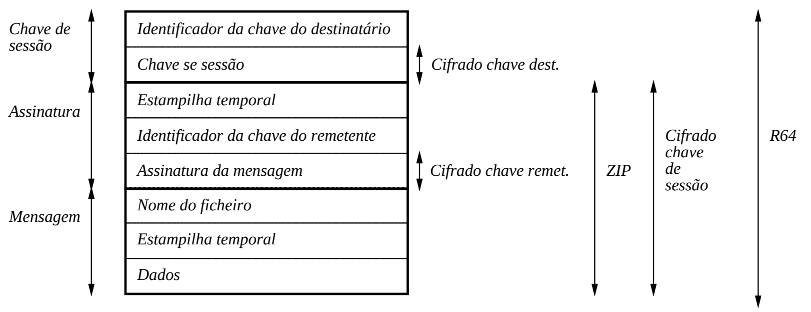
- Assinar a mensagem:

  O PGP obtém a chave secreta (cifrada) do remetende a partir do "private key ring"; pede ao utilizador a palavra chave para a decifrar; e constroi a assinatura da mensagem.

- Cifrar a mensagem.

  O PGP gera a chave de sessão e cifra a mensagem; obtém a chave pública do destinatário a partir do "public key ring" e cifra a chave de sessão.

PGP: formato das mensagem



PGP: gestão das chaves públicas

- Cada utilizador especifica a confiança que tem nos restantes utilizadores do ponto de vista da distribuição de chaves públicas.

  Máxima (as suas), de confiança total, confiança parcial, sem confiança ou desconhecido.

- As chaves públicas podem ser inseridas directamente pelo utilizador ou através da recepção de um certificado vindo de outro utilizador.

- O grau de confiança que se tem na chave depende do número de certificados já obtidos e do grau de confiança que se tem nos seus emissores.

Resumo

- Autenticação.

  Cifra. Soma de controlo criptográfica. Função de Hash.

- Canais seguros.

  Distribuição de chaves segura. Kerberos.

- Aplicações de segurança.

  PGP (Pretty Good Privacy).