## Sumário

- Facetas da segurança.
- Cifra.
- Canais seguros.







## Facetas da segurança

- Confidencialidade: *impedir um intruso acede a infomação confidencial.*
- Autenticidade: *impedir um intruso/utilizador de personifica um outro utilizador autorizado.*
- Integridade: *impedir um intruso de alterar (ou destruir) informação.*
- Disponibilidade: *garantir o acesso à informação por parte dos utilizadores legítimos.*







## Confidencialidade

- Tem por objectivo proteger a informação privada.

  *Embora por vezes seja socialmente justificável violar a confidencialidade (por exemplo, para combater o crime organizado).*

- Pode ser obtida:

  *Impedindo o acesso à informação.*

  *Cifrando a informação.*







## Autenticidade

- A informação foi produzida por quem?

  *Ter a certeza de quem foi o produtor da informação.*

- Impedir a repudiação:

  *Quem produziu a informação (ficheiro, mensagem) não poderá negar a sua autoria.*

## Integridade

- Evitar e ou detectar o forjamento ou alteração de informação (documentos ou mensagens).

  *Exemplo: garantir que uma mensagem não pode ser alterada no seu percurso até ao destinatário.*







## Disponibilidade

- Evitar acções que inpeçam o acesso à informação.

  *Bloqueamento, destruição de serviços sensíveis.*

  *Bloqueamento da comunicação.*

- Necessário redundância.

  *Nas redes e nos servidores.*







## Técnicas de criptografia para texto

- Cifra de césar.

  *Transforma uma letra na nª letra que a segue no campeonato.*

- Transposição simples.

  *Arranjar uma chave de 26 letras obtida por permutação do alfabeto e transformar o texto de acordo com a chave.*







## Para informação binária

- Função de cifrar (E) e de decifrar (D).
- Uma chave para cifrar e outra (possivelmente a mesma) para decifrar.

  *E(chave1, dados)= dados cifrados.*

  *D(chave2, dados cifrados) = dados.*

- Deve ser feito de modo a:

  *Não ser fácil obter os dados a partir dos dados cifrados sem saber a chave2. Conhecendo os dados e os dados cifrados não deve ser fácil descobrir a chave1.*

## Criptografia simétrica

- A mesma chave é usada para cifrar e decifrar.
- Existem algoritmos para cifar com grande eficiencia.
- Fica comprometida se a chave e descoberta.
- É necessário que os dois participantes conheçam a chave.







## Criptografia assimétrica

- As chaves são diferentes.

  *Tipicamente uma chave é conhecida e a outra é secreta.*

  *Chave pública e chave privada.*
- É necessário: *que não seja fácil encontrar uma chave dada a outra .*
- É útil: *que se possa cifrar com qualquer uma das chaves (e decifrar com a outra).*
- Desvantagem: *é muito mais lento que a chave simétrica.*







## Desempenho (rapidez)

| Operação | H/W, bit/s | S/W, bit/s/MIP |
|---|---|---|
| assimétrica cifrar (RSA) | 220 K | .5 K |
| assimétrica decifrar (RSA) | - | 32K |
| simétrica (DES) | 1.2 G | 400K |
| digest (MD4) | - | 1300K |







## Confidencialidade com criptografia assimétrica

- O remetente envia uma mensagem cifrada com a chave pública do destinatário.
- O destinatário é o único que pode decifrar a mensagem com a sua chave secreta.

  *Sejam $L_s$ e $L_p$ as chaves secreta e pública do receptor.*

  *D($L_s$, E($L_p$, dados)).*

## Autenticidade com criptografia assimétrica

- O remetente envia uma assinatura cifrada com a sua chave secreta.
- Só a sua chave pública é que consegue reconstruir a sua assinatura.
- O destinatário usa a chave pública do remetente para verificar a assinatura.

  *Sejam $K_s$ e $K_p$ as chaves secreta e pública do emissor.*

  *D($K_p$, E($K_s$, assinatura)).*







## Confidencialidade e autencidade

- O remetente assina a mensagem, cifra com a sua chave secreta e cifra o resultado com a chave pública do destinatário.
- O destinatário decifra tudo com a sua chave secreta e autentica-a com a chave pública do remetente.

  *D($K_p$, D($L_s$, E($L_p$,E($K_s$, dados)))).*







## Message digest (MD4)

- Arranjar um "resumo" (digest) que seja função do texto de modo que cada texto tenha um "resumo" diferente.

  *Diferente de CRC, pois podem haver diferentes mensagens com o mesmo CRC.*

- Tornar muito difícil arranjar outro texto com o mesmo digest.

  *Mesmo (sobretudo) se o texto for parecido.*

- Usado para garantir a integridade de um texto (bastando cifrar o digest).







## 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.

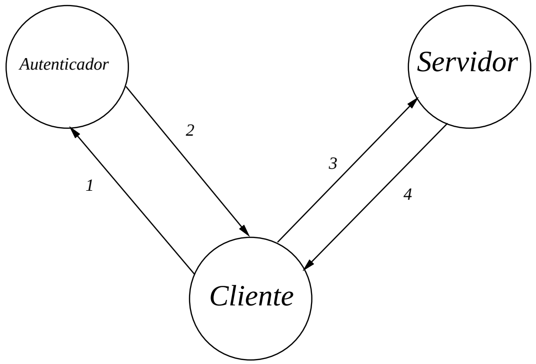
- Distingue três entidades.

  *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: $U, \textsf{server}$.
- 2. Kerberos → User:

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

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







## 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

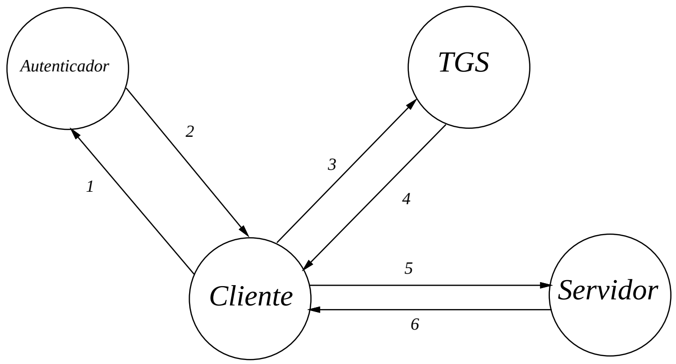
  *É necessário mais uma interacção.*







## Kerberos completo (ilustração)



## Canais com cripto mista

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

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

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

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







## Resumo

- Facetas da segurança.

  *Confidencialidade, autenticidade, integridade, disponibilidade.*

- Cifra.

  *Cifra clássica, cifra com chave simétrica, cifra com chaves assimétricas.*

- Canais seguros.

  *Distribuição de chaves segura. Kerberos.*