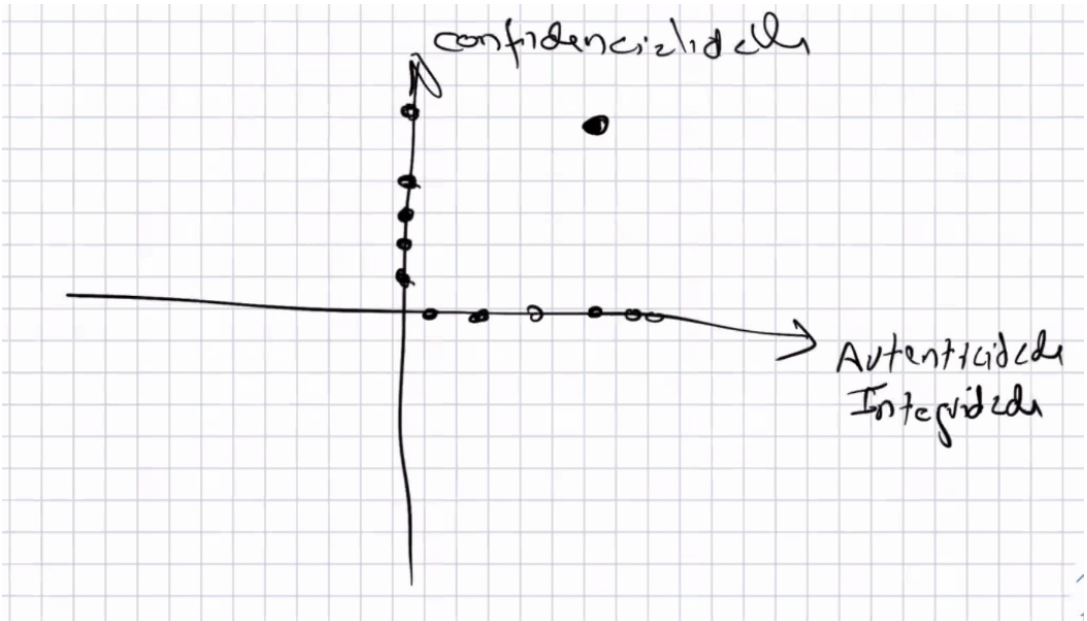
Key Exchange Protocols

Teórica #11 de Criptografia Aplicada

Key Management

A criptografia está dividida sobre dois eixos:



Em criptografia simétrica, as **cifras simétricas** focam-se na confidencialidade (e.g., AES-CTR), ao que os MACs focam-se na autenticidade (e.g., HMAC). As cifras autenticadas (AEAD) focam-se tanto na confidencialidade como autenticidade.

Criptografia simétrica funciona se a chave secreta seja partilhada, o que requer autenticidade e confidencialidade.

—

Em criptografia de chave pública, temos **assinaturas digitais** e **cifras assimétricas**. Existem duas chaves, a *privada* que é usada para **decifrar** e **assinar** mensagens, e a *pública* que é usada para **cifrar** e **verificar** mensagens.

Uma assinatura no contexto de cifras é uma prova que uma entidade consegue utilizar a chave secreta.

Por norma as cifras assimétricas são híbridas, pois tanto assinam como cifram uma mensagem.

Quantas chaves são necessárias e qual o seu propósito?

Existem dois tipos de chaves criptográficas, sejam estas públicas ou privadas:

- De sessão (temporárias), ou *data encapsulation keys*, que tem o propósito de proteger a informação. Estas são sempre simétricas, de curta duração e vivem temporariamente na memória do computador, sendo que se expostas, apenas comprometem um pequeno conjunto de informação;
- De longo termo (permanentes), ou *wrapping keys*, que tem o propósito de proteger as chaves de sessão. Estas podem ser assimétricas, simétricas, ou até passwords (passphrases), sendo que existem poucas destas chaves, requerem um armazenamento de longa duração (e.g., HSM) e requerem algum tipo de *trusted party* para serem estabelecidas.

Como é que as chaves de sessão são estabelecidas?

As chaves de sessão podem ser estabelecidas de três maneiras:

- Diretamente derivadas de outras chaves de sessão, que acontece tipicamente em protocolos complexos como o TLS, e também quando derivadas de uma chave mestre;
- Distribuídas através de um *trusted agent*, onde as chaves são pré-acordadas com um trusted server (e.g., Kerberos, Radius);
- Através de um acordo de chave autenticada, sendo chaves permanentes pré-acordadas entre duas entidades, realizando um handshake para construir a chave de sessão.

Como são geridas as chaves em criptografia de chave pública?

A utilização de chaves públicas resolve o problema de garantir a confidencialidade e autenticidade em sistemas abertos (e.g., Internet, onde não há uma entidade central que os registe).

O sistemas abertos tem um problema relativamente à autenticidade de quem comunica neste. Não é possível reconhecer com certeza que uma chave pública A seja par da chave privada B, da entidade C. Para resolver este problema é imposto uma **infra-estrutura de chave pública** (Public Key Infrastructure) que gerar as chaves públicas e o seu par privado, distribuindo as mesmas a quem deseja interagir com o sistema.

Segurança de Protocolos de Acordo de Chaves

Para estabelecer segurança em protocolos de acordo de chaves, deve ser impostas as seguintes propriedades:

- **Data Leak**: Apenas algumas chaves de sessão podem ser reveladas, mas nunca chaves permanentes;
- **Corruption**: Apenas as chaves de sessão atuais podem ser corrompidas, de forma a que as chaves anteriores se mantenham seguras e não relevando as mensagens anteriores (forward secrecy);
- **Channel-Only**: As chaves podem apenas ser transportadas no canal de comunicação em que pertencem;
- **Key Control**: Apenas uma entidade não pode determinar uma chave, sendo necessárias pelo menos duas para se chegar a um acordo;

- **Key Compromise Impersonation**: A falsificação de uma chave permanente de uma entidade A, não deve permitir falsificar a entidade B.

Eficiência em Protocolos de Acordo de Chaves

Para maximizar a eficiência em protocolos de acordos de chaves, deve-se ter em conta as seguintes propriedades:

- **Número de round-trips**, ou seja o número de viagens entre A e B até que seja satisfeito o acordo da chave (impacto da latência da rede);
- **Banda-larga**, ou seja o número de bits enviados e recebidos no acordo;
- **Pre-computação**, ou seja, realizar algum tipo de computação no lado das entidades A e B de forma offline, de forma a que a computação online requeira menos passos e seja mais rápida.

Segurança de Diffie-Hellman Autenticado

- Autenticação mútua (devido ao trace das assinaturas e a confirmação da chave adicional);
- Nenhuma entidade controla a chave final;
- Conhecer as chaves de sessão atuais não dá a conhecer as sessões anteriores;
- Corromper as chaves assinadas atuais, não dá a conhecer as chaves passadas.

Erros comuns cometidos em Diffie-Hellman

- Própria implementação do protocolo;
- Não utilizar a correta KDF para o valor de `g^ab`;
- Utilização de DH anónimo sobre canais não autenticados (sujeito a MiTM);
- Aceitação de chaves públicas não autenticadas.