Criptografia

Índice

Acordo de Chaves 3
Algoritmos eficientes para operações modulares 4
Alguns problemas (tidos por) intratáveis 4
Protocolo Diffie&Hellman 4
Descrição: 4
Segurança (perante adversários passivos) 5
Man-in-the-middle 5
Criptografia de Chave-Pública 5
Trapdoor Permutation 6
Cifra Assimétrica 6
Utilização (na prática) 6
Man-in-the-middle 7
Assinatura digital 7
Descrição 7
Utilização básica 8
Utilização (na prática) 8
Man-in-the-middle 9
Certificação das chaves 9
Protocolo Station-to-Station 9
RSA 10
Descrição 10
Segurança 10
Variantes aleatórias 11
RSA-OAEP 11
RSA-PSS 12
El-Gamal 12
Cifra 12
Correção e segurança 12
Correção 12
Segurança 12
Digital Signature Algorithm(DSA) 13
Criptografia em Curvas Elípticas (ECC) 14
Certificados Digitais 14
Enquadramento 15

Âmbito das ICP 16
EC in-house 16
EC comercial 17
EC em outsourcing 17
EC comercial com atribuições especiais 18
EC pública 18
Arquitetura de uma ICP 19
Arquitetura 19
Operações 20
Estrutura dos Certificados X509 20
Certificado de Chave Pública 20
Certificados X.509v3 21
Atributos básicos 21
Identificadores 21
Certificados X.509 (V3): Extensões 22
Cadeias de Certificação e de Confiança 23
Validação de Certificados 24
Âncoras de Confiança 24
Perfis e Políticas de Certificados 24
Perfis de Certificados (Certificate Profiles) 24
Políticas de Certificados (Certificate Policies) 25
Políticas de Certificação na validação de certificados 26
Declaração de Práticas de Certificação (Certification Practice Statements) 26
Exemplos de Perfis de Certificados 26
Certificados TLS 27
Classificação do Método de Validação (CA/browser forum) 27
Certificate Transparency 27
Revogação de Certificados 28
Certificate Revocation Lists (CRL) 28
Online Certificate Status Protocol (OCSP) 29
Aplicações 30
Protocolos de Comunicação 30
IPSec 30
TLS 33
TLS v1.3 36
Outros protocolos de Sessão 36
Correio Electrónico Seguro 37

S/Mime 37
Utilização típica 37
PGP 37
Assinatura de Documentos 38
PAdES 38
Time-Stamping 40
Tokens Criptográficos 40
Cartão de Cidadão 41

Acordo de Chaves

A utilização de criptografia (simétrica) obriga à existência de chaves partilhadas.

O pré-acordo de chaves é um procedimento custoso (requer a utilização de canais seguros...) e pouco flexível (e.g. considere-se a inclusão de mais um agente na comunidade...)

Exemplo: Admita-se que dispomos de uma cifra (simétrica) em que a operação de cifra é comutativa:

Ek1(Ek2(X))=Ek2(Ek1(X))

Para **A** comunicar **M** com **B** pode:

- **A** envia a **B EKA(M),** em que **KA** é só conhecida por **A**.

- **B** devolve a **A EKB(EKA(M))=EKA(EKB(M)),** em que KB só é conhecida por B.

- **A** decifra mensagem recebida e re-envia a **B** o resultado, i.e. EKB(M)

- **B** decifra mensagem M.

> ... ou seja, A e B comunicam de forma segura sem partilharem segredos... (a mensagem M circula sempre protegida com, pelo menos, uma operação de cifra)

Obs.: mas este esquema também exibe uma vulnerabilidade importante... (c.f. man-in-the-middle attack)

Pode-se contornar o problema da distribuição de chaves se ambas as partes acordarem num segredo comum:

- trocando mensagens sobre um canal público...
- mas sem que que seja possível derivar o segredo conhecendo apenas as mensagens trocadas.

Um esquema que acomoda estes requisitos surgiu no artigo de **Diffie-Hellman**

A segurança resulta de se acreditar que a exponenciação modular é uma função de sentido único.

Algoritmos eficientes para operações modulares

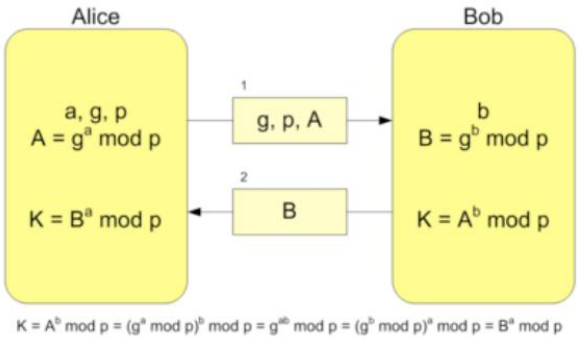
- Adição; multiplicação; resíduo (módulo) - adaptações dos algoritmos usuais;
- Exponenciação - (square and multiply);
- GCD - o famoso algoritmo de Euclides;
- Inversa multiplicativa (divisão) - generalização do algoritmo de Euclides;
- Primalidade (testar se um número é primo) - existem testes probabilísticos que nos permitem obter garantias (tão boas quanto necessárias) que um número é primo.

Alguns problemas (tidos por) intratáveis

- Factorização de um inteiro:
  - Dado um inteiro n determinar a sua factorização em números primos.
  - Ou seja, determinar números primos $p_1,...,p_i$ tal que $p_1 \times ... \times p_i = n$
- Logaritmo discreto: Dado a, b e n, determinar x tal que $a^x \bmod n = b$
- Raiz quadrada discreta:
  - Dado y e n, determinar x tal que $x^2 \bmod n = y$

Protocolo Diffie&Hellman

Parâmetros: Seja p um primo e g um gerador do grupo Zp*



Descrição:

A gera um inteiro 1<x<p e envia a B “gx mod p”

• B gera um inteiro 1<y<p e envia a A “gy mod p”

• Segredo partilhado: $g^{xy} \bmod p = (g^y)^x \bmod p = (g^x)^y \bmod p$

Segurança (perante adversários passivos)

• Se fosse possível calcular o logaritmo discreto, o intruso poderia calcular x a partir de g^x e, assim, atacar o protocolo.

• Em rigor, a segurança do protocolo exprime-se como uma assunção de segurança própria (Computational DiffieHellman problem): a partir de g^x e g^y não é possível determinar g^{xy}

Por vezes, refere-se aos valores envolvidos no algoritmo como pares de chaves:

- x, g^x: chave privada e pública de A;
- y, g^y: chave privada e pública de B.

Man-in-the-middle

Na presença de um adversário ativo, é possível este fazer-se passar por outro agente comprometendo a segurança da técnica: ataque vulgarmente designado por man-in-the-middle.

Suponhamos que A pretende acordar um segredo com B.

• A gera x, calcula g^x e envia este último valor a B;
• I intercepta a mensagem de A;
• I gera z e calcula g^z que envia para A:
• A adota o segredo $K=(g^z)^x=g^{xz}$ que presume acordado com B;
• I conhece o segredo $K=(g^x)^z=g^{xz}$ que A pensa partilhar com B.

I pode ainda executar uma sessão análoga com B e assim colocar-se "no meio" da comunicação entre A e B.

Este é um ataque a que estão sujeitas a generalidade das técnicas criptográficas assimétricas**: A utilização de técnicas criptográficas assimétricas requer uma associação fidedigna entre pares de chaves e identidades dos agentes.**

Criptografia de Chave-Pública

Conceito introduzido por *Diffie & Hellman* em 1976. Ideia base:

• Duas chaves distintas são utilizadas na operação de cifra Kc de decifragem Kd. **E(Kd,E(Kc,M)=M**

• O conhecimento de uma chave não permite retirar informação sobre a outra.

Isto leva ao conceito de função de sentido único com segredo:

- Cifra com uma das chaves deve ser uma função de sentido único - não deve ser computacionalmente viável inverter essa função.
- Mas informação adicional (outra chave) permite calcular operação inversa.
- Mais uma vez, é a Teoria de números que se tem revelado a principal fonte de problemas que se acredita satisfazerem critérios requeridos.

Assim, uma das chaves pode ser "tornada pública".

Trapdoor Permutation

Na criptografia de chave pública faz-se uso de uma especificação mais apertada nas funções de sentido único: Funções de Sentido Único Com Segredo (trapdoor permutation):

- Função injetiva que dispõe de um algoritmo eficiente para o seu cálculo;
- Cujo cálculo da inversa seja um problema intratável;
- Mas, quem dispuser de informação adicional (o segredo), pode calcular essa inversa.

Cifra Assimétrica

A utilização de chaves distintas para as operações de cifra e decifragem permite contornar o problema da pré-distribuição de chaves.
O ponto de partida é a observação só a chave para decifrar necessita ser mantida secreta.
Cada agente dispõe de um par de chaves (Kc,Kd).

Cifra:

- Chave pública: Kc; Chave privada: Kd
- Para A enviar mensagem M a B: envia E(KcB,M) (note que KcB é publicamente conhecida)
- B decifra a mensagem utilizando a sua chave privada: E(KdB,M)=M

A dispõe de garantias que só B pode extrair o conhecimento de M porque só ele dispõe do conhecimento da chave privada.

Utilização (na prática)

- Para o mesmo nível de segurança, as cifras assimétricas são várias ordens de grandeza menos eficientes do que as simétricas (e.g. 1000x).
- Por isso, são normalmente utilizadas em conjunção com estas (e não alternativamente).

Utilização típica:

Envelope digital, utilizado para garantir confidencialidade na transmissão de uma mensagem:

- A gera uma chave de sessão K (para uma cifra simétrica)
- A envia a B par com $E(Kc^B,K)$ e EK(M) - EK(-) é uma cifra simétrica
- B decifra K e utiliza essa chave para decifrar M.

Man-in-the-middle

Tal como no caso do acordo de chaves, também a cifra assimétrica é vulnerável perante um adversário activo (ataque man-in-the-middle).
Na sua essência, este ataque traduz-se por fazer uso da chave pública "errada".

Exemplo: Suponhamos que A deseja cifrar uma mensagem para B.

• Ao pedido de A relativo à chave pública de B, I responde com a sua própria chave pública Kcl .
• A envia E(Kcl ,M).
• I intercepta essa mensagem, decifra-a, e torna-a a cifrar utilizando a verdadeira chave pública de B
• B decifra mensagem.
Mais uma vez observa-se que existe necessidade de confiar na associação entre os pares de chaves e as identidades.

Assinatura digital

O principal contributo da criptografia assimétrica foi o de permitir a definição de um análogo digital do conceito de assinatura de um documento.

Em geral, podemos identificar uma assinatura digital como um "suplemento" à mensagem que nos permite verificar:

- Integridade: a mensagem não é modificada após a assinatura;
- Autenticidade: a identidade do assinante pode ser confirmada;
- Não repúdio: é possível demonstrar a identidade do assinante.

É a capacidade de a assinatura digital "autenticar documentos" que permitirá ultrapassar as limitações das técnicas assimétricas que tem vindo a ser identificadas.

Descrição

Na utilização de uma assinatura estão envolvidas duas entidades: o (S)ignatário e o (V)erificador.

Um esquema de assinaturas compreende duas operações:

- produção da assinatura: processo pelo qual o Signatário gera a assinatura $x=Sig^{S}(M)$ que anexa à mensagem - a mensagem assinada consiste assim num par (M,x);
- verificação da assinatura: processo em que o Verificador confirma que a origem da mensagem M é S, i.e. $Ver^{S}(M,x)=true$

Das propriedades requeridas pela assinatura resulta que, se o (S)ignatário produzir uma assinatura $x=Sig^{S}(M)$, o (V)erificador com o par (M,x):

- Pode verificar que a origem de M é S, i.e. VerS(M,x)=true
- Não pode produzir M'≠M tal que VerS(M',x)=true

Obs.1: na essência do conceito de assinatura digital está uma assimetria entre as capacidades do verificador e do assinante: o primeiro deve estar habilitado a verificar

as assinaturas produzidas pelo segundo sem dispor da capacidade de, ele próprio, as produzir.

Obs.2: note que os MACs garantem os dois primeiros requisitos mas falham no último (não repúdio) - nesse caso o verificador dispõe de tanta informação como o assinante.

Utilização básica

Em relação à cifra assimétrica, as operações num esquema de assinaturas:

-A produção da assinatura é restrita ao Signatário;

- A verificação pode ser pública.

Assim é concebível trocar os papeis das chaves públicas e privadas nas cifras assimétricas para codificar um esquema de assinatura:

Cada agente X dispõe de um par de chaves (KcX,KdX):

• Chave pública: Kd Chave privada: Kc

• SigA(M) = E(M, KcA)

• VerA(M, S) = E(S, KdA)==M

B (ou qualquer agente) dispõe de garantias que M foi realmente enviada por A porque só ele dispunha da chave privada.

Utilização (na prática)

- As considerações expostas anteriormente relativamente à eficiência das técnicas assimétricas, assim como outras relativas a aspetos de segurança)
- Faz com que se combine o padrão apresentado com a utilização de uma função de hash criptográfica.

Assim, na prática temos:

- A utiliza uma função de hash criptográfica para calcular H=hash(M)
- A envia a B o par constituído por M e S=E(H, KcA).
- B determina valor de hash e compara-o com resultado da decifragem de S.

Man-in-the-middle

Tal como as restantes técnicas assimétricas, também as assinaturas digitais são vulneráveis ao ataque man-in-the-middle.

Na assinatura, esse ataque traduz-se na falha de garantias de autenticação após a verificação da assinatura (a verificação é realizada com uma chave pública “errada”).

Mas é interessante observar que desta vez existe um certo grau de circularidade entre o que é o objetivo da técnica e a causa do problema:

- a assinatura digital pretende estabelecer a autenticidade de uma mensagem/ documento;

- e a falha na garantia de autenticidade da associação entre as chave públicas e identidades leva à possibilidade do ataque man-in-the-middle;
- ora, se considerar um documento que estabeleça essa associação;
- podemos utilizar uma assinatura digital para certificar esse documento (assunto que abordaremos adiante)

Certificação das chaves

Problema descrito mostra que nunca devemos utilizar cifras assimétricas sem uma confiança plena na associação entre pares de chaves e identidades dos agentes.

Obs.: Notar que o problema já está presente no esquema hipotético (com cifras simétricas) utilizado para motivar o conceito.

Evidentemente que tal garantia pode ser conseguida por uma pré-distribuição de chaves (mas então não estamos longe do problema inicial...)

Solução alternativa consiste em utilizar os próprios mecanismos disponibilizados pelas técnicas assimétricas (em particular a assinatura digital) para estabelecer a confiança entre as associações par-de-chaves/identidades:

- Todos os agentes dispõem da chave pública de um agente fidedigno - a Autoridade de certificação (CA). Essa chave pública deve ser obtida por via de um canal seguro...
- A CA garante (assinando digitalmente) a associação entre chave-pública/identidade do agente - o que designamos por certificado de chave pública. É responsabilidade da CA a correção da associação estabelecida.
- Um qualquer agente pode verificar a assinatura de um certificado (atestando assim a validade da associação pretendida).

Protocolo Station-to-Station

As técnicas apresentadas (envelope digital, assinatura digital, ...) devem então requerer um certificado sempre que necessitarem de uma chave pública. Dessa forma ficam com garantias que as chaves públicas utilizadas se encontram associadas às identidades presumidas (por via da confiança depositada na CA), e assim impossibilitam ataques do tipo man-in-the-middle.

Já no caso do protocolo de acordo de chaves Diffie-Hellman, não faz sentido certificar as chaves públicas utilizadas (já que estas são geradas para cada sessão do protocolo).

Em vez disso, faz-se uso de assinaturas digitais normais para garantir a autenticidade das mensagens trocadas durante certos passos do protocolo.

Ao protocolo resultante dá-se o nome Station-to-Station:

$A \rightarrow B$: $+x \,.\, g^x$
$B \rightarrow A$: $+y \,.\, g^y$, $E_K(Sig^B(g^x, g^y))$
$A \rightarrow B$: $E_K(Sig^A(gx, gy))$

Obs.1: Na apresentação do protocolo, " $A \rightarrow B$: $+x \,.\, M$" denota um passo em que A gera o valor x e envia a B a mensagem M; $E_K(M)$ denota o criptograma resultante de cifrar M com a chave K; e $Sig^A(M)$ a assinatura digital de M por A.

Obs.2: Estando envolvidas assinaturas digitais nos segundo e terceiro passos do protocolo, é normal incluírem-se nas mensagens trocadas também os certificados requeridos para a verificação dessas assinaturas (i.e. da chave pública de B e A respetivamente).

RSA

Algoritmo que realiza o conceito de criptografia de chave pública introduzido por Diffie & Hellman.

Baseada no problema da factorização de inteiros.

Descrição

1.Inicialização (produção do par de chaves):

- Geram-se dois números primos grandes p,q (faz-se $n=p*q$, logo $\phi(n)=(p-1)*(q-1)$), sendo $\phi(n)$, a totient de Euler
- Considera-se um valor e que seja primo relativo a $\phi(n)$
- Calcula-se d como a inversa de e no grupo multiplicativo $Z^* \phi(n)$
- Chave para cifrar: (n,e) Chave para decifrar: (n,d)

2. Utilização (como cifra):

Ambas as operações são a exponenciação modular.
Cifra do texto limpo x ($0 \leq x < n$) com chave (n,e):
$x^e \bmod n$
Decifragem do criptograma y ($0 \leq y < n$) com chave (n,d):
$y^d \bmod n$

Segurança

Derivar chave privada da chave pública:

- É possível definir um algoritmo (probabilístico) que permite calcular a factorização de n assumindo que dispomos de um oráculo para derivar a chave privada RSA da pública. Ou seja, os problemas são demonstrados equivalentes.

Extrair mensagem do criptograma:

- Se se escolher uma mensagem arbitrária (de entre todo o espaço de mensagens admissíveis), "acredita-se" que não é possível derivar essa mensagem do criptograma respetivo.

(Não) Indistinguibilidade de mensagens:

- Mas é muito simples derivar a mensagem cifrada se se souber que ela pertence a um conjunto restrito de possibilidades (e.g. um único bit).

Variantes aleatórias

As maiores críticas apontadas ao RSA resultam de ele ser determinístico (i.e. uma dada mensagem cifrada repetidas vezes resulta sempre no mesmo criptograma).

Já vimos que este facto pode comprometer completamente a segurança da técnica em determinadas utilizações.

Existem variantes aleatórias do RSA que ultrapassam esta limitação, prevendo a utilização de fatores aleatórios na produção do criptograma (ou assinatura).

É possível demonstrar (formalmente) que essas variantes cumprem requisitos de segurança mais apertados (e.g. IND-CCA, no modelo Random Oracle).

Exemplos:

- Cifra: RSA-OAEP

- Assinatura: RSA-PSS

RSA-OAEP

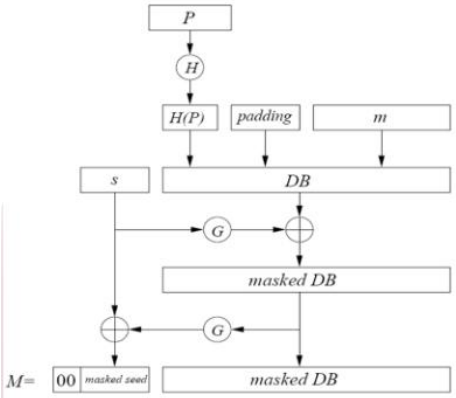


Figure 1: OAEP encoding function.

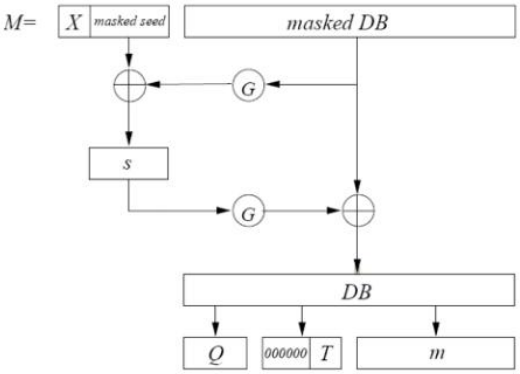
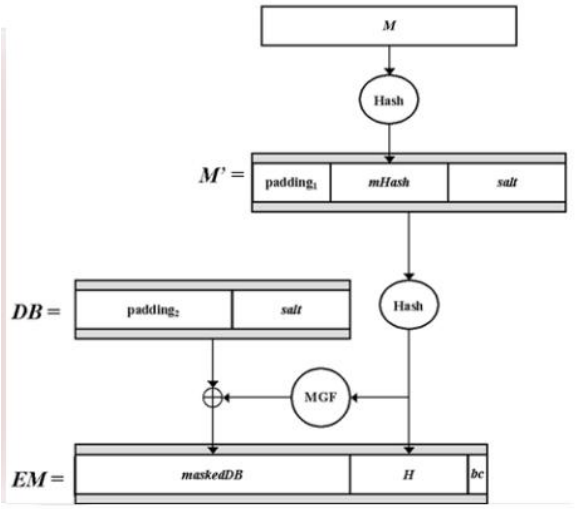


Figure 2: OAEP decoding function.

RSA-PSS



El-Gamal

Baseado no problema do logaritmo discreto.
Variantes para funcionar como cifra ou como assinatura.

Cifra

1. Inicialização

Escolher um primo p e dois inteiros, g e x, tal que g é gerador de Z*p e x<p
Calcular y = g^x mod p
[chave privada, chave pública] = [x, (y, g, p)]

2. Cifra de uma mensagem M

Escolher (aleatoriamente) um inteiro k, 0<k<p-1:
- tal que k não foi já utilizado e gcd(k, p-1)=1

Calcular a = g^kmod p e b = M*y^k mod p
criptograma: (a, b)

3.Decifragem

-Dada a chave pública (y,g,p), e o criptograma (a,b)
-M = b/a^x mod p

Correção e segurança

Correção

Escolher um primo p e dois inteiros, g e x, tal que g é gerador de Z*p e x<p

• calcular y = g^x mod p

• [chave privada, chave pública] = [x, (y, g, p)]

Segurança

- Derivar chave privada da chave pública: Corresponde precisamente ao problema do logaritmo discreto, que se crê intratável.

- Extrair mensagem do criptograma: Se se escolher uma mensagem arbitrária (de entre todo o espaço de mensagens admissíveis), “acredita-se” que não é possível derivar essa mensagem do criptograma respetivo.
- Indistinguibilidade de mensagens (IND-CPA): É possível demonstrar que, dado um criptograma c que se sabe resultante da cifra de uma de duas mensagens previamente escolhidas, não é possível saber qual a mensagem efetivamente cifrada (admitindo que o problema Diffie-Hellman é intratável).

Digital Signature Algorithm(DSA)

Desenvolvido pela NSA para a NIST (baseado no El-Gamal).

Desenhado para dispor de um procedimento de assinatura muito eficiente - e.g. muito mais eficiente do que o RSA... (em contrapartida, a verificação é muito mais pesada).

É, por isso, particularmente adaptada para ser executada em ambientes com recursos limitados (e.g. smartcards).

Desenhado para funcionar unicamente como assinatura (mas é possível desenvolver esquemas que permitem utilizar as rotinas de assinatura/verificação DSA para cifrar mensagens...)

1.Inicialização

- p é um primo de L bit (512≤ L ≤ 1024 ;L múltiplo de 64)
- q é um factor primo de (p-1) de 160 bit
- $g=h^{(p-1)/q} \bmod p$, onde $h<p-1$; $g>1$
- $y = g^x \bmod p$, em que $x<q$
- [chave privada, chave pública] = [x, (y, g, p,q)]

2. Assinatura de uma mensagem m (utiliza função de hash H)

- escolher (aleatoriamente) um inteiro k, 0<k<q

  - tal que k não foi já utilizado e gcd(k, p-1)=1

- calcular $r = (g^k \bmod p) \bmod q$ e $s = k^{-1}*(H(m)+x*r) \bmod q$
- assinatura: (r, s)

3.Verificação da assinatura

- Dada a chave pública (y,g,p,q), e a assinatura (r,s) da mensagem m
- Calcular $w=s^{-1} \bmod q$; $u1=(H(m)*w) \bmod q$; $u2=(r*w) \bmod q$
- Verificar se $(g^{u1}y^{u2} \bmod p) \bmod q = r$

Criptografia em Curvas Elípticas (ECC)

O problema do "logaritmo discreto" pode ser expresso em qualquer corpo finito (e.g. GF(p) ou GF(pn))

Em particular, podemos exprimir a exponenciação no grupo cíclico determinado por uma curva elíptica sobre o corpo considerado.

Permite representações compactas para níveis de segurança pretendidos (e.g. 163 bit para níveis de segurança análogos as 1024 bit em RSA) e realizações eficientes das operações pretendidas

Argumenta-se, por isso, ser particularmente adequado para dispositivos com recursos limitados (e.g. smartcards)

Certificados Digitais

A combinação de técnicas criptográficas simétricas e assimétricas permite ultrapassar um aspeto crítico limitativo da aplicabilidade em larga escala das primeiras – a pré-distribuição de chaves.

Mas a segurança das segundas depende da correta associação entre chaves públicas e identidades.

Para garantir essas associações, faz-se uso de uma entidade externa de confiança (EC):

- Os Certificados de Chave Pública, assinados pela EC, atestam essa associação;
- Os utilizadores aceitam como válida essa associação (por via da confiança depositada na EC).

Os utilizadores, de cada vez que necessitarem de uma chave pública, solicitam o respetivo certificado:

- Confirmam a validade do certificado verificando a assinatura nele contido
- Utilizando para isso a chave pública da EC (essa sim, terá de ser distribuída de forma segura)

Não são necessários canais seguros para transmitir os certificados

Interessa reforçar o papel crítico que as Entidade de Certificação exercem na segurança de todo o sistema.

É por isso normal impôr-se nas ECs padrões de segurança muito elevados (utilização de HSMs; segurança física; planos de segurança rigorosos; etc.)

Para operacionalizar todos estes aspetos, torna-se necessário fixar toda uma série de formatos, regras e procedimentos relativos aos mecanismos de certificação das chaves públicas que são adotados pela comunidade – **a Infraestrutura de Chave Pública (ICP) ou Public-Key Infrastructure - PKI**

Enquadramento

A utilização em larga escala de Certificados de Chave Pública pressupõe uma base sólida de fixação e sistematização de:

- Formatos e tecnologia;
- Convenções e assunções;
- Práticas e procedimentos;
- Enquadramento normativo e legal;
- Etc.

De facto, uma ICP faz a ponte entre conceitos e elementos de natureza tecnológica (chaves criptográficas) com aspetos que no limite detém um cariz "social" (identidade; individualidade; personalidade).

Por outro lado, e quando considerada em toda a sua abrangência, envolve questões que tocam aspetos tão diversos quanto:

- Tecnológicos (especificação de formatos, protocolos, etc.);
- Comerciais (direitos de propriedade, vantagens competitivas, cotas de mercado, etc.);
- Políticas (supervisão e controlo, jurisdição, soberania, etc.)

Trata-se pois de um área que, em grande medida, tem evoluído em resposta a estímulos localizados e bem sucedidos na solução de problemas concretos, que depois de suficientemente estabilizados/ amadurecidos são " adotados" para abordar questões mais abrangentes.

1. Definições básicas: âmbito e objetivos; formatos; codificações; etc.
2. Processos: interações ao nível dos intervenientes "locais" (EC e utilizadores); operações e ciclo de vida; protocolos de gestão; etc.
3. Inter-operacionalidade: mecanismos de interação e inter-relação entre diferentes utilizações e comunidades; impacto nas relações de confiança e validade; etc.
4. Regulamentação e acreditação: boas práticas, regras, organismos, etc.
5. Enquadramento legal: jurisdição, direitos e responsabilidades; valor jurídico de operações eletrónicos; etc.
6. ICPs púbicas: suporte aos organismos atividades dos estados e cidadãos

Âmbito das ICP

Interessa reforçar que, mesmo com todo o volume de normas e standards que preveem ICPs de âmbito verdadeiramente global, continua a fazer sentido considerarem-se ICPs de âmbito local com atribuições específicas.

Os requisitos e os procedimentos são nesses casos ajustados de acordo com a criticidade do sistema.

Diferentes âmbitos para uma EC(Certificado de impedimento):

- EC in-house (doméstica)
- EC comercial
- EC em outsourcing
- EC comercial com atribuições especiais
- EC pública

EC in-house

EC criada para dar resposta à necessidade de certificação locais à organização.

Requisitos e procedimentos associados à EC são normalmente muito simplificados, que resulta num ponto de falha crítico para segurança do sistema.

Por norma, é benéfico adotar os mesmos formatos/procedimentos/etc. estabelecidos pelos standards, por forma a permitir a (re)utilização de software standard.

Vantagens:

- Flexibilidade
- Cadeia de confiança não depende de terceiros
- Começa a existir suporte nos sistemas operativos

Desvantagens:

- Requer recursos humanos qualificados
- Tendência para “relaxar demasiado” aspectos da segurança

Utilizações típicas:

- Projetos piloto
- Segurança de Intranets

EC comercial

Empresas comerciais que fornecem o serviço de emitirem certificados de chave pública

Por regra, essas entidades estão acreditadas para o efeito, pelo que é credível que ofereçam níveis de credibilidade/segurança aceitáveis

Organismos “adquirem” os certificados que necessitam dessas ECs.

Vantagens:

- Simplicidade e baixo custo
- Grande oferta (escolha)
- Garantia de padrões de segurança/qualidade
- Certificados das ECs são por norma válidos na configurações standard dos sistemas operativos

Desvantagens:

- Dependência de terceiros na cadeia de confiança

Utilizações típicas:

- Sítio de comércio electrónico
- Email seguro

EC em outsourcing

Um serviço que as empresas certificadoras também oferecem é o de alojarem/gerirem ECs de clientes

Dessa forma, recursos e know-how é da empresa certificadora, sendo que o controlo sobre os certificados emitidos se mantém no cliente.

Vantagens:

- Controle sobre a EC
- Garantia de padrões de segurança/qualidade
- Certificados das ECs são por norma válidos na configurações standard dos sistemas operativos

Desvantagens:

- Custo
- Dependência de terceiros na cadeia de segurança

Utilizações típicas:

- Certificados para colaboradores/serviços de uma organização (email, TLS, etc.)

EC comercial com atribuições especiais

Certas empresas de certificação estão habilitadas a emitir certificados com atribuições especiais

Normalmente, pressupõe um processo de acreditação específico.

O suporte desses certificados é muitas vezes um token criptográfico (e.g. smartcard)

Vantagens:

- Ter acesso à atribuição especial concreta.
- ECs estão por norma obrigadas a requisitos específicos e/ou mais apertados.

Desvantagens:

- Escolha limitada (ou inexistente)

Utilizações típicas:

- Emissão de certificados qualificados (aptos para assinatura qualificada)
- Certificados para code-signing em sistemas fechados

EC pública

EC da responsabilidade de organismos públicos para suprir necessidades próprias

Ainda que, muitas vezes, sejam geridas em regime de outsourcing por entidades privadas.

Utilizações típicas:

- Emissão de certificados para documentos electrónicos (e.g. cartão cidadão, passaporte, etc.)

Organismos do Estado Português relacionados com a Certificação Digital:

Entidade responsável pela credenciação das Entidades de Certificação em Portugal é o **Gabinete Nacional de Segurança.**

Arquitetura de uma ICP

Infraestrutura de Chave Publica (ICP)

Uma **Infraestrutura de Chave Pública (ICP)** define-se como o conjunto de hardware, software, pessoas, políticas e procedimentos necessários para criar, gerir, armazenar, distribuir e revogar Certificados de Chave Pública.

Intervêm numa ICP diferentes entidades:

- **Titulares de Certificados**: possuem as respetivas chaves privadas que utilizam para decifrar mensagens ou produzir assinaturas digitais.
- **Clientes**: Utilizam a chave pública contida num certificado para cifrar mensagens e verificar assinaturas.
- **Entidades de Certificação**: Emitem/renovam/revogam certificados.
- **Autoridades de Registo**: Garantem a associação entre chaves públicas e identidades de titulares (opcionais).
- **Repositórios**: Armazenam e disponibilizam certificados e outra informação relevante (como certificados revogados, etc.).

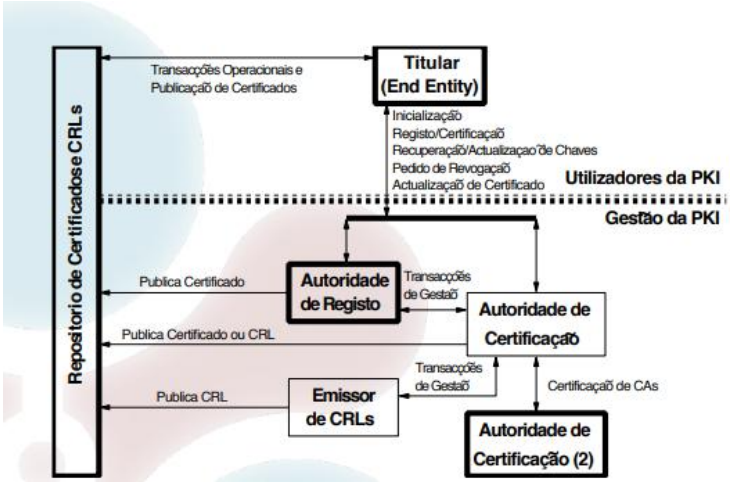
O funcionamento de uma PKI baseia-se em dois tipos de protocolos:

- **Protocolos Operacionais**-Estes protocolos são necessários para entregar certificados e CRLs aos sistemas que os utilizam. Estas

operações podem ser efetuadas de diversas formas, incluindo o LDAP, HTTP e FTP. Para todos estes meios estão especificados protocolos operacionais que definem, inclusivamente, os formatos das mensagens.

- **Protocolos de Gestão**- Estes protocolos são necessários para dar suporte às interações entre os utilizadores e as entidades de gestão da PKI, nomeadamente:
  - Inicialização
  - Registo e Certificação
  - Renovação e Atualização de pares de chaves
  - Pedido de revogação
  - Certificação de CAs.

Arquitetura



Operações

- **Inicialização:** Processo inicial que permite ao utilizador comunicar com a PKI: toma conhecimento das ECs em que confia e adquire as chaves públicas e certificados correspondentes.
- **Pedido de Certificado**: O utilizador, dispondo do seu par de chaves, solicita a uma EC a certificação da sua chave pública por meio de um pedido de certificado.
- **Registo**: Um utilizador dá-se a conhecer a uma EC (diretamente, ou através de uma RA) para que a EC lhe possa emitir um certificado; para isso fornece informação de identificação que deve ser verificada pela EC (RA).
- **Geração de Par de Chaves**: Em algumas implementações, as ECs encarregam-se de gerar os pares de chaves dos utilizadores, que enviam de forma segura junto com o certificado.
- **Certificação:** A CA recebe a chave pública do utilizador e a sua identificação e emite o respetivo certificado, segundo regras internas.
- **Publicação de Certificados e CRLs:** Esta tarefa pode ser feita diretamente pela CA, ou indiretamente por entidades como RAs. Além de colocar os certificados e CRLs em repositórios é muitas vezes necessário fazer estes documentos chegar aos utilizadores finais por outros meios (on-line ou não).
- **Revogação:** Quando um certificado é emitido o seu período útil de vida está pré-definido. No entanto, pode haver a necessidade de invalidar o certificado antes do fim desse período por diversos motivos (e.g. atributos deixam de ser aplicáveis, o comprometimento da chave privada, etc.).

- **Recuperação de um Par de Chaves:** Em algumas implementações as ECs armazenam de forma segura o par de chaves da entidade como back-up e proteção (e.g. no caso de uma empresa e os seus empregados). Nestes casos o par de chaves pode ser restaurado em caso de extravio ou danificação do seu suporte.
- **Renovação de certificados e/ou atualização de Par de Chaves:** Uma vez esgotada a validade de um certificado, existe necessidade de construir um novo certificado. Neste processo pode ou não ser mantido a chave pública do utilizador.

Estrutura dos Certificados X509

Certificado de Chave Pública

Conteúdo básico de um certificado de chave pública:

- Dados Informativos:
  - Identificação do titular do certificado (i.e. quem detém a chave privada associada a essa chave pública certificada);
  - Chave pública do titular;
  - A identificação da EC;
  - Outra informação relevante para a operacionalização do conceito (número de série; datas de validade; etc.)
- Assinatura dos dados realizada pela EC.

Os Certificados X509 são uma instância de certificados de chave pública que foram introduzidos para autenticar os nós do serviço de X500.

Os dados são representados como estruturas de dados “atributo/ valor” (dicionários).

As codificações desses dados está standardizada, garantindo a sua interoperabilidade:

- DER - formato binário definido pelo standard (notação ASN.1)
- PEM - representação da informação contido no formato DER em caracteres imprimíveis

A assinatura do certificado é efetuada pela Entidade de Certificação (EC) sobre a codificação DER dos dados nele contidos.

Certificados X.509v3

A versão de certificados utilizada atualmente (standardizada em 1996) veio colmatar as deficiências que as versões anteriores apresentavam em alguns domínios de aplicações, e que se traduziam essencialmente na necessidade de mais atributos.

Esta versão introduziu um novo campo do tipo Extensions, equipando assim os certificados com a flexibilidade necessária às novas utilizações.

Cada extensão é, ela própria, uma estrutura de dados com um identificador e um valor adequado ao tipo do atributo que representa.

Atributos básicos

- **version** – Versão do standard X509 (v3).
- **serialNumber** – Número único atribuído pela EC ao certificado.
- **subject** – Identificação do titular da chave pública contida no certificado.
- **subjectPublicKeyInfo** – Estrutura contendo a chave pública do titular do certificado e identificação do algoritmo correspondente.
- **issuer** – Identificação da EC que emite o certificado.
- **signature** – Estrutura que identifica o algoritmo utilizado para gerar a assinatura da EC que acompanha o certificado.
- **validity –** Estrutura com as duas datas que delimitam o período de validade do certificado.

Identificadores

Os atributos issuer e subject que identificam a EC e o titular do certificado respectivamente são do tipo Name.

O tipo Name provém da norma X.501 e é utilizado porque permite a compatibilidade com os sistemas de directório definidos nas normas X.500 (e.g. DAP e LDAP).

O tipo Name é uma colecção de atributos, geralmente strings da forma "⟨nome⟩ = ⟨valor⟩". Estes atributos definem um Distinguished Name (DN) para o agente titular.

O DN tem uma estrutura hierárquica. A norma X.520 standardiza alguns dos componentes de um DN. Os seguintes são de reconhecimento obrigatório e muito utilizados:

- Country (C)
- organization (O)
- organizational-unit (OU)
- common name (CN)
- serial number (SN)

Exemplo: "C=PT, O=UMINHO, OU=DI, CN=JOE"

Certificados X.509 (V3): Extensões

Permitem personalizar os dados contidos no certificado (e certificados pela EC por via da respetiva assinatura)

As extensões são marcadas como Critical ou Non Critical. Uma aplicação que encontre uma extensão crítica que não reconheça deve rejeitar o certificado.

O RFC-5280 da IETF normaliza as extensões recomendadas para utilização na Internet, definindo como estas devem ser codificados no certificado.

São desaconselhados desvios desta recomendação, nomeadamente no que diz respeito a extensões críticas, apesar de não haver qualquer limitação a nível do standard.

Basic Constraints permite assinalar um certificado como pertencendo a uma (sub-)EC, e limitar o comprimento de cadeias de certificados.

Certificate Policies permite incluir informação relativa às políticas de certificação aplicáveis ao certificado:

- Para certificados de utilizador, permite especificar em que condições o certificado foi emitido e quais as restrições associadas à sua utilização.
- Para certificados de CAs, permite definir as políticas de certificação aplicáveis por CAs hierarquicamente inferiores.

Key Usage - permite restringir as utilizações do par de chaves associado ao certificado e.g. quando uma chave apenas pode ser utilizada para verificar assinaturas digitais. Contempla as seguintes utilizações:

- **digitalSignature** - assinaturas digitais para autenticação e integridade de dados, excepto certificados e CRLs.
- **nonRepudiation** - assinaturas digitais para não repúdio.
- **keyEncipherment** - protecção da confidencialidade de chaves.
- **dataEncipherment** - protecção da confidencialidade de dados.
- **keyAgreement** - protocolos de acordo de chaves.
- **keyCertSign** - assinatura de certificados.
- **cRLSign** - assinatura de CRLs.
- **encipherOnly/decipherOnly** - restringem a funcionalidade keyAgreement.

Extended Key Usage

Permite especificar ou restringir as utilizações previstas para o par de chaves associado ao certificado, em adição ou em alternativa à extensão Key Usage. Estão definidas diversas utilizações, bem como a sua relação com as especificadas na extensão Key Usage:

- WWW server authentication
- WWW client authentication
- Signing of downloadable executable code
- E-mail protection

Cadeias de Certificação e de Confiança

Para utilizar um serviço que requeira o conhecimento de uma chave pública, é necessário obter e validar um certificado que a contenha.

A validação do certificado implica, por sua vez, o conhecimento da chave pública da Entidade de Certificação que o emitiu.

Aqui, existem duas alternativas:

1. A chave pública já é do conhecimento do utilizador (e.g. foi pré-instalada de forma segura)
2. ou é também fornecida por via de um certificado emitido por uma outra EC

Naturalmente que, no segundo caso, há necessidade de proceder à verificação de validade desse certificado.

Que resulta num procedimento de validação recursivo!

Que só termina quando se encontrar um certificado de uma EC que já se confia!

A esta sequência de certificados envolvidos no processo de validação dá-se o nome de **Cadeia de Certificação.**

Note que, numa cadeia de certificação bem formada, o issuer de um certificado deverá ser o subject do antecessor.

As cadeias de certificação refletem uma hierarquia de Entidades de Certificação: as ECs hierarquicamente superiores emitem os certificados das ECs hierarquicamente inferiores.

No(s) topo(s) da hierarquia reside uma EC denominada Root ou raiz. O certificado desta EC é emitido e assinado por ela própria – ou seja, um certificado auto-assinado, i.e. os campos subject e issuer do seu certificado são iguais.

A confiança na chave pública de uma Root EC é estabelecida por um meio externo à ICP.

Por exemplo: sistemas operativos comuns (e.g. MS Windows) incluem certificados de dezenas de Root ECs!

Validação de Certificados

Para cada certificado da cadeia de certificação bem formada, verificar:

1. validade da assinatura
2. a aplicabilidade do certificado (face às extensões)
3. se não foi revogado (e.g. consultando CRLs)

A raiz da cadeia de certificação deverá ser de uma EC que já se conheça a chave pública – designa-se por raiz ou âncora da relação de confiança.

A convenção é que os certificados de “raiz” são auto-assinados (subject é igual ao issuer).

Âncoras de Confiança

Um utilizador conhece um número limitado de chaves públicas pertencentes a ECs (em geral Root CAs) e que funcionam como raizes das relações de confiança.

Isso significa que o utilizador aceitará um certificado emitido por uma dessas CAs e que depositará um determinado nível de confiança no seu conteúdo.

A validação de uma cadeia de certificados termina então quando for encontrado um certificado com essa característica. Esses são normalmente certificados auto-assinados.

Conclusão: o grau de confiança depositada num certificado válido baseia-se, em última análise, na confiança depositada na EC que funcionou como raiz da relação de confiança.

A gestão da lista com âncoras de confiança, assim como do próprio processo de validação das cadeias de certificados, é normalmente assegurada pelo próprio Sistema Operativo.

Em particular, a compilação dos certificados Root adoptados, assim como a sua actualização/manutenção, é assegurada pelo fabricante.

Torna o processo de utilização de certificados praticamente transparente para o utilizador:

- *O que é bom!* porque, quer os conceitos envolvidos na certificação, quer a própria manipulação dos certificados é complexa.

- *O que é mau!* porque toda a segurança que supostamente eles suportam fica comprometida se não houver plena consciência das relações de confiança envolvidas.

Perfis e Políticas de Certificados

Perfis de Certificados (Certificate Profiles)

O standard X509v3 oferece flexibilidade para se definirem extensões à medida das necessidades

A semântica desses atributos é assim “aberta”, mas que deve ser fixada por regras que estabeleçam, num dado contexto (cenário de utilização, aplicação, protocolo, etc.), quais os atributos que devem estar presentes e qual o seu significado.

Um Perfil de Certificados denota uma classe de certificados, e compreende:

- Quais os atributos/extensões de podem ou devem estar presentes e qual a criticidade desses atributos;

- Qual o significado desses atributos e gama de valores admissível;
- Quais os algoritmos criptográficos suportados e tamanho de chaves correspondentes;
- Formato de nomes adotado e restrições que se lhe devem impor; - Política de Certificados associada e respetiva identificação;
- Regras de validação para as extensões críticas consideradas.
-

Políticas de Certificados (Certificate Policies)

A confiança depositada numa EC depende desde:

- Fatores externos, como a credibilidade da instituição ou empresa que suporta a EC e o seu país de origem; etc.
- Informação sobre as práticas adotadas pela EC, e garantias que elas cumprem os requisitos apropriados (e.g. por via de acreditação

Mas a confiança que é depositada num certificado individual depende, em última instância, do critério adotado pela EC na emissão do respetivo certificado.

Obs: note que uma EC pode emitir certificados para diferentes fins (perfis), sendo que é concebível que esses diferentes perfis ofereçam garantias distintas...

Numa ICP, prevê-se a forma de basear a confiança que se deposita num certificado incluindo nele explicitamente a referência para a respetiva Política de Certificados e respetiva documentação.

Uma Política de Certificados é um conjunto de regras que define a aplicabilidade de certificados a uma determinada comunidade ou classe de aplicações com requisitos de segurança comuns:

- A legislação em que se baseará a emissão e utilização dos certificados.
- Os requisitos e as responsabilidades (nomeadamente legais e financeiras) associados a ECs e RAs.
- Os requisitos e as responsabilidades associados a Titulares e Clientes.
- Restrições ao conteúdo e utilização dos certificados
- Procedimentos a serem implementados relativamente a diversos aspetos do funcionamento de ECs e RAs.

Políticas de Certificação na validação de certificados

Uma parte significativa do RFC-5280 é dedicada às políticas de certificação e ao efeito de uma política de certificação imposta num determinado ponto da hierarquia de certificação.

Como foi já referido, esta especificação define também as extensões que permitem incluir este tipo de informação nos certificados X.509.

De facto, associada a cada certificado pode estar uma lista de políticas aplicáveis à sua utilização ou, no caso do certificado de uma EC, uma lista das políticas aceitáveis para os certificados hierarquicamente inferiores.

Durante a validação de um certificado é necessário propagar as políticas impostas desde o topo da hierarquia até à sua base.

A política em vigor na base da hierarquia de certificação resulta da reunião das políticas em vigor nos níveis superiores, com a ressalva de que uma política inserida num determinado nível não pode contradizer uma política de nível superior.

Declaração de Práticas de Certificação (Certification Practice Statements)

Está ainda previsto que as ECs publiquem um documento onde explicitam as práticas seguidas na emissão e gestão dos certificados por si emitidos.

Cada EC publica então uma ou mais Declaração de Práticas de Certificação (CPS), nas quais publicita as suas normas de operação internas.

Em particular, explica a forma como a EC implementa um determinado conjunto de Políticas de Certificação.

A acreditação de uma EC de acordo com uma determinada CPS implica uma auditoria efetuada por (ou em nome de) uma Policy Management Authority.

Por exemplo, a PKI Governamental do Canadá define oito CPs correspondentes a quatro níveis de segurança na utilização de certificados em assinaturas digitais e proteção de dados. Uma CA que pretenda emitir certificados em conformidade com estas políticas tem de ser credenciada pelo estado Canadiano.

É também possível (e recomendado) incluir nos certificados referência explícita à CPS.

Exemplos de Perfis de Certificados

1. Proteção de Email (S/MIME)
2. Autenticação de Sítios (TLS-server)
3. Autenticação em Serviços (TLS-client)
4. Assinatura Qualificada de Documentos

Certificados TLS

Classificação do Método de Validação (CA/browser forum)

1. **Domain Validation (DV)** - identidade verificada unicamente com base em evidência de controlo do domínio DNS.

2. **Organization Validation (OV)** - verifica existência/controlo de uma organização (e.g. empresa, organismo público, etc.)

3. **Extended Validation (EV)** - critérios mais rigorosos de validação fornecendo evidência de controlo legal sobre a entidade.

Certificate Transparency

Motivação:

1. PKIX não prevê um mecanismo simples e efectivo de auditar se certificados são emitidos com lacunas no processo de validação (quer por omissões da CA, ou por comprometimento desta);

2. …esses problemas tem ocorrido com frequência crescente e com impacto significativo…

3. …tornando evidente que é "demasiadamente simples" uma CA incorrer em falhas sem terem consciência da gravidade das possíveis consequências.

Objectivos:

1. Impossibilita (dificulta) a capacidade de CAs emitirem certificados sem conhecimento dos detentores dos domínios DNS correspondentes;

2. Disponibiliza um mecanismo aberto de "registo" e "monitorização" que permite controlar se um certificado foi emitido de forma incorrecta ou maliciosa;

3. Protege os utilizadores de serem induzidos em erro por certificados incorrectamente/maliciosamente emitidos.

Arquitectura:

1. Logs:

- servidores que periodicamente contactam os servidor de Logs por forma validarem consistência e identificarem eventuais certificados suspeitos;
- mantém registo incremental assegurado criptograficamente e publicamente auditável de certificados;
- operados por intervenientes interessados na confiabilidade do sistema (CAs, ISPs, etc.)

2. Monitors:

- servidores que periodicamente contactam os servidor de Logs por forma validarem consistência e identificarem eventuais certificados suspeitos;

3. Auditors:

- componentes de software (incorporado em, e.g., web browsers) capazes de interrogar e verificar integridade dos Logs

• …verificando em particular se contém o certificado pretendido.

Revogação de Certificados

Por vezes há necessidade de revogar certificados que ainda se encontram no seu período de validade

Motivos para a revogação de certificados:

- Chave privada comprometida;
- Circunstância que justificava associação do issuer à chave pública já não se verifica (e.g. issuer é o detentor de um cargo temporário)
- Dados contidos no certificado deixam de ser correctos (e.g. atributo já não se aplica)

Mecanismo originalmente previsto para a revogação de certificados são as Listas de Revogação de Certificados (CRL)

Certificate Revocation Lists (CRL)

As Certificate Revocation Lists (CRL) são o canal previsto no X.509 para a revogação de certificados dentro do período de validade.

Uma CRL diz-se:

- Base CRL quando lista todos os certificados revogados por uma EC que ainda estão no seu período de validade.
- Delta CRL quando apenas lista os certificados revogados desde a publicação de uma Base CRL referenciada.

Cada CRL tem um contexto específico (o conjunto de certificados passíveis de aparecerem no seu conteúdo), que deve estar bem definido.

A segurança de uma ICP depende da eficácia com que são revogados os certificados que se tornaram inválidos. Este facto sugere que, assim que um certificado se torna inválido, uma nova CRL deva ser publicada.

No entanto, desta forma, um utilizador nunca saberia qual a CRL mais recente.

Admitindo que o atacante controla o meio de comunicação que liga o utilizador ao ponto de publicação de uma CRL, possibilitaria ataques do tipo:

- Vamos admitir que o utilizador pretende utilizar um certificado cuja chave privada foi comprometida, e que é conhecida pelo intruso.
- O utilizador tenta obter a CRL mais recente, que revogaria o certificado.

- Mas o intruso fornece uma versão antiga da CRL onde ainda não aparece a revogação desse certificado.
- O utilizador aceita o certificado porque não tem como saber que a CRL que utilizou estava desatualizada.

De facto, a utilidade de uma CRL depende do facto de ela ser publicada periodicamente (e.g. diariamente, semanalmente, mensalmente, etc.)

Isto permite também que a CRL seja pública, e distribuída por canais não seguros.

Compete ao utilizador estar ao corrente da frequência de publicação das CRLs, e definir uma política sobre o que é uma CRL "suficientemente recente".

O atributo *nextUpdate* permite indicar na própria CRL a altura a partir da qual é garantida a publicação de uma nova CRL.

O utilizador está consciente de que, a menos que obtenha a última versão da CRL, estará a correr o risco de aceitar certificados inválidos.

Isto não quer dizer que não possam ser publicadas CRLs extraordinárias, fora da frequência normal de publicação.

Isto pode ocorrer, por exemplo, se um certificado importante tem de ser revogado porque a chave privada correspondente foi comprometida (e.g. o certificado de uma EC hierarquicamente inferior).

No entanto, a granularidade garantida nunca é inferior ao período de publicação da CRL: não é possível garantir que os utilizadores obtenham a CRL extraordinária antes da data de publicação da próxima CRL periódica.

Online Certificate Status Protocol (OCSP)

Os riscos associados à utilização indevida de um certificado revogado podem não ser aceitáveis.

Em alternativa ou adição à consulta de uma CRL, pode ser necessária informação actual sobre o estado de revogação de um certificado.

O OCSP (definido no RFC-6960) permite a uma aplicação determinar o estado de um certificado com maior frescura temporal.

O Cliente OCSP emite um pedido a um Responder OCSP (Servidor) e suspende a aceitação do certificado até que este forneça uma resposta.

Estão ainda previstos serviços análogos ao OCSP para Delegated Path Validation e Delegated Path Discovery (RFC-3379).

Aplicações

Protocolos de comunicação:

- IPSec
- TLS

Correio Electrónico Seguro:

- S/Mime
- PGP

Assinatura de Documentos PAdES:

- Time-Stamping

Tokens Criptográficos:

- Cartão de Cidadão

Protocolos de Comunicação

IPSec

O IP (Internet Protocol) está ao nível da camada de rede do Modelo OSI. Fornece serviços de encaminhamento de pacotes através de redes heterogéneas.

A maior parte das infra-estruturas de comunicação na Internet são baseadas neste protocolo, conjuntamente com o TCP (TCP/IP).

O IPsec fornece o mesmo conjunto de serviços, mas inclui funcionalidade extra ao nível da segurança.

Estes serviços são oferecidos ao nível da camada de rede, oferecendo proteção não só a esse nível, mas também a todas as camadas superiores.

O IPsec está definido nas especificações RFC2401 e seguintes, do IETF.

O IPsec oferece os seguintes serviços seguros ao nível da camada de rede:

› Controlo de acessos

› Integridade ao nível do pacote

› Autenticação da origem de dados

› Proteção contra pacotes repetidos

› Confidencialidade

Estes serviços permitem proteger ligações de rede entre nós IP, entre gateways seguras, ou entre um nó IP e uma gateway segura.

Não substituem os serviços IP. São módulos adicionais que podem ser implementados e utilizados consoante o contexto e as necessidades das aplicações.

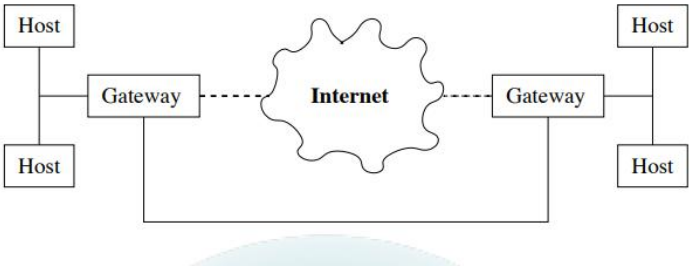
Protocolo IP

Funcionamento:

- Dentro de uma rede local, cada nó constrói um pacote IP incluindo os endereços de origem e destino no cabeçalho.

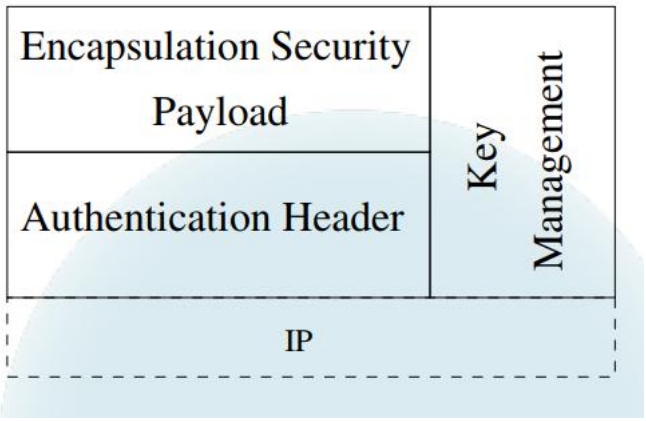
- A comunicação com redes remotas é feita passando os pacotes a uma gateway: o endereço da gateway encapsula o verdadeiro.

- A gateway substitui o encapsulamento reencaminhando o pacote. A gateway da rede remota retira o encapsulamento.



Estrutura do IPSec

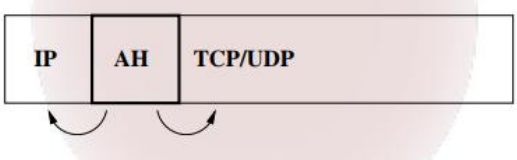
O IPsec está estruturado em duas sub-camadas:



As duas sub-camadas são apoiadas por procedimentos e protocolos de gestão de chaves criptográficas (manuais ou automáticos).

Os protocolos estão especificados de forma a serem independentes de algoritmos criptográficos. No entanto, alguns destes algoritmos estão pré-definidos.

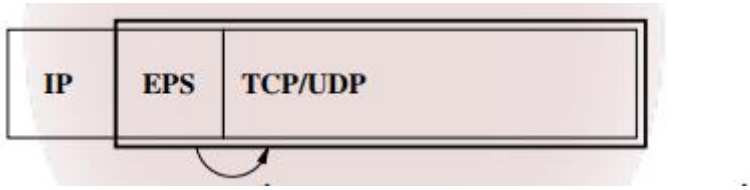
Authentication Header (AH)



A sub-camada IP Authentication Header (AH) inclui serviços de integridade ao nível dos pacotes, autenticação da origem de dados e, opcionalmente, proteção contra a repetição de pacotes.

Recorre-se ao AH quando se pretende autenticação da informação correspondente à camada de rede (cabeçalho IP), ou quando a confidencialidade não é necessária (ou permitida).

Encapsulating Security Payload (ESP)



A sub-camada Encapsulating Security Payload (ESP) fornece confidencialidade (cifragem) da totalidade ou de apenas parte do tráfego.

Como opções, o ESP oferece autenticação, verificação de integridade, e protecção contra pacotes repetidos. No entanto, esta funcionalidade abrange apenas informação correspondente às camadas de transporte e superiores (não trata o cabeçalho IP).

No entanto pode ser (geralmente é) usado isoladamente.

Modos de Funcionamento

Consoante o tipo de nós envolvidos, o IPSec pode funcionar em:

Transport Mode – Sobre os cabeçalhos IP originais:

- Apenas serve para ligações host-host
- Com ESP não há protecção dos cabeçalhos IP (interferiria com a infraestrutura IP).
- Com AH, há uma protecção parcial desses cabeçalhos.

Tunnel Mode – Sobre cabeçalhos IP encapsulados.

- Corresponde a um túnel IP (caminho virtual entre nós) — tipicamente utilizado em ligações com/entre gateways.
- A proteção alcança todo o pacote original.
- Diferenças entre AH e ESP mantêm-se, mas apenas para o cabeçalho exterior.

Algumas Críticas ao IPSec

O IPsec é muito criticado pela sua complexidade. Os principais problemas apontados são:

- A complexidade do IPsec dificulta a sua implementação e configuração. Além disso torna virtualmente impossível uma avaliação cabal da respetiva segurança.
- A documentação é muito dispersa e difícil de ler.

- Existe demasiada flexibilidade e funcionalidades aparentemente redundantes.

Por exemplo:

- Porquê dois modos de funcionamento (transporte e túnel): se o que se pretende é mais segurança, porque não utilizar apenas o túnel?
- porquê dois protocolos (AH e ESP) quando seria muito mais simples especificar um protocolo único, mais simples e mais consistente?

TLS

Originalmente concebido pela Netscape inc. como Secure Sockets Layer (SSL).

Está para o TCP como o IPsec está para o IP. É um upgrade da camada de transporte para incluir segurança nas comunicações.

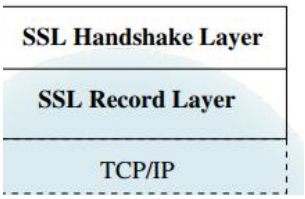
A versão 3 do SSL foi adoptada pela IETF sob a designação Transport Layer Security (TLS). O TLS está definido no RFC2246.

Os serviços fornecidos pelo SSL incluem:

- Confidencialidade baseada em cifras simétricas.
- Autenticação baseada em criptografia de chave pública.
- Integridade baseada em Message Authentication Code

Estrutura do TLS

O SSL está estruturado em duas sub-camadas:



A Handshake Layer permite a autenticação entre clientes e servidores, e a negociação de algoritmos e chaves criptográficos antes de se iniciar a troca de dados através da Record Layer. A Record Layer encapsula a informação correspondente às camadas superiores.

Sessões SSL/TLS

O funcionamento do TLS baseia-se em sessões estabelecidas entre um cliente e um servidor.

Cada sessão TLS pode incluir várias ligações seguras, e cada nó pode manter diversas sessões TLS. Durante o seu estabelecimento e operação, as sessões e ligações TLS atravessam uma sequência de estados.

Cliente e Servidor mantêm uma máquina de estados para cada sessão e ligação. A camada de Handshake sincroniza os estados no cliente e no servidor.

As transições entre estados efetuam-se em duas fases:

- Primeiro constrói-se/negoceia-se um pending state.
- Depois substitui-se o operating state pelo pending state.

Record Layer

Recebe informação arbitrária das camadas superiores, em blocos de dados de tamanho variável.

Os dados são fragmentados em blocos com um máximo de 2^{14} bytes denominados SSL Plaintext.

Os blocos SSL Plaintext são protegidos com a cifra e algoritmo de MAC definidos na CipherSpec da sessão. O resultado é um bloco do tipo SSL Ciphertext. (obs: até v1.2, o MAC é calculado antes da cifragem).

Estes blocos são trocados entre Cliente e Servidor que têm de reverter estas transformações para obter o texto limpo.

Handshake Layer

Os parâmetros de sessão e ligação utilizados pela Record Layer são estabelecidos pela Hanshake Layer.

As mensagens da Handshake Layer viajam elas próprias sob o controlo da Record Layer. Inicialmente não há qualquer protecção: é utilizada uma cipher spec nula até que a primeira negociação seja concluída.

Uma negociação é iniciada pelo Cliente com uma mensagem ClientHello. O Servidor deve responder com uma mensagem equivalente.

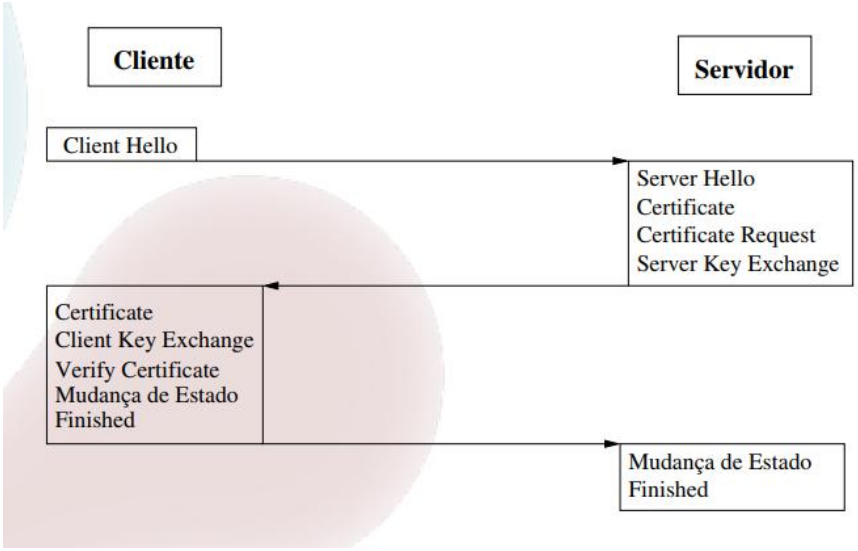
Ficam acordados:

- A versão do protocolo SSL a utilizar
- O identificador da sessão e os números aleatórios.
- Os algoritmos criptográficos a utilizar (os mais fortes dos suportados por ambos).

Modos de Funcionamento

O SSL/TLS suporta modos de autenticação para:

- autenticação de servidor — modo normalmente adotado em cenários web (servidor atesta sua identidade por intermédio de assinatura e com recurso a certificado X509 com atributos apropriados)
- autenticação mútua — servidor solicita também que o cliente se autentique criptograficamente, requisitando para isso que exiba certificado apropriado.



1. Na resposta ao clienteHello, o Servidor envia o seu certificado X.509. O cliente, além da validação habitual, assegura-se de que o nome de domínio do Servidor está correctamente indicado no certificado.
2. Parâmetros do Servidor específicos para acordo de chaves são também ser enviados nesta fase (ServerKeyExchange).
3. Caso o Servidor autentique o Cliente, solicita o certificado correspondente (CertificateRequest). Este pedido inclui um desafio para ser utilizado na autenticação do cliente.
4. O Servidor termina esta fase da negociação enviando uma mensagem ServerHelloDone.
5. Caso tenha recebido um pedido de certificado, o Cliente deve envia-lo (ou a negociação falha). Conjuntamente com o certificado deve enviar a assinatura digital do desafio, comprovando assim a posse da chave privada associada ao certificado.
6. Finalmente, o Cliente envia os seus parâmetros para acordo de chaves (ClientKeyExchange), altera o seu estado de sessão, e envia uma primeira mensagem cifrada que indica o seu estado de prontidão (finished).
7. O Servidor efectua o mesmo procedimento e a negociação termina tendo sido acordado o MasterSecret da sessão (de onde são derivados os segredos requeridos)

Segurança

Mesmo se a versão actual do protocolo TLS é considerada segura, a verdade é que o historial de ataques a que a família de protocolos SSL/TLS se viu confrontada ao longo dos últimos anos é constrangedora!

Em geral, os ataques dirigidos ao protocolo propriamente dito foram sendo ultrapassados pelas revisões sucessivas (e.g. “ciphersuit rollback”, possível na versão SSL.2, foi ultrapassado com a autenticação das mensagens de handshake requerida pelo SSL.3).

Os ataques mais recentes ao SSL/TLS dirigem-se maioritariamente a:

- aspectos de implementação “menos cuidada” do protocolo (ou de features obscuras do protocolo, como no ataque heartbleed);
- interacção com cipher suites específicas (e.g. utilização do RC4 ou padding oracle attack ao modo CBC).

Certas implementações optam deliberadamente por suportar um fragmento restrito do protocolo para minimizar potenciais problemas de segurança (e.g. Amazon’s AWS S2N).

Na prática, a enorme quantidade opções de configuração exige enorme cuidado no seu deployment.

TLS v1.3

Nova versão do protocolo TLS concluiu recentemente a fase final de standardização (v1.3 - RFC-5246).

Consiste numa redefinição substancial do protocolo:

- Assume quebra de "backward compatibility"para versões anteriores;
- Propõe-se atingir: maior eficiência e maior segurança (vs. TLS1.2).

Algumas características:

- Simplifica fase de handshake (1 roundtrip).
- Remove suporte a características "obscuras"ou pouco utilizadas; técnicas criptográficas obsoletas (e.g. DES, RC4, MD5, SHA-1, AES-CBC, ...); etc.
- Opta por suportar conjunto restrito de possibilidades (grupos DHE; modos AEAD; …)

Outros protocolos de Sessão

Secure Shell (SSH) — protocolo de acesso remote a máquinas (inclui variantes seguras dos protocolos telnet, ftp, etc.);

Virtual-Private Networks (VPNs)

- PPTP — protocolo com sérios problemas de segurança e já considerado obsoleto (mas ainda muito utilizado…)
- L2TP/IPSec — protocolo vpn recomendável, que recorre aos serviços IPSec para garantias de confidencialidade.
- openVPN, SSTP, …

Correio Electrónico Seguro

S/Mime

Secure/Multipurpose Internet Mail Extensions (S/Mime) oferece uma forma consistente de enviar e receber informação segura no formato MIME.

Disponibiliza combinações dos seguintes serviços:

- autenticação, integridade e não repúdio de origem das mensagens através da utilização de assinaturas digitais;
- confidencialidade dos dados, através de da utilização de um esquema de cifra de chave pública (mais precisamente, um esquema híbrido).

Utiliza certificados X509 para autenticação das chaves públicas.

Standardizado pelo Internet Engineering Task Force RFC-5751 (V3.2).

Suportado pela generalidade das aplicações de EMail (MUA) existentes.

Utilização típica

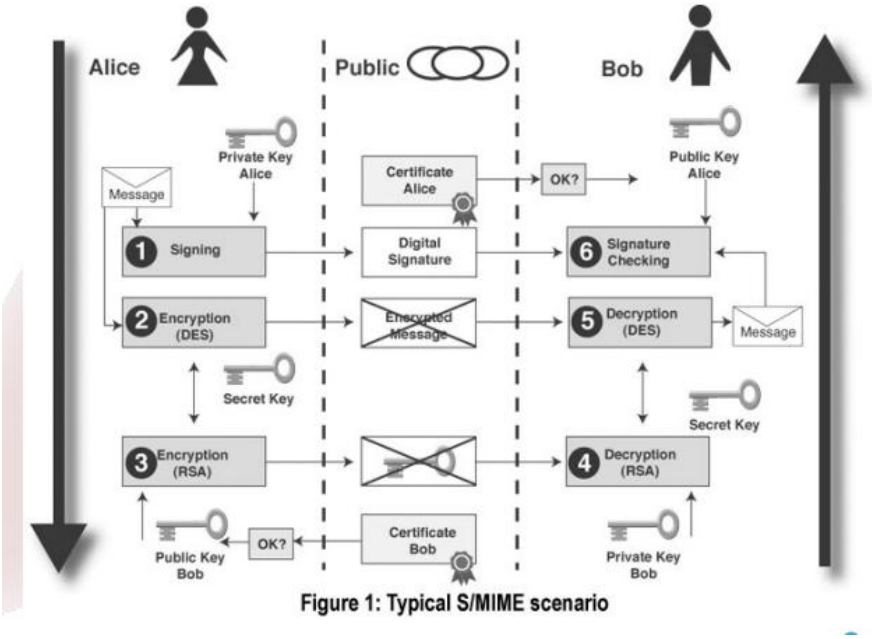


Figure 1: Typical S/MIME scenario

PGP

O PGP é uma aplicação freeware desenvolvida por Phil Zimmermann com o objectivo de disponibilizar uma infraestrutura para protecção de informação (i.e. privacidade) à disposição do cidadão comum.

O lançamento do software causou alguma polémica nos EUA devido às leis que restringiam a difusão e utilização generalizada da chamada strong cryptography. A batalha legal associada constituiu um marco importante no equilíbrio sempre instável entre privacidade individual e segurança da comunidade.

Um argumento apresentado a favor da utilização do PGP, e contra as restrições ao uso de sistemas de protecção da privacidade é: “se a protecção de informação é ilegalizada apenas os fora-da-lei conseguem ter privacidade!”

Utilização

O PGP é um sistema com três vertentes principais:

› **Privacidade** — utilização de algoritmos de compressão, cifras simétricas e assimétricas na proteção de informação, nomeadamente ficheiros e mensagens de e-mail.

› **Integridade e Autenticação** — utilização de funções de hash criptográficas e algoritmos de assinatura digital para a assinatura de mensagens e documentos.

› **Certificação** — estabelecimento de relações de confiança e distribuição de chaves públicas com base num esquema certificação próprio, alternativo à PKI e ao X.509

Assinatura de Documentos

PAdES

PDF é dos formatos mais utilizados como suporte a documentos digitais.

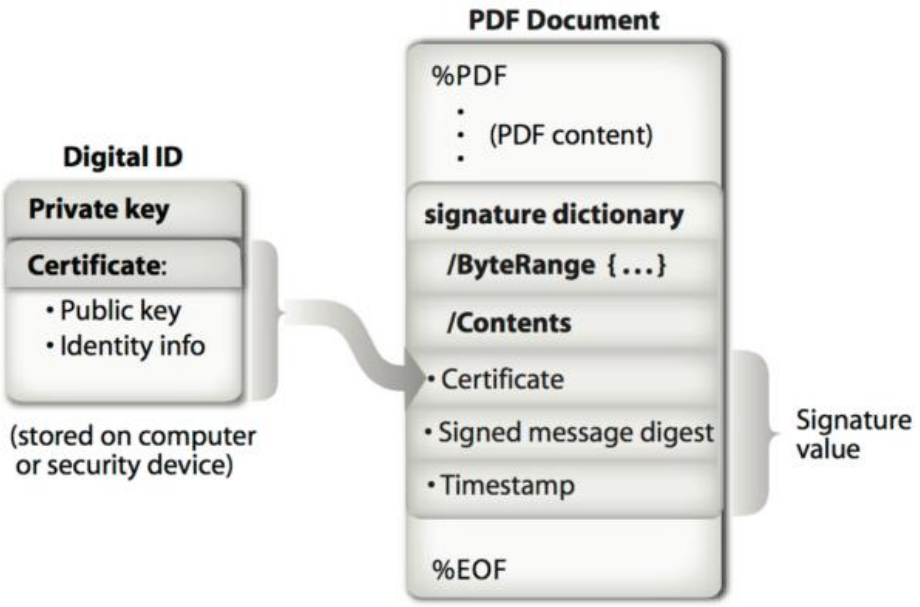
Originário de uma empresa privada (Adobe), mas entretanto standardizado por organismos competentes (ISO 32000-1, 32000-2).

Incluí já suporte a assinaturas digitais (desde V1.7).

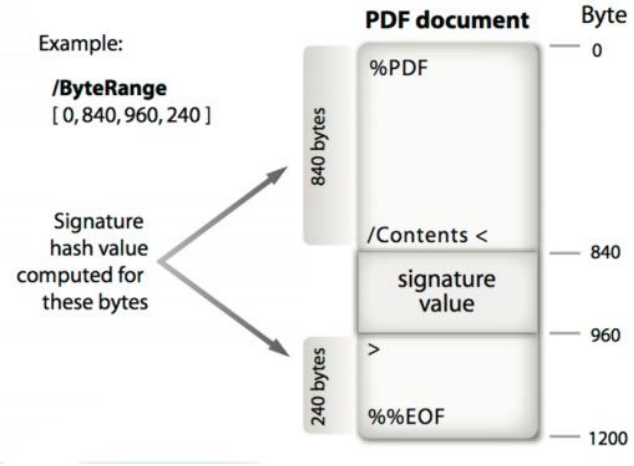
O que permite beneficiar das faculdades oferecidas pela assinatura digital com recurso a aplicações disponíveis livremente (e.g. Acrobat Reader).

Versões mais recentes atualizaram suporte às assinaturas digitais para ficarem conformes especificação dos standards PAdES (PDF Advanced Electronic Signatures), da ETSI (TS 102 778).

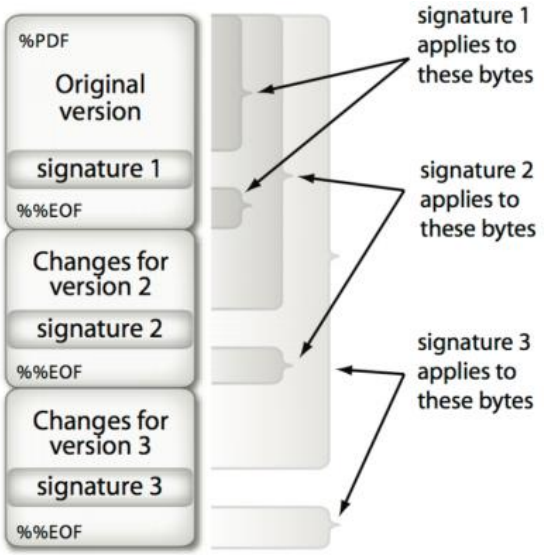
Assinatura de documentos PDF



Toda a informação do PDF (com excepção da própria assinatura) é considerada para efeitos do respectivo cálculo.



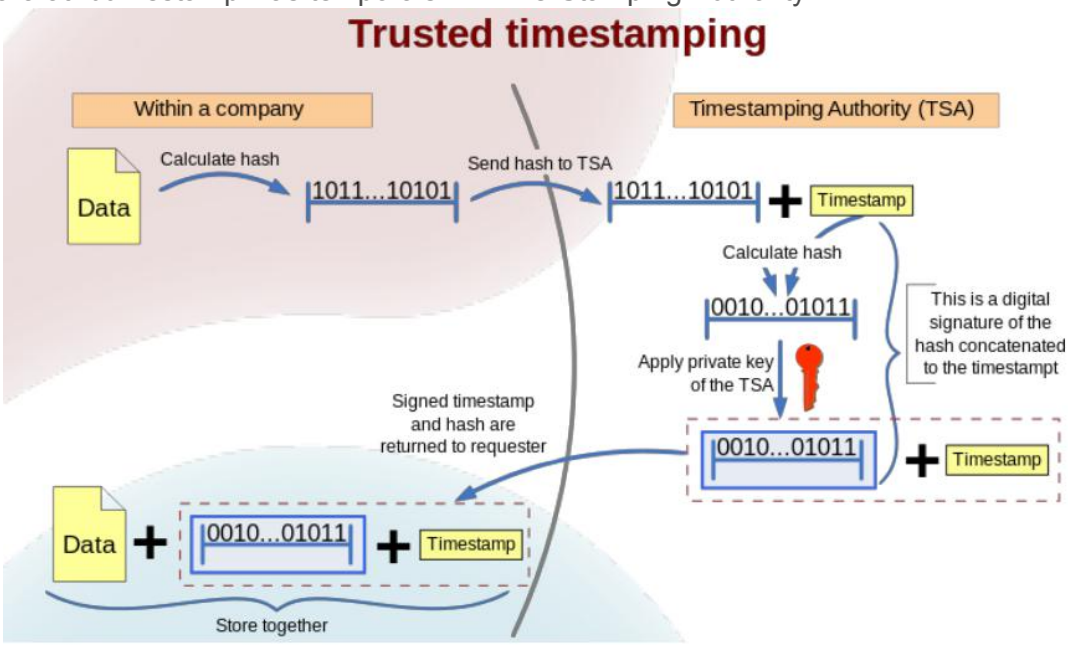
São suportadas multiplas assinaturas sobre um mesmo documento (assinaturas incrementais).



Time-Stamping

Como estabelecer que uma assinatura digital foi realizada num determinado instante? Como preservar o facto que, num determinado instante, a assinatura é considerada válida?

Para ultrapassar essas dificuldades, considera-se uma terceira parte de confiança para atribuir estampilhas temporais — Time-Stamping Authority.



Tokens Criptográficos

Dispositivos electrónicos portáteis que, pela sua natureza, constituem um ambiente protegido para:

- armazenar informação secreta;
- executar operações críticas de segurança.

Disponíveis em diferentes formatos/tecnologias (e.g. USB, bluetooth, smart-card, contactless tokens, ...)

Normalmente utilizados para autenticação/identificação, ou para realizar operações criptográficas críticas (e.g. assinaturas digitais).

Chaves criptográficas podem ser geradas, armazenadas, e utilizadas no próprio dispositivo, minimizando assim o risco da sua exposição a ambientes hostís.

Incluem normalmente medidas de protecção físicas (tamper resistant packaging).

Cartão de Cidadão

Smart-card com funcionalidades criptográficas, contendo informação em formato digital do utente

Funcionalidades Criptográficas

Dois certificados pessoais com respectivas chaves privadas

- autenticação
- assinatura qualificada

Certificados das CAs (cadeia de certificação dos certificados do cidadão)

Coprocessador criptográfico (RSA), que possibilitam que as assinaturas sejam realizadas no próprio cartão (chaves privadas protegidas por PIN)

Outras chaves para acesso privilegiado (e.g. serviços de segurança).