Digital signatures

Digital signatures: goals

- ▷ Authenticate the contents of a document
  - Ensure its integrity
- ▷ Authenticate its author
  - Ensure the identity of the creator/originator
- ▷ Non-repudiation
  - Prevent signing repudiation

Digital signatures: fundamental approach

- Signature generation
  - Production of a value using a private key
  - Signer (or signatory) is the private key owner
- Signature verification
  - Validation of an expression using the signature and a public key
  - Anyone can verify
    - Since public keys can be universally known
  - Signature can be linked to the public key owner

Signature schemes

- With message (or document) recovery
  - The message is fully recovered upon a signature validation
  - Signature validation is mandatory prior to message observation
- With appendix
  - The signature is detached from the message
  - The message can be observed anytime

Key elements of a digital signature

- The message (or document)
  - It only makes sense with the signed object
- The signature date
  - Because is usually required
  - Because key pairs have validity periods
- The identity of the signatory
  - Otherwise, it would not mean anything

The document to sign

- It may accommodate digital signatures as appendixes
  - PDF, XML
  - DOCX (archive of XML components)
- Other formats may group document and signature
  - S/MIME (mail)
  - JOSE (JSON Object Signing and Encryption)

The signature date

- It may be given by the signatory machine
  - Does not protect against time forgery attacks by the signatory
- It may be given by a Time Stamping Authority (TSA)
  - Does not protect against the future discovery of the private keys used

The identity of the signatory

- Usually provided by a X.509 public key certificate
  - It provides several attributes of the identity
  - It provides the public key for signature validation
  - It provides the acceptable signing time frame
    - Together with the respective CRL

Optional elements of a digital signature

- Attributes that can help to interpret it
  - Location
    - Where it was signed
  - Reason
    - Why it was signed
  - Appearance
    - Handwritten signature (usually without legal value)
    - Name of the signatory
    - Date of signature
    - Some kind of logo

Digital signatures' algorithms

- Message recovery scheme
  - Asymmetric encryption and decryption
  - Only for RSA
- Signing
  $A_x(doc) = info + E(K_x^{-1}, doc)$
- Verification
  $info \rightarrow K_x$
  $D(K_x, A_x(doc))$
  Check integrity of doc

- Message appendix scheme
  - Digest functions
  - Asymmetric signature and validation
  - RSA, ElGamal (DSA), EC
- Signing
  $A_x(doc) = info + E(K_x^{-1}, h(doc+info))$
  $A_x(doc) = info + S(K_x^{-1}, h(doc+info))$
- Verification
  $info \rightarrow K_x$
  $D(K_x, A_x(doc)) \equiv h(doc + info)$
  $V(K_x, A_x(doc), h(doc + info)) = True$

RSA signatures

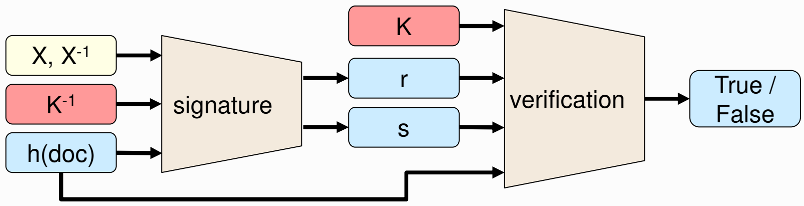
- Creation with private key
  - Validation with the corresponding public key
- Special padding for Signature Scheme w/ Appendix
  - RSASSA-PKCS#1 (v1.5)
    - Deterministic
  - RSASSA-PSS (Probabilistic Signature Scheme)
    - Randomized (EMSA-PSS)
- Hash function prefixing
  - ASN.1 algorithm OID

ASN.1 digest algorithm prefixes

Digest	ASN.1 OID	Perfix (bytes)
MD5	1.2.840.113549.2.5	30 20 30 0C 06 08 **2A 86 48 86 F7 0D 02 05** 05 00 04 10
RIPEMD-160	1.3.36.3.2.1	30 21 30 09 06 05 **2B 24 03 02 01** 05 00 04 14
SHA-1	1.3.14.3.2.26	30 21 30 09 06 05 **2B 0E 03 02 1A** 05 00 04 14
SHA-224	2.16.840.1.101.3.4.2.4	30 2D 30 0D 06 09 **60 86 48 01 65 03 04 02 04** 05 00 04 1C
SHA-256	2.16.840.1.101.3.4.2.1	30 31 30 0D 06 09 **60 86 48 01 65 03 04 02 01** 05 00 04 20
SHA-384	2.16.840.1.101.3.4.2.2	30 41 30 0D 06 09 **60 86 48 01 65 03 04 02 02** 05 00 04 30
SHA-512	2.16.840.1.101.3.4.2.3	30 51 30 0D 06 09 **60 86 48 01 65 03 04 02 03** 05 00 04 40

Digital Signature Standard (DSS)

- With a variant of ElGamal
  - Digital Signature Algorithm (DSA)
  - Uses a random value **X**, and its multiplicative inverse, $\mathbf{X^{-1}}$
  - **r** depends on **X**, **s** depends on $\mathbf{X^{-1}}$
- With elliptic curves (ECDSA)
  - Similar to DSA with EC



Blind signatures

- Signatures made by a "blinded" signer
  - Signer cannot observe the contents it signs
  - Similar to a handwritten signature on an envelope containing a document and a carbon-copy sheet
- Useful for ensuring anonymity of the signed information holder, while the signed information provides some extra functionality
  - Signer X knows who requires a signature (Y)
  - X signs T_1, but Y afterwards transforms it into a signature over T_2
    - Not any T_2, a specific one linked to T_1
  - Requester Y can present T_2 signed by X
    - But it cannot change T_2
    - X cannot link T_2 to the T_1 that it observed when signing

Chaum Blind Signatures

- Implementation using RSA
  - Blinding
    - Random blinding factor K
    - $k \times k^{-1} \equiv 1 \pmod N$
    - $m' = k^e \times m \bmod N$
  - Ordinary signature (encryption w/ private key)
    - $A_x(m') = (m')^d \bmod N$
  - Unblinding
    - $A_x(m) = k^{-1} \times A_x(m') \bmod$

Qualified electronic signature

- An electronic signature compliant with the EU eIDAS Regulation
  - Regulation No 910/2014
- Enables to verify the authorship of a declaration in electronic data exchange
  - Over long periods of time
- Can be considered as a digital equivalent to handwritten signatures

Qualified electronic signature

- Three main requirements:
  - The signatory must be linked and uniquely identified to the signature
  - The data used to create the signature must be under the sole control of the signatory
  - Must have the ability to identify if the data that accompanies the signature has been tampered with since the signing of the message

Qualified electronic signature

- Must be created using a qualified signature creation device
  - This device uses specific hardware and software that ensures that the signatory only has control of their private key
- A qualified trust service provider manages the signature creation data that is produced
  - But the signature creation data must remain unique, confidential and protected from forgery

Signature devices

- Crypto tokens
  - Smartcards
  - Cartão de Cidadão
- Cloud HSM (Hardware Secure Modules)
  - Mainly for mobile devices
  - Chave Móvel Digital

PKCS #11

- Crypto tokens' standard interface
  - Cryptoki
- Enables applications to use arbitrary PKCS #11 libraries
  - Developed for a specific set of crypto tokens
- Specification in C
  - There are interfaces for other languages

Microsoft Cryptographic API (CAPI)

- Unique OS security middleware hub
  - Applications use the abstractions it provides
- Cryptographic Services Providers (CSP)
  - Target-specific software module under the CAPI
    - It enables a particular functionality
  - Signature capabilities can be added with CSPs
    - For local crypto tokens
    - For remote, cloud-based HSMs

Long-Term Validation (LTV)

- A document signature may become invalid upon an initial verification
  - Due to a late certification revocation
- Signature algorithms may become vulnerable
  - Allowing signatures with old credentials to be forged
- LTV attempts to handle both issues
  - With successive signature layers
  - Performed by original signers or signed documents' holders
  - It leverages signed timestamps created by TSAs
    - Proof of Existence (POE)

LTV: Proof of Existence (POE)

A proof of existence is evidence that proves that an object
(a certificate, a CRL, signature value, hash value, etc.)
existed at a specific date/time, which may be a date/time in the past.

The possession of a certain object at current time is a proof of its existence at the current time.

A suitable way of providing proof of existence of an object at a time in the past is to generate a time-stamp on that object.

Electronic Signatures and Infrastructures (ESI); Signature validation procedures and policies, ETSI TS 102 853 V1.1.2 (2012-10)

- If
  a signed timestamp can be validated now
  and
  the timestamp is bounded to values that were valid when it was signed
  then
  those values are valid now

LTV Advanced Electronic Signatures (AdES)

- PAdES
  - PDF Advanced Electronic Signature
- CAdES
  - Cryptographic Message Syntax Advanced Electronic Signatures
- XAdES
  - XML Advanced Electronic Signatures