openHPI – Confidential Communication in the Internet

Feasible Digital Signatures

Prof. Dr. Christoph Meinel

Hasso Plattner Institute
University of Potsdam, Germany

Feasible Digital Signatures
Basic Scheme (1/2)

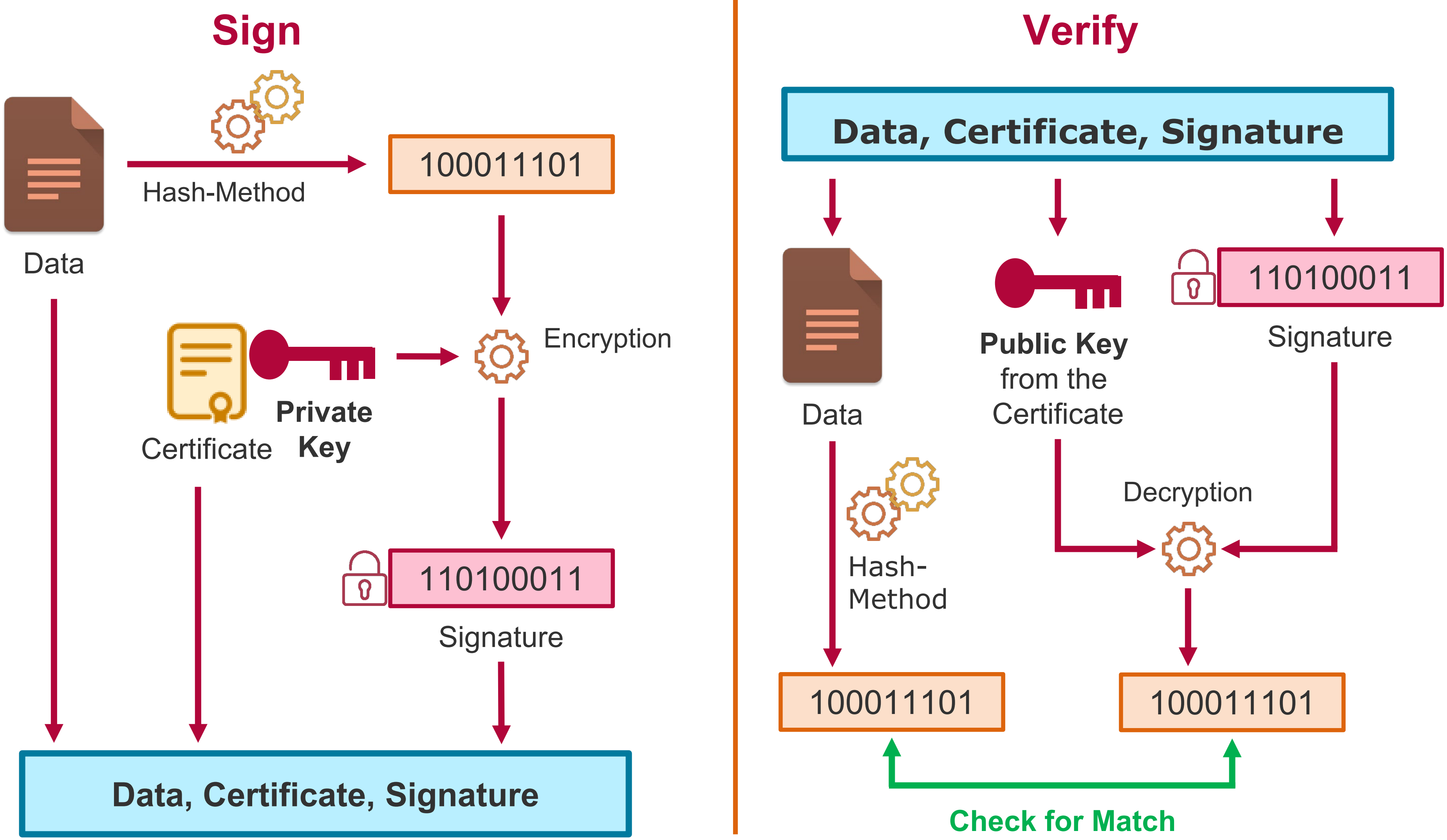
Main problem with digital signatures:

- The encryption of the complete document with a public-key cryptosystem for a digital signature **requires enormous computing efforts**

Idea:

- Not the document itself is signed, but only the cryptographic **hash of the document** is signed (i.e. asymmetrical encrypted)
- Any change in the document results in a change in its cryptographic hash. Therefore any manipulation in the document can be discovered when working with its hash

Reminder: **Digital Signatures** Overview (2/2)



RSA Signatures

In most digital signature protocols RSA is used as asymmetric cryptosystem

- All attacks against RSA encryption are thus also attacks on RSA-based digital signatures
- When RSA is used for signing longer keys are applied than for encryption with RSA:
  - Digital signatures generally have to be valid for a long time, e.g. several years
  - Verification is easy with RSA

RSA Signatures Algorithm

Let **p**, **q** large prime numbers, **n** = p·q and a·b = 1 mod φ(n)

Alice owns:

- Public Key: (n, b)
- Private Key: (p, q, a)

RSA signature algorithm

- Alice signs the hash value **h** = h(M) of the message M
  - **sig**(**h**) = h^a mod n
- Alice sends message M with the signature **sig**(**h**) to Bob
- Bob calculates the hash value h(M') of the received message M' and verifies the signature of Alice:
  - ver(h(M') , **sig**(**h**)) = "yes" if h(M') = **sig**(**h**)^b mod n

Example of a RSA Signature

Let **p = 6.997, q =7.927:**

1. Then n = p·q = 55.465.219 and φ(n) = 6.996·7.927 = 55.450.296
2. If b = 5, then a = 5^{-1} = 44.360.237 mod φ(n) .
3. Public Key of Alice: (55.465.219, 5)
   Private Key of Alice: (p, q, 44.360.237)
4. Alice signs hash value 31.229.978 of message M
   - 30.729.435 = $31.229.978^{44.360.237}$ mod 55.465.219
5. Alice sends the message M together with the signature to Bob
6. Bob calculates the hash value h(M') of the received Message M�much and verifies the signature of Alice:
   - ver(h(M'), 30.729.435) = "yes" h(M') = $30.729.435^5$ mod 55.465.219

Digital Signatures in Practice (1/2)

Each application that creates or deals with binding documents should obligatory equipped with a simple user interface for signing:

- ○ **Buttons to sign** and **to verify**

Signing with asymmetric cryptosystems requires a **private key** of the signee. Where does it come from?

- Storage in the main memory with password protection, but attention:
  - private key is only as secure as its password protection
- Storage on Memory Stick
- Storage on chip card with a crypto chip for encryption

Digital Signatures in Practice (2/2)

To verify signatures based on asymmetric cryptosystems, the signee's public key is required. Where does it come from?

- When binding public keys to their owner there is a trust problem
- To solve this trust problem a complex infrastructure (**PKI**) is necessary

Summary:

- **Digital signatures** are much more secure than signatures by hand
- Technology for digital signing is mature and ready
- Signature legislations creates legal framework (EU, D, UK)
- State responsibility for digital identification of citizens