Digital Signatures

Murat Kantarcioglu

Based on Prof. Li's Slides

Digital Signatures: The Problem

- Consider the real-life example where a person pays by credit card and signs a bill; the seller verifies that the signature on the bill is the same with the signature on the card
- Contracts, they are valid if they are signed.
- Can we have a similar service in the electronic world?

2

Digital Signatures

- Digital Signature: a data string which associates a message with some originating entity.
- Digital Signature Scheme: for each key, there is a SECRET signature generation algorithm and a PUBLIC verification algorithm.
- Services provided:
  - Authentication
  - Data integrity
  - Non-Repudiation (MAC does not provide this.)

3

Adversarial Goals

- **Total break**: adversary is able to find the secret for signing, so he can forge then any signature on any message.
- **Selective forgery**: adversary is able to create valid signatures on a message chosen by someone else, with a significant probability.
- **Existential forgery**: adversary can create a pair (message, signature), s.t. the signature of the message is valid.

- A signature scheme can not be perfectly secure; it can only be computationally secure.
- Given enough time and adversary can always forge Alice's signature on any message.

4

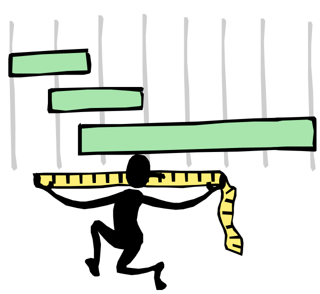
Attack Models for Digital Signatures

- **Key-only attack**: Adversary knows only the verification function (which is supposed to be public).
- **Known message attack**: Adversary knows a list of messages previously signed by Alice.
- **Chosen message attack**: Adversary can choose what messages wants Alice to sign, and he knows both the messages and the corresponding signatures.

5

Digital Signatures and Hash

- Very often digital signatures are used with hash functions, hash of a message is signed, instead of the message.
- Hash function must be:
  - Pre-image resistant
  - Weak collision resistant
  - Strong collision resistant



6

UTD

RSA Signature

Key generation (as in RSA encryption):

- Select 2 large prime numbers of about the same size, p and q
- Compute $n = pq$, and $\Phi = (q - 1)(p - 1)$
- Select a random integer e, $1 < e < \Phi$, s.t. $gcd(e, \Phi) = 1$
- Compute d, $1 < d < \Phi$ s.t. $ed \equiv 1 \mod \Phi$

Public key: (e, n)

Secret key: d, p and q must also remain secret

7

UTD

RSA Signature (cont.)

Signing message M

- M must verify $0 < M < n$
- Use private key (d)
- compute $S = M^d \mod n$

Verifying signature S

- Use public key (e, n)
- Compute $S^e \mod n = (M^d \mod n)^e \mod n = M$

Note: in practice, a hash of the message is signed and not the message itself.

8

RSA Signature (cont.)

Example of forging

- Attack based on the multiplicative property of property of RSA.

  $y_1 = sig_K(x_1)$
  $y_2 = sig_K(x_2)$, then
  $ver_K(x_1x_2 \bmod n, y_1y_2 \bmod n) = true$

- So adversary can create the valid signature

  $y_1y_2 \bmod n$ on the message $x_1x_2 \bmod n$
- This is an existential forgery using a known message attack.

9

El Gamal Signature

Key Generation (as in ElGamal encryption)

- Generate a large random prime p such that DLP is infeasible in Z_p and a generator α of the multiplicative group $\mathbb{Z}_p$ of the integers modulo p
- Select a random integer *a*, 1≤a ≤ p-2, and compute

  $\beta = \alpha^a \bmod p$
- Public key is (p, α, β)
- Private key is *a*
- Recommended sizes: 1024 bits for p and 160 bits for a.

10

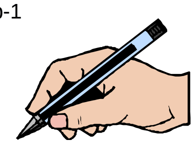
ElGamal Signature (cont.)

Signing message M

- Select random k, $1 \le k \le p-1$, $k \in Z_{p-1}^*$
- Compute

  $r = \alpha^k \bmod p$

  $s = k^{-1}(M - ar) \bmod (p-1)$



- Signature is: (r,s)
- Size of signature is double size of p

NOTE: In practice, instead of M , h(M) is used where h is a hash function

ElGamal Signature (cont.)

Signature is: (r, s)
$r = \alpha^k \bmod p$
$s = k^{-1}(M - ar) \bmod (p-1)$

Verification

- Verify that r is in Z_{p-1}^* : $1 \le r \le p-1$
- Compute

  $v_1 = \beta^r r^s \bmod p$

  $v_2 = \alpha^M \bmod p$

- Accept iff $v_1=v_2$

ElGamal Signature (cont.)

Security of ElGamal signature

- Weaker than DLP
- k must be unique for each message signed
- Hash function h must be used, otherwise easy for an existential forgery attack
  - without h, a signature on $M \in Z_p$, is (r,s) s.t. $\beta^r r^s = \alpha^M \bmod p$
  - choose u,v s.t. gcd(v,p-1)=1, then let $r = \alpha^u \beta^v \bmod p = \alpha^{u+av} \bmod p$, and let $s = -rv^{-1} \bmod (p-1)$
  - then $\beta^r r^s = \alpha^{ar} (\alpha^{u+av})^s = \alpha^{ar} g^{avs} g^{us}$
    $= \alpha^{ar} \alpha^{av(-rv^{-1})} \alpha^{us} = \alpha^{ar} \alpha^{-ar} \alpha^{us} = g^{us}$
  - i.e., (r,s) is a signature of the message u.s

13

ElGamal Signature (Continued)

- 0 < r < p must be checked, otherwise easy to forge a signature on any message if an valid signature is available.
  - given M, and $r = \alpha^k$, $s = k^{-1}(M - ar) \bmod (p-1)$
  - for any message M', let u=M'/M mod (p-1)
  - computes s'=su mod (p-1) and r' s.t.
    $r' \equiv ru \pmod{p-1}$ AND $r' \equiv r \pmod p$, then
    $\beta^{r'} r^{s'} = \beta^{ru} r^{su} = (\beta^r r^s)^u = (\alpha^M)^u = \alpha^{M'}$

14

Digital Signature Algorithm (DSA)

Specified as FIPS 186

Key generation

- Select a prime q of 160-bits
- Choose $0 \leq t \leq 8$
- Select $2^{511+64t} < p < 2^{512+64t}$ with q | p-1
- Let α be a generator of Z_p^*, and
- set $g = \alpha^{(p-1)/q} \bmod p$
- Select $1 \leq a \leq q-1$
- Compute $\beta = g^a \bmod p$

Public key: (p, q, g, β)
Private key: a

15

DSA

Signing message M:

- Select a random integer k, 0 < k < q
- Compute

  $k^{-1} \bmod q$

  $r = (g^k \bmod p) \bmod q$

  $s = k^{-1} (h(M) + ar) \bmod q$
- Signature: (r, s)

Note: FIPS recommends the use of SHA-1 as hash function.

16

DSA

Signature: (r, s)
$r = (g^k \bmod p) \bmod q$
$s = k^{-1} (h(M) + ar) \bmod q$

Verification

- Verify $0 < r < q$ and $0 < s < q$, if not, invalid
- Compute

  $w = s^{-1} \bmod q$

  $u_1 = w \bullet h(m) \bmod q$,

  $u_2 = r \bullet w \bmod q$

  $v = (g^{u_1} \beta^{u_2} \bmod p) \bmod q$
- Valid iff $v = r$

17

Schnorr Signature

Key generation (as in DSA, $h:\{0,1\}^* \rightarrow Z_q$)

- Select a prime q
- Select $1 \leq a \leq q-1$
- Compute $\beta = g^a \bmod p$

Public key: (p,q,g,β)

Private key: a

18

Schnorr Signature

Signing message M

- Select random secret k, $1 \leq k \leq q-1$
- Compute

  $r = g^k \bmod p$,

  $e = h(M \,||\, r)$

  $s = ae + k \bmod q$

- Signature is: (s, e)

19

Schnorr Signature

Verification

- Compute

  $v = g^s \beta^{-e} \bmod p$,

  $e' = h(m \,||\, v)$

- Valid iff $e' = e$

20