Department of Electronics

Cryptography

Fall 2019

Hasan Mahmood

hasan@qau.edu.pk

Week 9 (30 & 31 October 2019)

Digital Signatures

- With the development of electronics commerce and electronics documents, the traditional methods of signature no longer suffice
- Electronics forgery, changing the digitized signatures
- We require digital signature not to be separated from the document
- Cannot be attached to other message(s), the signature is tied only to the signer and message
- Digital signature needs to be easily verified by other parties
- Two distinct steps: the signing process and the verification process
- We are not trying to encrypt the message!

RSA Signature

Bob has a document m that Alice agrees to sign. They do the following:

1. Alice generates two large primes p, q, and computes $n = pq$. She chooses e_A such that $1 < e_A < \phi(n)$ with $\gcd(e_A, \phi(n)) = 1$, and calculates d_A such that $e_A d_A \equiv 1 \pmod{\phi(n)}$. Alice publishes (e_A, n) and keeps private d_A, p, q.

2. Alice's signature is
$$y \equiv m^{d_A} \pmod{n}.$$

3. The pair (m, y) is then made public.

RSA Signature Verification

Bob can then verify that Alice really signed the message by doing the following:

1. Download Alice's (e_A, n).
2. Calculate $z \equiv y^{e_A} \pmod{n}$. If $z = m$, then Bob accepts the signature as valid; otherwise the signature is not valid.

RSA Variant

- Signing of document without knowing its contents
- Suppose Bob has made an important discovery
- He wants to record publicly what he has done

1. Alice chooses an RSA modulus n ($n = pq$, the product of two large primes), an encryption exponent e, and decryption exponent d. She makes n and e public while keeping p, q, d private. In fact, she can erase p, q, d from her computer's memory at the end of the signing procedure.

2. Bob chooses a random integer $k \pmod{n}$ with $\gcd(k, n) = 1$ and computes $t \equiv k^e m \pmod{n}$. He sends t to Alice.

3. Alice signs t by computing $s \equiv t^d \pmod{n}$. She returns s to Bob.

4. Bob computes $s/k \pmod{n}$. This is the signed message m^d.

$$s/k \equiv t^d/k \equiv k^{ed}m^d/k \equiv m^d \pmod{n},$$

Caveats

- Bob could have Alice sign a promise to pay him a million dollars
- Safeguards are need to prevent such problems
- Blind signatures

The ElGamal Signature Scheme

- The ElGamal encryption method can be modified to give a signature scheme
- Many different signatures are valid for a given message
- Suppose Alice wants to sign a message
- She chooses a large prime p and a primitive root α
- Alice chooses a secret integer a such that $1 \leq a \leq p-2$
- Calculates $\beta \equiv \alpha^a \pmod{p}$
- The values of p and α and β are made public
- Security: a is kept private
- Difficult to determine a from (p, α, β), discrete log problem difficult

The ElGamal Signature Scheme, message signature method

1. Selects a secret random k such that $\gcd(k, p-1) = 1$
2. Computes $r \equiv \alpha^k \pmod{p}$
3. Computes $s \equiv k^{-1}(m - ar) \pmod{p-1}$

The signed message is the triple (m, r, s).

Bob can verify the signature as follows:

1. Download Alice's public key (p, α, β).
2. Compute $v_1 \equiv \beta^r r^s \pmod{p}$, and $v_2 \equiv \alpha^m \pmod{p}$.
3. The signature is declared valid if and only if $v_1 \equiv v_2 \pmod{p}$

The ElGamal Signature Scheme Verification

- Verification procedure
- Assume signature is valid
- Since $s \equiv k^{-1}(m - ar) \pmod{p-1}$
- We have: *sk*=*m*-*ar* (mod *p*-1), so *m*=*sk*+*ar* (mod *p*-1)

$$v_2 \equiv \alpha^m \equiv \alpha^{sk+ar} \equiv (\alpha^a)^r(\alpha^k)^s \equiv \beta^r r^s \equiv v_1 \pmod{p}$$