T-79.4501
Cryptography and Data Security

Lecture 9:
- Principles of authentication
- Digital signatures
- DSS

Stallings: Ch. 11.1-2; 13.1; 13.3

1

Principles of message authentication

Attacks against message security:

- Disclosure
- Traffic analysis
- Masquerade (impersonate); this is what a man-in-the-middle does
- Content modification
- Sequence modification
- Timing modification; replay
- Source repudiation
- Destination repudiation

These attacks can be prevented using message authentication

2

Authentication functions

- Authentication functions are cryptographic primitives which are used by message authentication protocols between two parties, sender and receiver. Sender attaches to the message an authenticator. Receiver uses the authenticator to verify authenticity of the message.
- Authentication functions:
  - Message encryption
  - Message authentication code (MAC function)
  - Hash function
  - Digital signature

3

Message Authentication Protocols

Messages are sent from Alice to Bob:

Authenticity requirements:

1. Bob can verify that Alice sent the message
2. Bob can verify that the contents of the message is as it was when Alice sent it.
3. Bob can prove to Carol that Alice sent the message
4. Bob can prove to Carol what the message contents was when Alice sent it.
5. Alice cannot deny that she sent the message.

Requirements 1 and 2 can be fulfilled using protocols based on symmetric key authentication functions.

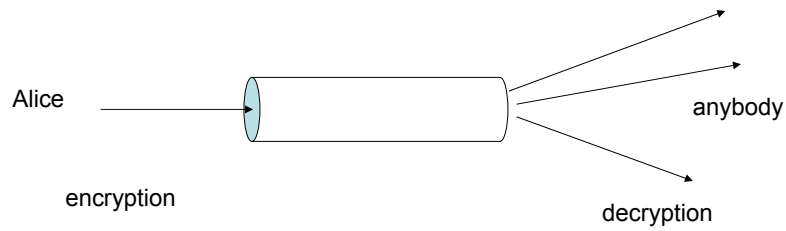
Requirements 3-5 can be fulfilled only using protocols based on asymmetric (public key) cryptosystems: Digital Signatures

4

Asymmetric encryption as authentication function

Encryption operation is private

Decryption is a public operation



Alice's key for a public key cryptosystem is a pair: (K_{pub},K_{priv}) where K_{pub} is public and K_{priv} cannot be used by anybody else than Alice.

5

Digital Signature

Two types

- Digital signature with message recovery: the entire message is encrypted using the private key; before encryption some verifiable redundancy must be added to the message. The message authenticator is the entire ciphertext.
- Digital signature with appendix: First a hash code is computed from the message. Then the hash code encrypted using private key. The encrypted hash code is the authenticator, which is appended to the cleartext message.

6

The RSA Digital Signature

- Key derivation: the same as in RSA encryption:

  $n = pq$, p, q two different primes, e public exponent, d private exponent, $ed \bmod \phi(n) = 1$
- RSA authenticator generation function: given data D the authenticator S of D is computed as $S = D^d \bmod n$
- RSA verification function: given S, the RSA verification function is computed as $S^e \bmod n$
- Hash function: any hash function allowed
- EMSA-PKCS1-v1_5 Formatting of D is specified in PKCS#1 (octet string):

  D = 0x00 || 0x01 || {at least eight times 0xFF } || 0x00 || T ,

where T is the ASN.1 encoding of the hash type and the hash code of the message. || denotes concatenation of octet strings. The number of all-one octets 0xFF in the middle is chosen to adjust the length of D at most equal to the length of the modulus n.

7

The Digital Signature Algorithm DSA

- FIPS 186-2 (2000)
- DSA is a digital signature with appendix
- The complete specification defines:
  - The asymmetric cryptosystem: Key derivation, private key operation (for signature creation), public key operation (for signature verification)
  - Prime number generation
  - The hash function
  - Pseudo-random number generator

8

The DSA public key cryptosystem

Global public key components

p is a 1024-bit prime (old (until 2000): prime number where $2^{L-1} < p < 2^L$, for $512 \le L \le 1024$ and L is a multiple of 64)

q a prime divisor of $p-1$, where q is a 160-bit number

$g = h^{(p-1)/q} \bmod p$, where h is any integer such that $1 < h < p-1$ and $h^{(p-1)/q} \bmod p \neq 1$. (Then the order of the group $\langle g \rangle$ generated by g in Z_p^* is equal to q.)

User's private key

x random or pseudo-random integer with $0 < x < q$

User's public key

$y = g^x \bmod p$

9

DSA: Signature generation

- Message M;
- Hash code H = SHA-1(M) taken as integer
- per-message randomizer k : k is a secret random or pseudorandom integer $0 < k < q$
- The first part r of the signature :

  $r = (g^k \bmod p) \bmod q$
- The second part s of the signature:

  $s = k^{-1} \cdot (H + r \cdot x) \bmod q$ ← Private key used here!
- The signed message is M,(r,s), where (r,s) is the authenticator appended to the message M

10

DSA: Signature verification

Verfier has p, q, g and y.

Verifier receives: M',(r',s') and computes:

H' = SHA-1(M')

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

$u_1 = w \cdot H' \bmod q$

$u_2 = w \cdot r' \bmod q$

$v = g^{u_1} y^{u_2} \bmod p$ ← Public key used here!

and checks if $v \equiv r' \pmod q$.

11