CHAPTER 20

Public-Key Digital Signature Algorithms

Digital Signature Algorithm(DSA)

- In August 1991, The National Institute of Standards and Technology (NIST) proposed the Digital Signature Algorithm (DSA) for use in their Digital Signature Standard (DSS)

- In all, NIST received 109 comments by the end of the first comment period on February 28, 1992.
- Lets look at the criticisms against DSA one by one.

1. DSA cannot be used for encryption or key distribution.

2. DSA was developed by the NSA, and there may be trapdoor in the algorithm.

DSA

3. DSA is slower than RSA.
4. RSA is a de facto standard.
5. The DSA selection process is not public; sufficient time for analysis has not been provided.
6. DSA may infringe no other patents.
7. The key size is too small.

Description of DSA

- DSA is a variant of the Schnorr and ElGamal Signature Algorithms. The algorithm uses the following parameters:

1. p = a prime number L bits long, where L ranges from 512 to 1024 and is a multiple of 64. (In the original standard, the size of p was fixed at 512 bits).

2. q = a 160-bit prime factor of $p-1$.

3. $g = h^{(p-1)/q} \bmod p$, where h is any number less than $p-1$ such that $h^{(p-1)/q} \bmod p$ is greater than 1.

4. x = a number less than q.

5. $y = g^x \bmod p$.

Description of DSA Continue…

- The algorithm also makes use of a one way hash function: H(m).
- The parameters p, q and g are public and can be common across the network of users.
- The private key is x; the public key is y.

- To sign a message m:

1. Alice generates a random number k, less than q.
2. Alice generates:
   a. $r = (g^k \bmod p) \bmod q$.
   b. $s = (k^{-1}(H(m)+xr)) \bmod q$.

   The parameters r and s are her signature; she sends these to Bob.

3. Bob verifies the signature by computing.
   a. $w = s^{-1} \bmod q$.
   b. $u_1 = (H(m)* w) \bmod q$.
   c. $u_2 = (rw) \bmod q$.
   d. $v = ((g^{u1}*y^{u2}) \bmod p) \bmod q$.

   If v = r, then the signature is verified.