Lecture notes for 2.7-2.8 public key crypto

[Hand back homework 3]

RSA
- Most? popular public key crypto system
- K length: variable ~512 to ~2048 bits
- P length: variable, must be smaller than K
- C length: same as P
- Theoretical foundation: factoring a big number is hard

Math

- Euler's totient function
  - $\phi(n)$ = # integers < n relatively prime to n
  - If p, q prime, then $\phi(n) = (p-1)(q-1)$
- Modular exponentiation
  - Inverse of x: y s.t. $\forall m : m^{xy} = m \bmod n$
    - This is the core of RSA
  - Square-free: n is square-free if it has no factor p^2, p prime
  - Lemma 1: If n is square-free, then $x^y \bmod n = x^{(y \bmod \phi(n))} \bmod n$
    ==> if $y = 1 \bmod \phi(n)$, then $x^y \bmod n = x \bmod n$

RSA key pairs

- Pick large primes p, q (~256 bits each)
- Let $n = pq$
- Choose e relatively prime to $\phi(n) = (p-1)(q-1)$
  - Relatively prime so has a mult inverse
  - Common value of e: 65537
  - <e, n> is public key
- Let d be s.t. $de = 1 \bmod \phi(n)$
  - Note that $\forall m$: $m^{de} = m^{ed} \bmod n$
    $= m^{(ed) \bmod \phi(n)} \bmod n$
    $= m \bmod n$ [by Lemma 1]
    $= m$ [since $m < n$]
  - <d, n> is public key
- p, q can be discarded (must NOT be shared!)

RSA operation

- Encryption: $c = m^e \bmod n$
- Decryption: $m = c^d \bmod n$
- Signing: $s = m^d \bmod n$
- Verification: $m = s^e \bmod n$

Public key distribution problem

- Secure distribution of public keys
- Public key infrastructure (PKI)
  - SSL, PGP

Certificate

- Signed message asserting <name, pubkey> match
- Signer must be trusted

Root of trust

- Public key that user fundamentally trusts
- Distributed via out-of-band channel (OS updates to root certs)
- “Certificate authority” (CA)

Certificate chain

- Series of certificates from root to target
- Delegation of trust
- Draw vertically:
  - root: e_c --> [<u_1, e_{u_1}>]d_c
    --> [<u_2, e_{u_2}>]d_{u_1}
    --> ...
    --> target: [<u_n, e_{u_n}>]d_{u_{n-1}}
- Research systems: SPKI SDSI

Who are roots of trust?

- SSL: many roots of trust
  - Manufacturer, not user, decides roots of trust
  - Opportunities for malicious insiders
- PGP: every user is their own root of trust
  - User decides who they certify
  - Transitivity: chain happens via sequence of user certificates
  - Do you really trust your friend's friends' friends?

Certificate revocation

- Cancel the trust assertion
  - Stolen key, fired employee
- Expiration date
- Certificate revocation lists
- Online revocation servers

Certificate formats

- Specification of fields & their meanings
- Algorithms for path verification
- Standards for revocation
- X.509: SSL, TLS, S/MIME (email), Ipsec, SSH, LDAP