Digital Signature and Secret Sharing

Dawn Song

dawnsong@cs.berkeley.edu

Review

- Hash functions
- Message authentication codes (MACs)
  - What security property is it designed to provide?
- Digital signatures
  - What security property is it designed to provide?

Today

- Sample constructions of digital signatures
- Secret sharing schemes
- Questionnaire

One-time Signature

- Lamport, 1979
- Let h be a cryptographic hash function
- To sign a n-bit document m_0, …, m_n, Alice picks
  - Private key: $x_{i,0}$, $x_{i,1}$
  - Public key: $y_{i,0} = h(x_{i,0})$, $y_{i,1} = h(x_{i,1})$
  - Signature: $s_i = x_{i,0}$ if $m_i = 0$;
    $x_{i,1}$ if $m_i = 1$
- How to verify?
- What's the security of this scheme?
  - How many messages can Alice sign with the same public key

RSA Signature

- Idea:
  - Let p, q be large secret primes, N = pq
  - Given e, find d, such that ed ≡ 1 mod $\phi(N)$, where $\phi(N)=(p-1)(q-1)$
  - public key: e, N
  - private key: d, p, q
  - Signature: $s = h(m)^d \bmod N$
  - Verification: $s^e \mathrel{?=} h(m) \bmod N$
- What if h is not collision-resistant?
- In practice, RSA-PKCS (public-key cryptography standards)

ElGamal Signatures & DSA (I)

- RSA signing: similar to "encryption with a private key"
- ElGamal signing is different
  - Relates to zero-knowledge proofs (later in class)
- Set up: Let
  - p be a large prime
  - g be an integer of order p-1 mod p
  - a be private key, public key $y = g^a$
- To sign m, Alice
  - picks a random number k, s.t. gcd(k, p-1) = 1
  - Computes $r = g^k \bmod p$
  - Solves s such that $a*r + k*s \equiv m \bmod p-1$
  - Signature = (r,s)

ElGamal Signatures & DSA (II)

- Recall: a be private key, public key $y = g^a$
- To sign m, Alice
  - picks a random number k, s.t. gcd(k, p-1) = 1
  - Computes $r = g^k \bmod p$
  - Solves s such that $a*r + k*s \equiv m \bmod p-1$
  - Signature = (r,s)
- How to verify?
  - $y^r\ r^s\ ?= g^m \bmod p$
- What is the security of the scheme?
  - Homework 2
- In practice, Digital Signature Algorithm (DSA)

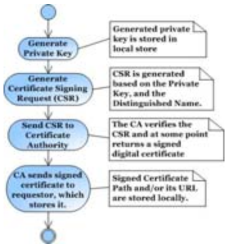
Administrative Matters

- Homework 1 due
- Homework 2 out
- Everyone should have gotten class accounts by now
- Group signup is done
  - Anyone who still has issues should come see me after class
- svn will be set up next week

2-minute Break

How do we know a public key?

- **One approach – the big directory (white pages)**
  - **Need to make secure big directory**
  - **Need to keep it updated**
- **Better approach: allow one party to attest to another**
  - **Public key infrastructure (PKI)**
  - **Public key certificate (PKC)**
  - **Certificate authority (CA)**


Generate Private Key
Generated private key is stored in local store
Generate Certificate Signing Request (CSR)
CSR is generated based on the Private Key, and the Distinguished Name.
Send CSR to Certificate Authority
The CA verifies the CSR and at some point returns a signed digital certificate
CA sends signed certificate to requestor, which stores it.
Signed Certificate Path and/or its URL are stored locally.

A hypothetical public-key hierarchy

Rusty Sears' public key is …
Love, Arnold Schwarzenegger

Digitally signed by AS

A hypothetical public-key hierarchy

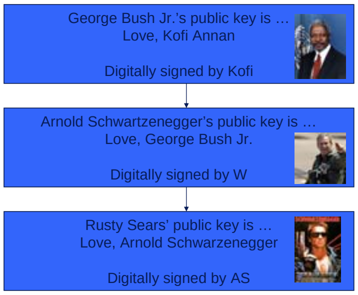
Arnold Schwartzenegger's public key is …
Love, George Bush Jr.

Digitally signed by W

Rusty Sears' public key is …
Love, Arnold Schwarzenegger

Digitally signed by AS

A hypothetical public-key hierarchy

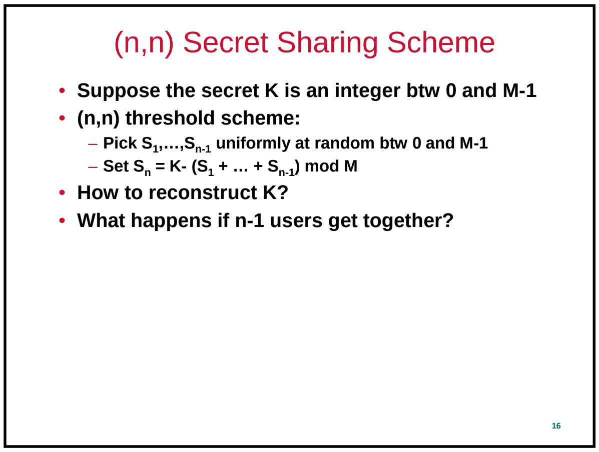

George Bush Jr.'s public key is …
Love, Kofi Annan
Digitally signed by Kofi
Arnold Schwartzenegger's public key is …
Love, George Bush Jr.
Digitally signed by W
Rusty Sears' public key is …
Love, Arnold Schwarzenegger
Digitally signed by AS

Replay attacks

- **Cryptosystems are vulnerable to replay attacks**
- **Record message; playback later identically**
  - **"Yes"/"No"**
- **Solution: use nonces (random bits; timestamp) etc.**
  - **Freshness property**
- **Message is <text, timestamp>**

Secret Sharing

- **A trusted authority TA has a secret K**
- **Wants to split K into n shares S1, …, Sn, distributing to n users U1,…,Un respectively, s.t.**
  - **A reconstruction algorithm can be used to efficiently reconstruct K from any t of the n shares**
  - **Any t-1 of the n shares reveal no information about K**
- **Such a scheme is called an (n,t) threshold secret sharing scheme**



(n,n) Secret Sharing Scheme

- Suppose the secret K is an integer btw 0 and M-1
- (n,n) threshold scheme:
  - Pick $S_1,\ldots,S_{n-1}$ uniformly at random btw 0 and M-1
  - Set S_n = K- (S_1 + … + S_{n-1}) mod M
- How to reconstruct K?
- What happens if n-1 users get together?