6.857 Computer and network Security

Lecture 15

Today:

- Digital signatures
- Security definition for digital signatures
- Hash and sign
- RSA-PKCS
- RSA-PSS
- El Gamal digital signatures
- DSA-(NIST standard)

TOPIC: | DATE:
FILE UNDER: | PAGE: L16.11

Digital Signatures (compare "electronic signature", "cryptographic signature")

- Invented by Diffie & Hellman in 1976 ("New Directions in Cryptography")
- First implementation: RSA (1977)
- Initial idea: switch PK/SK
  (enc with secret key ⇒ signature)
  (if PK decrypts it & looks ok then sig ok ??)

Current way of describing digital signatures

- $\mathrm{Keygen}(1^\lambda) \longrightarrow (PK, SK)$
  verification key → PK ; signing key → SK
- $\mathrm{Sign}(SK, m) \rightarrow \underbrace{\sigma_{SK}(m)}_{\text{signature}}$ [may be randomized]
- $\mathrm{Verify}(PK, m, \sigma) = \text{True}/\text{False}$ (accept/reject)

Correctness:

$$(\forall m)\ \mathrm{Verify}(PK, m, \mathrm{Sign}(SK, m)) = \text{True}$$

TOPIC:

DATE:

FILE UNDER:

Security of digital signature scheme:

Def: (weak) existential unforgeability under adaptive chosen message attack.

① Challenger obtains (PK, SK) from Keygen(1^λ)
Challenger sends PK to Adversary

② Adversary obtains signatures to a sequence

$$m_1, m_2, \ldots, m_q$$

of messages of his choice. Here $q = \text{poly}(\lambda)$, and m_i may depend on signatures to $m_1, m_2, \ldots, m_{i-1}$.

Let $\sigma_i = \text{Sign}(SK, m_i)$.

③ Adversary outputs pair (m, σ_*)

Adversary wins if $\text{Verify}(PK, m, \sigma_*) = \text{True}$ and $m \notin \{m_1, m_2, \ldots, m_q\}$

Scheme is secure (i.e. weakly existentially unforgeable under adaptive chosen message attack) if

$$\text{Prob}[\text{Adv wins}] = \text{negligible}$$

Scheme is <u>strongly</u> secure if adversary can't even produce new signature for a message that was previously signed for him.

I.e. Adv wins if $\text{Verify}(PK, m, \sigma_*) = \text{True}$ and $(m, \sigma_*) \notin \{(m_1, \sigma_1), (m_2, \sigma_2), \ldots, (m_q, \sigma_q)\}$.

Digital signatures

- Def of digital signature scheme
- Def of weak/strong existential unforgeability under adaptive chosen message attack.

see notes from last lecture

Hash & Sign:

For efficiency reasons, usually best to sign cryptographic hash $h(M)$ of message, rather than signing M. Modular exponentiations are slow compared to (say) SHA-256.

Hash function h should be collision-resistant.

Signing with RSA — PKCS

- PKCS = "Public key cryptography standard"
  (early industry standard)
- Hash & sign method. Let H be C.R. hash fn.
- Given message M to sign:

  Let $m = H(M)$

  Define $pad(m) =$

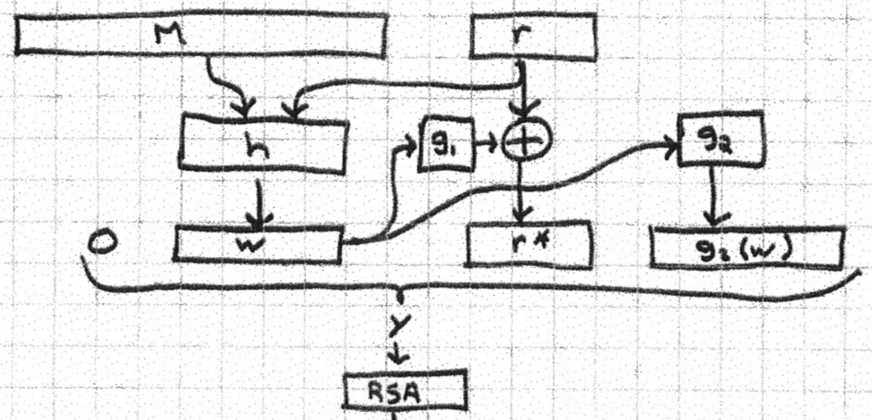
  $$0x\ 00\ 01\ FF\ FF \ldots FF\ 00 \,\|\, \text{hash-name} \,\|\, m$$

  where # FF bytes enough to make $|pad(m)| = |n|$ in bytes.

  where hash-name is given in ASN.1 syntax (ugly!)
- Seems secure, but no proofs (even assuming H is CR and RSA is hard to invert)
- $\sigma(M) = (pad(m))^d \pmod n$

PSS - Probabilistic Signature Scheme [Bellare & Rogaway 1996]

- RSA-based
- "Probabilistic" ≡ randomized [one M has many sigs]



$$\sigma(m) = y^d \pmod n$$

Sign(M): $r \xleftarrow{R} \{0,1\}^{k_0}$

$w \leftarrow h(M \| r)$ $\qquad |w| = k_1$

$r^* \leftarrow g_1(w) \oplus r$ $\qquad |r^*| = k_0$

$y \leftarrow 0 \| w \| r^* \| g_2(w)$ $\qquad |y| = |n|$

output $\sigma(M) = y^d \pmod n$

Verify(M, σ): $y \leftarrow \sigma^e \pmod n$

Parse y as $b \| w \| r^* \| \gamma$

$r \leftarrow r^* \oplus g_1(w)$

return True iff $b = 0$ & $h(M \| r) = w$ & $g_2(w) = \gamma$

- We can model h, g_1, and g_2 as random oracles.

Theorem:

PSS is (weakly) existentially unforgeable against a chosen message attack in random oracle model if RSA is not invertible on random inputs.

El Gamal digital signatures

Public system parameters: prime p

generator g of $\mathbb{Z}_p^*$

Keygen: $x \xleftarrow{R} \{0,1,\ldots,p-2\}$ $\quad SK = x$

$y = g^x \pmod p$ $\quad PK = y$

Sign(M):

$m = \text{hash}(M)$ $\quad$ CR hash fn into $\mathbb{Z}_{p-1}$

$k \xleftarrow{R} \mathbb{Z}_{p-1}^*$ $\quad$ [$\gcd(k, p-1) = 1$]

$r = g^k$ $\quad$ [hard work is indep of M]

$s = \dfrac{(m - rx)}{k} \pmod{p-1}$

$\sigma(M) = (r, s)$

Verify(M, y, (r, s)):

Check that $0 < r < p$ $\quad$ (else reject)

Check that $y^r r^s = g^m \pmod p$

where $m = \text{hash}(M)$

Correctness of El Gamal signatures:

$$y^r r^s = g^{rx} g^{sk} = \underbrace{g^{rx+sk} \stackrel{?}{=} g^m \pmod{p}}_{\equiv}$$

$$rx + ks \stackrel{?}{=} m \pmod{p-1}$$

or $$s \stackrel{?}{=} \frac{(m - rx)}{k} \pmod{p-1}$$

(assuming $k \in \mathbb{Z}_{p-1}^{*}$) $\blacksquare$

Theorem: El Gamal signatures are existentially **forgeable** (without h, or h = identity (note: this is CR!))

Proof: Let $e \xleftarrow{R} Z_{p-1}$

$$r \leftarrow g^e \cdot y \pmod p$$

$$s \leftarrow -r \pmod p$$

Then (r,s) is valid El Gamal sig. for message $m = e \cdot s \pmod{p-1}$.

Check: $y^r r^s \stackrel{?}{=} g^m$

$$g^{xr}(g^e y)^{-r} = g^{-er} = g^{es} = g^m \checkmark \qquad \blacksquare$$

But: It is easy to fix.

Modified El Gamal (Pointcheval & Stern 1996)

Sign (M): $k \xleftarrow{R} Z_p^*$

$$r = g^k \pmod p$$

$$m = h(M \| r) \quad \Longleftarrow ***$$

$$s = (m - rx)/k \pmod{p-1}$$

$$\sigma(M) = (r,s)$$

Verify: check $0 < r < p$ & $y^r r^s = g^m$ where $m = h(M \| r)$.

Theorem: Modified El Gamal is existentially unforgeable against adaptive chosen message attack, in ROM, assuming DLP is hard.

TOPIC:

FILE UNDER:

DATE:

Digital Signature Standard (DSS - NIST 1991)

Public parameters (same for everyone):

q prime $\qquad |q| = 160$ bits

$p = nq+1$ prime $\qquad |p| = 1024$ bits

g_0 generates $\mathbb{Z}_p^*$

$g = g_0^n$ generates subgroup G_q of $\mathbb{Z}_p^*$ of order q

Keygen:

$x \xleftarrow{R} \mathbb{Z}_q \qquad$ SK $\qquad |x| = 160$ bits

$y \leftarrow g^x \pmod p \qquad$ PK $\qquad |y| = 1024$ bits

Sign (m):

Note: if k is reused for different messages m, one could solve for x so it is not secure. If k is reused for the same m, we obtain the same signature so this is not a problem. If k is different for the same m, it should be random and unknown (any known relation between the two k-s allows to solve for x)

Bottomline: All of the above are enforced by k chosen at random from Z_q^* for large enough q

$k \xleftarrow{R} \mathbb{Z}_q^* \qquad$ (i.e. $1 \le k < q$)

$r = (g^k \bmod p) \pmod q \qquad |r| = 160$ bits

$m = h(M)$

$s = (m + rx)/k \pmod q \qquad |s| = 160$ bits

redo if $r=0$ or $s=0$

$\sigma(M) = (r, s)$

TOPIC

FILE UNDER

DATE

Verify:

Check $0 < r < q$ & $0 < s < q$

Check $y^{r/s} g^{m/s} \pmod p \pmod q = r$

where $m = h(M)$

Correctness:

$$g^{(rx+m)/s} \stackrel{?}{=} r \pmod p \pmod q$$

$$\equiv \quad g^k = r \pmod p \pmod q \quad \checkmark$$

As it stands, existentially forgeable for h = identity.

Provably secure (as with Modified El Gamal)

if we replace $m = h(M)$ by $m = h(M \| r)$, as before.

Note: As with El Gamal, secrecy & uniquness of k

is essential to security.