6.857 Rivest
3/16/11 L13.1

Admin:

Outline:

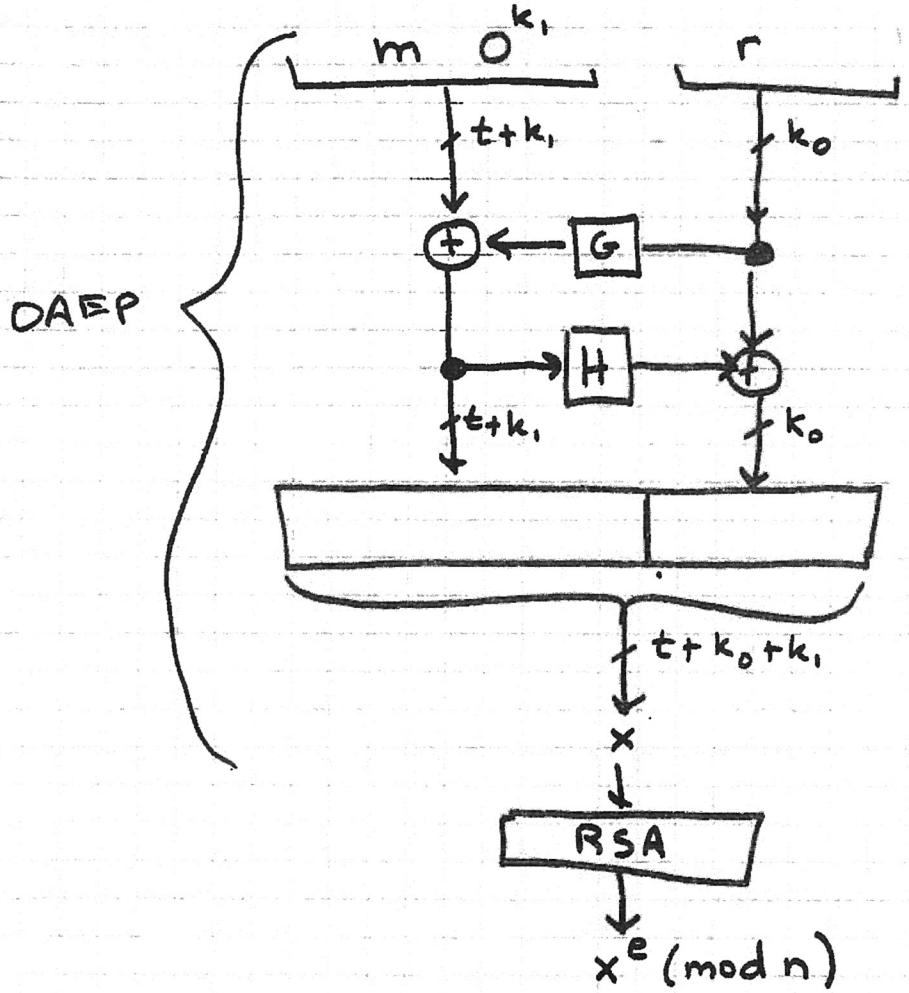
- RSA-OAEP
- Digital Signatures
  - def
  - security def
  - RSA-PSS
  - El Gamal
  - DSS } ← didn't do

How to make RSA IND-CCA2 secure?

"OAEP" = Optimal asymmetric encryption padding [BR 94]

Given m, $|m| = t$ bits

Pick r at random, $|r| = k_0$



$G: \{0,1\}^{k_0} \to \{0,1\}^{t+k_1}$

$H: \{0,1\}^{t+k_1} \to \{0,1\}^{k_0}$

G, H: "random oracles"

like UFE (!) of Desai

On decryption: invert RSA
invert OAEP
reject if 0^{k_1} not present

Thm: RSA with OAEP secure against ACCA, assuming RO model & that RSA hard to invert on random inputs. } bug in proof, but one with slightly different assumptions... (or OAEP+)

OAEP: used in practice

theory: (we don't have random oracles...)

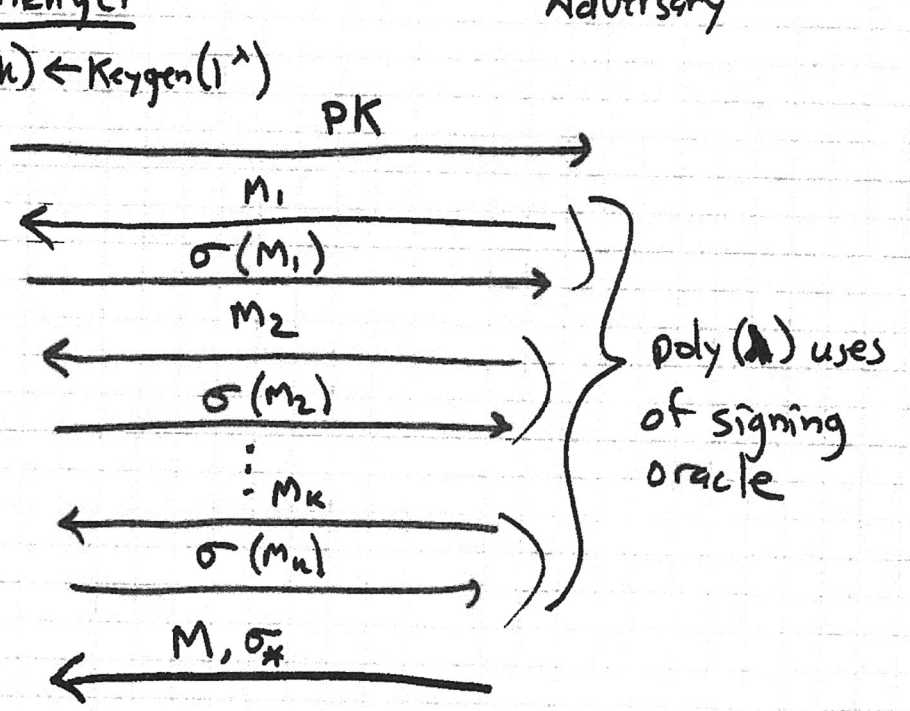
Digital Signatures

- Invented by Diffie/Hellman in 1976 (New Directions)
- First implementation: RSA (1977) [key motivation for me for PK... !]
- Initial idea: switch PK/SK - enc with secret key ⇒ sig
  - if PK decrypts it - then sig ok
- Current way of describing digital signatures
  - (Note: law is confused (includes hashes, MACs, etc...) - ignore it
  - $\underline{\text{Keygen}}(1^\lambda) \to (PK, SK)$ — PK: verification key; SK: signing key
    λ = "security parameter" all lengths are polynomial in security may be negligible fn of λ.
  - ignore for now "PKI" issue: knowing that you have "right" PK
  - $\underline{\text{Sign}}(SK, M) \longrightarrow \sigma_{SK}(M)$ (may be randomized)
    $M \in \{0,1\}^*$
  - $\underline{\text{Verify}}(PK, M, \sigma) = \text{True/False}$

$\underline{\text{Correctness:}}$ $(\forall M)\ \underline{\text{Verify}}(PK, M, \underline{\text{Sign}}(SK, M)) = \text{True}$

Security: (Weak) existential unforgeability under adaptive chosen message attack:

Game:



Adv wins if $\text{Verify}(PK, M, \sigma^*) = \text{True}$
& $M \notin \{M_1, \ldots, M_n\}$

Scheme is secure (ie. weakly existentially unforgeable against adaptive chosen message attack)
if Prob[Adv wins] is negligible (i.e. $\leq 1/\lambda^c$ for all c & all sufficiently large λ)

Scheme is **strongly** secure if adversary can't even produce new sig for ~~previous~~ message previously signed
I.e. Adv wins if $\text{Verify}(PK, M, \sigma^*) = \text{True}$
& $(M, \sigma^*) \notin \{(M_1, \sigma_1), (M_2, \sigma_2), \ldots, (M_n, \sigma_n)\}$

Signing with RSA

(1) Hash & sign with PKCS

Let [redacted] $H(M) = SHA256(M)$

Let $H'(M) = 0x\ 00\ 01\ FF\ FF \ldots FF\ 00 \;\|\; $ [redacted] (name of hash) ASN.1 $\| H(M)$

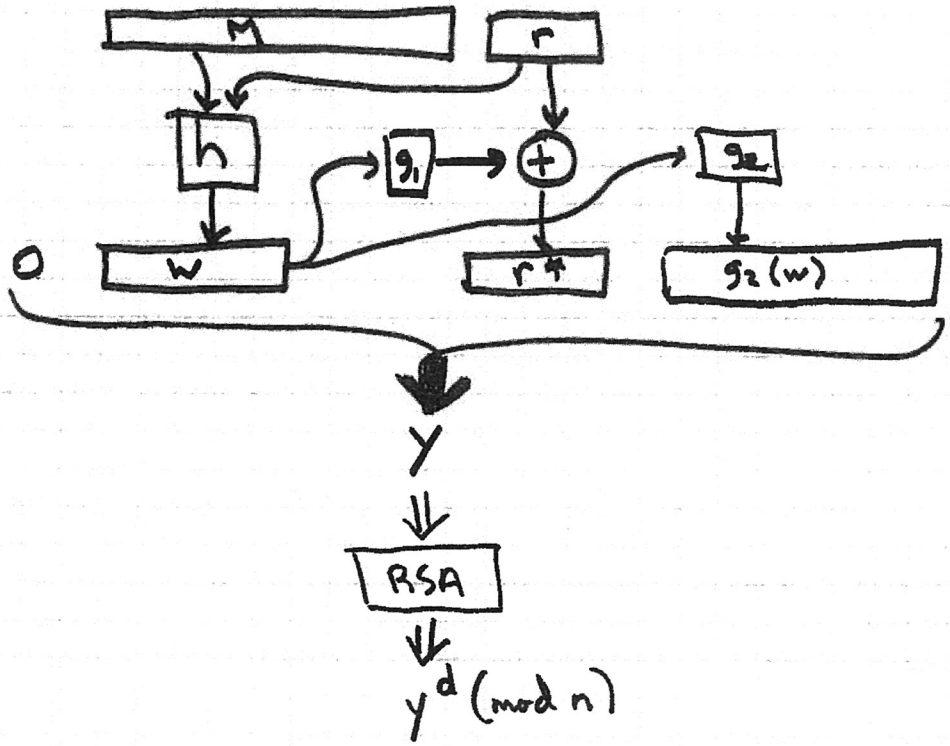
$\sigma(M) = (H'(M))^d \bmod n$

Some problems with $e=3$ (bad implementations can find 0, parse ASN.1, take $H(M)$, miss other stuff after $H(M)$)

Otherwise seems OK, but no proofs. (even assuming collision resistance & RSA hard to invert...)

commonly used, nonetheless...

(2) PSS [Bellare & Rogaway 1996]



Sign(M):

$r \xleftarrow{R} \{0,1\}^{k_0}$

$w \leftarrow h(M \| r)$ ⟸ note! (compare with El Gamal later)

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

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

return $y^d \pmod n$

$|w| = k_1$

$|r^*| = k_0$

$|y| = k = |n|$

Verify(M, x):

$y \leftarrow x^e \pmod n$

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

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

if $h(M \| r) = w$ & $g_2(w) = \gamma$ & $b = 0$

return True

else return False

Theorem: PSS is (weakly) ~~secure~~ existentially unforgeable against chosen message attack in ROM if RSA is not invertible on random inputs.

($\neg\exists$ Adv who can produce x^d given x.)

El Gamal signatures

Public system parameters p prime
g generator

Keygen: $x \in_R \{0,1,\ldots,p-2\}$ $\quad SK = x$
$y = g^x$ $\quad PK = y$

Sign(M): $m = h(M)$
$k \in_R Z^*_{p-1}$ $\quad [\gcd(k, p-1) = 1]$
randomized signing
$r = g^k$ $\quad$ [hard work is indep of M]

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

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

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

Return True if both checks pass else return False

Correctness: $g^{rx} g^{sk} = g^{rx+sk} \stackrel{?}{=} g^m \pmod p$

$\equiv$

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

$\equiv$

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

(if $\gcd(k, p-1) = 1$)

[El Gamal signatures, cont.d]

That was original version.

Theorem: El Gamal is existentially forgeable (without h fn or h = identity)
↑ note: CR!

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

$$r \leftarrow g^e y \pmod p$$
$$s \leftarrow -r \pmod{p-1}$$

(r,s) is sig for message $m = es \pmod{p-1}$

$$y^r r^s \stackrel{?}{=} g^m$$
$$g^{xr}(g^e y)^{-r} = g^{-er} = g^{es} = g^m \quad \text{for } m = es \pmod{p-1}$$

But: It is easy to fix.

Modified El Gamal (Pointcheval/Stern 1996)

sign(M): $k \in_R Z_p^*$

$$r = g^k \pmod p$$
$$m = h(M \| r) \quad \Leftarrow \text{***}$$
$$s = \frac{(m - rx)}{k} \pmod{p-1}$$

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

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

10/23/06 L12.12

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

Digital Signature Standard (DSS - NIST 1991)

Public parameters: q prime $\quad |q| = 160$ bits

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

g_0 generates Z_p^*

$g = g_0^n$ generates G_q - subgroup of Z_p^* of order q

Keygen: $x \in_R Z_q$ $\quad$ SK $\quad |x| = 160$ bits

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

Sign(M): $k \in_R Z_q^*$ $\quad$ (i.e. $1 \le k < q$)

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

$m = h(M)$

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

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

$\sigma(M) = (r, s)$ $\quad |\sigma| = 320$ bits

Verify $(PK, M, (r,s))$

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

where $m = h(M)$

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

$g^k \stackrel{?}{=} r \pmod p \pmod q$ ✓

Security proof works if we had done $m = h(M||r)$, as before.

As it stands, existentially forgeable for h = identity.