TOPIC: 6.857 - Applied Cryptography

DATE: 3/14/22

FILE UNDER:

PAGE: L13

Admin: Projects

Today: RSA encryption

Wed: Digital signatures

Diffie-Hellman model of PK crypto

"New Directions in Cryptography" (Nov. 1976)

- $Gen(1^\lambda) \to (PK, SK, M, C)$

  (public key, secret key, message space, ciphertext space)

  Here $|M| = |C|$

- $Enc(PK, \cdot)$ is an efficient (poly-time) computable map that is one-to-one & deterministic from M to C

  $c = Enc(PK, m)$ is (unique) ciphertext for m

- $Dec(SK, \cdot)$ is efficiently computable inverse

  $Dec(SK, c) = Dec(SK, Enc(PK, m)) = m$
  $(\forall m \in M)$

- It is hard/infeasible to decrypt with knowledge of PK but without knowledge of SK.
  SK represents "trapdoor" information that enables inversion of the (otherwise one-way) function $Enc(PK, \cdot)$.

D&H proposed model, but didn't have implementation!

RSA – invention (Rivest, Shamir, Adleman, 1977)

RSA method:

- Gen (1^λ):
  find two large primes p, q of length λ
  (e.g. $\lambda = 1024$ bits)
  p, q could be randomly chosen to guard against factoring attacks
  $n = p \cdot q$
  $\varphi(n) = |\mathbb{Z}_n^*| = (p-1)(q-1)$
  $e \leftarrow \mathbb{Z}^*_{\varphi(n)}$ e.g. $\gcd(e, \varphi(n)) = 1$
  $e = 65{,}537$ common choice
  $d = e^{-1} \pmod{\varphi(n)}$ [via Euclid's extended alg]
  $PK = (n, e)$
  $SK = (d, p, q)$
  $M = C = \mathbb{Z}_n$ (note: $\mathbb{Z}_n^* \subsetneq \mathbb{Z}_n$!)
- Enc $(PK, m) = m^e \pmod{n}$ $(\forall m \in \mathbb{Z}_n)$
- Dec $(SK, c) = c^d \pmod{n}$ $(\forall c \in \mathbb{Z}_n)$

Correctness of RSA

Lemma: (Chinese remainder theorem or CRT)

Let $n = p \cdot q$ where p, q are distinct primes.

$(\forall x, y \in \mathbb{Z}_n)$

$x = y \pmod n \iff x = y \pmod p \ \& \ x = y \pmod q$ ∎

Thus it suffices to prove RSA correct mod p. Correctness mod q is proved the same way & CRT $\Rightarrow$ correctness mod n.

Given $e \cdot d = 1 \pmod{\varphi(n)}$ since $d = e^{-1} \pmod{\varphi(n)}$

$\Rightarrow e \cdot d = 1 + t \cdot (p-1) \cdot (q-1)$ for some t, over $\mathbb{Z}$

$\Rightarrow e \cdot d = 1 \pmod{(p-1)}$ i.e. $d = e^{-1} \pmod{p-1}$

Correctness of RSA means

$(\forall m \in \mathbb{Z}_n) \ (m^e)^d = m \pmod n$

By CRT we only need to prove

$(\forall m \in \mathbb{Z}_p) \ (m^e)^d = m \pmod p$

We consider 2 cases:

Case 1: $m = 0 \pmod p$

Trivial: $0^{ed} = 0 \pmod p$

Case 2: $m \neq 0 \pmod p$

$\equiv \quad m \in \mathbb{Z}_p^*$

so $m^{p-1} = 1 \pmod p$ [Fermat]

Then $m^{ed} = m^{1+u\cdot(p-1)} \pmod p$

where $u = t\cdot(q-1)$

$m^{ed} = m\cdot(m^{p-1})^u \pmod p$

$= m \cdot 1^u$

$= m$

$\therefore \quad m^{ed} = m \pmod p \quad (\forall m \in \mathbb{Z}_p)$

& $m^{ed} = m \pmod q \quad (\forall m \in \mathbb{Z}_q)$ (similarly)

$\Rightarrow \quad m^{ed} = m \pmod n \quad (\forall m \in \mathbb{Z}_n)$ (CRT)

Thus $(\forall m \in \mathbb{Z}_n)$

$$Dec(SK, Enc(PK, m)) = m \qquad \blacksquare$$

RSA is 1 to 1; $Dec(SK, \cdot)$ is the inverse of $Enc(PK, \cdot)$

Security of RSA

Factoring attack:

If an Adversary can factor n, then the Adversary can compute $\varphi(n)$, and thus compute $d = e^{-1} \pmod{\varphi(n)}$

Key insight: size $\varphi(n)$ of group $\mathbb{Z}_n^*$ is unknown & unknowable to (bounded) Adversary.

How hard is factoring?

- Time $\exp\{c \cdot (\ln n)^{1/3} (\ln \ln n)^{2/3}\}$
- RSA challenge RSA-250 factored 2020 (829 bits)
- RSA keys of ≥ 2048 bits secure for a long time, unless there are algorithmic breakthroughs (including possibility of building a quantum computer).

RSA problem

(Factoring being hard may not be enough)

RSA Problem (or RSA assumption)

Given (n, e) and $c = m^e \pmod{n}$, it is hard to find m.

Is RSA CPA secure?

No. (It is deterministic!)

∴ not CCA secure, either!

How to make RSA CCA secure?

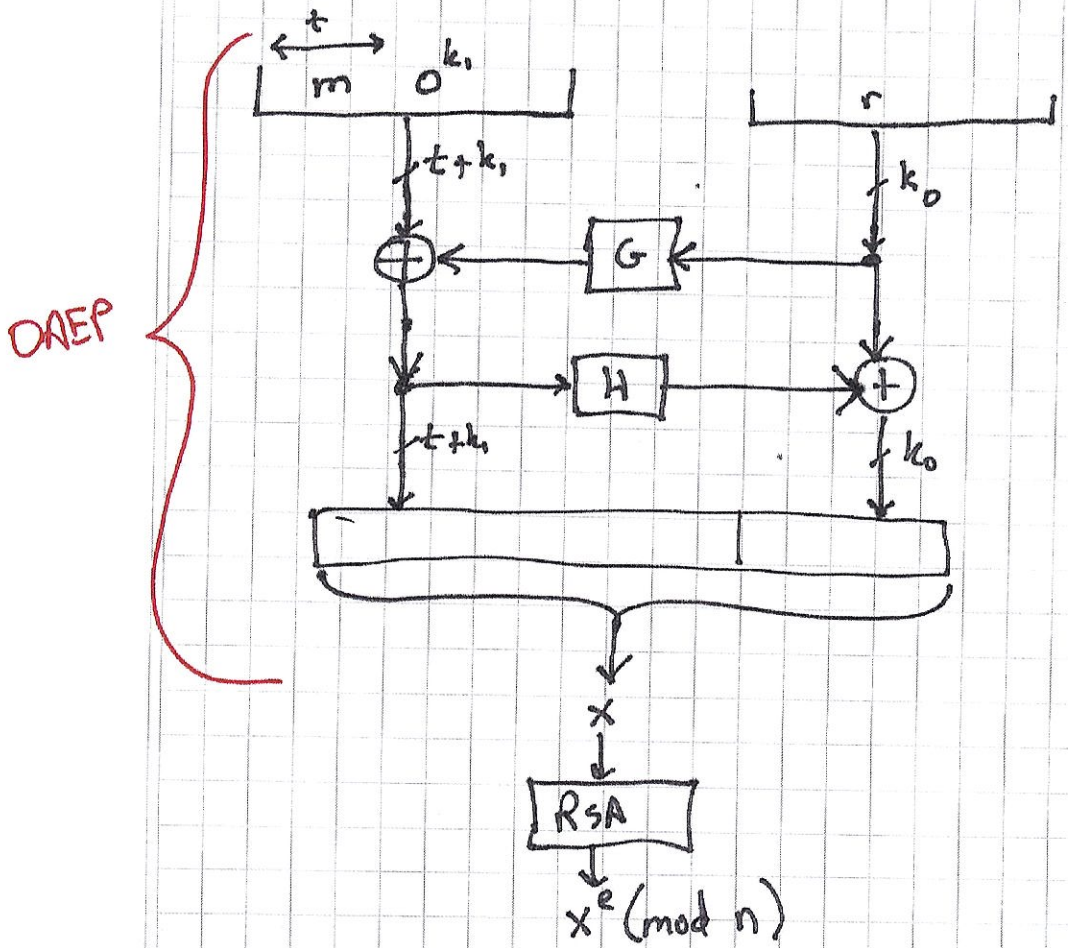
"Pad" message with 0's & randomness, using OAEP (Optimal Asymmetric Encryption Padding), then encrypt.

Decryption rejects if 0's not present.

OAEP:

Let $t = |m|$
Let r be randomness of length k_0
Add k_1 bits of 0s 0^{k_1} (to check)

Assume: $G: \{0,1\}^{k_0} \to \{0,1\}^{t+k_1}$
$H: \{0,1\}^{t+k_1} \to \{0,1\}^{k_0}$
} are RO's



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

Theorem: RSA with OAEP is CCA secure, assuming ROM for G & H, & assuming RSA problem (RSA hard to invert on random inputs)

[Needs slightly modified assumptions, or OAEP+ for general trapdoor permutation, but OK with RSA]

OAEP used in practice.

Other aspects of RSA security

Weak keys

small d (e.g. $< n^{1/4}$) is insecure

Implementation:

Side channels: power

timing

Fault injection: (esp. if CRT is used)

Quantum computing

Peter Shor (MIT) has a polynomial-time factoring algorithm on a quantum computer.