RSA encryption

Greatest common divisor

- $GCD(x, y) := \max_{z}(z \mid x) \ \ and \ (z \mid y)$
  - $a \mid b \ \ := a \ divides \ b$

- $Z_n^* := \{ \ m \ \mid GCD(m, n) = 1, \ m < n \ \}$

Modular inverse

- Let (x,n) be integers such that $GCD(x, y) = 1$

$$\{x^{-1}\ (mod\ n)\} \coloneqq \{y \mid y \cdot x = 1\ (mod\ n), y < n\ \}$$

Powers

- $g^x := g \cdot \ldots \cdot g$ (n times)

- $2^5 mod\ 7 = 2 * 2 * 2 * 2 * 2\ (mod\ 7) = 32\ (mod\ 7) = 4$

Public key-encryption

- How can people send encrypted messages to google, steam, your bank, even though they have never exchanged secret keys with those companies?

- Public-key encryption allows you to do it
  - Public key is revealed publicly so that everyone can encrypt messages
  - Secret key is kept hidden and only the owner is allowed is able to decrypt the ciphertext

Public-key encryption

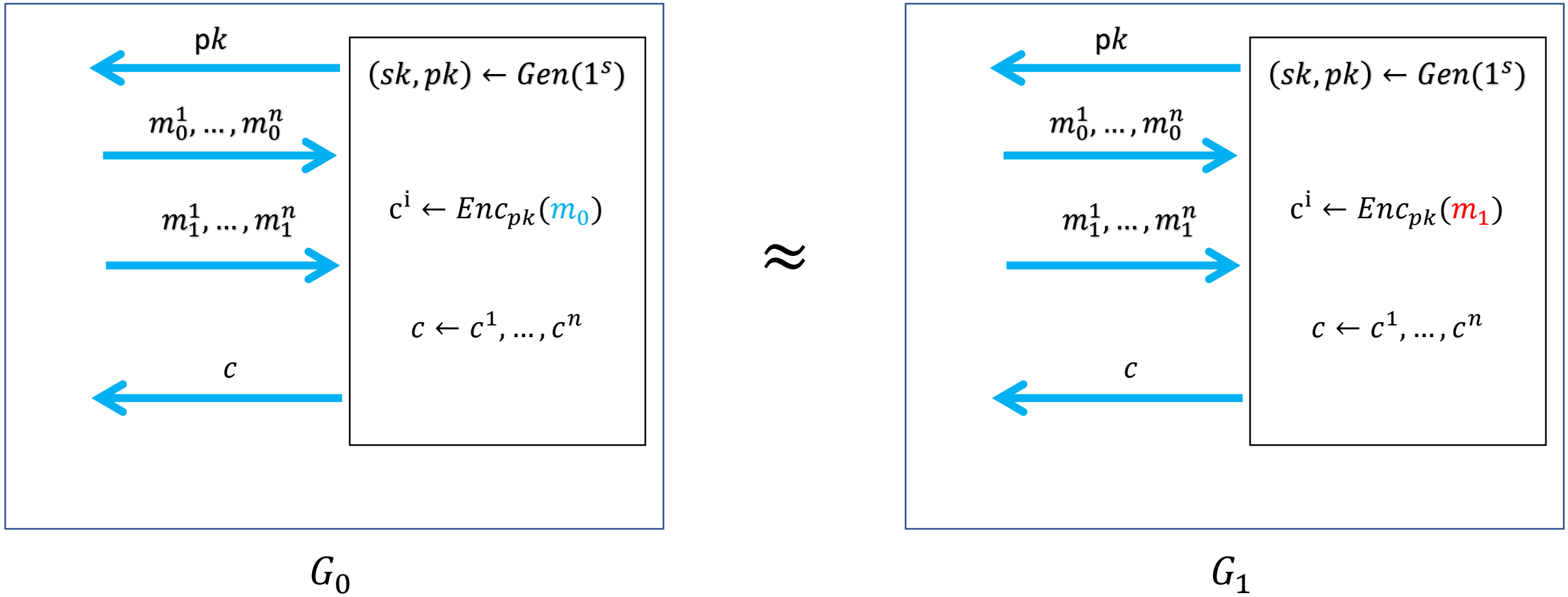
- The Gen algorithm takes security parameter 1^s and outputs both a secret key and a public key
- The encrypt algorithm takes a public key pk and a message m and outputs a ciphertext c
- The decrypt algorithm takes a secret key sk and a ciphertext c and outputs the message m

Formal definition

- $Gen(1^s) \rightarrow (sk, pk)$
- $Enc_{pk}(m) \rightarrow c$ where $m \in M, c \in C$
- $Dec_{sk}(c) \rightarrow m$ where $m \in M, c \in C$
- Correctness:

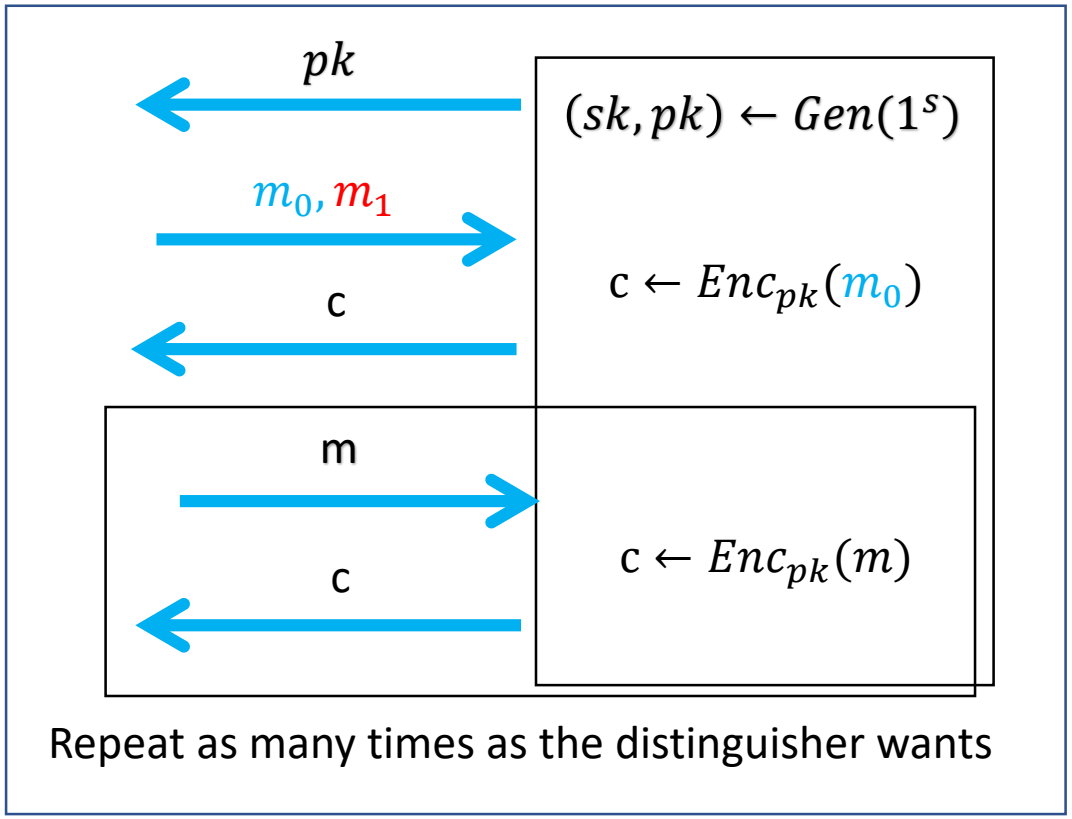
$$\Pr[\mathrm{Dec}_{\mathrm{sk}}\left(Enc_{pk}(m)\right) = m \mid (sk, pk) \leftarrow Gen(1^s)] = 1$$

Multi-message indistinguishability



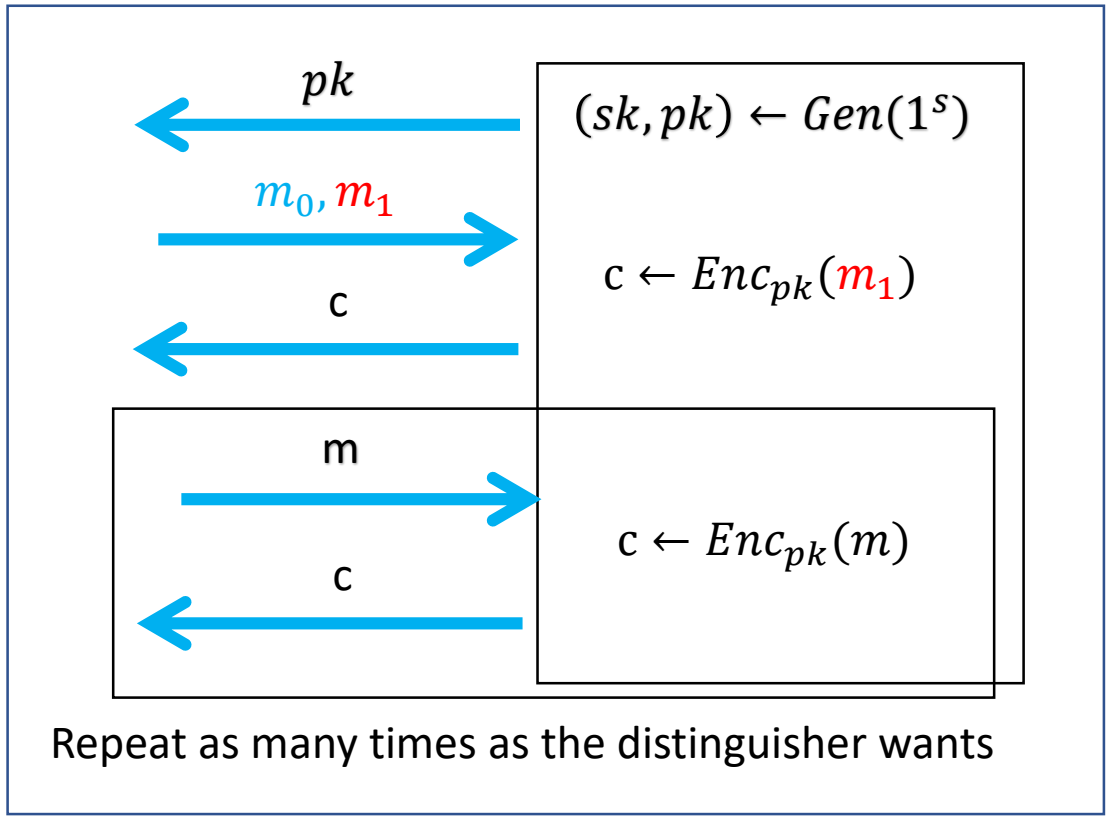
G_0 G_1

Chosen-plaintext security



G_0

$\approx$



G_1

RSA keygen

- Sample at random large primes p, q
  - Security depends on the primes being long

- $N \leftarrow pq$
- $\phi(n) \coloneqq (p-1) \cdot (q-1)$
- $e \in_R \{x \mid GCD(x, \phi(N)) = 1\}$
- $d \leftarrow e^{-1} \, mod \, \phi(n)$
- $PK \coloneqq (N, e)$
- $SK \coloneqq (N, d)$

Example

- $p = 3$
- $q = 5$
- $\phi(n) = (p-1) \cdot (q-1) = 2 \cdot 4 = 8$
- $e = 7$
- $GCD(e, 8) = 1$
- $d = e^{-1} = 7$
- $PK \leftarrow (7,15)$
- $SK \leftarrow (7,15)$

Example

- $p = 3$
- $q = 11$
- $\phi(n) = (p-1) \cdot (q-1) = 2 \cdot 10 = 20$
- $e = 7$
- $GCD(e, 20) = 1$
- $d = e^{-1} = 3$
- $PK \leftarrow (7,33)$
- $SK \leftarrow (3,33)$

Textbook RSA (not secure at all)

- $Enc_{pk}(m)$
  - $(e, N) \leftarrow pk$
  - $c \leftarrow m^e \pmod N$
  - Output c

- $Dec_{sk}(c)$
  - $(d, N) \leftarrow sk$
  - $m \leftarrow c^d \pmod n$
  - Output m

Example

- $p = 3$
- $q = 5$
- $PK \leftarrow (7,15)$
- $SK \leftarrow (7,15)$
- $Enc_{(7,15)}(2) = 2^7 (mod\ 15) = 8\ (mod\ 15)$
- $Dec_{(7,15)}(8) = 8^7\ (mod\ 15) = 2\ (mod\ 15)$

CPA-secure RSA

- $Enc_{pk}(m; r)$
  - $(e, N) \leftarrow pk$
  - $r \in \{r' | \, (r' || m) \in Z_n\}$
  - $\tilde{m} \leftarrow \; r \,||\, m$
  - $c \leftarrow \; \tilde{m}^e \;\; (mod \; N)$
  - Output c

- $Dec_{sk}(c)$
  - $(d, N) \leftarrow pk$
  - $\tilde{m} \leftarrow c^d \; (mod \;\; n)$
  - $r \,||\, m \leftarrow \tilde{m}$
  - Output m

Validation oracles / error oracles

- When encrypting message using public-key encryption, it might be that the website sends you an error if the message is not valid.

- Homomorphic properties of certain encryption schemes

$$Enc_{pk}(m_1) * Enc_{pk}(m_2) = Enc_{pk}(m_1 * m_2)$$

RSA is homomorphic

- $Enc_{pk}(m)$
  - $(e, N) \leftarrow pk$
  - $c \leftarrow\ m^e\ \ (mod\ N)$
  - Output c

- $Enc_{pk}(m_1) \cdot Enc_{pk}(m_2) \coloneqq (m_1)^e \cdot (m_2)^e = (m_1 \cdot m_2)^e$

Validation oracle attack using homomorphism

- $M = \{x \mid x \; mod \; 3 = 0, x < n\} \cup \{x \mid x \; mod \; 3 = 1, x < n\}$

$Dec_{sk}\left(Enc_{pk}(x) * Enc_{pk}(2)\right) \notin M \Rightarrow$
$Dec_{sk}\left(Enc_{pk}(x * 2)\right) \notin M \Rightarrow$
$x \; mod \; 3 = 1$

Exams distribution