Introducción a la Criptografía y a la Seguridad de la Información

Part 6
Public-key Cryptosystems

Jorge E. Camargo, PhD

Agenda

- Public-key Cryptosystems
  - A Postal Analogy
  - RSA
    - Example
  - The Discrete Logarithm Problem
  - ElGamal
    - Example

Public-key Cryptography

The key management is a Public-key Cryptosystem is much simpler:

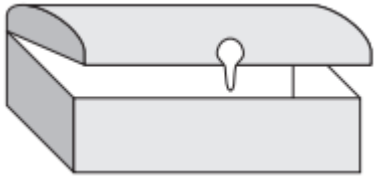
- only decryption key must be kept secret
- encryption key can be published
- computing private key from their corresponding public key is infeasible

Public-key Cryptography is also known as asymmetric cryptography.

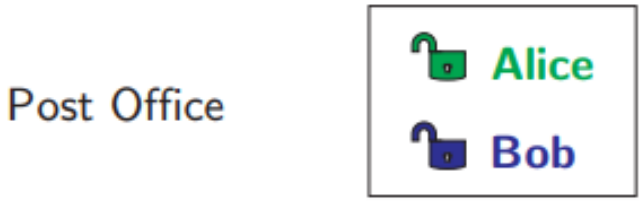
A Postal Analogy

In this example Alice has the secret message and wants to send it to Bob, after which Bob sends a secret reply.

Assume that the message is sent in a box with a clasp ring.



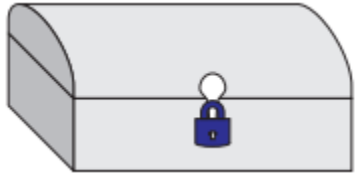
Bob and Alice have separate open padlocks with their names, freely available in a place such as a post office.



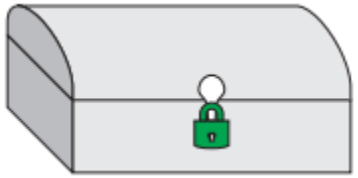
Firstly, Alice gets Bob's open padlock from the postoffice.

A Postal Analogy (cont.)

Alice uses Bob's padlock to lock a box containing her message, and sends the locked box to Bob.



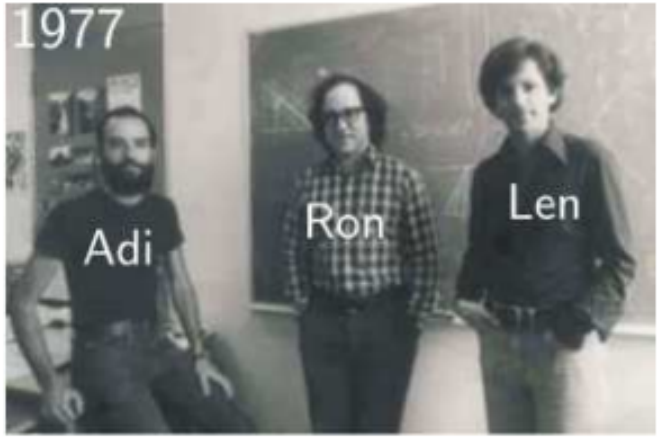
Bob can then unlock the box with his key and read the message. To reply, Bob must similarly get Alice's open padlock to lock the box before sending it back to her.

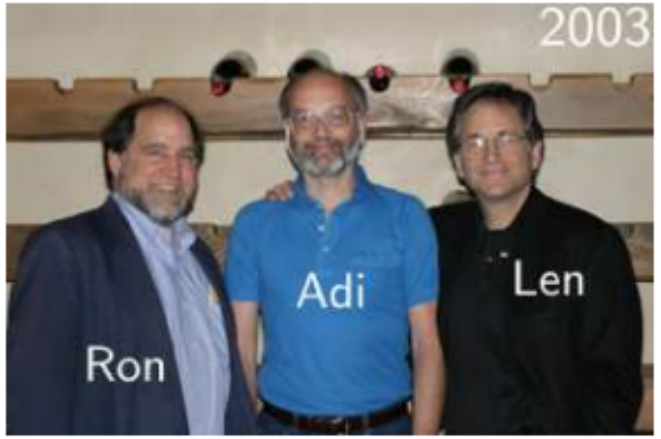


RSA Public-Key Cryptosystem

Ronald Rivest, Adi Shamir, Leonard Adleman, 1977

RSA was named after its developers Ronald Rivest, Adi Shamir, and Leonard Adleman.

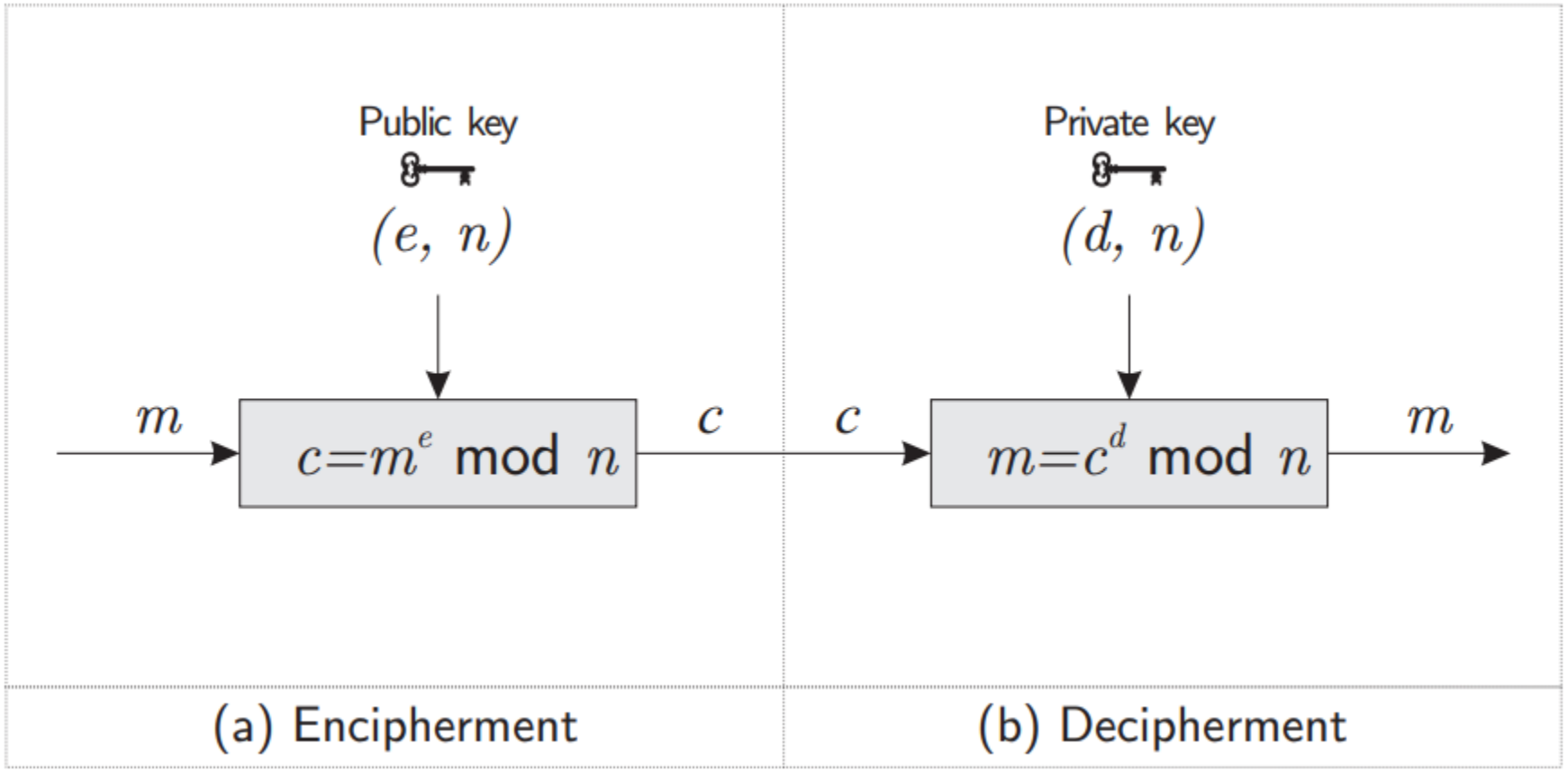




RSA Security released the patent into the public domain in 2000.

Its security is based on the intractability of the integer factorization problem.

RSA Crytosystem



RSA Crytosystem (cont.)

(a) Key Generation:

1) Generate two large random primes p and q ($p \neq$ q and $| p | \approx | q |$)
2) Compute $n = pq$ and ϕ = $(p - 1)(q - 1)$
3) Select a random integer e, 1 < e < ϕ such that GCD(e, ϕ)=1
4) Use EEA(ϕ, e) algorithm to find *y, that is* (d), such that $ed \equiv 1 (\text{mod } \phi)$
5) Publish (e, n) as RSA public key
6) Keep (d, n) as RSA secret key

RSA Crytosystem (cont.)

(b) Encryption:

1) Compute $m = m_1, m_2, \ldots, m_t$ such that m_i < n ∀i ∈ {1, . . . , t}

2) Compute c_i = PowerMod(m_i , e, n) ∀i ∈ {1, . . . , t}

(c) Decryption:

1) Compute mi = PowerMod(c_i , d, n) ∀i ∈ {1, . . . , t}

Example

Doing business on the internet requires you (*client*) to send personal information (e.g. credit card numbers) to a business (*server*). To do this securely, your client software must encrypt your credit card number so that others cannot intercept it. It is currently done with RSA encryption software. The following is a much too simple example, but it doesn't burden us with large numbers.

(a) Key Generation:

1) Let p = 47, q = 71

2) Let n = pq = 3337, ϕ = 46 × 70 = 3220

3) Choose e (at random) to be 79

4) Compute EEA(3220, 79) to find d

Φ	e	q	gcd(Φ, e)	x	y	
3220	79	40	1	-25	1019	-> d=1019
79	60	1	1	19	-25	
60	19	3	1	-6	19	
19	3	6	1	1	-6	
3	1	3	1	0	1	
1	0	-	1	1	0	

5) Public key = (79, 3337)

6) Private key = (1019, 3337)

(b) Encryption:

To encrypt a credit card m=6882 3268 7966 6683 we break it into smaller numbers such that $m_i < n$ $\forall i \in \{1, \ldots, t\}$. For example.

m_1=688, m_2=232, m_3=687, m_4=966, m_5=668, m_6=3
c_i=PowerMod(m_i , 79, 3337) $\forall i \in \{1, \ldots, 6\}$
$b = 79_{10} = \langle 1001111 \rangle_2$
n = 3337

i	6	5	4	3	2	1	0	
b_i	1	0	0	1	1	1	1	
m_1	688	2827	3151	2564	1574	595	1570	$=c_1$
m_2	232	432	3089	3253	1862	391	2756	$=c_2$
m_3	687	1452	2657	3085	2647	1308	2091	$=c_3$
m_4	966	2133	1358	1637	2463	80	2276	$=c_4$
m_5	668	2403	1399	364	77	2890	2423	$=c_5$
m_6	3	9	81	2998	1052	3134	158	$=c_6$

c = 1570 2756 2091 2276 2423 158
This message is then sent across the network to the server

(c) Decryption:

Decrypting the message is done at the server. The server knows d, which the client doesn't, so the server can decrypt the above message.

m_i=PowerMod(c_i , 1019, 3337) ∀i ∈ {1, . . . , 6}
b = 1019_{10}= $\langle 1111111011 \rangle_2$
n = 3337

i	9	8	7	6	5	4	3	2	1	0	
b_i	1	1	1	1	1	1	1	0	1	1	
c_1	1570	796	735	1308	1733	2752	2880	1955	2535	688	$=m_1$
c_2	2756	2560	2206	740	654	57	1073	64	2842	232	$=m_2$
c_3	2091	605	640	2517	179	782	1665	2515	2814	687	$=m_3$
c_4	2276	2407	2752	1582	890	2013	2784	2142	292	966	$=m_4$
c_5	2423	374	480	2459	2889	1445	2146	256	2583	668	$=m_5$
c_6	158	3315	3058	2033	521	554	2781	2132	1200	3	$=m_6$

Getting back the original message m=6882 3268 7966 6683

DLP - The Discrete Logarithm Problem

Problem Instance: I = (p, α, β), where p is prime, α ∈ $\mathbb{Z}_p$ is a *primitive element modulo p*, and β ∈ $\mathbb{Z}_p^*$.

Objective: Find the unique integer a ∈ $\mathbb{Z}_{\phi(p)}$ such that

$$\alpha^a \equiv \beta \pmod{p}$$

We will denote this integer *a* by $log_\alpha \beta$ (mod *p)*

Clearly, the DLP problem can be solved by exhaustive search in O(p) time and O(1) space. By precomputing all possible values α^a, and sorting the ordered pairs (a, α^amod p) with respect to their second coordinates, we can solve the DLP in O(1) time and O(p) space.

Example: Given β = 11 ∈ $\mathbb{Z}_{13}^*$ find the unique exponent $a \in \mathbb{Z}_{11}$ such that $\alpha^a \equiv 11 \pmod{13}$ for α = 2, 6, 7 and 11.

	a											
α	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
2	1	2	4	8	3	6	12	11	9	5	10	7
6	1	6	10	8	9	2	12	7	3	5	4	11
7	1	7	10	5	9	11	12	6	3	8	4	2
11	1	11	4	5	3	7	12	2	9	6	8	10

$\alpha^a \equiv \beta \pmod{p}$

a = 7, 11, 5 and 1 for α = 2, 6, 7 and 11, respectively

ElGamal Public-Key Cryptosystem
Taher ElGamal, 1984

Let p be a prime number such that the DLP problem is intractable, and let $\alpha \in \mathbb{Z}_p^*$ be a *primitive element modulo p*

Let $\mathcal{M} = \mathbb{Z}_p^*$, $C = \mathbb{Z}_p^* \times \mathbb{Z}_p^*$, and define

$$\mathcal{K} = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a \pmod{p}\}$$

The values p, α and β are public, and a is secret.

For $K = (p, \alpha, a, \beta)$ and for some random $k \in \mathbb{Z}_p^*$, define

($\rightarrowtail$) $c = (\gamma, \delta) = (\alpha^k \bmod p, m * \beta^k \bmod p)$
($\leftarrowtail$) $m = \delta * (\gamma^a)^{-1} \bmod p$

ElGamal Cryptosystem

(a) Key Generation:

1) Generate a large random prime p and a generator α of $\mathbb{Z}_p^*$
2) Select a random integer a, $1 \leq a \leq p - 2$
3) Compute $\beta = \text{PowerMod}(\alpha, a, p)$
4) Publish (p, α, β) as ElGamal public key
5) Keep a as ElGamal secret key

ElGamal Cryptosystem (cont.)

(b) Encryption:

1) Compute $m = m_1, m_2, \ldots, m_t$ such that $m_i \in \mathbb{Z}_p^*$
2) Select a random integer k, $1 \leq k \leq p - 2$
3) Compute γ^i= PowerMod(α, k, p), ∀i ∈ {1, . . . , t}
4) Compute δ_i' = PowerMod(β, k, p), ∀i ∈ {1, . . . , t}
5) Compute δ_i = m∗ δ_i' mod p ∀i ∈ {1, . . . , t}
6) Send the cipher text c_i=(γ^i , δ_i), ∀i ∈ {1, . . . , t}

ElGamal Cryptosystem (cont.)

(c) Decryption:

1) Compute $p' = p - 1 - a$

2) Compute m_i' = PowerMod(γ, p' , p), ∀i ∈ {1, . . . , t}

3) Compute m_i = δ*m_i' mod *p,* ∀i ∈ {1, . . . , t}

Example

(a) Key Generation:

1) Let p=2579, α=2

2) Let a=765=<1011111101>$_2$

3) β=PowerMod(2, 765, 2579)

i	9	8	7	6	5	4	3	2	1	0	
b_i	1	0	1	1	1	1	1	1	0	1	
α	2	4	32	2048	1700	461	2086	1246	2537	949	->β

4) Public key=(2579, 2, 949)

5) Private key=765

(b) Encryption:

1) Let m=1299

2) k=853=$<1101010101>_2$

3) γ=PowerMod(2, 853, 2579)

i	9	8	7	6	5	4	3	2	1	0	
b_i	1	1	0	1	0	1	0	1	0	1	
α	2	8	64	455	705	1135	1304	1710	2093	435	->γ

4) Compute δ'=PowerMod(949, 853, 2579)

i	9	8	7	6	5	4	3	2	1	0	
b_i	1	1	0	1	0	1	0	1	0	1	
β	949	65	1646	1676	445	1732	447	345	391	2424	->δ'

5) δ=(1299 × 2424) mod 2579=2396

6) c=(γ, δ)=(435, 2396)

(c) Decryption:

1) p' = 2579 - 1 - 765 = 1813 = $< 11100010101 >_2$

2) m' = PowerMod(435, 1813, 2579)

i	10	9	8	7	6	5	4	3	2	1	0	
b_i	1	1	1	0	0	0	1	0	1	0	1	
γ	435	1511	209	2417	454	2375	959	1557	373	2442	1980	->m'

3) m= (2396 × 1980) mod 2579 = 1299

References

- Pinzon, Yoan. “Introducción a la criptografía y a la seguridad de la información”, 2013.
- Menezes A., Handbook of applied Cryptografy, 5th Edition. CRC Press, 2001.
- A Graduate Course in Applied Cryptography by D. Boneh and V. Shoup
- H. Delfs and H. Knebl, Introduction to Cryptography, 3rd ed. 2015.
- J. A. Buchmann, Introduction to Cryptography. 2004.
- T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Ch. 31 in Introduction to Algorithms, 3rd ed. 2009.
- B. Lomas de Zamora: Gradi. Hacking desde cero, Fox Andina, 2011.
- Secure Coding Working Group, Japan Smartphone Security Association (JSSEC), Android Application Secure Design/Secure Coding Guidebook, 2017.