Lesson 8
Public key cryptosystems
ElGamal - RSA

UNIT 1
The ElGamal Public key Cryptosystem

The ElGamal Public key Cryptosystem

DO NOT REMEMBER THE RULE

Bob wants to send a message to Alice in such a way that just Alice is able to read.

ElGamal Public key cryptosystem (1984 - 7 years later than RSA)

SETUP

1. Alice or a trusted party chooses a large prime p and $g \in \mathbb{F}_p^*$ of large order. PUBLIC

2. Alice chooses a secret number a PRIVATE KEY

3. Alice publishes $A = g^a$ in $\mathbb{F}_p$. PUBLIC KEY

ENCRYPTION: ^k

PRIVATE, used once to randomize the encryption process

4. Bob wants to send a message $m \in \mathbb{F}_p$: he chooses $1 < k < p-1$ randomly and sends $(c_1 = g^k, c_2 = mA^k) \in \mathbb{F}_p \times \mathbb{F}_p$. Might be PUBLIC

DECRYPTION:^a

5. Alice computes $m = (c_1^a)^{-1} \cdot c_2$.

CORRECTNESS

The ElGamal Cryptosystem

ElGamal Public key cryptosystem (1984)

SETUP

1. Alice or a trusted party chooses a large prime p and $g \in \mathbb{F}_p^*$ of large order. PUBLIC
2. Alice chooses a secret number a PRIVATE KEY
3. Alice publishes $A = g^a$ in $\mathbb{F}_p$. PUBLIC KEY

ENCRYPTION: ^k

PRIVATE, used once to randomize the encryption process

4. Bob wants to send a message $m \in \mathbb{F}_p$: he chooses $1 < k < p-1$ randomly and sends $(c_1 = g^k, c_2 = mA^k) \in \mathbb{F}_p \times \mathbb{F}_p$. Might be PUBLIC

DECRYPTION:^a

5. Alice computes $m = (c_1^a)^{-1} \cdot c_2$.

EXAMPLE

SETUP

Alice chooses $p = 467$, $g = 2$ (primitive root)

$a = 153$ PRIVATE KEY

$A = g^a[p] = 2^{153} = 224.$ PUBLIC KEY

ENCRYPTION

Bob decides to send the message $m = 331$. Random element: $k = 197$.

DECRYPTION

L7 U1

The ElGamal Cryptosystem

ElGamal Public key cryptosystem (1984)

SETUP

1. Alice or a trusted party chooses a large prime p and $g \in \mathbb{F}_p^*$ of large order. PUBLIC
2. Alice chooses a secret number a PRIVATE KEY
3. Alice publishes $A = g^a$ in $\mathbb{F}_p$. PUBLIC KEY

ENCRYPTION: ^k

PRIVATE, used once to randomize the encryption process

4. Bob wants to send a message $m \in \mathbb{F}_p$: he chooses $1 < k < p-1$ randomly and sends $(c_1 = g^k, c_2 = mA^k) \in \mathbb{F}_p \times \mathbb{F}_p$. Might be PUBLIC

DECRYPTION:^a

5. Alice computes $m = (c_1^a)^{-1} \cdot c_2$.

REMARK How to compute $(c_1^a)^{-1}$:

Method 1

i) Compute c_1^a with Fast Powering Algorithm

ii) Compute $(c_1^a)^{-1}$ with Extended Euclidean algorithm

Method 2

Compute $(c_1^a)^{-1} = c_1^{-a} = c_1^{p-1-a}$ with Fast Powering Algorithm $\sim 3\log_2(p-1-a)$

Method 1 is more convenient if a is small.

The ElGamal Cryptosystem

on EVE's side: solving DLP

ElGamal Public key cryptosystem (1984)

SETUP

1. Alice or a trusted party chooses a large prime p and $g \in \mathbb{F}_p^*$ of large order. PUBLIC
2. Alice chooses a secret number a PRIVATE KEY
3. Alice publishes $A = g^a$ in $\mathbb{F}_p$. PUBLIC KEY

ENCRYPTION: ^k

PRIVATE, used once to randomize the encryption process

4. Bob wants to send a message $m \in \mathbb{F}_p$: he chooses $1 < k < p-1$ randomly and sends $(c_1 = g^k, c_2 = mA^k) \in \mathbb{F}_p \times \mathbb{F}_p$. Might be PUBLIC

DECRYPTION:^a

5. Alice computes $m = (c_1^a)^{-1} \cdot c_2$.

REMARK: Solving DLP allows to decrypt ElGamal

Eve task in trying to decrypt.

Eve knows p, g and $A = g^a$.

If Eve knows how to solve the DLP $g^x = A$ then she finds

$$a (\text{mod. } \operatorname{ord}_p(g)) : \ a + u \operatorname{ord}_p(g),\ u \in \mathbb{Z}.$$

She computes $c_1^{a+u \operatorname{ord}_p(g)} = c_1^a \cdot c_1^{u \operatorname{ord}_p(g)} = c_1^a \cdot g^{ku \operatorname{ord}_p(g)} = c_1^a,$

thus she computes $m = (c_1^a)^{-1} \cdot c_2$

The ElGamal Cryptosystem

on EVE's side: knowing k

ElGamal Public key cryptosystem (1984)

SETUP

1. Alice or a trusted party chooses a large prime p and $g \in \mathbb{F}_p^*$ of large order. PUBLIC
2. Alice chooses a secret number a PRIVATE KEY
3. Alice publishes $A = g^a$ in $\mathbb{F}_p$. PUBLIC KEY

ENCRYPTION: ^k

PRIVATE, used once to randomize the encryption process

4. Bob wants to send a message $m \in \mathbb{F}_p$: he chooses $1 < k < p-1$ randomly and sends $(c_1 = g^k, c_2 = mA^k) \in \mathbb{F}_p \times \mathbb{F}_p$. Might be PUBLIC

DECRYPTION:^a

5. Alice computes $m = (c_1^a)^{-1} \cdot c_2$.

REMARK (you decrypt if you know k)

If Eve knows k, then she is able to decrypt the message.

Indeed, she knows A, k and $c_2 = mA^k$. Thus $m = c_2 \cdot (A^k)^{-1}$.

REMARK (the need of changing k)

If k is not changed then the cipher is subject to known plaintext attack!

Imagine Eve knows $m, (c_1 = g^k, c_2 = mA^k)$, with $A = g^a$.

If now m' is encrypted as $(c_1 = g^k, c_2' = m'A^k)$ then

$$m' \cdot m^{-1} = c_2' \cdot (c_2)^{-1} \Rightarrow m' = mc_2' \cdot (c_2)^{-1}.$$

UNIT 3
The Rivest - Shamir - Adleman (RSA) Public Key Cryptosystem

Back to Fermat's little Theorem

RECALL (Fermat THM, version 2) **Let p prime, $a \in \mathbb{Z}$. Then $a^p \equiv a[p]$.**

COROLLARY **Let p prime, $a \in \mathbb{Z}$. If $x \in \mathbb{N}_{\geq 1}, x \equiv 1[p-1] \Rightarrow a^x \equiv a[p]$.**

REMARK **If $p \mid a$ and $a \in \mathbb{Z}$ the term a^x makes sense just if $x > 0$.**

Proof.

Euler's formula

EULER's FORMULA for a product of two distinct primes

It holds with any number of distinct primes

Let p, q be distinct primes, $a \in \mathbb{Z}$. If $x \in \mathbb{N}_{\geq 1}$, $x \equiv 1[\phi(pq)]$ then $a^x \equiv a[pq]$.

Proof.

Euler's Formula: applications

EXAMPLE $n = 15 = 3 \cdot 5,\ \phi(n) = 8.$ $\quad 9 \equiv 1[8] \Rightarrow$

EXAMPLE

Let $m \in \mathbb{Z}$.

Let $c := m^{41}$ in $\mathbb{Z}/(7 \cdot 13)\mathbb{Z}$.

Compute $c^{65}[7 \cdot 13]$:

The RSA Public key Cryptosystem

Bob wants to send a message to Alice in such a way that just Alice is able to read.

Rivest-Shamir-Adleman Public key cryptosystem (1977, 1 year after Diffie-Hellmann)

SETUP

1. **Alice chooses two large distinct primes p, q and computes $N = pq$,** SECRET

   $\phi(N) = (p-1)(q-1)$. SECRET

2. **Alice chooses $1 < e < \phi(N)$ with $\gcd(e, \phi(N)) = 1$ and computes $d := e^{-1}[\phi(N)]$.** (encryption exponent: e; decryption exponent: d)

3. **Alice makes N (modulus) and e (encryption key) public.** PUBLIC

ENCRYPTION

4. **Bob sends $c = m^e[N]$ to Alice.** Might be PUBLIC

DECRYPTION

5. **Alice computes $m = c^d[N]$.**

CORRECTNESS $c^d = (m^e)^d = m^{ed} = m$ **in $\mathbb{Z}/N\mathbb{Z}$: indeed $ed \equiv 1[\phi(N)]$.**

EULER'S FORMULA

The RSA Public key Cryptosystem

SETUP

1. Alice chooses two large distinct primes p, q and computes $N = pq$, SECRET
   $\phi(N) = (p-1)(q-1)$. SECRET
2. Alice chooses $1 < e < \phi(N)$ with $\gcd(e, \phi(N)) = 1$ and computes $d := e^{-1}[\phi(N)]$. (encryption exponent / decryption exponent)
3. Alice makes N (modulus) and e (encryption key) public. PUBLIC

ENCRYPTION

4. Bob sends $c = m^e[N]$ to Alice. Might be PUBLIC

DECRYPTION

5. Alice computes $m = c^d[N]$.

EXAMPLE

1. Alice chooses $p = 1223, q = 1987$ and computes
$N = p \cdot q = 1223 \cdot 1987 = 2430101$

2. Alice chooses a public key exponent $e = 948047$ with the property:
$\gcd(e, (p-1)(q-1)) = 1$.

She computes $d = e^{-1}$ in $\mathbb{Z}/(p-1)(q-1)\mathbb{Z} = \mathbb{Z}/2426892\mathbb{Z}$: $d =$

3. Alice shows $N = 2430101, e = 948047$.

4. Bob converts its plaintext $m = 1070777$: he computes $c \equiv m^e[N] \equiv$ and sends it to Alice.

5. Alice takes c and computes $c^d[N] \equiv$

L8 U3

The RSA Public key Cryptosystem

SETUP

1. Alice chooses two large distinct primes p, q and computes $N = pq$, SECRET
   $\phi(N) = (p-1)(q-1)$. SECRET
2. Alice chooses $1 < e < \phi(N)$ with $\gcd(e, \phi(N)) = 1$ and computes $d := e^{-1}[\phi(N)]$. (encryption exponent: e; decryption exponent: d)
3. Alice makes N (modulus) and e (encryption key) public. PUBLIC

ENCRYPTION

4. Bob sends $c = m^e[N]$ to Alice. Might be PUBLIC

DECRYPTION

5. Alice computes $m = c^d[N]$.

WARNING **A frequent error**

When $\gcd(e, N) = \gcd(e, \phi(N)) = 1$ an error is to compute $e^{-1}[N]$ instead of $e^{-1}[\phi(N)]$. When $\gcd(e, N) > 1$ nobody would compute e^{-1} in $\mathbb{Z}/N\mathbb{Z}$...

EXAMPLE $N = 3 \times 5 = 15$, $\phi(N) = 8$. Take $m = 13, e = 7$.

The RSA Public key Cryptosystem

SETUP

1. Alice chooses two large distinct primes p, q and computes $N = pq$, SECRET
   $\phi(N) = (p-1)(q-1)$. SECRET
2. Alice chooses $1 < e < \phi(N)$ with $\gcd(e, \phi(N)) = 1$ and computes $d := e^{-1}[\phi(N)]$. (encryption exponent, decryption exponent)
3. Alice makes N (modulus) and e (encryption key) public. PUBLIC

ENCRYPTION

4. Bob sends $c = m^e[N]$ to Alice. Might be PUBLIC

DECRYPTION

5. Alice computes $m = c^d[N]$.

REMARK

$e \neq 2$ because $(p-1)(q-1)$ is even.

The smallest possible value is $e = 3$.

If $\gcd(3, (p-1)(q-1)) = 1$ then $e = 3$ is as secure as a high value of e.

People who want fast encryption but worried $e = 3$ too small, often take $e = 2^{16} + 1 = 65537$.

Actually, it would be better to begin by taking $d >> N^{1/4}$.

The RSA: Security Issues

SETUP

1. Alice chooses two large distinct primes p, q and computes $N = pq$, SECRET
   $\phi(N) = (p-1)(q-1)$. SECRET
2. Alice chooses $1 < e < \phi(N)$ with $\gcd(e, \phi(N)) = 1$ and computes $d := e^{-1}[\phi(N)]$. (encryption exponent, decryption exponent)
3. Alice makes N (modulus) and e (encryption key) public. PUBLIC

ENCRYPTION

4. Bob sends $c = m^e[N]$ to Alice. Might be PUBLIC

DECRYPTION

5. Alice computes $m = c^d[N]$.

1. Factor N allows to find p, q and thus d. **(Lesson 11)**

   Typical size of N: 2048 to 4096 bit typically. An 829-bit N has been factored.

2. Finding roots: Find x such that $x^e \equiv c[N]$. **Next Unit...**

UNIT 4
Security issues of RSA Public Key Cryptosystem
Computing modular e-th roots

Security of RSA

SETUP

1. Alice chooses two large distinct primes p, q and computes $N = pq$, SECRET
   $\phi(N) = (p-1)(q-1)$. SECRET
2. Alice chooses $1 < e < \phi(N)$ with $\gcd(e, \phi(N)) = 1$ and computes $d := e^{-1}[\phi(N)]$. (encryption exponent / decryption exponent)
3. Alice makes N (modulus) and e (encryption key) public. PUBLIC

ENCRYPTION

4. Bob sends $c = m^e[N]$ to Alice. Might be PUBLIC

DECRYPTION

5. Alice computes $m = c^d[N]$.

PROBLEM

Knowing N, e, c is deciphered if, either,

i) One knows p, q, or, FACTORIZATION of N: DISCUSSED IN A LESSON 11

ii) The equation $x^e = c$ has one solution $[N]$ and it is easy to solve. Compute e-th roots

iii) (Woman-in-the-middle attack) Eve has full control on the communication network: she chooses her own exponent e' and sends it to Bob. Certificate authorities for public keys

Existence and computation of e-th roots mod. products of primes

PROPOSITION
Existence of e-th root [N]

Let $p_1, \ldots, p_r$ $(r \geq 1)$ be distinct primes, $N = p_1 \cdots p_r$.
Let $e \geq 1$ with $\gcd(e, \phi(N)) = 1$.

Then for any $c \in \mathbb{Z}/N\mathbb{Z}$ the equation $x^e = c[N]$ has a unique solution: $x = c^d[N]$,

where $d = e^{-1}$ in $\mathbb{Z}/\phi(N)\mathbb{Z}$.

Proof. $ed \equiv 1[\phi(N)]$:

i) Uniqueness: $x^e = c \Rightarrow \left(x^e\right)^d = c^d \Leftrightarrow x^{ed} = c^d$ **in** $\mathbb{Z}/N\mathbb{Z}$.

$\Leftrightarrow x = c^d$ **in** $\mathbb{Z}/N\mathbb{Z}$.

EULER'S FORMULA

ii) Existence: $x = c^d$ **is a solution.**

Indeed, $\left(c^d\right)^e = c^{de} = c$ **in** $\mathbb{Z}/N\mathbb{Z}$. $\square$

EULER'S FORMULA

Computing e-th roots mod. a prime

THE CASE of N PRIME

EXAMPLE Solve $x^{1583} = 4714$ in $\mathbb{F}_{7919}$ (7919 is prime).

1. $\gcd(1583, 7919 - 1) = \gcd(1583, 7918) =$
2. Find the inverse of 1583 in $\mathbb{Z}/7918\mathbb{Z}$:
3. $x \equiv 4714^{5277}[7919] \equiv$

EXAMPLE (the case GCD(e, p-1)>1) $p = 11, e = 2$: $\gcd(e, p - 1) = \gcd(2, 10) = 2$.

$\mathbb{F}_{11}$

x	0	1	2	3	4	5	6	7	8	9	10
x^2	0	1	4	9	5	3	3	5	9	4	1

Computing e-th roots mod. N

The case of a product of 2 distinct primes

EXAMPLE

Solve $x^{17389} = 43927$ in $\mathbb{Z}/64349\mathbb{Z}$.

Computing e-th roots mod. N

The case of a product of 2 distinct primes

EXAMPLE **Alice and Bob decide to use RSA to exchange a message.**

Alice shows $N = 30069476293$, $e = 9843$.

Bob sends $c = 134872[N]$.

Eve tries to decrypt the cipher, and wants to solve $x^e = c$. **FIND *x***

Computing e-th roots mod. N

The case of a product of more than 2 distinct primes

EXAMPLE **Solve $x^{73} = 23$ in $\mathbb{Z}/(5 \cdot 7 \cdot 11)\mathbb{Z}$.** N is a product of 3 distinct primes

$$N = 5 \cdot 7 \cdot 11 = 385, \phi(N) = (5-1)(7-1)(11-1) = 240.$$

Computing e-th roots mod. N

THE CASE OF A PRODUCT OF PRIMES WITH MULTIPLICITIES

DO NOT REMEMBER THE RULE

REMARK **If N is not a product of distinct primes, the equation $x^e = c$ in $\mathbb{Z}/N\mathbb{Z}$ ($\gcd(e, \phi(N)) = 1$):**

- **If $\gcd(c, N) = 1$ it has the unique solution $x = c^d$, where $de = 1$ in $\mathbb{Z}/\phi(N)\mathbb{Z}$;**
- **If $\gcd(c, N) > 1$ it may have several solutions or no solutions.**

EXAMPLE **For instance, $N = 48 = 2^4 \cdot 3$. Then $\phi(N) = 16$. Take $e = 11$:**

Computing e-th roots mod. N

The case of a product of prime with multiplicities >1

EXAMPLE Solve $x^{959} = 1583 [5^3 \cdot 13]$

N is not a product of distinct primes

$$N = 5^3 \cdot 13,\ \phi(N) = (5^3 - 5^2)(13 - 1) = 100 \cdot 12 = 1200.$$

$$\gcd(959, \phi(N)) = 1.$$

Further check: $c = 1583$, $\gcd(1583, N) = 1$?

L8 U4

Computing e-th roots mod. N

EPILOGUE

THE METHOD Solve $x^e = c$ in $\mathbb{Z}/N\mathbb{Z}$. N product of distinct primes.

1. Compute $\phi(N)$.
2. Check that $\gcd(e, \phi(N)) = 1$ and find $d = e^{-1}$ in $\mathbb{Z}/\phi(N)\mathbb{Z}$.
3. If N is a product of distinct primes: $x = c^d[N]$.

REMARK In order to find the solution x the method shows that you must know the prime factors of N (though some research papers indicate it is slightly easier to find e-th roots than prime factorization)

END of Lesson 8
Public key cryptosystems