Information Systems Security

Lecture 4

Asymmetric cryptography

Dr. En. Bader Ahmad

references

[1] K. Martin's Lecture (www.rhul.ac.uk).

[2] Cryptography and Network Security, By W. Stallings. Prentice Hall, 2003.

[3] Handbook of applied Cryptography by A. Menezes, P. Van Oorschot and S. Vanstone. 5th printing, 2001
http://www.cacr.math.uwaterloo.ca/hac

[4] Cryptography: A Very Short Introduction (Very Short Introduction S.), by Fred Piper and Sean Murphy, Oxford University Press, 2002.

Outline

1. Basic mathematical concepts
2. Public key cryptography
3. OWF
4. RSA
5. ElGamal

1. The modulo operation

- Definition
  - Let a, r, n be integers and let $q > 0$
  - We write $a \equiv r \bmod n$ if n divides $a - r$ (or $r - a$) and $0 \leq r < n$
  - n is called the ***modulus***
  - r is called the ***remainder***
    - Note that r is positive or zero
  - Note that $a = n.q + r$ where q is another integer (***quotient***)

- Example: $42 \equiv 6 \bmod 9$
  - 9 divides 42 - 6 = 36
  - 9 also divides 6 - 42 = -36
  - Note that 42= 9x4 + 6
    - (q = 4)

Number Theory

- Natural numbers $N = \{1,2,3,\ldots\}$
- Whole numbers $W = \{0,1,2,3, \ldots\}$
- Integers $Z = \{\ldots,-2,-1,0,1,2,3, \ldots\}$

- Divisors
  - A number b is said to divide a if $a = mb$ for some m where $a,b,m \in Z$
  - We write this as $b|a$
    - Read as "b divides a"

SYRIAN PRIVATE UNIVERSITY
الجامعة السورية الخاصة

Divisors

- Some common properties
  - If $a|1$, $a = +1$ or -1
  - If $a|b$ and $b|a$ then $a = +b$ or $-b$
  - Any $b \in Z$ divides 0 if $b \neq 0$
  - If $b|g$ and $b|h$ then $b|(mg + nh)$ where $b,m,n,g,h \in$ Z

- Examples:
  - The positive divisors of 42 are 1,2,3,6,7,14,21,42
  - $3|6$ and $3|21$ => $3|21m+6n$ for $m,n \in Z$

Prime Numbers

- An integer *p* is said to be a prime number if its only positive divisors are 1 and itself
  - Examples 2, 3, 7, 11, ..
- Any integer can be expressed as a ***unique*** product of prime numbers raised to positive integral powers
  - $n=p_1^{e_1} p_2^{e_2} ...p_k^{e_k}$ // n: ingterger, p_i:prime, e_i: positive integer
- Examples
  - $7569 = 3 \times 3 \times 29 \times 29 = 3^2 \times 29^2$
  - $5886 = 2 \times 27 \times 109 = 2 \times 3^3 \times 109$
- This process is called ***Prime Factorization***

Greatest common divisor (GCD)

- Definition: ***Greatest Common Divisor***
  - This is the largest divisor of *both a* and *b*
- Given two integers *a* and *b*, the positive integer *c* is called their GCD or greatest common divisor if and only if
  - $c \mid a$ and $c \mid b$
  - Any divisor of both *a* and *b* also divides *c*
- Notation: $gcd(a, b) = c$
- Example: $gcd(49,63) = ?$

- $gcd(a,b)=gcd(b, a \bmod b)$
- Exception: $gcd(0,0)=0$

SYRIAN PRIVATE UNIVERSITY
الجامعة السورية الخاصة

Relatively Prime Numbers

- Two numbers are said to be ***relatively prime*** if their *gcd* is 1
  - Example: 63 and 22 are relatively prime
- How do you determine if two numbers are relatively prime?
  - Find their *gcd* or
  - Find their prime factors
    - If they do not have a common prime factor other than 1, they are relatively prime
  - Example: $63 = 9 \times 7 = 3^2 \times 7$ and $22 = 11 \times 2$

Modular Arithmetic Again

- We say that $a \equiv b \bmod m$ if $m \mid a - b$
  - Read as: *a* is ***congruent*** to *b* modulo *m*
  - *m* is called the *modulus*
  - Example: $27 \equiv 2 \bmod 5$
- Note that *b* is the ***remainder*** after dividing *a* by *m*
  - Example: $27 \equiv 2 \bmod 5$ and $7 \equiv 2 \bmod 5$
- $a \equiv b \bmod m \Rightarrow b \equiv a \bmod m$
  - Example: $2 \equiv 27 \bmod 5$
- We usually consider the *smallest positive remainder* which is sometimes called the ***residue***

SYRIAN PRIVATE UNIVERSITY
الجامعة السورية الخاصة

Modulo Operation

- The modulo operation "**reduces**" the infinite set of integers to a finite set

- Example: modulo 5 operation
  - We have five sets
    - {…,-10, -5, 0, 5, 10, …} => $a \equiv 0 \bmod 5$
    - {…,-9, -4, 1, 6, 11,…} => $a \equiv 1 \bmod 5$
    - {…,-8, -3, 2, 7, 12,…} => $a \equiv 2 \bmod 5$
    - {…,-7, -2, 3, 8, 13,…} => $a \equiv 3 \bmod 5$
    - {…,-6, -1, 4, 9, 14…} => $a \equiv 4 \bmod 5$
  - The set of residues of integers modulo 5 has five elements {0,1,2,3,4} and is denoted Z_5.

SYRIAN PRIVATE UNIVERSITY الجامعة السورية الخاصة

Euler phi (or totient) function

- For $n \geq 1$, $\phi(n)$: is the number of integers in [1,n] which are relatively prime to n // $\phi(n)$ is the ***Euler phi*** or ***totient function***
- If p is prime, then $\phi(p)=p-1$
- If $gcd(m,n)=1$, then $\phi(mn)= \phi(m).\phi(n)$
- Examples:
  - $\phi(21)= \phi(3).\phi(7) = (3-1) * (7-1) =12$

multiplicative group $\mathbf{Z_n}^*$

- Definition: the ***multiplicative group*** *Z_n^* of Z_n*
  - $Z_n^* = \{a \in Z_n \mid gcd(a,n)=1\}$
  - If n is prime then $Z_n^* = \{a \in Z_n \mid 1 \leq a \leq n\text{-}1\}$
  - $\phi(n) = |Z_n^*|$

- Let n ≥ 2 be an integer
  - Euler's theorem: If $g \in Z_n^*$ then $g^{\phi(n)} \equiv 1 \ (mod\ n)$
  - If n is a product of distinct primes, and if $r \equiv s \ mod\ (\phi(n))$, then $g^r \equiv g^s \ (mod\ n)$ for all integers g
  - *i.e.*, when <u>working modulo an *n*, exponents can be reduced modulo $\phi(n)$</u>

multiplicative group Z_n^*

- Let *p* be a prime nubmer
  - Fermat's theorem: If $gcd(a,p)=1$, then $g^{p-1} \equiv 1 \ (mod\ p)$
  - If $r \equiv s \ mod\ (p\text{-}1)$, then $g^r \equiv g^s \ (mod\ p)$ for all integers *g*
    - *i.e.*, when working modulo a prime *p*, exponents can be reduced modulo *p-1*
  - Particular case: $g^p \equiv g \ (mod\ p)$ for all integers *g*

SYRIAN PRIVATE UNIVERSITY

Generator of Z_n^*

- Let $g \in Z_n^*$, the ***order*** of g is the least positive integer t such that $g^t \equiv 1 \bmod n$

- If the order of $g \in Z_n^*$ is t, and $g^s \equiv 1 \pmod n$, then t divides s
  - A particular case: $t | \phi(n)$

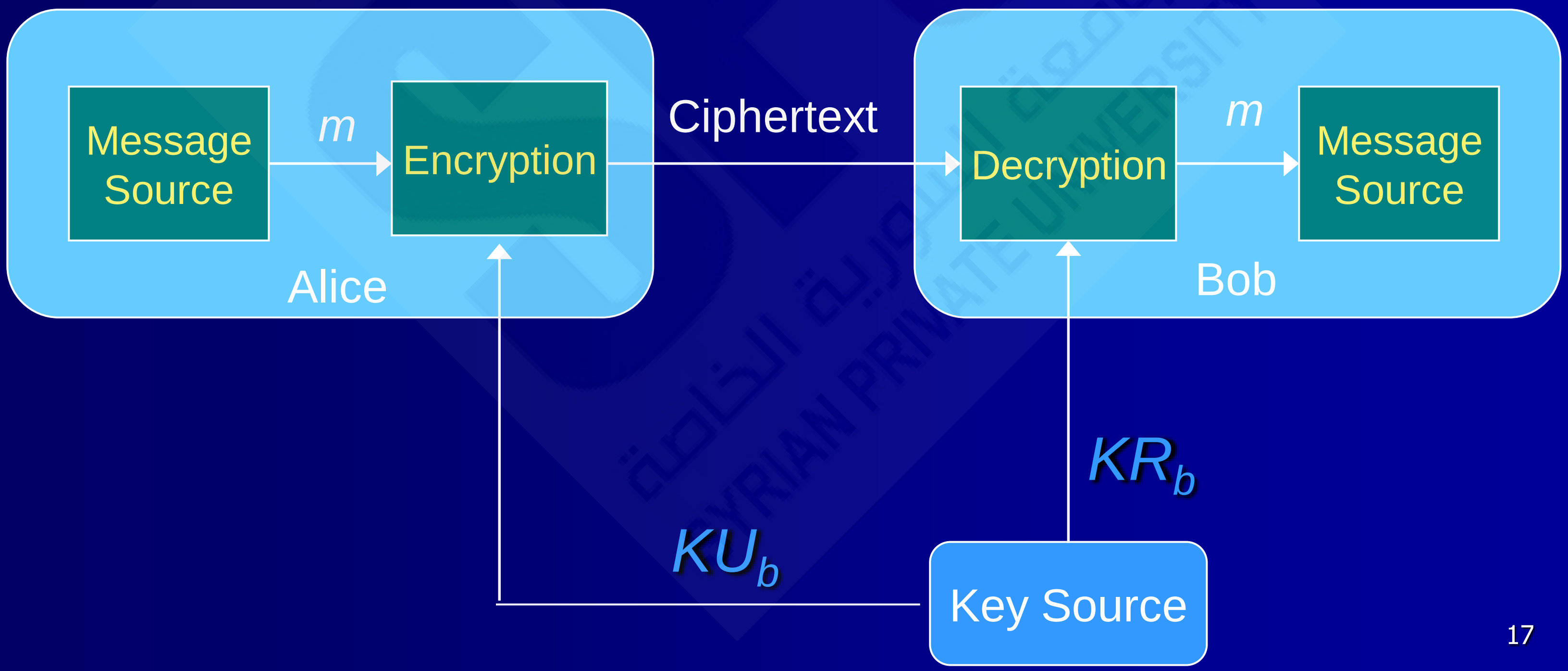
- Let $g \in Z_n^*$, if the order of g is $\phi(n)$, then g is said to be a ***generator*** or a ***primitive element*** of Z_n^*.
  - If g is a generator of Z_n^*, then $Z_n^* = \{g^i \bmod n \mid 0 \le i \le \phi(n) - 1\}$

2. Public-key cryptography

- Called also ***asymmetric cryptography***
- The keys used to encrypt and decrypt are different.
- Anyone who wants to be a receiver needs to "publish" an encryption key, which is known as the ***public key,*** *KU*.
- Anyone who wants to be a receiver needs a unique decryption key, which is known as the ***private key,*** *KR*.
- If B wants to send an enciphered text to A, B should knows ***the encryption algorithm*** and A's public key.

Confidentiality via Public key cryptography

- Alice wants to send a secret message *m* to Bob
- Bob should have 2 keys: public KU_b and private KR_b
- Prior to message encryption, <u>Alice gets by some means an authentic copy of Bob's public key</u> (*i.e.*, the encryption key)



Public-key cryptography

- It should not be possible to deduce the plaintext from knowledge of the ciphertext and the public key.

- It should not be possible to deduce the private key from knowledge of the public key.

- Public-key cryptography is based on ***One-Way Functions***

SYRIAN PRIVATE UNIVERSITY

3. One-Way Functions (OWF)

- A **one-way function** is a function that is "easy" to compute and "difficult" to reverse

- Examples of OWF that we'll use in this lecture to explain public-key systems:
  - Multiplication of two primes
  - Modular exponentiation

الجامعة السورية الخاصة
SYRIAN PRIVATE UNIVERSITY

OWF: Multiplying two primes

- Multiplication of two prime numbers is **believed** to be a one-way function.

- Given two prime numbers *p* and *q*
  - It's easy to find $n=p.q$
  - However, starting from n, it's difficult to find *p* and *q*

- Is it prime factorization?

الجامعة السورية الخاصة
SYRIAN PRIVATE UNIVERSITY

OWF: Modular exponentiation

- The process of ***exponentiation*** just means raising numbers to a power.
- Raising a to the power b, normally denoted a^b just means multiplying a by itself b times. In other words:

$$a^b = a \times a \times a \times \ldots \times a$$

- ***Modular exponentiation*** means computing a^b modulo some other number n. We tend to write this as

$$a^b \bmod n.$$

- Modular exponentiation is "easy".

OWF: Modular exponentiation

- However, given a, and $a^b \bmod n$ (when n is prime), calculating b is regarded by mathematicians as a hard problem.

- This difficult problem is often referred to as the **discrete logarithm problem**.

- In other words, given a number a and a prime number n, the function

$$f(b) = a^b \bmod n$$

is believed to be a one-way function.

4. RSA

- It is named after it inventors Ron Rivest, Adi Shamir and Len Adleman.
- Published in 1978
- It is the most widely used public-key encryption algorithm today.
- It provides confidentiality and digital signatures.
- Its security is based on the difficulty of integer factorization

RSA algorithm (key generation for RSA public-key encryption)

- Each entity A creates a public key and a corresponding private key by doing the following
  - Generate two large (at least 1024 bits) primes p and q
  - Compute $n=pq$ and $\phi(n)=(p-1)(q-1)$.
  - Choose $e < \phi$ relatively prime to ϕ (i.e., $gcd(e, \phi)=1$)
  - Compute d such that $ed \bmod \phi(n) \equiv 1$
- A's Public key: (e, n) // to be published.
- A's private key: d (or (d, n)) // to be kept secretly by A
- Who is capable of computing d?

SYRIAN PRIVATE UNIVERSITY

RSA Encryption/decryption

- Summary: B encrypts a message *m* for A. Upon reception, A decrypts it using its private key.
- Encryption: B should do the following
  - Obtain A's authentic public key *(n,e)*.
  - Represent the message as an integer in the interval [*0*,*n*-*1*]
  - Compute $c = m^e \bmod n$ // **<u>Encryption</u>**
  - Send the ciphertext c to A
- Decryption: to recover plaintext *m* from *c*, A does the following
  - Use the private key *d* to recover $m = c^d \bmod n$ // **<u>Decryption</u>**
- *How does B obtain A's authentic key?*

الجامعة السورية الخاصة
SYRIAN PRIVATE UNIVERSITY

Example: confidentiality

- Take $p = 7$, $q = 11$, so $n = 77$ and $\phi(n) = 60$
- Say Bob chooses (KU_b) $e = 17$, making (KR_b) $d = 53$
  - 17 x 53 mod 60 = ?
- Alice wants to secretly send Bob the message HELLO [07 04 11 11 14]
  - 07^{17} mod 77 = 28
  - 04^{17} mod 77 = 16
  - 11^{17} mod 77 = 44
  - 11^{17} mod 77 = 44
  - 14^{17} mod 77 = 42
- Alice sends ciphertext [28 16 44 44 42]

Example: confidentiality

- Bob receives [28 16 44 44 42]

- Bob uses private key (KR_b), $d = 53$, to decrypt the message:
  - $28^{53} \bmod 77 = 07$ H
  - $16^{53} \bmod 77 = 04$ E
  - $44^{53} \bmod 77 = 11$ L
  - $44^{53} \bmod 77 = 11$ L
  - $42^{53} \bmod 77 = 14$ O

- No one else could read it, as only Bob knows his private key and that is needed for decryption

Attacking RSA

1. Trying to decrypt a ciphertext without knowledge of the private key
   - The encryption process in RSA involves computing the function $c = m^e \bmod n$, which is regarded as being easy
   - An attacker who observes this ciphertext c, and has knowledge of e and n, needs to try to work out what m is.
   - *i.e.*, find m such that $\boldsymbol{m^e} = c \bmod n$
   - In other words, find the e^{th} root of c mod n

- Computing m from c, e and n is regarded as a hard problem and known as ***RSA problem***.

Attacking RSA

2. If the attacker knows the public key of a user (e,n), what would she/he need to do in order to obtain the corresponding private key?
   - He/she needs to find *d* such that $ed \bmod \phi(n) = 1$
   - *i.e.*, needs to know *p* and *q*
   - In other words, he/she must factor *n* (problem of prime factorization)

- Recommended size of n:
  - 768-bit is recommended
  - 1024-bit or larger is required for long term security
  - it is believed that factoring a 512 bit number is about as hard as searching for a 56 bit symmetric key.

SYRIAN PRIVATE UNIVERSITY

5. El Gamal

- ElGamal is another public-key encryption
- We will also take a look at the ElGamal public key cipher system for a number of reasons:
  - To show that RSA is not the only public key system
  - To exhibit a public key system based on a different one way function
  - ElGamal is the basis for several well-known cryptosystems

ElGamal algorithm (key generation)

- Key generation for ElGamal public-key encryption
- Each entity *A* creates a public key and a corresponding private key.
  - Generate a large prime number *p* (1024 bits)
  - Generate a generator *g* of the multiplicative group Z_p^* of the integers modulo *p*
  - Select a random integer *x*, $1 \le x \le p-2$
  - Compute $y = g^x \bmod p$
  - A's public key is (p, g, y)
    - To be published
  - A's private key is *x*
    - To be kept secret by A

ElGamal algorithm (key generation)

- Example
- Step 1: Let $p = 2357$
- Step 2: Select a generator $g = \mathbf{2}$ **of** $\mathbf{Z}_{2357}^{*}$
- Step 3: Choose a private key $x =$ **1751**
- Step 4: Compute $y = 2^{1751} \pmod{\mathbf{2357}}$

  $= \mathbf{1185}$

- Public key is **(2357,2,1185)**
- Private key is **1751**

ElGamal algorithm (Encryption/decryption)

- Summary: B encrypts a message *m* for A, which A decrypts
- Encryption: B should de the following
  - Obtain A's authentic public key *(p, g, y).*
  - Represent the message as an integer in the interval $[0,p-1]$
  - Select an integer *k*, $1 \leq k \leq p-2$
  - Compute $\gamma = g^k \bmod p$ and $\delta = m.(y)^k \bmod p$
  - *Send the ciphertext* $c = (\gamma, \delta)$ *to* A
- *Decryption*
  - A uses the private key *x* to compute $z = \gamma^{p-1-x} \bmod p$
  - A computes $z.\delta \bmod p\ (=m)$

ElGamal algorithm (Encryption/decryption)

- Encryption
  - To encrypt m = 2035 using Public key **(2357,2,1185)**
  - Generate a random number k = **1520**
  - Compute $\gamma = 2^{1520} \bmod 2357 = 1430$

    $\delta = 2035 \times 1185^{1520} \bmod 2357 = 697$

  - Ciphertext c = (**1430** , **697)**

- Decryption
  - $z = \gamma^{p-1-x} \bmod p = 1430^{605} \bmod 2357 = 872$
  - 872x697 mod 2357 = 2035

ElGamal Properties

- There is a *message expansion* by a factor of 2
  - *i.e.*, the ciphertext is twice as long as the corresponding plaintext
- *Requires a random number generator (k)*
- Relies on discrete algorithm problem, *i.e.*, having $y = g^x \bmod p$ it's hard to find x (the private key)
- ElGamal encryption is randomized (coming from the random number *k*), RSA encryption is deterministic.
- ElGamal is the basis of many other algorithms (*e.g.*, DSA)

Summary

- RSA is a public key encryption algorithm whose security is believed to be based on the problem of factoring large numbers.
- ElGamal is a public key encryption algorithm whose security is believed to be based on the discrete logarithm problem.
- RSA is generally favoured over ElGamal for practical rather than security reasons.
- RSA and ElGamal are <u>less efficient and fast</u> to operate than most symmetric encryption algorithms because they involve modular exponentiation.
  - Public key cryptography confined to **key management** and **signature applications**.

Questions?