

COMPUTER SECURITY

PUBLIC KEY CRYPTOGRAPHY
KNAPSACK

Willy Sudiarto Raharjo

Teknik Informatika
Universitas Kristen Duta Wacana

Public-Key Cryptography

- Known as asymmetric-key cryptography
- Encryption and decryption is carried out using two different keys
 - ◆ The two keys in such a key pair are referred to as the **public key** and the **private key**
 - ◆ The use of two keys will solve the key distribution problem
- With public key cryptography, all parties interested in secure communications can publish their public keys and keep the private keys on their system

public key



plaintext

encryption



ciphertext

private key



decryption



plaintext

Public Key Cryptography

- Invented in late 1960s and early 1970s
- Based on
 - ◆ Mathematical structures
 - ◆ Trap door one-way function
- Can be used for
 - ◆ Encryption/Decryption
 - ◆ Digital Signing (Digital Signature)
- Solves key management problem

How Many Keys Needed?



$$\frac{N(N-1)}{2} = 190 \text{ keys}$$

Diffie-Hellman Key Exchange

- Proposed by Diffie & Hellman (1976)
- Enables two parties to exchange a shared key over an unsecure medium without having any prior shared secrets
- Standard: RFC 2631
- Anonymous (non-authenticated) key agreement protocol

Requirement for Diffie-Helman

- It is computationally easy for a party B to generate a pair of public key KU_b and private key KR_b
- It is computationally easy for a sender A, knowing the public key and the message M , to generate ciphertext

$$C = E_{KU_b}(M)$$

- It is computationally easy for a receiver B to decrypt the ciphertext using the private key

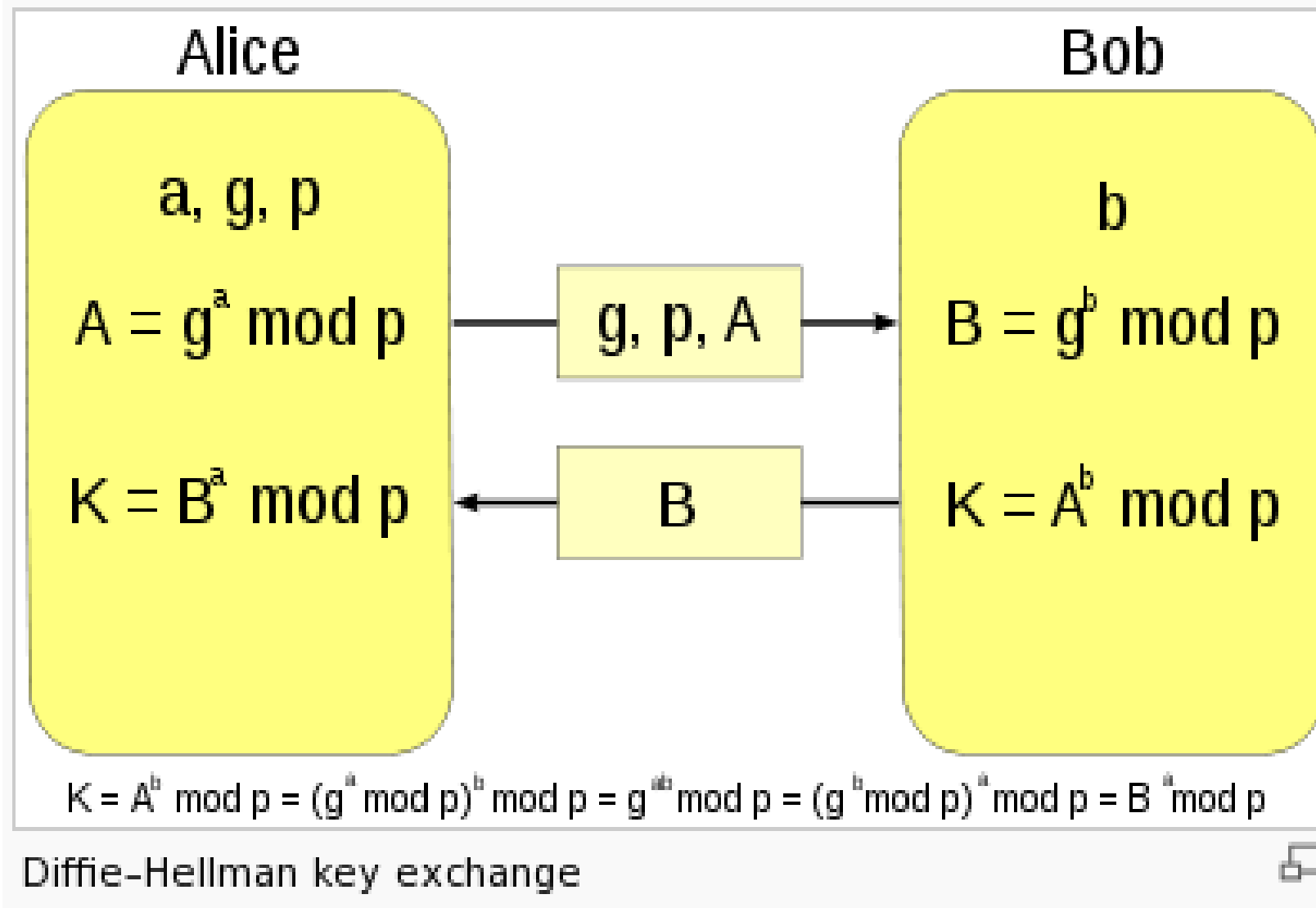
$$M = D_{KR_b}(C) = D_{KR_b}(E_{KU_b}(M))$$

- It is computationally infeasible for opponent, knowing public key KU_b and ciphertext C to recover M
- It is computationally infeasible for opponent, knowing public key Ku_b to determine the private key KR_b

Diffie-Helman Example

- Scenario
 - ◆ Sender: Alice
 - ◆ Recipient: Bob
- Parameters:
 - ◆ p : large prime integer (1024 bits)
 - ◆ g : an integer less than p
 - ◆ a : secret integer for Alice
 - ◆ b : secret integer for Bob

Diffie-Hellman Protocol



Alice		
Secret	Public	Calculus
	p, g	
a		
		$g^a \bmod p$
	...	
	$(g^b \bmod p)^a \bmod p$	

→
←
=

Bob		
Calculus	Public	Secret
	p, g	
		b
	...	
$g^b \bmod p$		
	$(g^a \bmod p)^b \bmod p$	

1. Alice and Bob agree to use a prime number $p=23$ and base $g=5$.
2. Alice chooses a secret integer $a=6$, then sends Bob $A = g^a \bmod p$
 - $A = 5^6 \bmod 23 = 8$.
3. Bob chooses a secret integer $b=15$, then sends Alice $B = g^b \bmod p$
 - $B = 5^{15} \bmod 23 = 19$.
4. Alice computes $s = B^a \bmod p$
 - $19^6 \bmod 23 = 2$.
5. Bob computes $s = A^b \bmod p$
 - $8^{15} \bmod 23 = 2$.

- Both Alice and Bob have arrived at the same value, because g^{ab} and g^{ba} are equal mod p
- a , b and $g^{ab} = g^{ba} \bmod p$ are kept secret
- p , g , $g^a \bmod p$, and $g^b \bmod p$ – are sent in the clear
- Once Alice and Bob compute the shared secret they can use it as an encryption key, known only to them, for sending messages across the same open communications channel

Online Diffie-Hellman Example



Bill's Security Material



Home [Bill's Home]

Diffie-Hellman Example

Diffie-Hellman is a standard method of Alice and Bob being able to communicate, and end up with the same secret encryption key. It is used in many applications. Initially the values of G and N are defined:

G:

N:

Click here first

Next Bob and Alice will calculate an X value and a Y value, respectively:

Bob X Value Alice's Y value

Next ...

Bob's A value Alice's B value

$A = G^x \text{ mod } N$

$B = G^y \text{ mod } N$

and Bob will send his A value to Alice, and Alice will send her B value to Bob, and they now re-calculate the values to generate the same shared key:

Finally ...

Bob's Key Alice's Key

$\text{Key} = B^x \text{ mod } N$

$\text{Key} = A^y \text{ mod } N$

<http://buchananweb.co.uk/security02.aspx>

Knapsack

- Imagine you have a set of different weights which you can use to make any total weight that you need by adding combinations of any of these weights together.
- Given set of n weights

$$W_0, W_1, \dots, W_{n-1}$$

- And Sum S , find $a_0, a_1, \dots, a_{n-1}$ where $a_i \in \{0, 1\}$, so that

$$S = a_0 W_0 + a_1 W_1 + \dots + a_{n-1} W_{n-1}$$

Example

- Weights

85, 13, 9, 7, 47, 27, 99, 86

$$S = 172$$

- Solution:

$$a = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) = (11001100)$$

$$85 + 13 + 47 + 27 = 172$$

Superincreasing Knapsack

- One in which the next term of the sequence is greater than the sum of all preceding terms

3, 6, 11, 25, 46, 95, 200, 411

$$3 + 6 < 11$$

$$3 + 6 + 11 < 25$$

$$3 + 6 + 11 + 25 < 46$$

- Example of non superincreasing knapsack

1, 2, 3, 9, 10, 24

Solving Superincreasing Knapsack

- Take the total weight and compare with largest weight
 - ◆ If total weight $<$ number $\rightarrow$ not in knapsack
 - ◆ If total weight $\geq$ number $\rightarrow$ it's in the knapsack
 - ◆ Subtract the number from total
 - ◆ Compare the next highest number
 - ◆ Repeat until total reaches zero
 - ◆ If zero is not reached, then there's no solution for it

Example

- Superincreasing knapsack

1, 2, 4, 9, 20, 38

$$S = 23$$

- ◆ Does not contain 38 since $38 \geq 23$
 - ◆ Contain the weight 20; leaving 3
 - ◆ Does not contain the weight 9 still leaving 3
 - ◆ Does not contain the weight 4 still leaving 3
 - ◆ Contains the weight 2, leaving 1; which contains the weight 1
- The binary code is therefore 110010

Idea of Knapsack Problem

- Create two different knapsack problems
 - ◆ One is easy to solve, the other is not
 - ◆ Using the easy knapsack, the hard knapsack is derived from it
 - ◆ The hard knapsack becomes the public key
 - ◆ The easy knapsack is the private key

Construct a Knapsack Cryptosystem

- Generate superincreasing knapsack
- Convert superincreasing knapsack into general knapsack
- General knapsack $\rightarrow$ Public Key
- Superincreasing knapsack $\rightarrow$ Private Key

Example

- Superincreasing Knapsack

2, 3, 7, 14, 30, 57, 120, 251
- Convert into general knapsack
 - ◆ Pick multiplier m and modulus n
 - m and n must be coprime
 - $n > \text{sum of all elements in superincreasing knapsack}$
 - ◆ $m = 41$
 - ◆ $n = 491$
- General knapsack is computed using modular multiplication

$$2m = 2 \times 41 = 82 \bmod 491$$

$$3m = 3 \times 41 = 123 \bmod 491$$

$$7m = 7 \times 41 = 287 \bmod 491$$

$$14m = 14 \times 41 = 83 \bmod 491$$

$$30m = 30 \times 41 = 248 \bmod 491$$

$$57m = 57 \times 41 = 373 \bmod 491$$

$$120m = 120 \times 41 = 10 \bmod 491$$

$$251m = 251 \times 41 = 471 \bmod 491$$

General knapsack is

82, 123, 287, 83, 248, 373, 10, 471

Based on previous information

- Public key: general knapsack

82, 123, 287, 83, 248, 373, 10, 471

- Private key: superincreasing knapsack with conversion factor of m

2, 3, 7, 14, 30, 57, 120, 251

and

$$m m^{-1} = 1 \pmod{n}$$

$$41 m^{-1} = 1 \pmod{491} \rightarrow 12$$

Encryption

- A want to send $M = 150$ to B
- Converts into binary $\rightarrow 10010110$
- A uses B's public key to get the ciphertext
82, 123, 287, 83, 248, 373, 10, 471
- $C = 82 + 83 + 373 + 10 = 548$
- 548 is sent to B

Decryption

- B receives 548 from A
- B finds m^{-1} using his private key:
- $Cm^{-1} \bmod n = 548 \times 12 \bmod 491 = 193$
- Recover 193 from private key

2, 3, 7, 14, 30, 57, 120, 251

$$2 + 14 + 57 + 120 = 193$$

Binary = 10010110

Decimal = 150

Practice

Superincreasing Knapsack

1, 2, 4, 10, 20, 40

$m = 31$

$n = 110$

$M = 100100111100101110$

What is the ciphertext?

Answer

Normal Knapsack Sequence

$$31 \bmod(110) = 31$$

$$62 \bmod(110) = 62$$

$$124 \bmod(110) = 14$$

$$310 \bmod(110) = 90$$

$$620 \bmod(110) = 70$$

$$1240 \bmod(110) = 30$$

public key is: {31, 62, 14, 90, 70, 30}

the private key is {1, 2, 4, 10, 20, 40}

$$m^{-1} = 71$$

Answer

- The knapsack contains six weights so we need to split the message into groups of six:
 - ◆ 100100
 - ◆ 111100
 - ◆ 101110
- This corresponds to three sets of weights with totals as follows
 - ◆ $100100 = 31 + 90 = 121$
 - ◆ $111100 = 31 + 62 + 14 + 90 = 197$
 - ◆ $101110 = 31 + 14 + 90 + 70 = 205$
- Ciphertext : 121 197 205

Proof of Decryption

- Use the known m^{-1} number: 71

$$121 \times 71 \bmod(110) = 11 = 100100$$

$$197 \times 71 \bmod(110) = 17 = 111100$$

$$205 \times 71 \bmod(110) = 35 = 101110$$

- The plaintext is 100100111100101110

Practice

Let a private key $(1, 3, 7, 15, 28, 57)$ with modulus $n = 119$ and $m^{-1} = 53$.

Find m , public key, and plaintext for ciphertext $C = 50$