19CSE311- Computer Security- Unit II-Part II

February 1, 2022

Dr. Remya S/Mr. Sarath R
Assistant Professor
Department of Computer Science and Engineering

AMRITA VISHWA VIDYAPEETHAM
DEEMED TO BE UNIVERSITY
School of Engineering

Contents

Introduction

- A Public Key Algorithm
- Only for Key Exchange
- Does NOT Encrypt or Decrypt
- Based on Discrete Logarithms
- Widely used in Security Protocols and Commercial Products

Discrete Logarithms

- What is logarithm?
- $log_{10}100 = 2$ because $10^2 = 100$
- In general if $log_m b = a$ then $m^a = b$, where m is called the base of the logarithm.
- A **discrete logarithm** can be defined for integers only.
- In fact we can define **discrete logarithm** mod p also where p is any prime number.
- The security of the Diffie-Hellman algorithm depends on the difficulty of solving the discrete logarithm problem (DLP) in the multiplicative group of a finite field.

Primitive Roots

- If $x^n = a$ then a is called the n-th root of x.
- For any prime number p, if we have a number a such that powers of 'a mod p' generate all the numbers between 1 to p-1 then a is called the **Primitive Root** of p.
- For any integer b and a primitive root a of prime number p, we can find a unique exponent *i* such that

$$b = a^i modp$$

Diffie-Hellman Algorithm

- Five Parts:
  1. Global Public Elements
  2. User A Key Generation
  3. User B Key Generation
  4. Generation of Secret Key by User A
  5. Generation of Secret Key by User B

Global Public Elements

- q Prime Number
- α $\alpha < q$ and α is a primitive root of q
- The global public elements are also sometimes called the domain parameters.

User A Key Generation

- Select Private Key X_A
  $X_A < q$
- Calculate Public Key Y_A
  $Y_A = \alpha^{X_A}$ mod q

User B Key Generation

- Select Private Key X_B
  $X_B < q$
- Calculate Public Key Y_B
  $Y_B = \alpha^{X_B}$ mod q

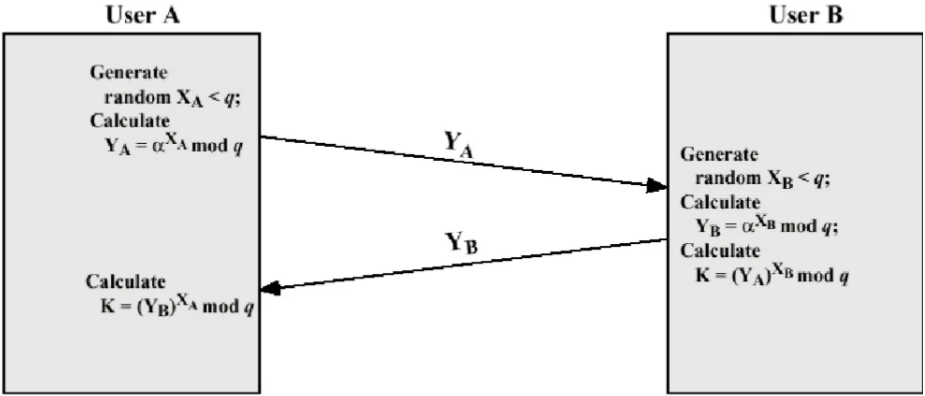
Generation of Secret Key by User A

$$K = (Y_B)^{X_A} \text{mod q}$$

Generation of Secret Key by User B

$$K = (Y_A)^{X_B} \text{mod q}$$

Diffie-Hellman Key Exchange



Diffie-Hellman Example.I

- q=97
- α=5
- X_A=36
- X_B=58
- Y_A=5^{36} mod 97 = 50 mod 97 = 50
- Y_B=5^{38} mod 97 = 44 mod 97 = 44
- $K = (Y_B)^{X_A}$mod q = 44^{36} mod 97 = 75 mod 97 = 75
- $K = (Y_A)^{X_B}$mod q = 50^{58} mod 97 = 75 mod 97 = 75

Diffie-Hellman Example.II

- q=23
- α=5
- X_A=6
- X_B=15
- Y_A=5^6 mod 23 = 8 mod 23 = 8
- Y_B=5^{15} mod 23 = 19 mod 23 = 19
- $K = (Y_B)^{X_A}$mod q = 19^6 mod 23 = 2 mod 23 = 2
- $K = (Y_A)^{X_B}$mod q = 8^{15} mod 23 = 2 mod 23 = 2

Man-in-the-Middle Attack

- Most serious weakness in Diffie-Hellman
- Assume Darth has ability to:
  - **Intercept** messages between Alice and Bob.
  - **Masquerade** as Alice or Bob to send message to the other.
- Darth generates own random value X_D.
- Computes own $\mathbf{Y}_D = \mathbf{q}^{X_D}$ **mod q** from the public values of q and α.
- **Goal**: Trick Alice and Bob into using keys he has created from X_D.

Station-to-Station Key Agreement

- Participants in Diffie-Hellman must **authenticate** their identities
  - Only solution to Man-in-the-Middle attack
- Authentication usually based on certificates:
  - Signed by trusted authorities
  - Contain public keys for participation

Applications

- Diffie-Hellman is currently used in many protocols, namely:
  - Secure Socket Layer(SSL)/ Transport Layer Security(TLS)
  - Secure Shell
  - Internet Protocol Security(IPSec)
  - Public Key Infrastructure(PKI)

Pros & Cons Of Diffie-Hellman Algorithm

- **Advantages**:
  - The sender and receiver have no prior knowledge of each other
  - Communication can be takes place through an insecure channel.
  - Sharing of secret key is safe.
- **Disadvantages**:
  - Can not be used for asymmetric key exchange.
  - Can not be used for signing digital signatures.
  - The nature of the Diffie-Hellman key exchange does make i susceptible to man-in-the-middle attacks since it doesn't authenticate either party involved in the exchange.