Assignment no ICS 3

Roll no 41239

Title: Implementation of Diffie-Hellman key exchange

Problem Definition: Implementation of Diffie-Hellman key exchange

Software Requirements:

Python 3.7,Colab

Hardware Requirement:

8GB RAM, 500 GB HDD,Keyboard,Mouse

Learning Objectives:
Learn Diffie-Hellman key exchange

Theory :

1. Diffie Hellman (DH) key exchange algorithm is a method for securely exchanging cryptographic keys over a public communications channel. Keys are not actually exchanged – they are jointly derived. It is named after their inventors Whitfield Diffie and Martin Hellman.

2. Working of Diffie-Hellman Algorithm:

1. In Public key encryption schemes are secure only if authenticity of the public key is assured.

2. Diffie-Hellman key exchange is a simple public key algorithm.

3. The protocol enables 2 users to establish a secret key using a public key scheme based on discrete algorithms.

4. The protocol is secure only if the authenticity of the 2 participants can be established.

5. There are 2 publicly known numbers :A prime number q and an integer α that is a primitive root of q.

For example:

2 is a primitive root mod 5, because for every number a relatively prime to 5, there is an integer z

such that 2z≡a.

All the numbers relatively prime to 5 are 1, 2, 3, 4, and each of these (mod 5) is itself (for instance 2 (mod 5) = 2):

- 20=1, 1 (mod 5)=1, so 20≡1
- 21=2, 2 (mod 5)=2, so 21≡2
- 23=8, 8 (mod 5)=3, so 23≡3
- 22=4, 4 (mod 5)=4, so 22≡4.

4 is not a primitive root mod 5, because for every number relatively prime to 5 (again, 1, 2, 3, 4) there is not a power of 4 that is congruent. Powers of 4 (mod 5) are only congruent to 1 or 4. There is no power of 4 that is congruent to 2 or 3.

The Diffie-Hellman algorithm is being used to establish a shared secret that can be used for secret communications while exchanging data over a public network using the elliptic curve to generate points and get the secret key using the parameters.

- For the sake of simplicity and practical implementation of the algorithm, we will consider only 4 variables, one prime P and G (a primitive root of P) and two private values a and b.
- P and G are both publicly available numbers. Users (say Alice and Bob) pick private values a and b and they generate a key and exchange it publicly. The opposite person receives the key and that generates a secret key, after which they have the same secret key to encrypt.

Step 1: Alice and Bob get public numbers P = 23, G = 9

Step 2: Alice selected a private key a = 4 and
Bob selected a private key b = 3

Step 3: Alice and Bob compute public values
Alice: x =(9^4 mod 23) = (6561 mod 23) = 6
Bob: y = (9^3 mod 23) = (729 mod 23) = 16

Step 4: Alice and Bob exchange public numbers

Step 5: Alice receives public key y =16 and
Bob receives public key x = 6

Step 6: Alice and Bob compute symmetric keys
Alice: ka = y^a mod p = 65536 mod 23 = 9
Bob: kb = x^b mod p = 216 mod 23 = 9

Step 7: 9 is the shared secret.

Conclusion : Successfully learned and implemented DH key exchange