Experiment 8

Date of Performance : 10-04-2023
Date of Submission : 17-04-2023
SAP Id: 60004200107 **Name :** Kartik Jolapara
Div: B **Batch :** B1

AIM

To implement Diffie Hellman Key exchange protocol. Demonstrate man in middle attack.

THEORY

Diffie Hellman Key exchange

Diffie-Hellman Key Exchange is a cryptographic protocol that allows two parties to establish a shared secret key over an insecure communication channel. The shared key can then be used for encryption, decryption, or other cryptographic operations. The protocol was invented by Whitfield Diffie and Martin Hellman in 1976 and is widely used in modern cryptography.
The basic idea behind the Diffie-Hellman protocol is that both parties agree on a large prime number and a generator, which is a smaller number that generates a cyclic group of numbers modulo the prime. Each party then selects a private key, which is a randomly chosen number, and computes a public key by raising the generator to the power of the private key modulo the prime. The parties exchange their public keys over the insecure channel, and then use them to compute a shared secret key.
The Diffie-Hellman protocol is used in many cryptographic applications, such as secure communication over the Internet (e.g., in the TLS/SSL protocol), key exchange in symmetric encryption schemes (e.g., in the SSH protocol), and digital signatures.

Man in the Middle

A man-in-the-middle (MITM) attack is a type of cyber-attack where an attacker intercepts and alters communications between two parties who believe they are communicating directly with each other. The attacker can eavesdrop on the communication, modify the content of the messages, and even impersonate one or both of the parties. In order to carry out a MITM attack, the attacker must be able to intercept the communication between the two parties. This can be done in several ways, such as by compromising a network device (e.g., a router or switch), by using a rogue access point to intercept wireless communications, or by using malware to intercept communications on a compromised computer.

Once the attacker has intercepted the communication, they can then modify the content of the messages. For example, they may insert malicious code or malware into a download, or modify a financial transaction to redirect funds to their own account.

PROGRAM

a) Diffie Hellman Key exchange Alice Program import socket import random as r

```
def alice():
   host = socket.gethostname()
port = 5000      s =
socket.socket()
   s.bind((host, port))
   s.listen(2)     conn, address = s.accept()
print("Connection from: " + str(address))

   p = int(input("Enter p = "))     g
= int(input("Enter g = "))
conn.send(str(p).encode('ascii'))
conn.send(str(g).encode('ascii'))

   a = r.randint(3, 1000)     Xa =
int(pow(g, a, p))     print("Xa
computed = ", Xa)
conn.send(str(Xa).encode('ascii'))

   Xb = int(conn.recv(1024).decode('ascii'))
print("Xb from Bob = ", Xb)

   Ak = int(pow(Xb, a, p))     print('Secret
key for Alice is = %d' % (Ak))
conn.close()

alice()
```

Bob Program import socket import random as r def bob(): host = socket.gethostname( ) port = 5000 s = socket.socket()

```
   s.connect((host, port))
```

```
   p =
int(s.recv(1024).decode('ascii'))
print("p = ", p)     g =
int(s.recv(1024).decode('ascii'))
print("g = ", g)
   Xa = int(s.recv(1024).decode('ascii'))
print("Xa from Man in the middle = ", Xa)

   b = r.randint(3, 1000)     Xb
= int(pow(g, b, p))
print("Xb computed = ", Xb)
   s.send(str(Xb).encode('ascii'))

   Bk = int(pow(Xa, b, p))     print('Secret
key for Bob is = %d' % (Bk))     s.close()

bob()
```

b) Man in the Middle

Attack Man in the Middle

Program

```
import socket import
random as r

def mitm():     host =
socket.gethostname()     port =
5000     s = socket.socket()
   s.bind((host, port))
   s.listen(10)     alice, address1 = s.accept()
bob, address2 = s.accept()
print("Connection from: " + str(address1))
print("Connection from: " + str(address2))

   p = int(alice.recv(1024).decode('ascii'))     print("p = ", p)     g =
int(alice.recv(1024).decode('ascii'))     print("g = ", g)

   bob.send(str(p).encode('ascii'))
bob.send(str(g).encode('ascii'))

   Xa =
int(alice.recv(1024).decode('ascii'))
print("Xa from Alice = ", Xa)     e =
r.randint(3, 1000)     Xe = int(pow(g, e,
p))     bob.send(str(Xe).encode('ascii'))
```

```
    Xb =
int(bob.recv(1024).decode('ascii'))
print("Xb from Bob = ", Xb)     f =
r.randint(3, 1000)     Xf = int(pow(g, f,
p))     alice.send(str(Xf).encode('ascii'))

    Ak = int(pow(Xa, f, p))     Bk =
int(pow(Xb, e, p))     print("Key
generated by Alice = ", Ak)
print("Key generated by Bob = ", Bk)

mitm()
```

Alice Program

```
import socket import
random as r

def alice():     host =
socket.gethostname()      port =
5000      s = socket.socket()
    s.connect((host, port))

    p = int(input("Enter p = "))
g = int(input("Enter g = "))
    s.send(str(p).encode('ascii'))
    s.send(str(g).encode('ascii'))

    a = r.randint(3, 1000)     Xa
= int(pow(g, a, p))
print("Xa computed = ", Xa)
    s.send(str(Xa).encode('ascii'))

    Xb = int(s.recv(1024).decode('ascii'))
print("Xb from Man in the middle = ", Xb)

    Ak = int(pow(Xb, a, p))     print('Secret
key for Alice is = %d' % (Ak))     s.close()

alice()
```

Bob Program import
socket import
random as r

```
def bob():     host =
socket.gethostname()      port =
5000      s = socket.socket()
   s.connect((host, port))

   p =
int(s.recv(1024).decode('ascii'))
print("p = ", p)     g =
int(s.recv(1024).decode('ascii'))
print("g = ", g)
   Xa = int(s.recv(1024).decode('ascii'))
print("Xa from Man in the middle = ", Xa)

   b = r.randint(3, 1000)     Xb
= int(pow(g, b, p))
print("Xb computed = ", Xb)
   s.send(str(Xb).encode('ascii'))

   Bk = int(pow(Xa, b, p))     print('Secret
key for Bob is = %d' % (Bk))     s.close()

bob()
```

INPUT AND OUTPUT

a) Diffie Hellman Key exchange

```
PS D:\Riya\DJ Sanghvi\Information and Network System\Experiments\Experiment8> python alice.py
Connection from: ('192.168.29.186', 54094)
Enter p = 23
Enter g = 9
Xa computed =  18
Xb from Bob =  12
Secret key for Alice is = 2
```

```
PS D:\Riya\DJ Sanghvi\Information and Network System\Experiments\Experiment8> python bob.py
p =  23
g =  9
Xa from Man in the middle =  18
Xb computed =  12
Secret key for Bob is = 2
```

b) Man in the Middle Attack

```
PS D:\Riya\DJ Sanghvi\Information and Network System\Experiments\Experiment8> python mitm.py
Connection from: ('192.168.29.186', 53986)
Connection from: ('192.168.29.186', 53987)
p =  23
g =  9
Xa from Alice =  2
Xb from Bob =  16
Key generated by Alice =  3
Key generated by Bob =  12
```

```
PS D:\Riya\DJ Sanghvi\Information and Network System\Experiments\Experiment8> python alice.py
Enter p = 23
Enter g = 9
Xa computed =  2
Xb from Man in the middle =  13
Secret key for Alice is = 3
```

```
PS D:\Riya\DJ Sanghvi\Information and Network System\Experiments\Experiment8> python bob.py
p =  23
g =  9
Xa from Man in the middle =  13
Xb computed =  16
Secret key for Bob is = 12
```

CONCLUSION

Thus, we have successfully implemented Diffie Hellman Key exchange protocol and demonstrated man in middle attack.