

Name : Harshal Kodgire
PRN : 2019BTECS00029
Batch : B1
Topic : CNS Assignment 13

Aim : Diffie-helman key exchange Algo

Theory :

The Diffie-Hellman algorithm is one of the most important algorithms used for establishing a shared secret. At the time of exchanging data over a public network, we can use the shared secret for secret communication.

We use an elliptic curve for generating points and getting a secret key using the parameters.

1. We will take four variables, i.e., P (prime), G (the primitive root of P), and a and b (private values).
2. The variables P and G both are publicly available. The sender selects a private value, either a or b, for generating a key to exchange publicly. The receiver receives the key, and that generates a secret key, after which the sender and receiver both have the same secret key to encrypt

Code :

```
#include <bits/stdc++.h>
using namespace std;

long long powM(long long a, long long b, long long n)
{
    if (b == 1)
        return a % n;
    long long x = powM(a, b / 2, n);
    x = (x * x) % n;
    if (b % 2)
        x = (x * a) % n;
    return x;
}

bool checkPrimitiveRoot(long long alpha, long long q)
{
    map<long long, int> m;

    for (long long i = 1; i < q; i++)
    {
        long long x = powM(alpha, i, q);
```

```

        //cout << x << endl;
        if (m.find(x) != m.end())
            return 0;
        m[x] = 1;
    }
    return 1;
}

int main()
{
    long long q, alpha;

    q = 71; // A prime number q is taken
    alpha = 7; // A primitive root of q

    if (checkPrimitiveRoot(alpha, q) == 0)
    {
        cout << "alpha is not primitive root of q";
        return 0;
    }
    else
    {
        cout << alpha << " is private root of " << q << endl;
    }

    long long xa, ya;
    xa = 4; // xa is the chosen private key
    ya = powM(alpha, xa, q); // public key of alice
    cout << "\n private key of alice is " << xa << endl;
    cout << "\n public key of alice is " << ya << endl << endl;

    long long xb, yb;
    xb = 3; // xb is the chosen private key
    yb = powM(alpha, xb, q); // public key of bob
    cout << "\n private key of bob is " << xb << endl;
    cout << "\n public key of bob is " << yb << endl << endl;

    //key generation
    long long k1, k2;
    k1 = powM(yb, xa, q); // Secret key for Alice
    k2 = powM(ya, xb, q); // Secret key for Bob

```

```
cout << "\n generted key by a is " << k1 << endl;  
cout << "\n generted key by b is " << k2 << endl << endl;  
  
return 0;  
}
```

Output:

```
D:\WCE_ENGINEERING\BTECH_SEM1\CNS lab\LA2>g++ assignment12_diffHell.cpp  
  
D:\WCE_ENGINEERING\BTECH_SEM1\CNS lab\LA2>a.exe  
7 is private root of 71  
  
private key of alice is 4  
public key of alice is 58  
  
private key of bob is 3  
public key of bob is 59  
  
generted key by a is 4  
generted key by b is 4  
  
D:\WCE_ENGINEERING\BTECH_SEM1\CNS lab\LA2>
```