

# **IT NETWORK SECURITY**

## **LAB WORK- DIFFIE HELLMAN**

### **EXCHANGE**

NAME: **OJASVI SINGH CHAUHAN**

ROLL NO. **R134218111**

BRANCH: BTECH CSE-**CSF-B3**

SAP ID: **500068394**

### **INDEX**

- AIM
- INPUT/OUTPUT
- THEORY
- ALGORITHM
- CODE
- OUTPUT

# **EXPERIMENT – 4**

## **DIFFIE HELLMAN KEY EXCHANGE**

**AIM:** To implement Diffie Hellman Key Exchange

**INPUT:** PRIME NUMBER – Q , ALPHA- PRIMITIVE ROOT FOR Q

**OUTPUT:** KEYS FOR SENDER AND RECIEVER

**THEORY:** **Diffie-Hellman key exchange** is a method of securely exchanging cryptographic keys over a public channel and was one of the first public-key protocols as conceived by Ralph Merkle and named after Whitfield Diffie and Martin Hellman. DH is one of the earliest practical examples of public key exchange implemented within the field of cryptography.

Traditionally, secure encrypted communication between two parties required that they first exchange keys by some secure physical means, such as paper key lists transported by a trusted courier. The Diffie–Hellman key exchange method allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure channel. This key can then be used to encrypt subsequent communications using a symmetric key cipher.

Diffie–Hellman is used to secure a variety of Internet services. However, research published in October 2015 suggests that the parameters in use for many DH Internet applications at that time are not strong enough to prevent compromise by very well-funded attackers, such as the security services of large governments.

### **ALGORITHM:**

Suppose users A and B wish to exchange the key.

1.  $k = (Y_A)^{x_B} \bmod q$  -> same as calculated by B

2. Global Public Elements

**q; prime number**

**$\alpha$ ;  $\alpha < q$  and it is primitive root of q**

### **3. USER A KEY GENERATION**

|   |                         |  |
|---|-------------------------|--|
| <b>Select Private key</b>                         | <b><math>X_A</math></b> | <b><math>X_A &lt; q</math></b>                 |
| <b>Calculation of Public key <math>Y_A</math></b> |                         | <b><math>Y_A = \alpha^{X_A} \bmod q</math></b> |

### **4. USER B KEY GENERATION**

|   |                         |  |
|---|-------------------------|--|
| <b>Select Private key</b>                         | <b><math>X_B</math></b> | <b><math>X_B &lt; q</math></b>                 |
| <b>Calculation of Public key <math>Y_B</math></b> |                         | <b><math>Y_B = \alpha^{X_B} \bmod q</math></b> |

**Calculation of Secret Key by A**  
 **$k = (Y_B)X_A \bmod q$**

### **5. Calculation of Secret Key by B**

**$k = (Y_A)X_B \bmod q$**

The result is that two sides have exchanged a secret value.

### **CODE:**

```
#include<stdio.h>
long long int power(int a,int b,int mod)
{
    long long int t;
    if(b==1)
        return a;
    t=power(a,b/2,mod);
    if(b%2==0)
        return (t*t)%mod;
    else
        return (((t*t)%mod)*a)%mod;
}
long long int calculateKey(int a,int x,int n)
{
    return power(a,x,n);
}
int main()
{
```

```

int n,g,x,a,y,b;

printf("Enter the value of n and g : ");
scanf("%d%d",&n,&g);

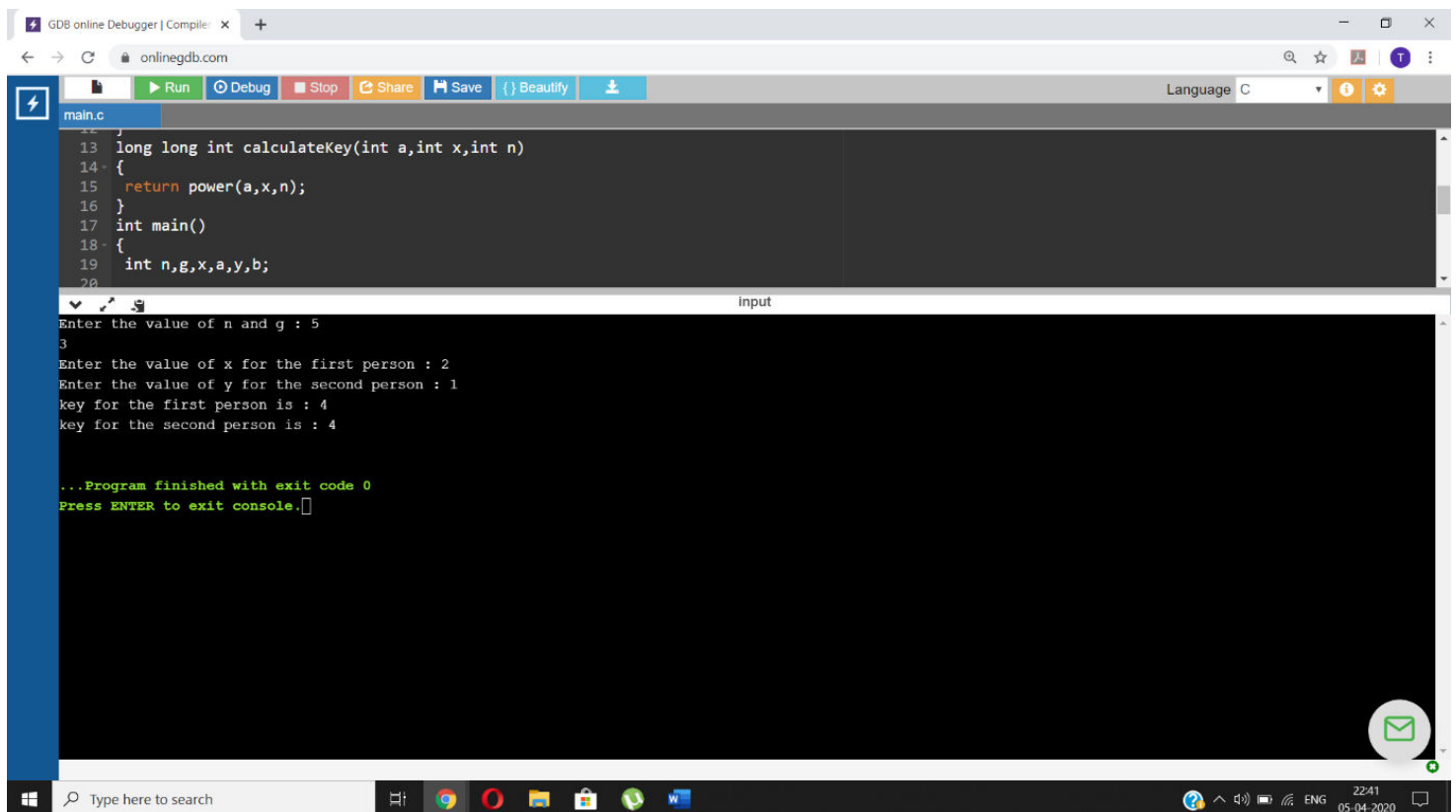
printf("Enter the value of x for the first person : ");
scanf("%d",&x);
a=power(g,x,n);

printf("Enter the value of y for the second person : ");
scanf("%d",&y);
b=power(g,y,n);

printf("key for the first person is : %lld\n",power(b,x,n));
printf("key for the second person is : %lld\n",power(a,y,n));
return 0;
}

```

### OUTPUT SNAPSHOT:



The screenshot shows the GDB online Debugger interface. The code editor displays the following C code:

```

main.c
13 long long int calculateKey(int a,int x,int n)
14 {
15     return power(a,x,n);
16 }
17 int main()
18 {
19     int n,g,x,a,y,b;
20

```

The console output shows the program's execution:

```

Enter the value of n and g : 5
3
Enter the value of x for the first person : 2
Enter the value of y for the second person : 1
key for the first person is : 4
key for the second person is : 4

...Program finished with exit code 0
Press ENTER to exit console.

```

The interface includes a toolbar with buttons for Run, Debug, Stop, Share, Save, and Beautify. The language is set to C. The bottom status bar shows the time as 22:41 on 05-04-2020.