Source Code :

```
import random
from time import sleep

if __name__ == "__main__":
    p = int(input("Enter value for p (it must be prime number ) : "))
    g = int(input("Enter value for g i.e. base : "))

    alice_a = random.randint(1, 20)
    alice_A = (g ** alice_a) % p
    print("Alice sent key {} to bob".format(alice_A))
    sleep(3)

    bob_b = random.randint(1, 20)
    bob_B = (g ** bob_b) % p
    print("Bob sent key {} to Alice".format(bob_B))
    sleep(3)


    print("Darth got keys {} and {} from both and he didn't sent the actula
keys".format(alice_A, bob_B))
    sleep(3)

    eve_c = random.randint(1, 20)
    eve_d = random.randint(1, 20)

    eve_C = (g ** eve_c) % p
    eve_D = (g ** eve_d) % p
    print("Darth sent {} key to Bob and {} key to Alice".format(eve_C, eve_D))
    sleep(3)

    alice_Calculates = (g ** (eve_d * alice_a )) % p
    bob_calculates = (g ** ( bob_b * eve_c )) % p

    print("Now alice calculates last key {} and bob calculates {} key after Darth
intervention.".format(alice_Calculates, bob_calculates))
    sleep(3)

    alice_acutal_key = (bob_B ** alice_a) % p
    bob_actual_key = (alice_A ** bob_b) % p
    print("Actually alice should have key {} and bob should have key {} if Darth
didn't intervine".format(alice_acutal_key, bob_actual_key))
```

Output :

```
Enter value for p (it must be prime number ) : 23
Enter value for g i.e. base : 12
Alice sent key 8 to bob
Bob sent key 4 to Alice
Darth got keys 8 and 4 from both and he didn't sent the actula keys
Darth sent 9 key to Bob and 13 key to Alice
Now alice calculates last key 2 and bob calculates 2 key after Darth intervention.
Actually alice should have key 9 and bob should have key 9 if Darth didn't intervine

Process finished with exit code 0
```

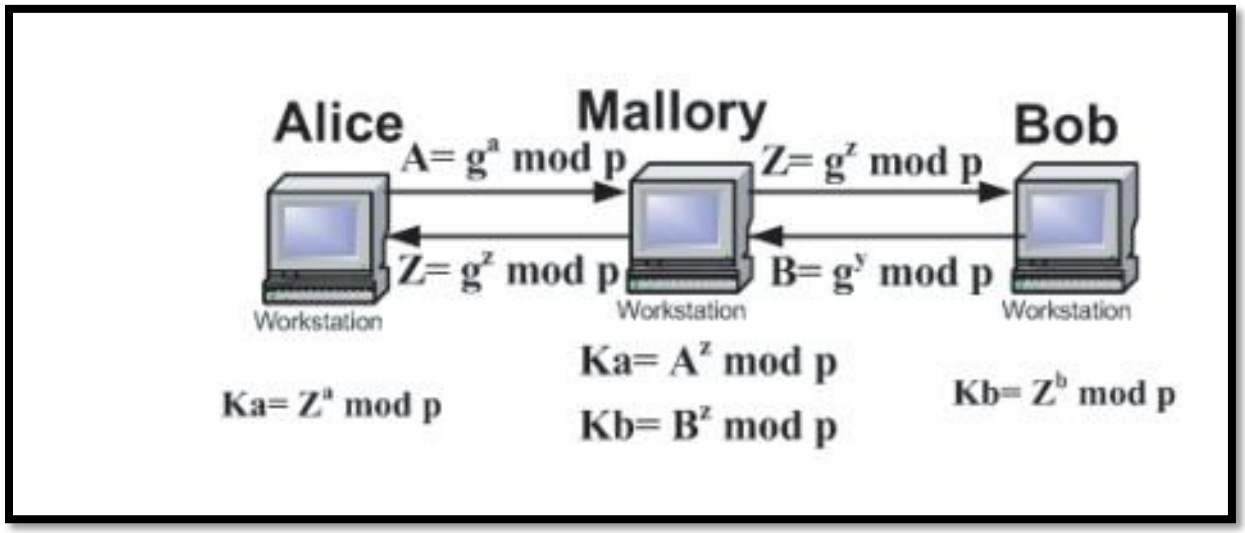

Fig. Man in middle Attack against Diffe-Hellman algorithm