GOA COLLEGE OF ENGINEERING

"Bhausaheb Bandodkar Technical Education Complex"

Experiment No: 5

GSM Authentication

Aim: To write and execute a program to implement GSM authentication

Theory:

The GSM network authenticates the identity of the subscriber through the use of a challenge- response mechanism. A 128-bit Random Number (RAND) is sent to the MS. The MS computes the 32-bit Signed Response (SRES) based on the encryption of the RAND with the authentication algorithm (A3) using the individual subscriber authentication key (Ki). Upon receiving the SRES from the subscriber, the GSM network repeats the calculation to verify the identity of the subscriber.

The individual subscriber authentication key (Ki) is never transmitted over the radio channel, as it is present in the subscriber's SIM, as well as the AUC, HLR, and VLR databases. If the received SRES agrees with the calculated value, the MS has been successfully authenticated and may continue. If the values do not match, the connection is terminated and an authentication failure is indicated to the MS.

The calculation of the signed response is processed within the SIM. It provides enhanced security, as confidential subscriber information such as the IMSI or the individual subscriber authentication key (Ki) is never released from the SIM during the authentication process.

Code:

Client Program

```
import socket
ki=181105010
port = 12345
s = socket.socket()

def  A3(RAND,ki):
   return RAND^ki

s.connect((’127.0.0.1’, port))

RAND=int(s.recv(1024).decode())
result=A3(RAND,ki)
print("A3 authentication result on client:",result)
s.send(str(result).encode())

s.close()
```

Server Program

```
import socket
from random import randint
```

```
maxBufSize=1000
ki=181105010
def A3(RAND,ki):

    return RAND^ki

s = socket.socket()
print ("Socket successfully created")

port = 12345

s.bind((’’, port))
print ("socket binded to %s" %(port))

s.listen(5)
print ("socket is listening")

while True:
    c, addr = s.accept()
    print (’Got connection from’, addr )
    RAND=randint(1000,100000)

    c.send(str(RAND).encode())
    SRES_s=A3(RAND,ki)
    SRES_c=int(c.recv(maxBufSize).decode())
    if SRES_s==SRES_c:

        print("A3 authentication result on server:",SRES_s)

        print("Therefore client is authenticated")
    else:

        print("Client could not be authenticated")
    c.close()
```

Output:

Server output

```
$ p server.py
Socket successfully created
socket binded to 12345
socket is listening
Got connection from (’127.0.0.1’, 54938)
```

A3 authentication result on server: 181029307 Therefore client is authenticated

Client output

$ p client.py
A3 authentication result on client: 181029307

Conclusion: A program to implement and execute GSM authentication was successfully written and executed.