Common DDoS Attack

Jayesh Patel
(647-271-9971)
Toronto, ON
jay.net.in@gmail.com

This DoS and DDoS Analysis project as a research project written in python. A Denial-of-service attack (DoS) of DDoS is an attempt to make a computer and network resources unavailable to its intended users. DoS Attacks typically target sites or services hosted on high profile web servers such as bank, credit card payment gateways and even root name servers. There are two general forms of DoS attacks: those that crash services and those that flood services.

One Common method of attack involves the target machine with external communications requests, such that it cannot responded to legitimate traffic, or response so slowly as to be rendered effectively unavailable. In General term DoS Attacks are implemented by either forcing the targeted computer to reset, or consuming its resources so that it can not longer provide its intended services.

In This project we cover common DoS attack introduction and also provide mechanism to run DoS and DDoS Attack. (This is only for testing purpose) we use scapy python library and other tools for these attack.

Project Details :

Step 1 : Install Ubuntu / Kali / BackTrack Linux :

apt-get install nikto

apt-get install sushi

apt-get install ssh

apt-get install tcpdump graphviz imagemagick python-gnuplot

sudo apt-get install slowhttptest

apt-get install subversion

apt-get install aclocal

apt-get install automaker

pip install scapy

pip install iptools

pip install ipaddress

pip install ipaddr

pip install pinject

 # Git clone "https://github.com/umasolution/DDoS.git"

root@bt:~/dos# python run.py -h

usage: run.py [-h] c_num victim_ip victim_port source_ip source_port

Pass Arguments

positional arguments:

c_num   Enter Number of request

victim_ip   Enter Victim IP

victim_port Enter Victim Port

source_ip   Enter Source IP

source_port Enter Source Port

python run.py 10 192.168.10.1 100 random 80

ls

run.py (main file)

scapy_dos.py (scapy file where we mentioned all attack)

slowhttptest (Tools for DoS)

tool_dos.py (Tools feel where we mentioned all tools based attack)

Please refer README File for use script.

Attack List

SYN Flood :

When a connection is made from client to server through TCP it is initialise with three way handshake.

1. client send SYN to server
2. server send SYN-ACK to client
3. client sends ACK to server

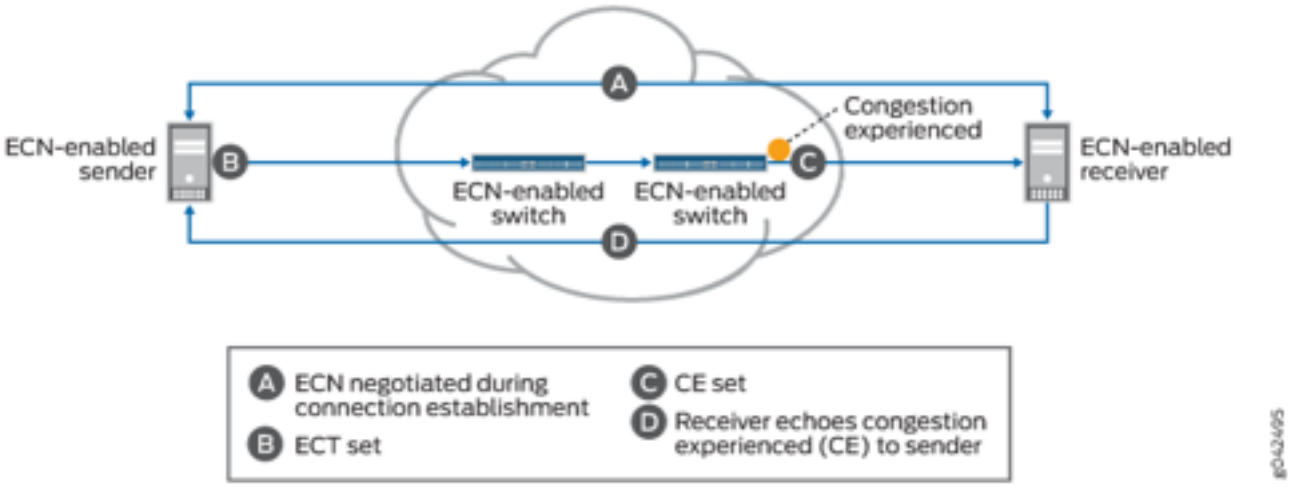
This three way handshake establish rest of connection between the client and server. When performing SYN Flood, you only performing first two part of the three way handshake.

SYN-ACK Flood :

In SYN-ACL Flood attack, Attacker sends spoofed ACK packets at very high packet rates that fail to belong to any current session within the firewall's state-table and/or server's connection list.The ACK flood attack exhausts victim's firewalls by forcing state-table lookups and depletes server resources used to match these incoming packets to an existing flow.

TOS Flood :

To perform this kind of attack, malefactors use TOS Field of the IP packet header. In Ethernet networks this field is used for Explicit congestion Notification (ECN) and setting the traffic service priority (DiffServ). There are two different TOS Flood Scenarios. In First case attacker fakes the ECN filed of the packets. creation the illusion of the congested network, and the server automatically limits the traffic capacity of certain transmission. And In second case attacker use Diffserve flags to change traffic priority.



Two TCP Flags ECN and ECE.

Sender – –-(IP)ECT––> Router –––(IP)CE–––-> Receiver
Sender – –-(TCP)ECE––> Router –––(TCP)ECE<–––- Receiver
Sender – –-(TCP)CWR––> Router –––(TCP)CWR–––-> Receiver

ECT - ECN Capable Transport
CE - congestion Experienced
CWR - Congestion Windows Reduce
ECE - ECN Echo

NTP Flood :

In a NTP Flood, attackers use NTP as a variant of a UDP flood. Attacker send valid but spoofed NTP Request packets at a very high packet rate and from a very large group of source IP addresses. since these appear as valid requests. The victim NTP server processed to responded to all requests. The NTP server can be overwhelmed by the vast number of requests. This attack consume large amount of network resources that exhausts the NTP infrastructure until it goes down.

Smurf Attack :

In a Smurf Attack, attackers send large numbers of ICMP packets with the incited spoofed source IP Address and are broadcast to a computer network using an IP Broadcast address. This cause all hosts on the network to reply to ICMP request, causing significant.

Fraggle Attack :

In a Fraggle Attack, attackers send spoofed UDP packets instead of ICMP echo reply (ping) packets to broadcast address of large network resulting in a denial of service.

NTP Amplified (Reflective) :

Attacker spoofing a victim NTP Infrastructure and use Open NTP Servers, which send small requests resulting in a very high-volume of NTP response.

DNS Amplified (Reflective) :

Attacker spoofing a victim DNS Infrastructure and use DNS Servers, which send small requests resulting in a very high-volume of DNS response.

SNMP Amplified (Reflective) :

Attacker spoofing a victim SNMP Infrastructure, which send small requests resulting in a very high-volume of SNMP response.

DNS Flood :

In a DNS Flood, attackers use DNS as a variant of a UDP flood. Attacker send valid but spoofed DNS request packet at very high packet rate and from very large group of source IP addresses. The victim's DNS server proceeds to responses to all requests. The DNS server can be overwhelmed by the vast number of requests. This attack consume large amount of network resources that exhausts the DNS infrastructure until it goes offline.

RST Flood :

In RST Flood, attacker send highly-spoofed RST packets at extremely high rate that do not belong to any session within the firewall state-table and server's session tables. The RST DDoS attack exhausts a victim's firewalls and/or servers by depleting its systems resources used to look up and match these incoming packets to an existing session.

FIN Flood :

in a RST Flood, attacker send highly-spoofed FIN packets at extremely high rate that do not belong to any session within the firewall state-table and server's session tables. The FIN DDoS attack exhausts a victim's firewall and/or servers by depleting its systems resources used to look up and match these incoming packets to an existing session.

Same Source/Dest Flood (LAND Attack) :

In LAND DDoS Attack, a victim receives spoofed SYN packets at a very high rate the victim IP ranges same in both source and destination IP fields in the IP Header. This attack exhausts a victim's firewall servers by exhausting a system resources use to computer these protocol violation.

UDP Flood :

In UDP Flood, DDoS attacker send highly-spoofed packets at very high packet rate using a large source IP range. The victim network is overwhelmed by large number of UDP Packets. This attack normally consume network resources and available bandwidth.

ICMP Flood :

In an ICMP Flood attacker send highly-spoofed ICMP packets at large enough volumes to flood a network. The victim network resources are overwhelmed by large number of incoming ICMP packets. The attacker consume resources and available bandwidth.

ICMP Fragmentation Flood :

In an ICMP Fragmentation Flood, attacker send highly-spoofed, large fragment ICMP Packets are very high packets rates and these packets can not be reassembled.

PING Flood :

In Ping Flood, attacker use “ping” which is variant of ICMP and send high-spoofed ping (ICMP echo request) packets at very high rate and from random source ip ranges or as the IP address of the victim. Attacker can consume all network resources bandwidth exhausted the network until it goes offline.

PING and Amplification :

You send ICMP Echo Request with the source address spoofed to that of your target, to the broadcast address of several routers. The systems behind those routers all send their ICMP echo replies to that of your target.

TCP Null

in TCP NULL Attack, attacker send packets that have the no TCP segment flags set which is invalid. This type of segment may be used in reconnaissance, such as a port scanning. You also try XMAS Scan, FIN Scan,

ACK or ACK-PUSH Flood

In ACK or ACK-PUSH Flood, attackers send spoofed ACK (or ACK-PUSH) Packets are very high rates that failed to belong to any current session within the firewall’s state-table and/or server’s connection list. ACK Flood exhausts a victim firewalls by forcing state table lookups and servers by depleting their system resources use to match these incoming packets to an existing flow.

IP NULL :

In IP NULL Attack, Attacker send packets whereby the IPv4 header field use to specify which transport protocol which being use in its payload (like TCP or UDP) and sets this field to a value of zero. Firewall configure for just TCP, UDP, and ICMP may allow this type of packet through. If these type of packet arrived as a flood, a victim server cpu resources may be wasted handling these packets.

Slow Read Attack :

In Slow Read DDoS Attack, attacker send valid TCP-SYN packet and perform TCP Three-Way Handshakes with the victim machine to establish valid sessions between the attacker and victim. The attacker first establish large number of valid sessions and begins to request to download a document or large object from each attaching machine. Once the download begins the attacking machine begin to slow down the ACK of received packet. The attacker will continue to slow down the receipt of packet, which consume excess resources on server.

Slowloris Attack :

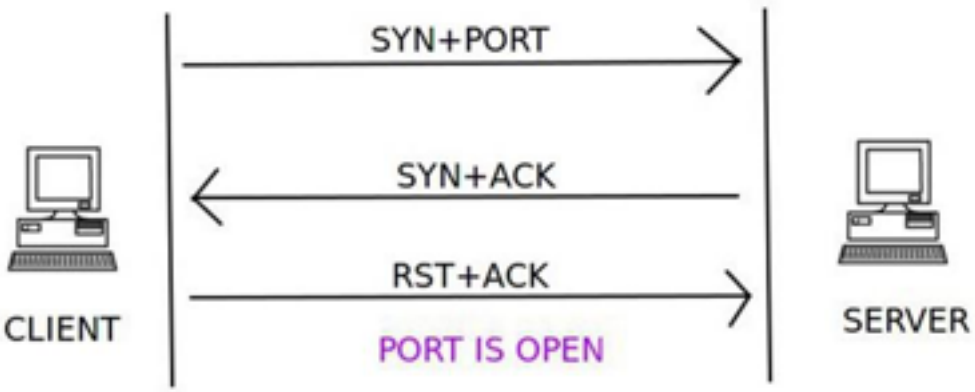
Slowloris attack send partial request to target server, opening connection, then sending header augmenting but never completing the request. Slow HTTP POST send header to signal who much data to be sent, but sends the data very slowly using thousand of HTTP POST Connection with server.

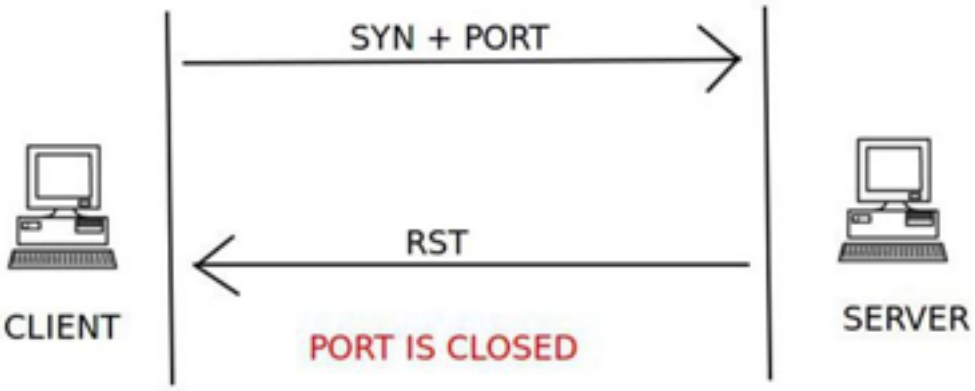
Slow Session Attack :

In Slow Session Attack, attacker send valid TCP-SYN packet and perform TCP Three Way handshakes with the victim to establish valid session between attacker and victim. The attacker first establish large number of valid sessions, then slowly response with an ACK packet and incomplete requests to keep the session open for long time of period.

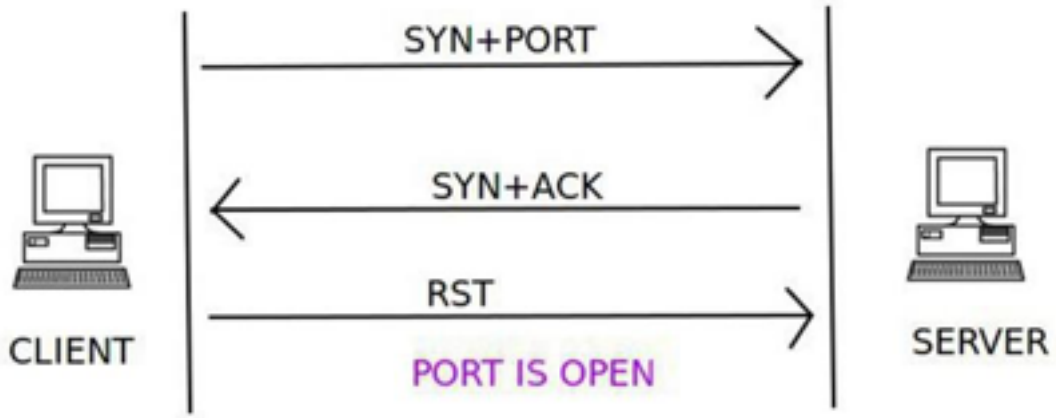
Scanning Technique

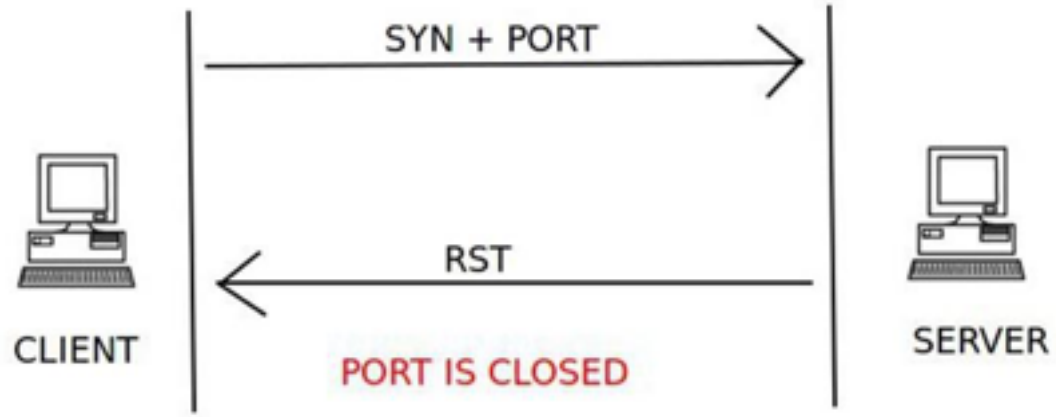
TCP Connect Scan :



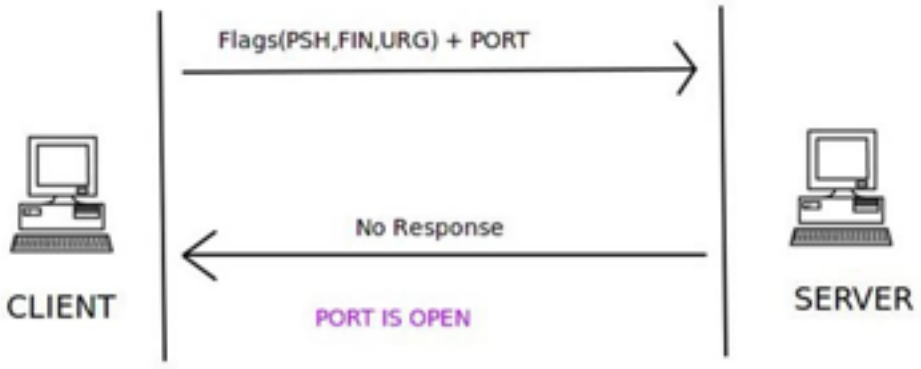


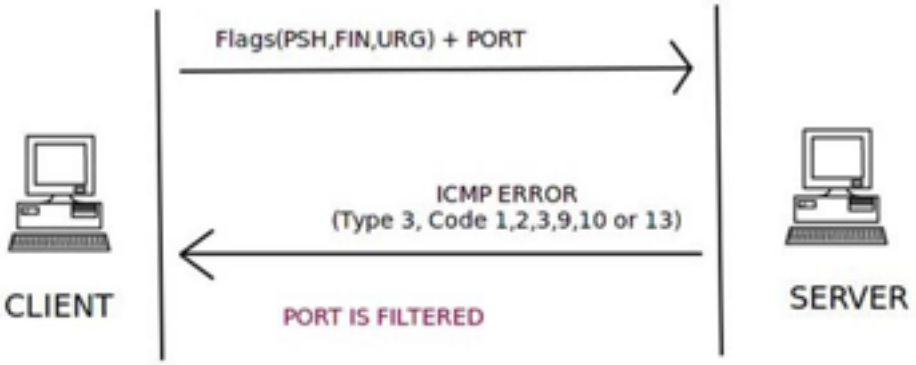
TCP Stealth Scan :

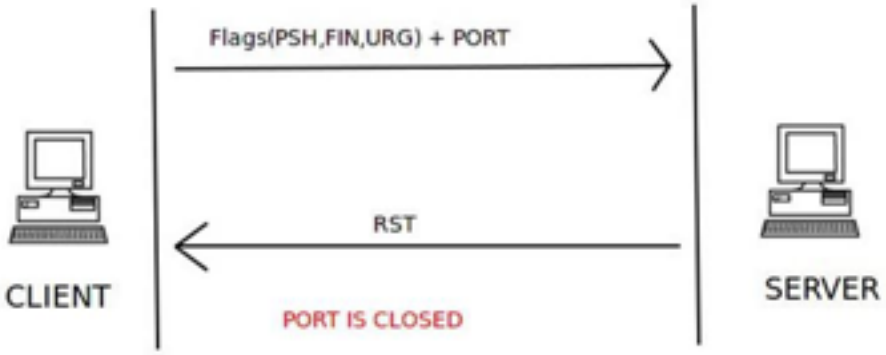

SYN+PORT
SYN+ACK
RST
CLIENT
SERVER
PORT IS OPEN


SYN + PORT
RST
CLIENT
SERVER
PORT IS CLOSED

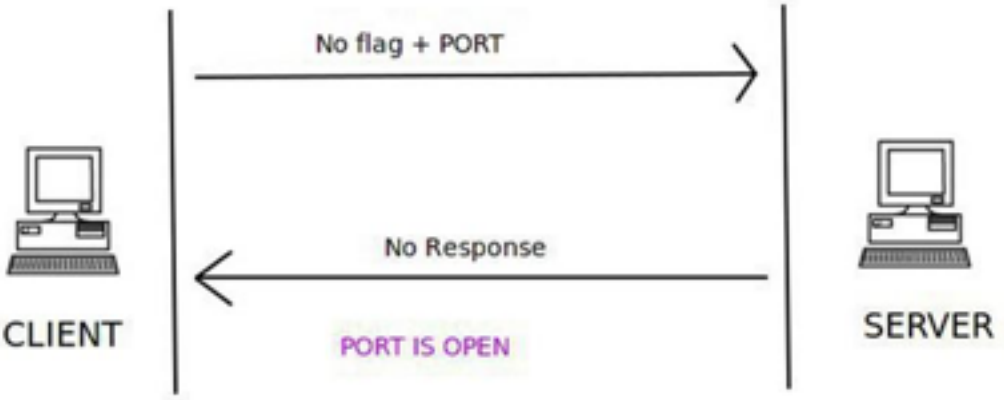
XMAS Scan :

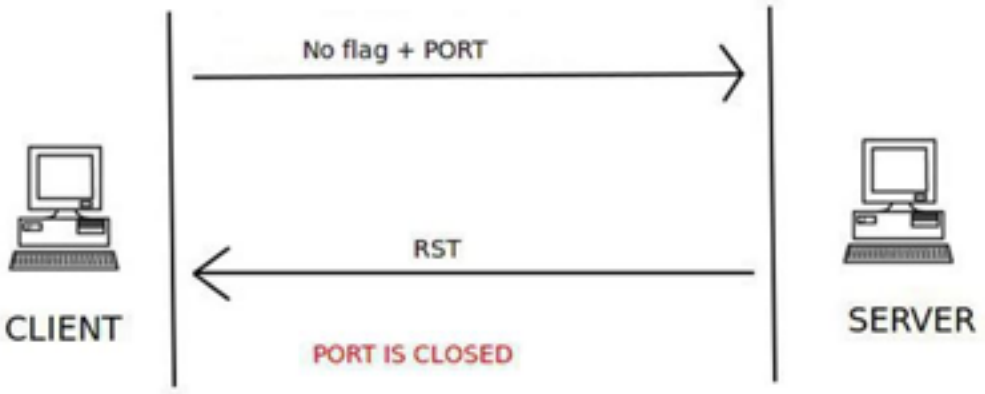

Flags(PSH,FIN,URG) + PORT
No Response
CLIENT
PORT IS OPEN
SERVER

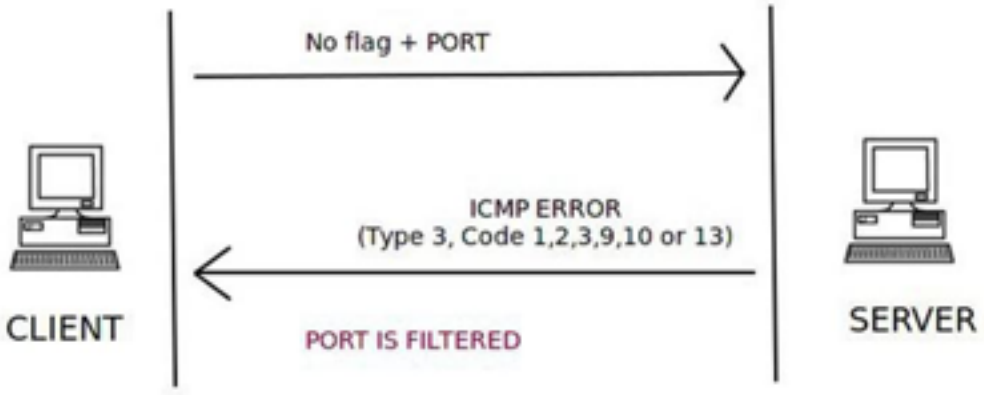

Flags(PSH,FIN,URG) + PORT
ICMP ERROR
(Type 3, Code 1,2,3,9,10 or 13)
CLIENT
PORT IS FILTERED
SERVER


Flags(PSH,FIN,URG) + PORT
RST
CLIENT
PORT IS CLOSED
SERVER

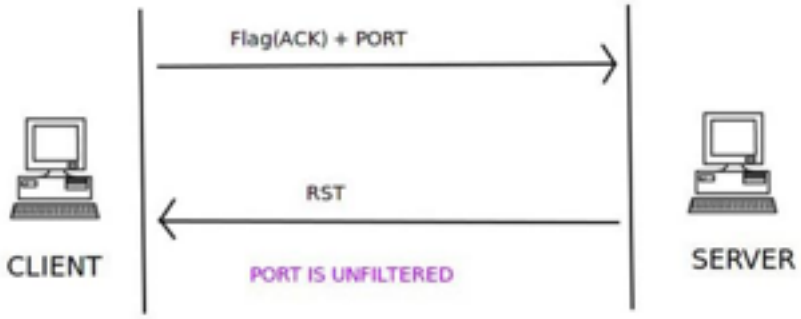
NULL Scan :

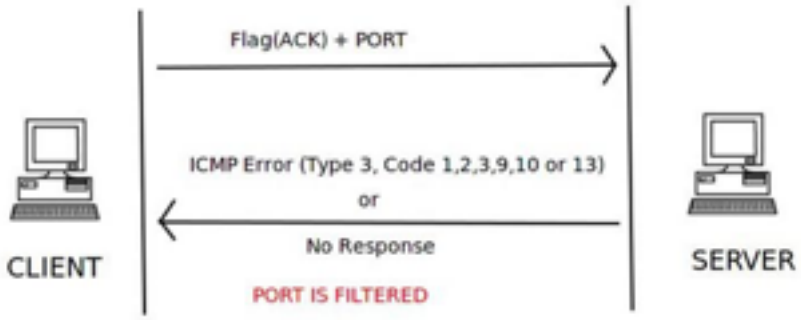

No flag + PORT
No Response
CLIENT
SERVER
PORT IS OPEN


No flag + PORT
RST
CLIENT
SERVER
PORT IS CLOSED

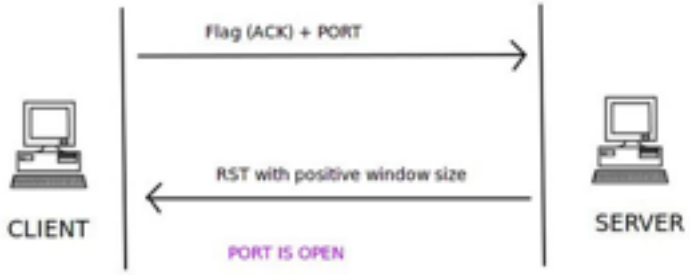

No flag + PORT
ICMP ERROR
(Type 3, Code 1,2,3,9,10 or 13)
CLIENT
SERVER
PORT IS FILTERED

TCP ACK Scan :

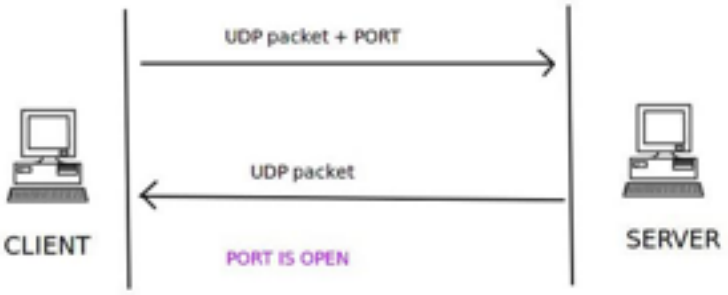


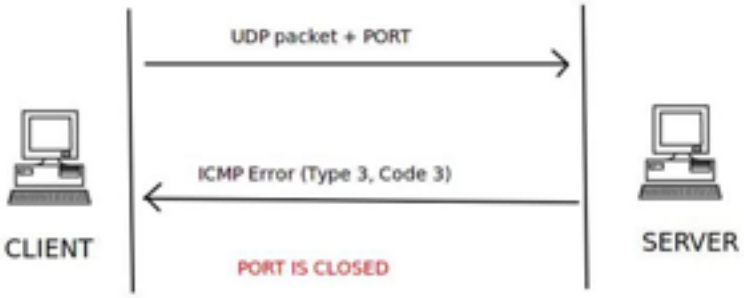


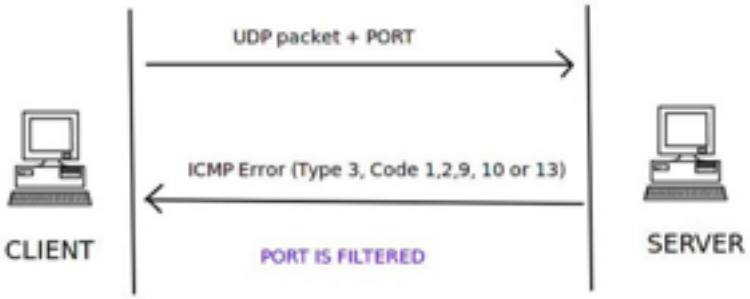
TCP Windows Scan :



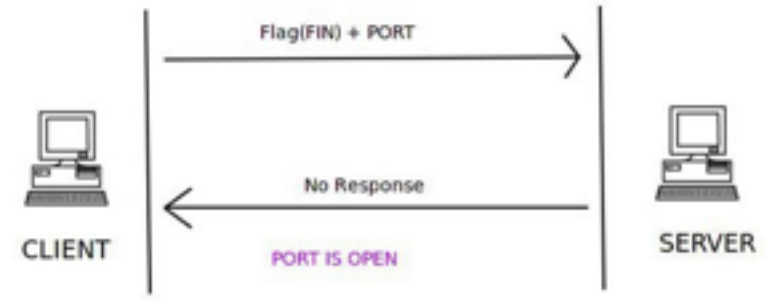
UDP Scan :

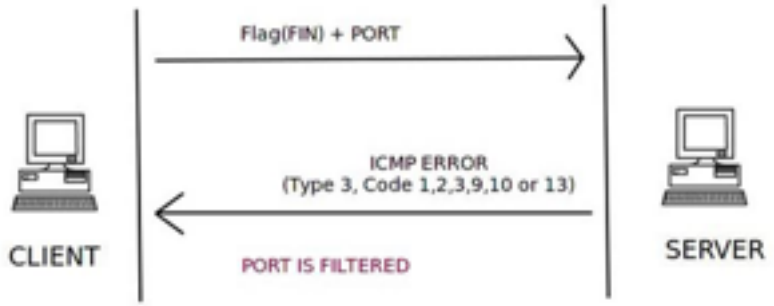


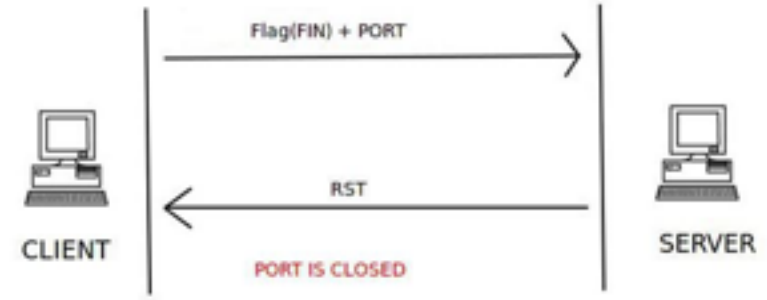




FIN Scan :


Flag(FIN) + PORT
No Response
CLIENT
SERVER
PORT IS OPEN


Flag(FIN) + PORT
ICMP ERROR
(Type 3, Code 1,2,3,9,10 or 13)
CLIENT
SERVER
PORT IS FILTERED


Flag(FIN) + PORT
RST
CLIENT
SERVER
PORT IS CLOSED