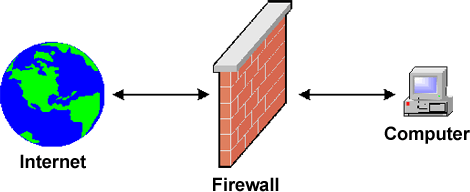
**Practical – 4**

**Aim:** Understand the concept of firewall and configure the State full Packet Inspection (SPI) firewall IPTABLES.

**Firewall:**

* A firewall is a network security system, either hardware or software based, that controls incoming and outgoing network traffic based on a set of rules. Acting as a barrier between a trusted network and other un-trusted networks such as the Internet or less-trusted networks such as a retail merchant's network outside of a cardholder data environment a firewall controls access to the resources of a network through a positive control model. This means that the only traffic allowed onto the network defined in the firewall policies all other traffic is denied.
* The National Institute of Standards and Technology (NIST) 800-10 divides firewalls into three basic types:
  1. Packet filters
  2. Stateful inspection
  3. Proxys

1. **Packet filters**

The earliest firewalls functioned as packet filters, inspecting the packets that are transferred between computers on the Internet. When a packet passes through a packet-filter firewall, its source and destination address, protocol, and destination port number are checked against the firewall's rule set. Any packets that aren't specifically allowed onto the network are dropped (i.e., not forwarded to their destination). For example, if a firewall is configured with a rule to block Telnet access, then the firewall will drop packets destined for TCP port number 23, the port where a Telnet server application would be listening.

Packet-filter firewalls work mainly on the first three layers of the OSI reference model (physical, data-link and network), although the transport layer is used to obtain the source and destination port numbers. While generally fast and efficient, they have no ability to tell whether a packet is part of an existing stream of traffic. Because they treat each packet in isolation, this makes them vulnerable to spoofing attacks and also limits their ability to make more complex decisions based on what stage communications between hosts are at.

1. **Stateful firewalls**

In order to recognize a packet's connection state, a firewall needs to record all connections passing through it to ensure it has enough information to assess whether a packet is the start of a new connection, a part of an existing connection, or not part of any connection. This is what's called "stateful packet inspection." Stateful inspection was first introduced in 1994 by Check Point Software in its FireWall-1 software firewall, and by the late 1990s, it was a common firewall product feature. This additional information can be used to grant or reject access based on the packet's history in the state table, and to speed up packet processing; that way, packets that are part of an existing connection based on the firewall's state table can be allowed through without further analysis. If a packet does not match an existing connection, it's evaluated according to the rule set for new connections.

1. **Proxy firewalls**

Firewall proxy servers also operate at the firewall's application layer, acting as an intermediary for requests from one network to another for a specific network application. A proxy firewall prevents direct connections between either sides of the firewall; both sides are forced to conduct the session through the proxy, which can block or allow traffic based on its rule set. A proxy service must be run for each type of Internet application the firewall will support, such as an HTTP proxy for Web services.

* **IP Tables**

IPtables is an extremely flexible firewall utility built for Linux operating systems. IPtables is a command-line firewall utility that uses policy chains to allow or block traffic. When a connection tries to establish itself on our system, iptables looks for a rule in its list to match it to. If it doesn’t find one, it resorts to the default action.

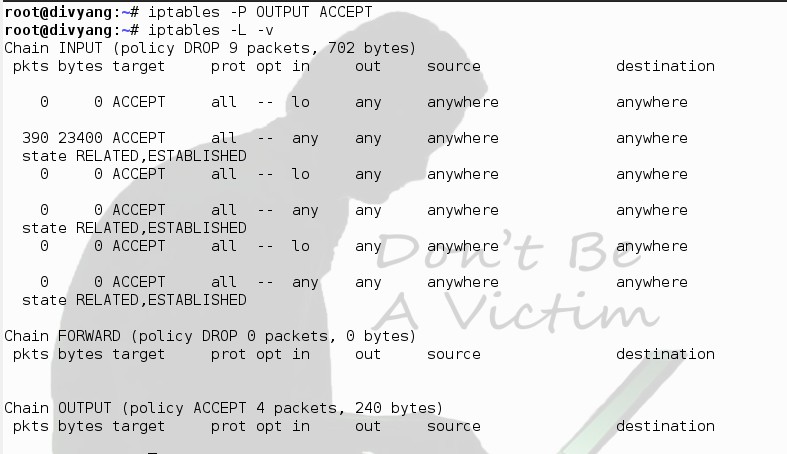
* **Types of Chains**

IPtables uses three different chains: input, forward, and output.

* + - Input: This chain is used to control the behavior for incoming connections. For example, if a user attempts to SSH into your PC/server, iptables will attempt to match the IP address and port to a rule in the input chain.
    - Forward: This chain is used for incoming connections that aren’t actually being delivered locally. Think of a router – data is always being sent to it but rarely actually destined for the router itself; the data is just forwarded to its target. Unless you’re doing some kind of routing, NATing, or something else on your system that requires forwarding, you won’t even use this chain.
    - Output: This chain is used for outgoing connections. For example, if you try to ping howtogeek.com, iptables will check its output chain to see what the rules are regarding ping and howtogeek.com before making a decision to allow or deny the connection attempt.
    - To see if iptables is running iptables -L

It is list the rules in chain or all chains.

* **Setup a SPI firewall that:**
  1. Allow all outgoing connection
  2. Block all unwanted incoming connection



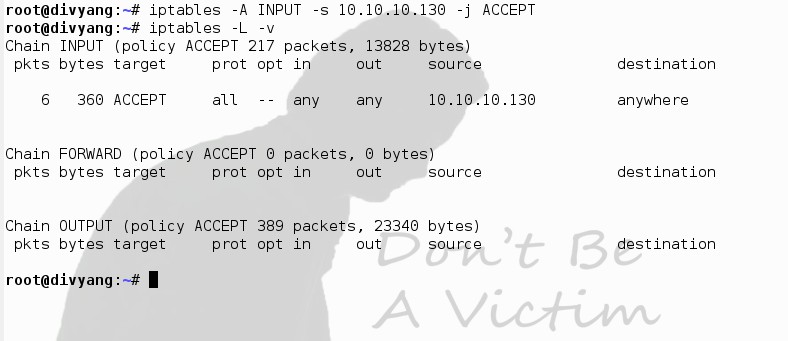
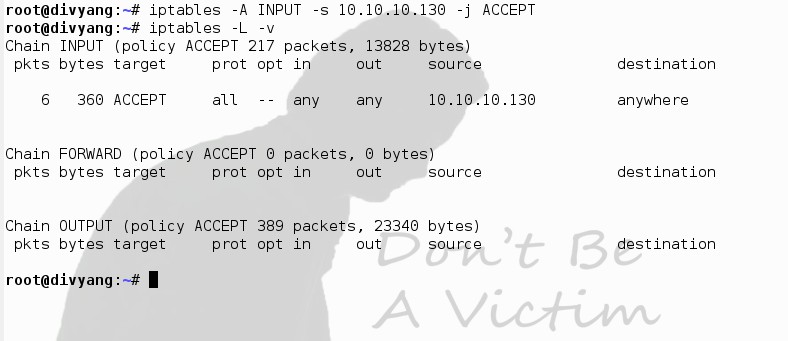
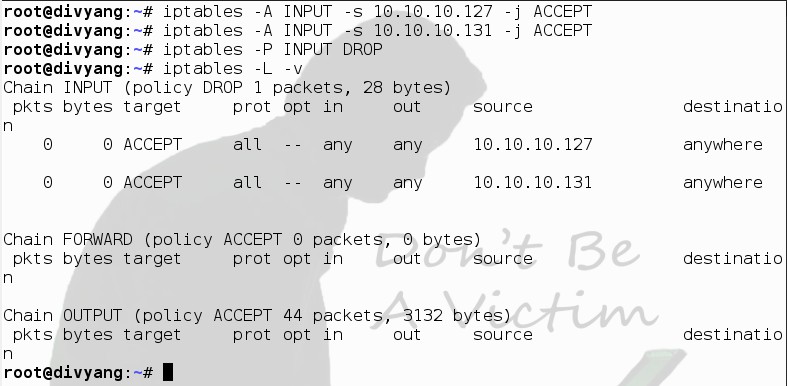
**iptables –F :** switch to flush all existing rules so we start with a clean state from which to add new rules

iptables -P INPUT DROP : -P switch sets the default policy on the specified chain which sets the default policy on the INPUT table to drop. If an incoming packet does not match one of the following rules it will be dropped. iptables –P FORWARD DROP

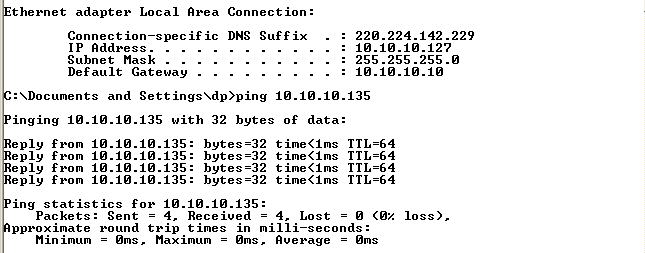
: set the default policy on the FORWARED chain to DROP since we’re not using our computer as a router, there should not be any packets passing through our computer

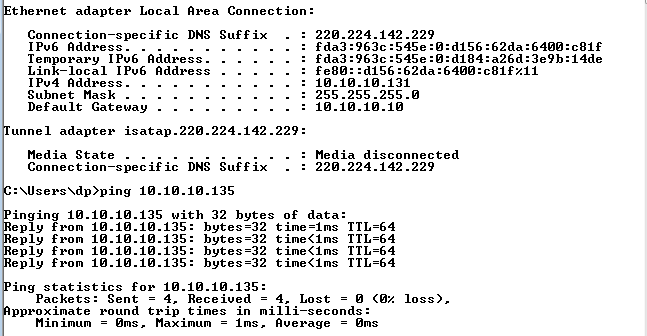
**iptables -P OUTPUT ACCEPT** : set the default policy on the OUTPUT chain to accept. This allow outgoing traffic

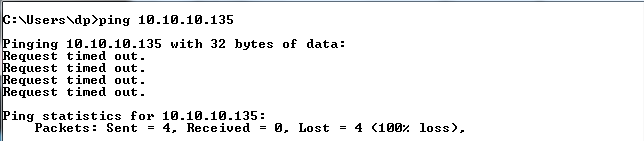
**iptables –A INPUT –i lo –j ACCEPT** : -A switch to append (or add) a rule to specific chain, the INPUT chain in this instance. –i switch (for interface) to specify Packed matching or destined for the lo(or localhost, 127.0.0.1) interface –j (jump) to the target action for packet maching the rule – in case ACCEPT.

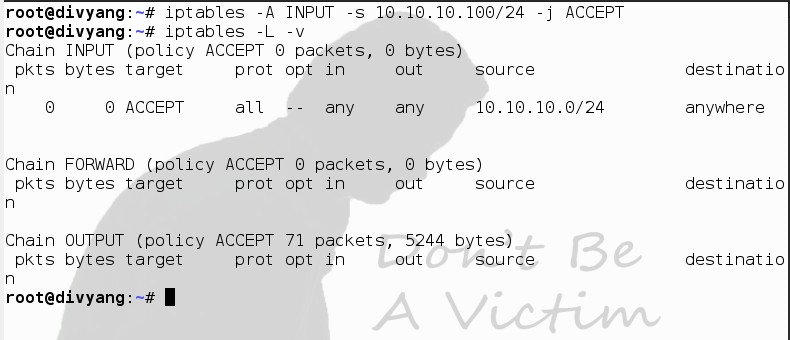
* Allow incoming only from one IP
* iptables –A INPUT –s 10.10.10.130 –j ACCEPT

After applying rules result from Different Machines are as bellow:



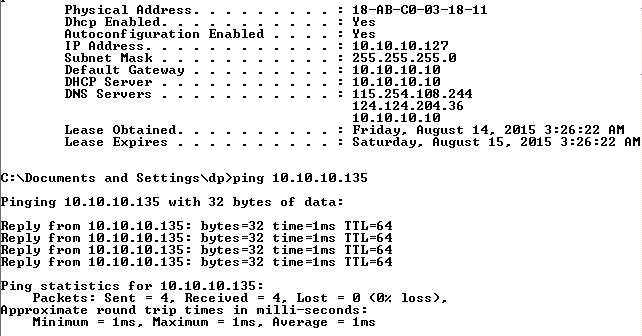
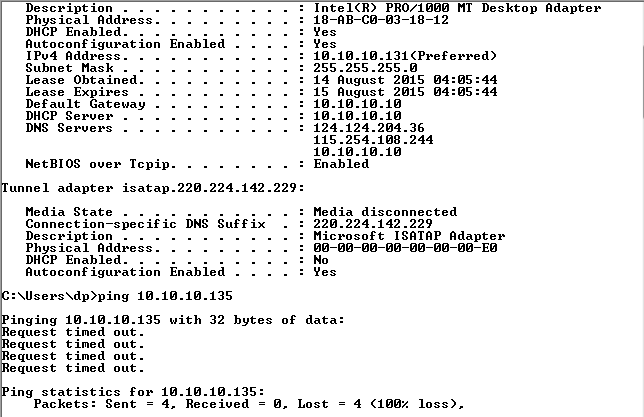


From Another system which are not listed in Rules

* iptables –A INPUT –s 10.10.10.100/24 –j ACCEPT(For whole Subnet)
* **Accept packet from trusted IP address with MAC**

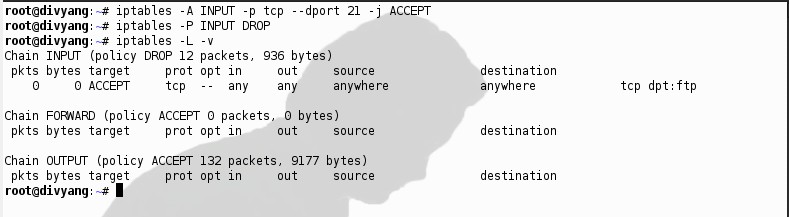
Rules for Accept packet from trusted IP address with MAC are described in image.

After applying rules we got the following output.

* 1. With Same ip and mac address
  2. Same IP but Different mac Address.
* **Port Address Filtering**

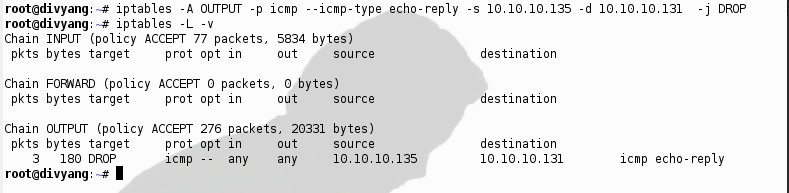
Single port

iptables –A INPUT –p tcp --dport 21 –j ACCEPT we can also set rules for Prot Range by applying

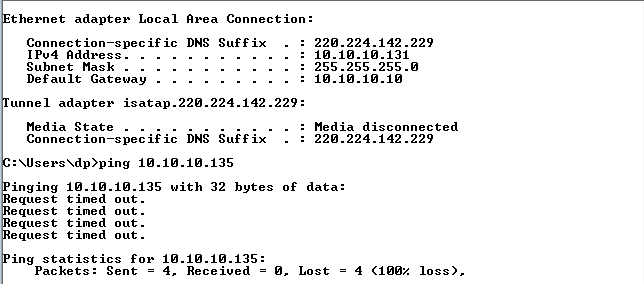
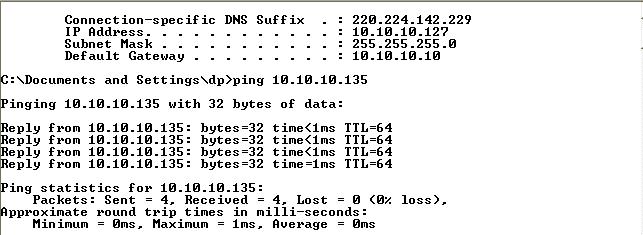
iptables –A INPUT –p tcp - -dport 6881:6890 –j ACCEPT

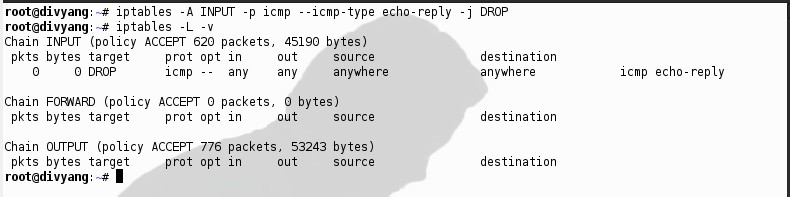
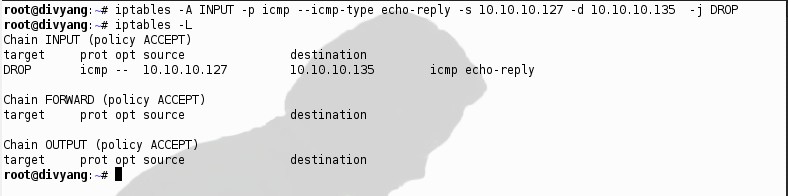
* **LAB Assignments:**

1. Block ICMP ping using OUTPUT and echo-reply

Solution:

After applying the rules result are as bellow.

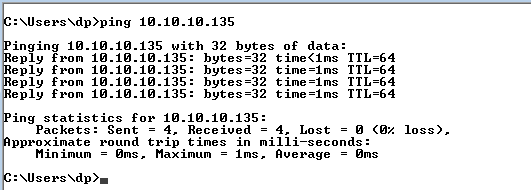
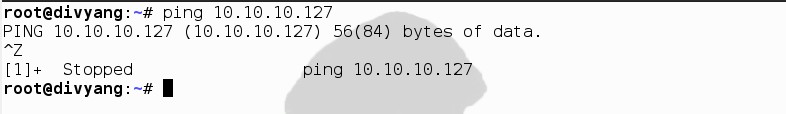
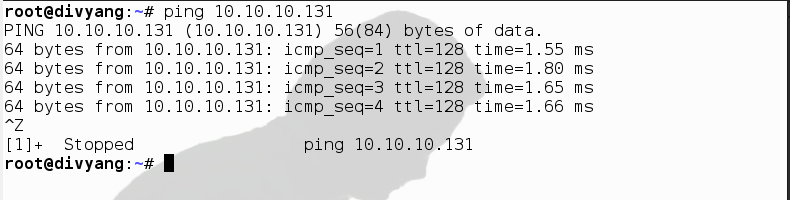
* 1. From given Destination IP
  2. From another IP

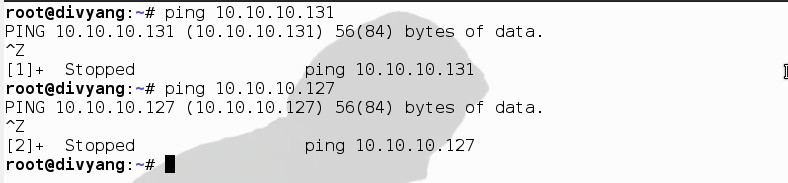
1. ******Block ICMP ping using INPUT and echo-request Solution 1**

**Solution 2**

After applying the rules result are as bellow.

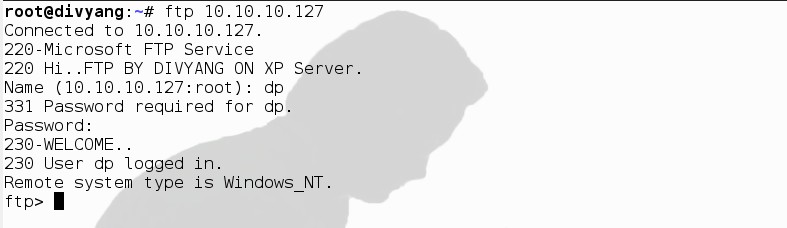
**For Solution 1**

* 1. from another Machine.
  2. From Linux Machine on which firewall rules applie

**For Solution 2**

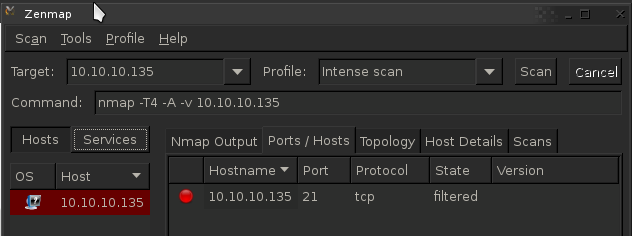
* 1. Block FTP using OUTPUT or INPUT (allow ftp server for your subnet only)

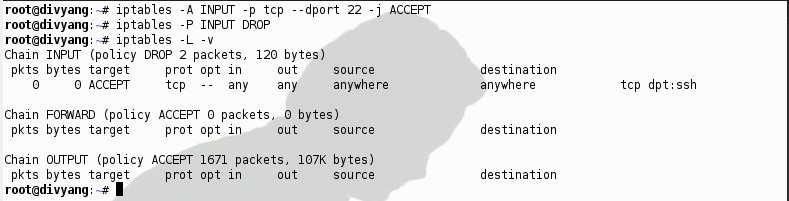
****Solution 1(For INPUT)

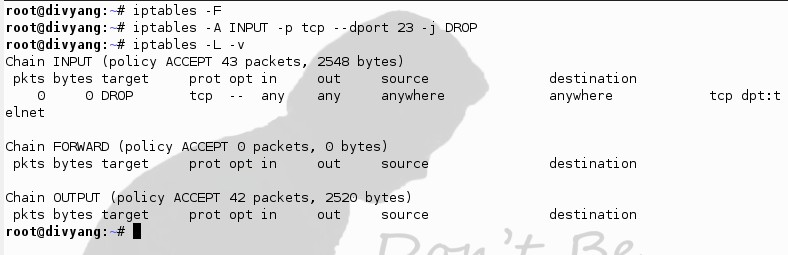
After applying the rules result are as bellow.

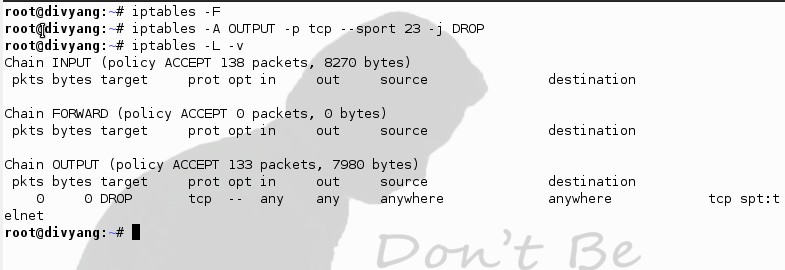
**Solution 2 (For OUTPUT)**

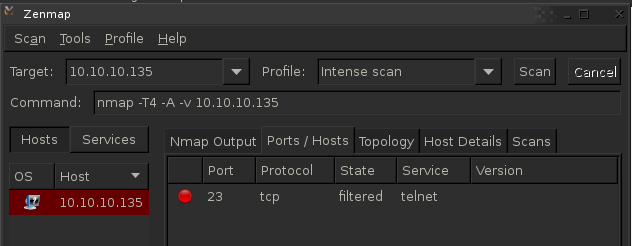
Result after applying the Rules.

**Checked the port using Zen map tools**

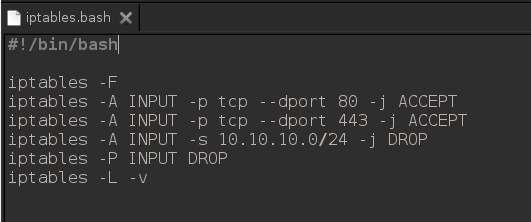
* 1. **ALLOW ssh using INPUT Solution**
  2. **Block TELNET using OUTPUT and INPUT.**

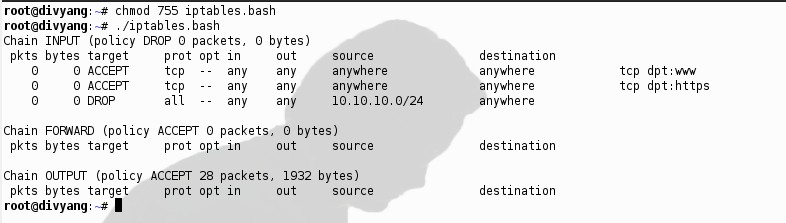
**Solution 1 (For INPUT)**

**Solution 1 (For OUTPUT)**

**After applying the rules result are as bellow.**

* 1. **Allow web server only to outside world.**

**Solution using bash file**

**After applying the rules result are as bellow.**