Security

What’s security?

It is a set of rules and configurations designed to protect the integrity, confidentiality and accessibility of computer networks and data using both software and hardware technologies. Every organization, regardless of size, industry or infrastructure, requires a degree of network security solutions in place to protect it from the ever-growing landscape of cyber threats in the wild today.

Firewalls, as their name suggests, act as a barrier between the untrusted external networks and your trusted internal network. Administrators typically configure a set of defined rules that blocks or permits traffic onto the network.

Types Of Attacks:

- Application layer attack
- Passive attack
- Backdoors
- Denial of services(DoS)
- Distributed denial of services(DDoS)

Application Layer Attack:

Application layer attacks or layer 7 (L7) DDoS attacks refer to a type of malicious behavior designed to target the “top” layer in the OSI model where common internet requests such as HTTP GET and HTTP POST occur. These layer 7 attacks, in contrast to network layer attacks such as DNS Amplification, are particularly effective due to their consumption of server resources in addition to network resources.

Passive attack:

Attempt to learn or make use of information from the system, it’s an unauthorized reading of messages from sender to receiver, it doesn’t affect system resources.

Denial of services(DoS):

Putting too much load (overloading) the server, so it can’t take any more clients, so whoever asks for the service, will be denied from getting it.

This is done by creating a malicious program/bot program is the system that allows a lot of TCP connections to be established is the server.

Attacker going to send a lot of TCP connections from his computer, but it will be seen as it from different destinations.

Backdoors:

A backdoor is a means to access a computer system or encrypted data that bypasses the system's customary security mechanisms.

A developer may create a backdoor so that an application or operating system can be accessed for troubleshooting or other purposes. However, attackers often use backdoors that they detect or install themselves as part of an exploit.

Distributed Denial Of Services(DDoS):

A distributed denial-of-service (DDoS) attack is a malicious attempt to disrupt the normal traffic of a targeted server, service or network by overwhelming the target or its surrounding infrastructure with a flood of Internet traffic.

Appliances:

- IDS - Intrusion Detection System
- IPS - Intrusion Prevention System
- Stateful IOS Firewall Inspection Engine
- Firewall Voice Traversal
- Authentication Proxy
- ICMP Inspection

Intrusion Detection System (IDS)

An Intrusion Detection System (IDS) is a system that monitors network traffic for suspicious activity and issues alerts when such activity is discovered. It is a software application that scans a network or a system for harmful activity or policy breaching. Any malicious venture or violation is normally reported either to an administrator or collected centrally using a security information and event management (SIEM) system.

Intrusion Prevention System (IPS)

An Intrusion Prevention System (IPS) is a form of network security that works to detect and prevent identified threats. Intrusion prevention systems continuously monitor your network, looking for possible malicious incidents and capturing information about them. The IPS reports these events to system administrators and takes preventative action, such as closing access points and configuring firewalls to prevent future attacks.

Stateful IOS Firewall Inspection Engine

Stateful inspection firewalls keep track of connection status. Ports can be dynamically opened and closed if necessary for completing a transaction. For example, when you make a connection to a server using HTTP, the server will initiate a new connection back to your system on a random port. A stateful inspection firewall will automatically open a port for this return connection.

Firewall Voice Traversal

Any firewall, including Cisco ASA or an application layer gateway (ALG), is expected to provide certain mechanisms so that voice and video traffic can traverse through the firewall/ALG to reach the destination. Firewall traversal is provided in multiple ways, including NAT traversal, IPsec tunnels, IP ACLs, or port-based ACLs.

Authentication Proxy

Proxy Authentication enables you to configure the authentication method used by the proxy. This determines how client machines are validated when accessing the Internet. Proxy Authentication must be enabled to be able to create new policies for users or groups.

ICMP Inspection

ICMP’s primary functions are error reporting at layer-3, and troubleshooting. In fact, two of the most useful networking tools, ping and traceroute, rely upon it. However, it is tightly bound to the IP stack at layer-3, so it’s no surprise that the ASA firewall treats it differently to other protocols like TCP or UDP.

Access Lists

Access Lists

It is a **List of Conditions** That Categorize Packets .It is used in Filtering unwanted packets and also has many purposes:

- Used to permit or deny packets moving through The Router.
- Permit or deny Telnet (VTY) access to or from a router.

Creating Access List like **Programming a series** of (If–Then) statements a given condition is met ,then a given action is taken . And applying an Access List causes The Router to analyze every packet crossing that interface in the specific direction and take the appropriate action.

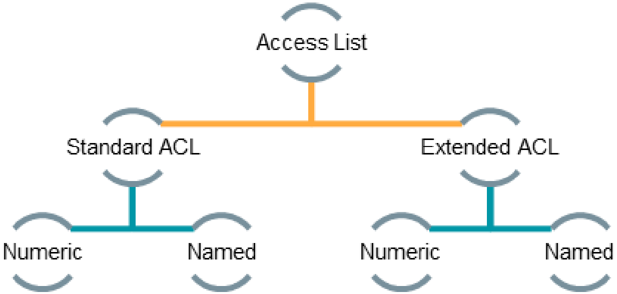
Important Rules

Packets follows these Rules when it's being compared with an access list:

- packets compared with each line of access list in **a sequential order** it start with first line on it and then go to line 2 and so on .
- it's **compared with lines of access list only** until a match is made once the packet matches the condition on a line of access list the packet is acted upon and no further comparison take place.
- there is an **implicit "deny"** is at the end of each access list this means if a packet doesn't match the condition on any lines in the access list, the packet will be discarded.

Each of these rules has some powerful implications when **filtering IP packets** with access list so creating effective access list takes some practice.

Types of Access Lists:



- **Standard Access Lists:** These use **only the source IP address** in IP packet as the condition test all decision are made based on the source IP address this mean that standard access list basically Filter by source IP addresses only .they don’t distinguished between any of the many types of IP traffic such as Web ,TELNET ,UDP and so on

- **Extended Access Lists:** it can evaluate many of other fields in the layer 3 and layer 4 headers of an IP Packet .they can evaluate **source** and **destination IP addresses**, the **protocol** field in the network layer header and the **port number** at the transport layer header .this gives the extended access list the ability to make much more granular decision when controlling traffic.

Named Access List : Functionally the same as standard and extended access lists , not actually a new type.

Applications of Access Lists:

Once you create access list it not going to do anything until you apply it they on the router but they are in active until you tell the router what to do with them. To use access list as a packet filter, we **need to apply it to an interface** on the router where you want the traffic filtered.

We should **specify which direction of traffic** we want access list applied to and by specifying it we can use different access list for inbound and outbound traffic on a single interface.

- **Inbound Access Lists:** When an access list is applied to inbound packets on an interface, Packets are processed before being routed to the outbound interface. Any packets that are denied won’t be routed because they are discarded before the routing process.
- **Outbound access lists:** when an access list is applied to outbound packets on an interface, those packets are routed to the outbound interface and then processed through the access list before being queued.

There are some *general access-list guidelines* that should be followed when you're creating and implementing access lists on a router.

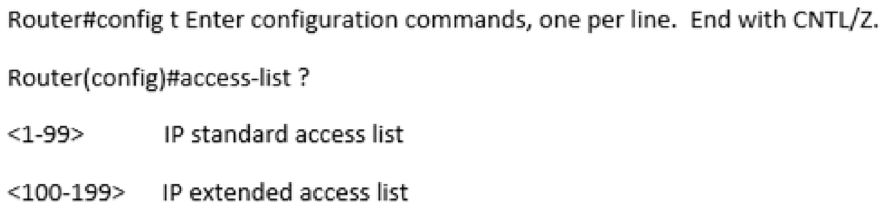
ACL Guidelines:

1. **One access list per direction.** Inbound and Outbound access list.
2. **More specific tests at the top of the ACL**. cause access list is a sequential list.
3. **New lists are placed at the bottom of the ACL.**
4. **Individual lines cannot be removed.** if we want to edit the list so we have to delete the list and create another one.
5. **Create ACLs & then apply them to an interface.** creating the ACL without applying it onto interface will not be useful.
6. **ACLs must have at least one permit command.**
7. **Put Standard ACLs close to the destination**. because it filters traffic base on the source IP address. As ACL work in sequence, when standard ACL is placed closest to the source it may stop the host to access other resources in the network that you do want to stop.
8. **Put Extended ACLs close the source**. because it filters based on much more specific criteria such as source, destination IP address, protocol and port number. Therefore by placing it closest to the source will only affect the specific host you are pointing and also will avoid the unwanted traffic to consume the bandwidth in your network.

Standard Access Lists

Standard Access Lists

- Standard IP access lists are used to permit/deny traffic only based on **source IP address** of the IP datagram packets.
- **ACL configurations:**
  1. Create ACL
  2. Apply ACL to a certain interface



```
Router#config t Enter configuration commands, one per line.  End with CNTL/Z.

Router(config)#access-list ?

<1-99>        IP standard access list

<100-199>     IP extended access list
```

- Standard Access Control Lists (ACLs) can be **created** by using the **"access-lists" IOS command**. The syntax of "access-list".
- **Maximum** number of ACL can be applied in each interface per protocol = **2**
- **Access list number**: For Standard Access Control List, Access list number must be between **1–99**
- ACL can do one of this actions:
  1. Permit
  2. Deny
- **permit | deny**: Whether to permit or deny traffic.
- **IP address**: An IP address to filter the traffic.

Standard Access Lists

Configuration:

1. **Creation of Standard ACL:**

Router(config) #access-list 10?

deny   Specify packets to reject

permit Specify packets to forward

- Permit or Deny ?

Router(config) #access-list 10 deny?

A.B.C.D   Address to match

any   Any source host

host   A single host address

- General: **Router(config) #access-list {1-99} {Permit / Deny} Source IP [Source Wildcard Mask]**
- Using the Host command: **Router(config) #access-list {1-99} {Permit / Deny} host Host-IP**

1. **Applying:**

Router(config) #interface {interface name}

Router(config-if) #ip access-group {199} {In / Out}

Understanding Wildcard Masks

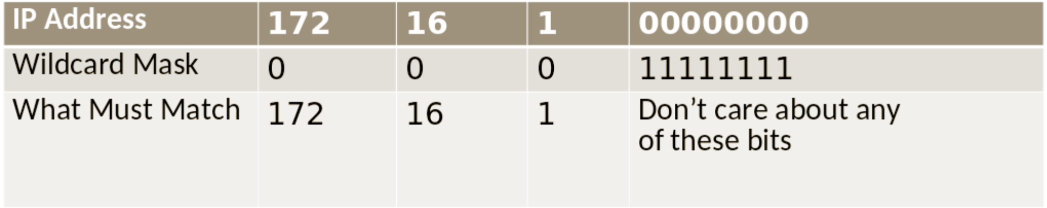
The point of a wildcard mask is to specify which bits in an IP address should be ignored when comparing that address with another IP address and which bits should match exactly, **‘Don't care bits'** are represented by **binary 1's** whilst the **'Do care bits'** are represented by **binary 0's**.

Example 1 : 172.16.1.2   0.0.0.0

The **0s** in the mask indicate that all bit positions **must match exactly**. Therefore, the ACL will only be applied to host **172.16.1.2**. Another way of specifying a particular host is by using the **host command**. So these two commands specify the same thing and will be applied to a particular host.

access-list 10 deny 172.16.1.2   0.0.0.0

access-list 10 deny host 172.16.1.2



IP Address	172	16	1	00000000
Wildcard Mask	0	0	0	11111111
What Must Match	172	16	1	Don't care about any of these bits

Example 2 : 172.16.1.0 0.0.0.255

The **0's** in the in the **first three octets** of the wildcard mask indicate that all bit positions **must match exactly**, but the **last octet** can be **any valid number**. In this case, the ACL will apply to all hosts in the 172.16.1.0 subnet.

Example (CONTD.)

Calculating Wildcard Masks from Subnet Masks:

SUBTRACT EACH OCTET IN THE SUBNET MASK FROM 255

let's find the wildcard mask for the subnet:

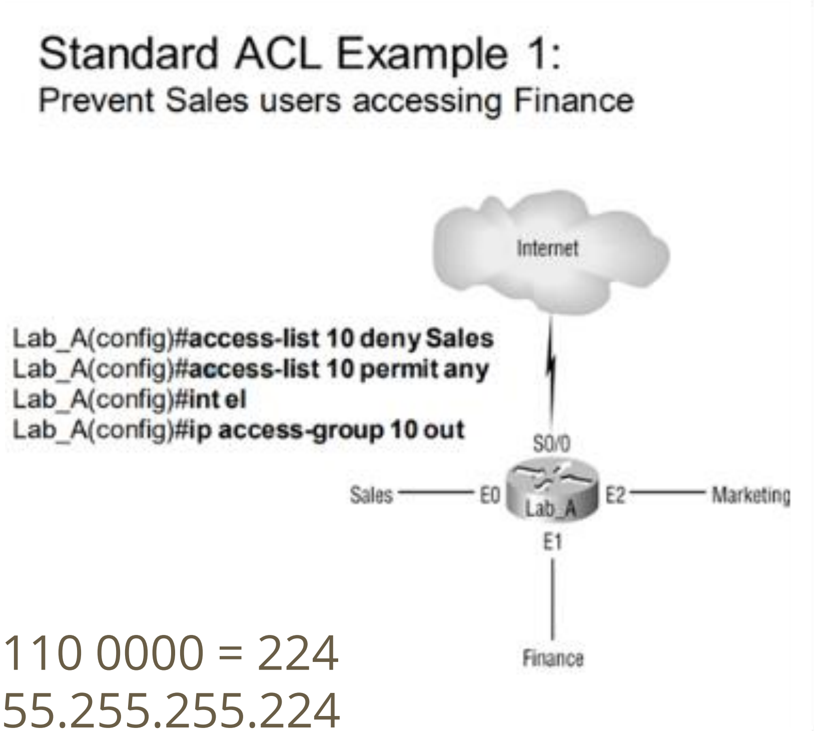
Ex1: 255.255.255.240.

	255	255	255	255
Subnet Mask	255	255	255	240
WildCard Mask	0	0	0	15

Ex2: 255 255 255 224

	255	255	255	255
Subnet Mask	255	255	255	224
WildCard Mask	0	0	0	31

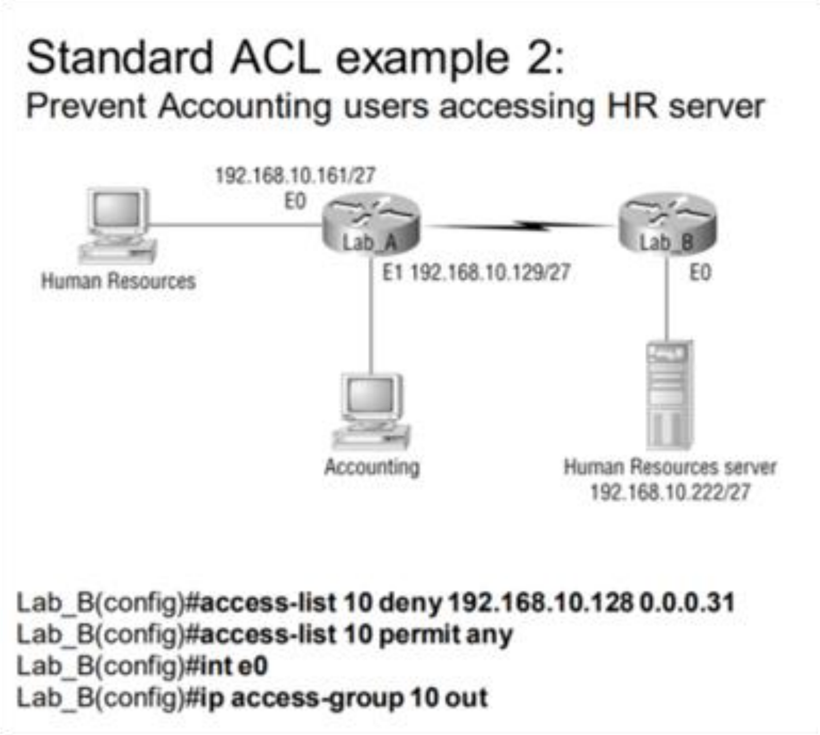
Standard ACL Examples:



Standard ACL Example 1:

Prevent Sales users accessing Finance

```
Lab_A(config)#access-list 10 deny Sales
Lab_A(config)#access-list 10 permit any
Lab_A(config)#int el
Lab_A(config)#ip access-group 10 out
```



Standard ACL example 2:

Prevent Accounting users accessing HR server

```
Lab_B(config)#access-list 10 deny 192.168.10.128 0.0.0.31
Lab_B(config)#access-list 10 permit any
Lab_B(config)#int e0
Lab_B(config)#ip access-group 10 out
```

1110 0000 = 224

255.255.255.224

Controlling VTY (Telnet) Access:

- Active interfaces on a **network router** can be **accessed by** users on the network if not properly secured. Users or Hackers might try **telnetting** the network router through the VTY access.

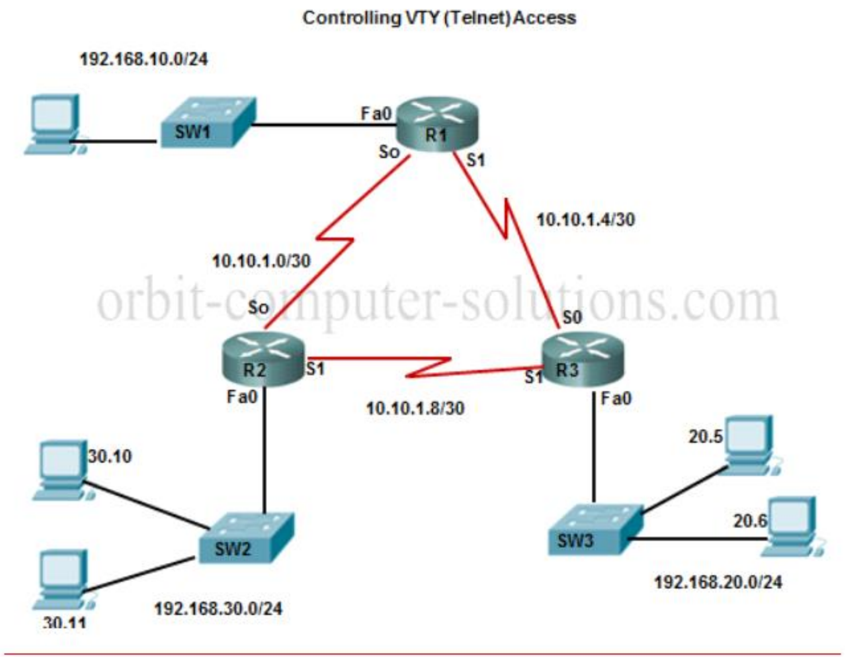
- To stop this from happening, the best practice is for you to use a **standard IP access list** to **limit telnet access** to every network or IP address on the router. Applying a standard IP access list to the VTY lines **eliminates** the option of using telnet protocols and destination address since it does not matter which interface address a user or hacker is using as a target for the telnetting session.

- Using a standard IP access list to restrict VTY access enables you to define which IP addresses are allowed telnet access to the router EXEC process. You can control which workstation or network access your router with an ACL and an access-class statement to your VTY lines

We are going to create a standard IP access list that permits only a host 192.168.30.2 (or hosts) to be able to telnet to the router R1, the command and configuration look like this:

- **R1#config t**
- **R1(config)#access-list 10 permit 192.168.30.10**
- **R1(config)#lines vty 0 4** **5 simultaneous virtual connection**
- **R1(config-line)#access-class 10 in**

The configuration simply means that **only** the **IP address 192.168.30.10** or host is allowed to Telnet or **access to the R1** router.


Controlling VTY (Telnet) Access

Extended IP Access Lists

Extended IP Access Lists

Extended IP access lists works on :

1. IP Source Address.
2. IP Destination Address.
3. Protocol.
4. Port number.

Extended IP Access Lists Configuration

1. Creation of Extended ALC:

(config)#Acess-list {100:199}{Permit/Deny}{Protocol} Source IP ,Source Wild card{Operator , Source Port number}, Destination IP, Destination Wild card {Operator , Destination Port number}.

Operator can be:

1. eq = equal
2. lt = less than or equal
3. gt = greater than or equal

1. Applying:

(Config)#ip access-group{100:199}{In/Out}

Extended IP ACL Steps

1. **Select the access list**: RouterA(config)#access-list 110
2. **Decide on deny or permit**: RouterA(config)#access-list 110 deny
3. **Choose the protocol type**: RouterA(config)#access-list 110 deny tcp
4. **Choose source IP address of the host or network**: RouterA(config)#access-list 110 deny tcp any
5. **Choose destination IP address**: RouterA(config)#access-list 110 deny tcp any host 172.16.30.2
6. **Choose the type of service, port, & logging**: RouterA(config)#access-list 110 deny tcp any host 172.16.30.2 eq 23

- **Example:**

- RouterA(config)#access-list 110 permit ip any any
- RouterA(config)#ip access-group 110 in **OR** RouterA(config)#ip access-group 110 out

Named Access List

Standard

```
Router > enable

Router # config t

Router (config) # access-list 1 deny host 172.16.10.5

Router (config) # access-list 1 permit any

Router (config) # interface fast Etherent 0/0

Router (config-if) # IP access-group 1 out

Router (config-if) # exit
```

Standard Name ACL

```
Router > enable

Router # config t

Router (config) # ip access-list standard internet

Router (config-std-nacl) # deny host 172.16.10.5

Router (config-std-nacl) # permit any

Router (config) # exit

Router (config) # interface fast Ethernet 0/0

Router (config-if) # IP access-group internet out

Router (config-if) # exit
```

Extended

```
Router > enable

Router # config t

Router (config) # access-list 100 deny host 172.16.10.5 host 192.168.1.1 eq http

Router (config) # access-list 100 permit ip any any

Router (config) # interface Fast Etherent 0/0

Router (config-if) # IP access-group 100 in

Router (config-if) # exit
```

Extended Name ACL

```
Router > enable

Router # config t

Router (config) # IP access-list extended http

Router (config-std-nacl) # deny tcp host 172.16.10.5 host 192.168.1.1 eq http

Router (config-std-nacl) # permit IP any any

Router (config) # exit

Router (config) # interface Fast Ethernet 0/0

Router (config-if) # IP access-group http in

Router (config-if) # exit
```

Why use the Name ACL?

Why It is preferable to use it by numbered method?

- Allows the use of descriptive names to ease network management.
- Deletes a command line from access list by placing the words ‘no’ before the command unlike numbered.

Monitoring IP Access Lists

- Display all access lists & their parameters

  show access-list

- Show only the parameters for the access list 110

  show access-list 110

- Shows only the IP access lists configured

  show ip access-list

- Shows which interfaces have access lists set

  show ip interface

- Shows the access lists & which interfaces have access lists set

  show running-config

Standard ACL on packet tracer

1. create access list (standard or extended)
2. apply access list to interface on the router(inbound or outbound)

Standard ACL:

- Applied closest to destination
- Denies or permits source IP address

```
R(config)#access list 1 deny 192.168.2.101 0.0.0.0

R(config)#access list 1 permit any

R(config)#interface gigabit Ethernet 0/1

R(config-if)#ip access-group 1 out
```

Extended ACL : applied closest to the source.

- Denies or permits source Ip address
- Denies or permits destination ip address
- Denies or permits port(service)

```
R(config)#access list 100 deny ip 192.168.2.100  0.0.0.0  192.168.1.0  0.0.0.255

R(config)#access list 100 permit ip any any

R(config)#interface gigabitethernet 0/1

R(config-if)#ip access-group 100 in
```

EXTENDED ACCESS LIST: SERVICE(PORT)

```
R(config)#access_list 100 permit tcp 192.168.2.0 0.0.0.255 host 192.168.1.254 eq 80

R(config)#int interface gigabitethernet 0/1

R(config)#ip access-group 100 in
```

Named ACL: allows standard and extended ACLs to be given names instead of numbers.

```
R(config)#int f/1

R(config)#ip access list extended (name)

R(config_ext_nacl)#permit tcp 192.168.2.0 0.0.0.255 host 192.168.1.254 eq 80

R(config_ext_nacl)#end

R(config)interface gigabitethernet 0/1

R(config)ip access-group (name) in
```