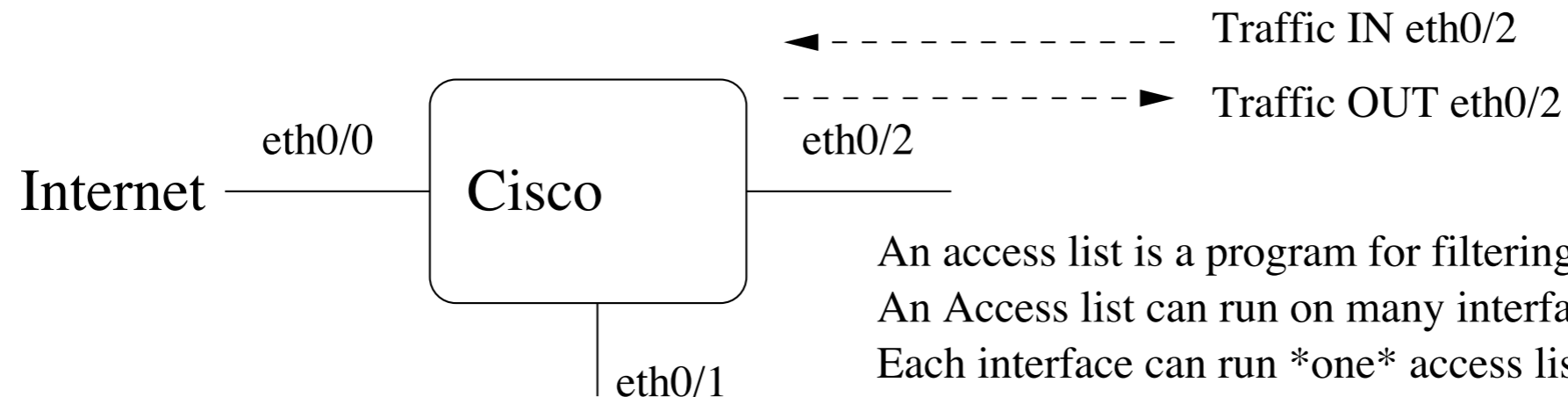


Network Security

ITL

Cisco Terminology



An access list is a program for filtering
An Access list can run on many interfaces
Each interface can run *one* access list for IN
and *one* access list for OUT
IN and OUT are relative to the interface
Wiring defines whether IN and OUT are to
the Internet or host

The `access-group` command

- Interface Configuration command, e.g.
 - `Interface eth 0/0`
 - `ip access-group <number> in|out`
- `<number>` is the access [control] list number
 - for historical reasons, access lists 1-99 are reserved for an older, simplified syntax; we will use 100-199.
- The code “in” or “out” is relative to the interface
 - Two access-group commands can be installed, one for inbound packets, one for outbound ones.

The `access-list` command

- Entered in configure mode
 - Adds one rule to the end of the numbered access list
- Filtering based on IP information
 - `access-list access-list-number [dynamic dynamic-name [timeout minutes]] {deny | permit} protocol {source source-wildcard | any} {destination destination-wildcard | any} [precedence precedence] [tos tos] [log]`
 - **we will only use a sub-set of the command**
 - `access-list access-list-number {deny | permit} protocol {source source-wildcard | any} {destination destination-wildcard | any}`
 - `access-list 197 deny ip any 10.0.0.0 0.255.255.255`

The “Wildcard” parameter

- The access-list command likely predates the concept of network masks
 - Even before network masks were widely used, it was useful in access lists to specify a range of networks
- Cisco picked the opposite representation from the one that became the network mask, i.e. the “wildcard”
 - In a network mask, the part you do not look at is set to zeroes (the host portion)
 - In wildcards, that part you do not look at is filled with ones

Masks and Wildcards

- Example: a /24 prefix
 - Mask: 255.255.255.0
 - Wildcard: 0.0.0.255
- Example: a /30 prefix
 - Mask: 255.255.255.252
 - Wildcard: 0.0.0.3

The ordering of rules

- Rules in an access list are examined in the order they were entered in
- Example: Permit only packets from 132.235.201.2 into the 10.0.0.0/8 network
 - How about:
 - `access-list 197 deny ip any 10.0.0.0 0.255.255.255`
 - `access-list 197`
`permit ip 132.235.201.2 0.0.0.0 10.0.0.0 0.255.255.255`
 - What we really need is
 - `access-list 197`
`permit ip 132.235.201.2 0.0.0.0 10.0.0.0 0.255.255.255`
 - `access-list 197 deny ip any 10.0.0.0 0.255.255.255`

More about rules

- What if we just enter

- `access-list 197`
`permit ip 132.235.201.2 0.0.0.0 10.0.0.0 0.255.255.255`

- This works as intended because every access lists has an implicit “deny ip any any” at the end
 - However, most security managers will put that rule in explicitly
 - Side effect: if you install an access-group command before writing the access-list, the interface will block all packets (in the direction set by access-group)

Reordering rules

- The short answer, you cannot!
 - Cisco's IOS has no commands that move rules within a list
 - Our approach in the lab:
 - Enter and edit the rules in a local text editor on your workstation
 - In your terminal window, enter
 - no access-list <access list number>
 - then paste the entire set of rules from the text editor to the terminal window
 - watch for error messages

Some useful commands

- `show running-config`
 - Displays the entire router configuration
- `show access-lists`
 - Display all access lists
- `show access-list 101`
 - Display one access list (note there is no “s”)
- The syntax “`show ip access-list(s)`” will also work

The “protocol” field

- We can restrict a rule to a particular protocol
 - `access-list 112 permit icmp any any echo`
 - `access-list 112 permit icmp any any echo-reply`
 - **permits the use of ping**
- You can put comments into lists
 - `access-list 101 remark this is my access list`
 - **Only visible in the show running-config command**
- The protocol can be TCP or UDP
 - `access-list 103 deny tcp any any`

TCP and UDP filters

- For TCP and UDP, we want to work with port numbers, so the syntax is expanded
 - `access-list access-list-number [dynamic dynamic-name [timeout minutes]] {deny | permit} {tcp | udp} {source source-wildcard | any} [operator port [port]] {destination destination-wildcard | any} [operator port [port]] [established] [precedence precedence] [tos tos] [log]`
 - **We will use mostly**
 - `access-list access-list-number {deny | permit} tcp {source source-wildcard | any} [operator port [port]] {destination destination-wildcard | any} [operator port [port]] [established]`
 - **Operators: lt gt eq neq range**
 - **“established” (only for TCP) - matches a packet only if it is not trying to create a new connection**

More examples

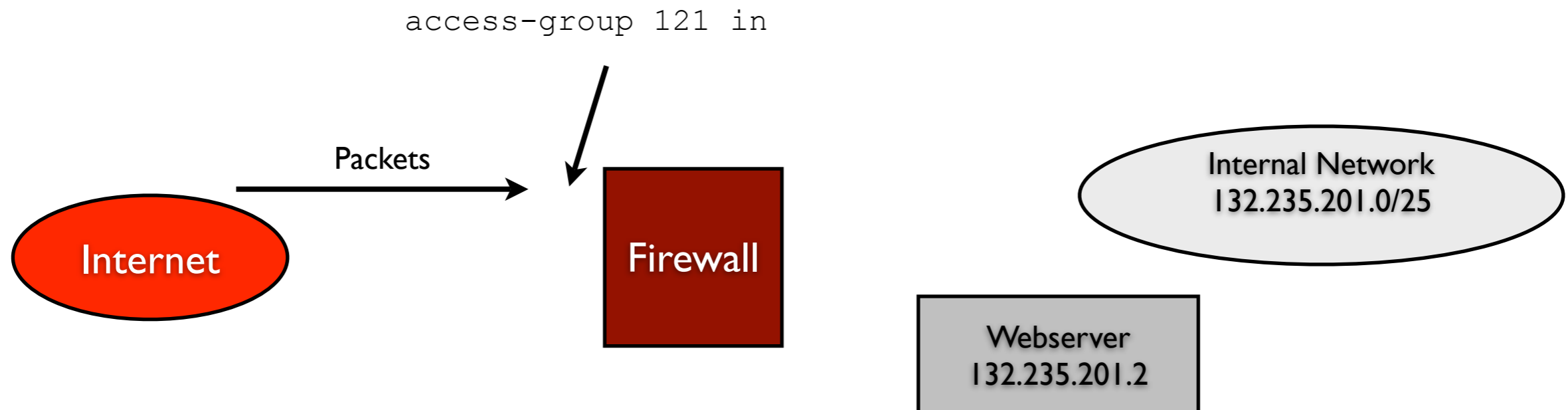
- `access-list 101 permit udp any any`
- `access-list 101 permit tcp any any established`
- `access-list 101 remark 22=ssh`
- `access-list 101 permit tcp any any eq 22`

Back to the firewall policy

- Permit TCP from anywhere, any port to port 80 on 132.235.201.2/32

```
access-list 121 permit tcp any 132.235.201.2 0.0.0.0 eq 80
```

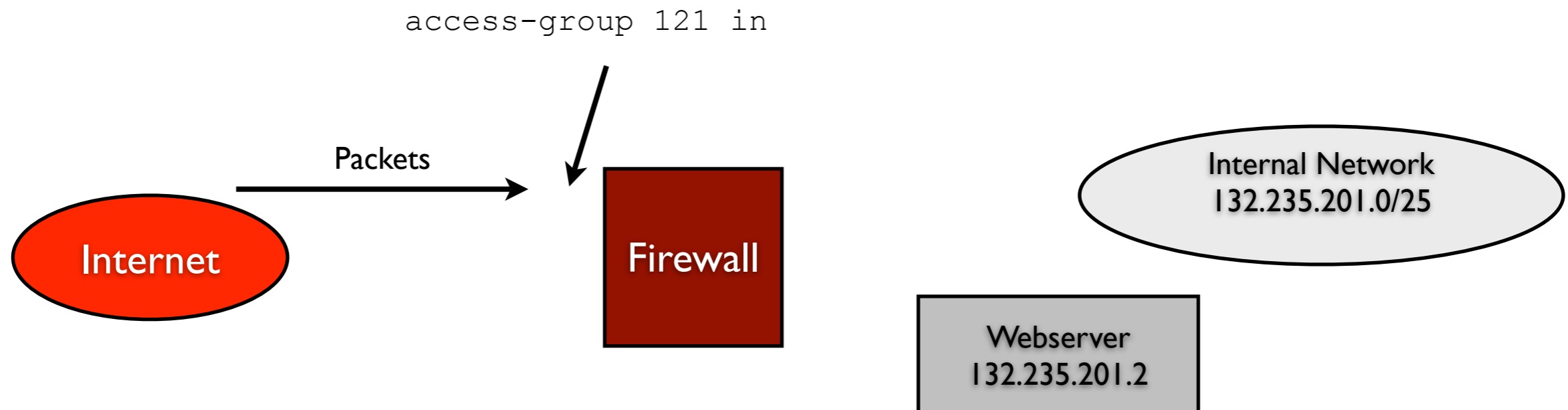
Note:
you can replace
132.235.201.2 0.0.0.0 with
host 132.235.201.2



Add a policy

- Permit TCP from anywhere, any port to port 80 on 132.235.201.2/32
- Permit any address in 132.235.201.0/25 access to ssh servers on the internet

```
access-list 121 permit tcp any host 132.235.201.2 eq 80
access-list 121 permit tcp any eq 22 132.235.201.0 0.0.0.127
```



Some examples

- Permit TCP from anywhere, any port to port 80 on 132.235.201.2/32
- Permit TCP from anywhere, any port to 132.235.201.0/25, any port, only as part of an established connection
- Permit ICMP echo from anywhere to 132.235.201.2/32
- Permit ICMP echo-reply from anywhere to 132.235.201.0/25
- Deny IP from anywhere to anywhere

Next lecture -- how to do this on a cisco router

