CALIFORNIA STATE UNIVERSITY
LONG BEACH
College of Engineering

CECS 303: Networks and Network Security

Firewalls and NAT Overview

Chris Samayoa

Week 5 – 2nd Lecture
2/17/2022

Course Information

CALIFORNIA STATE UNIVERSITY
LONG BEACH
College of Engineering

- CECS 303
  - Networks and Network Security – 3.0 units
- Class meeting schedule
  - TuTH 5:00PM to 7:15PM
  - Lecture Room: VEC 402
  - Lab Room: ECS 413
- Class communication
  - chris.samayoa@csulb.edu
  - Cell: 562-706-2196
- Office hours
  - Thursdays 4pm-5pm
  - Other times by appointment only

Objectives

- Overview of router access lists
- Introduction to firewalls
- Introduction to iptables
- Overview of Network Address Translation (NAT)

Security in Network Design

- Breaches may occur due to poor LAN or WAN design
  - Address though intelligent network design
- Preventing external LAN security breaches
  - Restrict access at every point where LAN connects to rest of the world

Router Access Lists

- Control traffic through routers
- Router's main functions
  - Examine packets
  - Determine destination
    - Based on Network layer addressing information
- ACL (access control list)
  - aka. access list
  - Routers can decline to forward certain packets
- Stateless
  - Access lists look at packets independent of what traffic has come before

Router Access Lists (cont'd)

- ACL variables used to permit or deny traffic
  - Network layer protocol (IP, ICMP)
  - Transport layer protocol (TCP, UDP)
  - Source IP address
  - Source netmask
  - Destination IP address
  - Destination netmask
  - TCP or UDP port number

Router Access Lists (cont'd)

- Router receives packet, examines packet
  - Refers to ACL for permit / deny criteria
  - Drops packet if deny characteristics match
  - Forwards packet if permit characteristics match
- Access list statement examples
  - Deny all traffic from source address with netmask 255.255.255.255
  - Deny all traffic destined for TCP port 23
- Separate ACL's for:
  - Interfaces; inbound and outbound traffic

ACL Example

CALIFORNIA STATE UNIVERSITY
LONG BEACH
College of Engineering

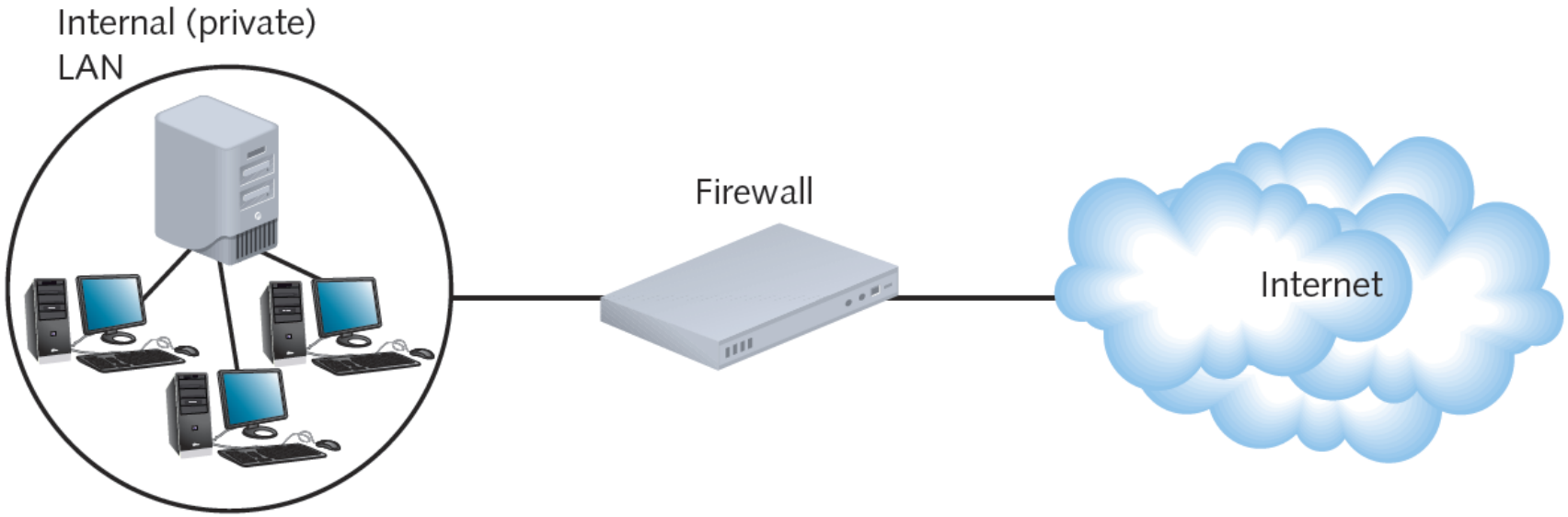
```
R1(config-ext-nacl)#do sh access-list OutBoundAccess
Extended IP access list OutBoundAccess
    10 permit ip 192.168.1.0 0.0.0.255 any
    11 deny tcp 192.168.2.0 0.0.0.127 any eq smtp
    12 deny tcp 192.168.2.0 0.0.0.127 any eq sunrpc
    13 deny tcp 192.168.2.0 0.0.0.127 any eq pop2
    14 deny tcp 192.168.2.0 0.0.0.127 any eq nntp
    15 deny tcp 192.168.2.0 0.0.0.127 any eq ftp
    16 deny tcp 192.168.2.0 0.0.0.127 any eq ftp-data
    17 deny tcp 192.168.2.0 0.0.0.127 any eq telnet
    18 deny tcp 192.168.2.0 0.0.0.127 any eq cmd
    19 deny tcp 192.168.2.0 0.0.0.127 any eq irc
    20 permit ip 192.168.2.0 0.0.0.255 any
    30 permit ip 192.168.3.0 0.0.0.255 any
    40 permit ip 192.168.4.0 0.0.0.255 any
    50 permit ip 192.168.5.0 0.0.0.255 any
R1(config-ext-nacl)#
```

Source: CCIE-or-Null.net

Firewalls

- Specialized device or computer installed with specialized software
  - Selectively filters and blocks traffic between networks
  - Involves hardware and software combination
  - Stateful
    - Decisions can be made based on previous traffic
    - e.g. Allowing return traffic from a web server
- Firewall locations
  - Between two interconnected private networks
  - Between private network and public network (network-based firewall)
  - Between two hosts (host based firewall)

Firewall Example



Source: Course Technology / Cengage Learning

Firewall Types

- Packet Filters
  - Stateful packet filters are the norm
- Host-based software firewalls
  - Manage connection policies for individual nodes
  - Can be centrally managed
- Application level gateways or proxies
  - Common for corporate / business intranets
  - Includes Next Generation Firewalls (NGFW)

Firewalls (cont'd)

- Optional firewall functions
  - Encryption
  - User authentication
  - Central management
  - Easy rule establishment
  - Filtering based on data contained in packets (DPI)
  - Logging, auditing capabilities
  - Protect internal LAN's address identity
  - Monitor data stream from end to end (stateful firewall)
- Tailoring a firewall
  - Consider type of traffic to filter
  - Consider exceptions to rules

Firewalls – Packet Filter

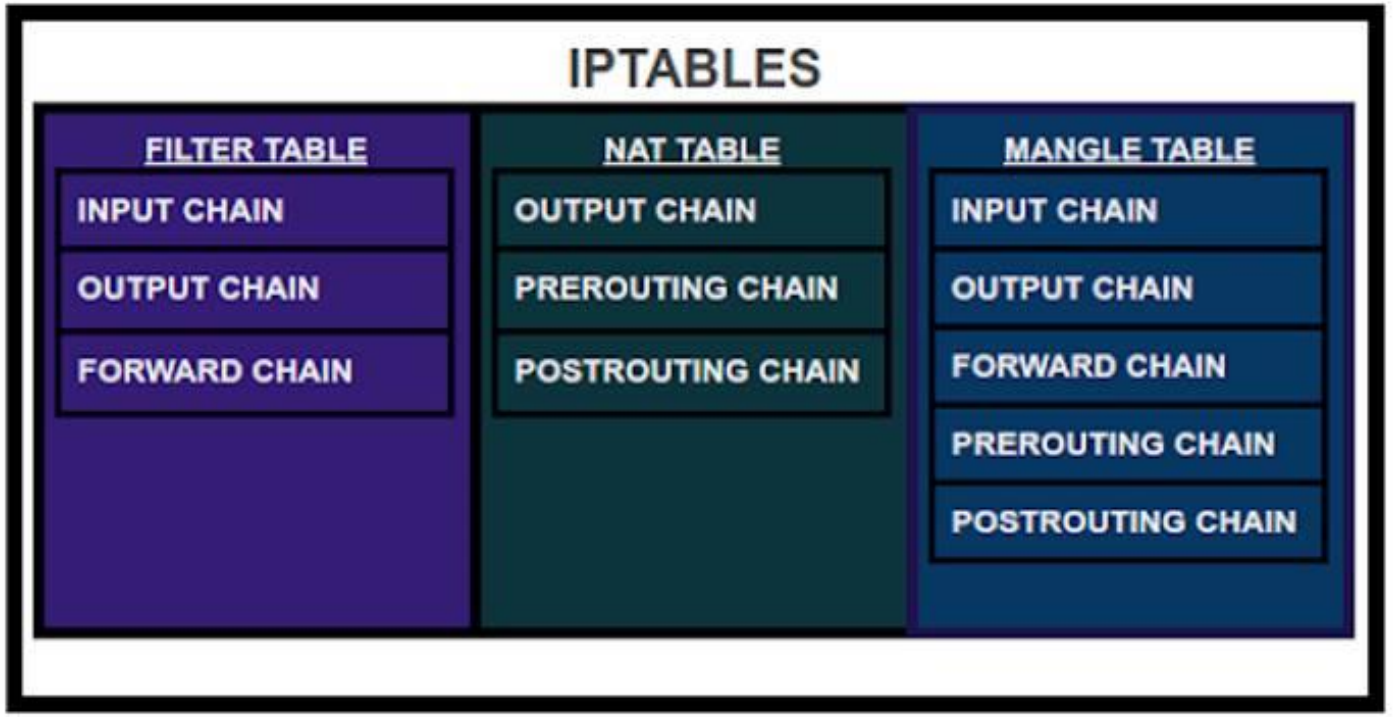
- Most common form of firewall
  - Typically thought of as default type of firewall
- Rules based (Port and IP)
  - Static rules allow packets on particular ports and to/from IP addresses
  - Dynamic rules track destinations based on connections originating for inside
- Common packet-filtering firewall criteria
  - Source / destination IP addresses and subnet masks
  - Source / destination ports
  - Flags set in the IP header
  - Transmissions using TCP, UDP or ICMP protocols
  - Packet's status as first packet in new data stream vs. subsequent packets
  - Packet's status as inbound to or outbound from private network
- Packet-filtering firewalls
  - Cannot distinguish user trying to breach firewall from authorized user

iptables

- What is iptables
  - Firewall utility built for Linux operating systems
  - Stateful
    - But can be configured in a stateless manner
  - Uses policy chains to allow or block traffic
    - List based
- Types of chains
  - Input: used to control behavior for incoming connections
  - Forward: used for rerouting of traffic or NAT
  - Output: used to control behavior for outgoing connections
    - Need to consider return data as well

Iptables (cont'd)

- Filter table
  - Control flow of packets to and from the system
- NAT table
  - Redirect connections to other interfaces on network
- Mangle table
  - Modify packet headers



Source: linode.com

iptables (cont'd)

- Policy chain default behavior
  - What should iptables do if the connection doesn't match any existing rules?
    - ACCEPT
    - DROP (deny)
    - REJECT (deny)

```
user1@cecshost1:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
user1@cecshost1:~$
```

iptables List Example – Verbose


```
user1@cecshost1:/var/log$ sudo iptables -L -v
Chain INPUT (policy DROP 61 packets, 4918 bytes)
 pkts bytes target     prot opt in     out     source               destination
  125 10238 ACCEPT     all  --  lo     any     anywhere             anywhere
 1064 80632 ACCEPT     all  --  any    any     anywhere             anywhere             ctstate RELATED,ESTABLISHED
    1    84 ACCEPT     icmp --  any    any     anywhere             anywhere             state NEW,RELATED,ESTABLISHED
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp spt:ssh state ESTABLISHED
   51  4138 LOG        all  --  any    any     anywhere             anywhere             limit: avg 5/min burst 5 LOG level debug prefix "iptab
les denied: "
    1    52 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:ssh state NEW,ESTABLISHED

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy DROP 113 packets, 6780 bytes)
 pkts bytes target     prot opt in     out     source               destination
  125 10238 ACCEPT     all  --  any    lo      anywhere             anywhere
  937  171K ACCEPT     all  --  any    any     anywhere             anywhere             ctstate ESTABLISHED
    2   168 ACCEPT     icmp --  any    any     anywhere             anywhere             state NEW,RELATED,ESTABLISHED
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:ssh state NEW,ESTABLISHED
   16  1199 ACCEPT     udp  --  any    any     anywhere             anywhere             udp dpt:domain ctstate NEW
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:domain ctstate NEW
```

iptables List Example – Verbose and Numeric

CALIFORNIA STATE UNIVERSITY
LONG BEACH
College of Engineering

```
user1@cecshost1:/var/log$ sudo iptables -L -n -v
Chain INPUT (policy DROP 61 packets, 4918 bytes)
 pkts bytes target     prot opt in     out     source               destination
  121  9930 ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
  710 50344 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           ctstate RELATED,ESTABLISHED
    1    84 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0           state NEW,RELATED,ESTABLISHED
    0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp spt:22 state ESTABLISHED
   51  4138 LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0           limit: avg 5/min burst 5 LOG flags 0 level 7 prefix "i
ptables denied: "
    1    52 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:22 state NEW,ESTABLISHED

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy DROP 113 packets, 6780 bytes)
 pkts bytes target     prot opt in     out     source               destination
  121  9930 ACCEPT     all  --  *      lo      0.0.0.0/0            0.0.0.0/0
  625 93869 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           ctstate ESTABLISHED
    2   168 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0           state NEW,RELATED,ESTABLISHED
    0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:22 state NEW,ESTABLISHED
   16  1199 ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0           udp dpt:53 ctstate NEW
    0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:53 ctstate NEW
```

iptables List Example – Verbose and Line Numbers

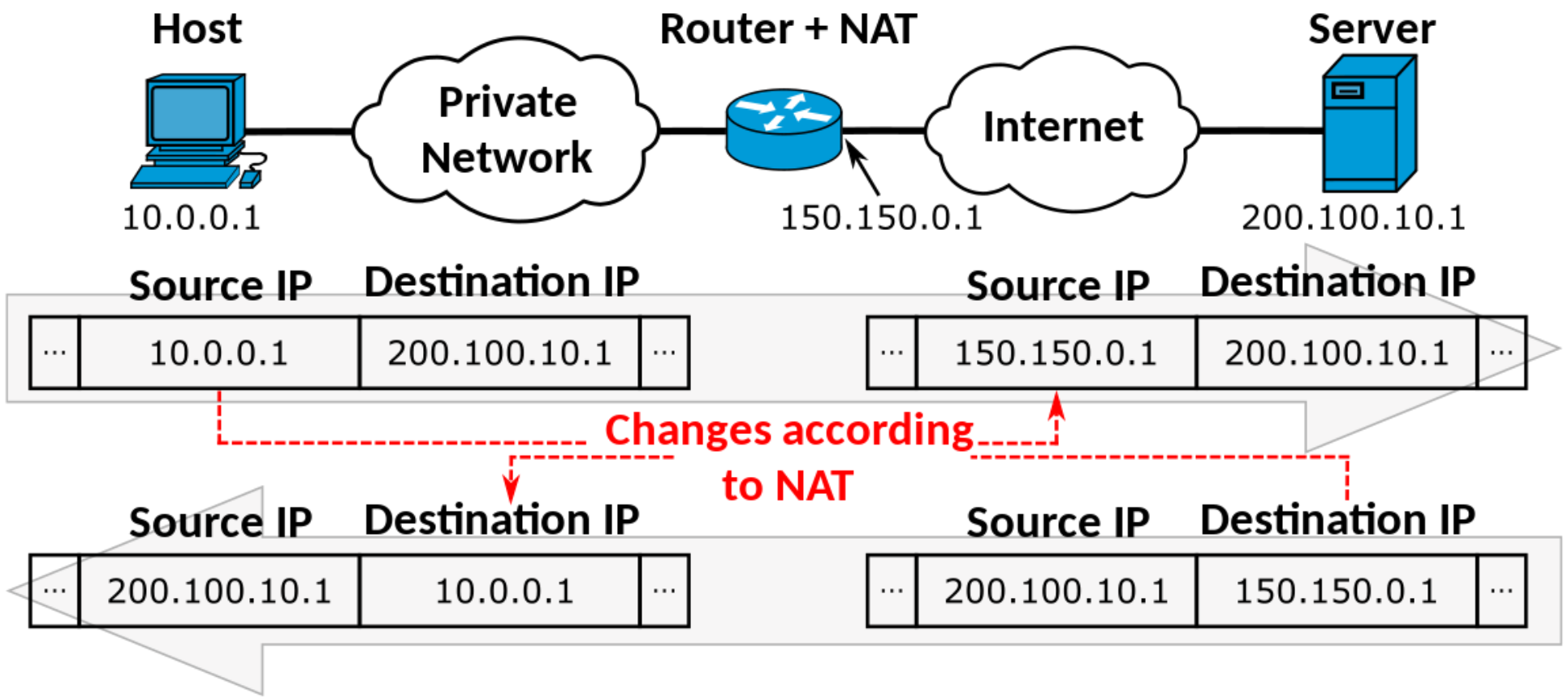
CALIFORNIA STATE UNIVERSITY
LONG BEACH
College of Engineering

```
user1@cecshost1:/var/log$ sudo iptables -L --line-numbers
Chain INPUT (policy DROP)
num  target     prot opt source               destination
1    ACCEPT     all  --  anywhere             anywhere
2    ACCEPT     all  --  anywhere             anywhere             ctstate RELATED,ESTABLISHED
3    ACCEPT     icmp --  anywhere             anywhere             state NEW,RELATED,ESTABLISHED
4    ACCEPT     tcp  --  anywhere             anywhere             tcp spt:ssh state ESTABLISHED
5    LOG        all  --  anywhere             anywhere             limit: avg 5/min burst 5 LOG level debug prefix "iptables denied: "
6    ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh state NEW,ESTABLISHED

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination

Chain OUTPUT (policy DROP)
num  target     prot opt source               destination
1    ACCEPT     all  --  anywhere             anywhere
2    ACCEPT     all  --  anywhere             anywhere             ctstate ESTABLISHED
3    ACCEPT     icmp --  anywhere             anywhere             state NEW,RELATED,ESTABLISHED
4    ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh state NEW,ESTABLISHED
5    ACCEPT     udp  --  anywhere             anywhere             udp dpt:domain ctstate NEW
6    ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:domain ctstate NEW
user1@cecshost1:/var/log$
```

Address Translation

- Private Network
  - Access typically restricted
    - ➢ Clients and machines have proper authentication mechanisms
  - Hiding IP addresses
    - ➢ Provides more flexibility in assigning addresses
- NAT (Network Address Translation)
  - Gateway replaces client's private IP address with public (internet-recognized) IP address
    - ➢ Occurs in packet header
  - Separates private / public transmissions on TCP/IP network
- Reasons for using address translation
  - Overcome IPv4 address availability limitations
  - Add small level of security to private networks that need connectivity to public networks

NAT Example



Source: https://upload.wikimedia.org/wikipedia/commons/c/c7/NAT_Concept-en.svg

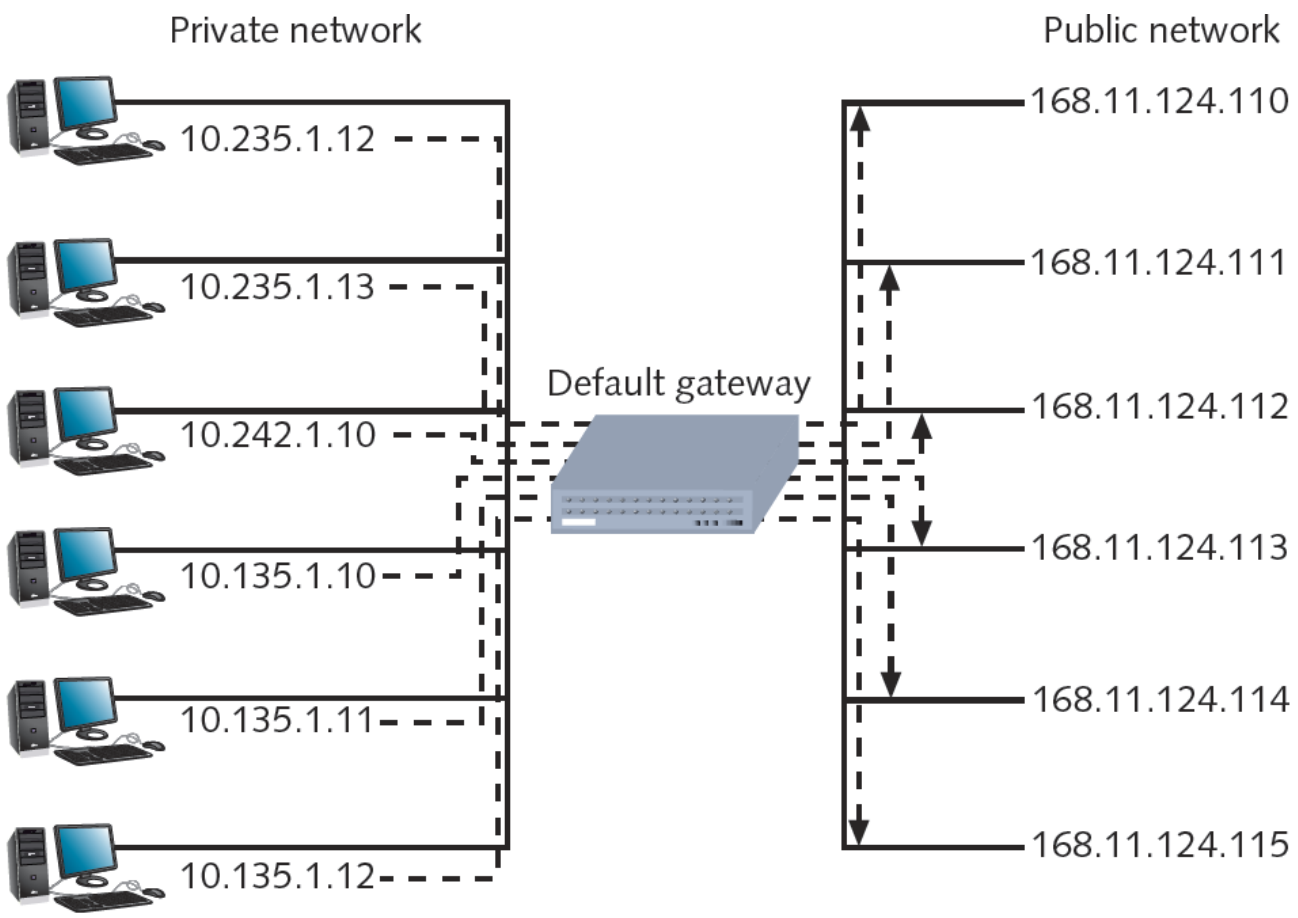
NAT Types

- SNAT (Static Network Address Translation)
  - Client associated with one private IP address and one public IP address
  - Addresses never (rarely) change
  - Useful when running services such as a mail server
    - Helps to avoid IP blacklisting
- DNAT (Dynamic Network Address Translation)
  - Also called IP masquerading
  - Internet-valid IP address might be assigned to any client's outgoing transmission
- PAT (Port Address Translation)
  - Each client session with a server on the Internet is assigned a separate TCP port number
    - Client-server packets (headers) contain this port number
  - Internet server responds to packet's source address using same port

Private and Link-Local Addresses (review)

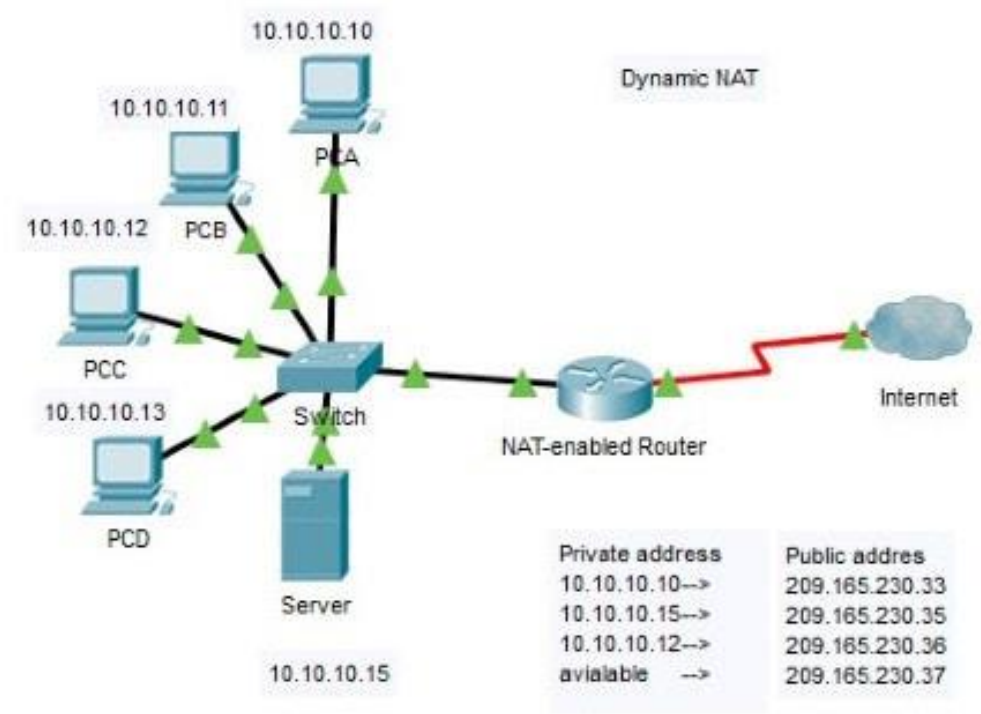
- Private addresses
  - Allow hosts in organization to communicate across internal network
  - Cannot be routed on public network
- Specific IPv4 address ranges reserved for private addresses
  - Class A: 10.0.0.0 to 10.255.255.255
  - Class B: 172.16.0.0 to 172.31.255.255
  - Class C: 192.168.0.0 to 192.168.255.255
- Link-local address
  - Provisional address
  - Capable of data transfer only on local network segment

SNAT Example



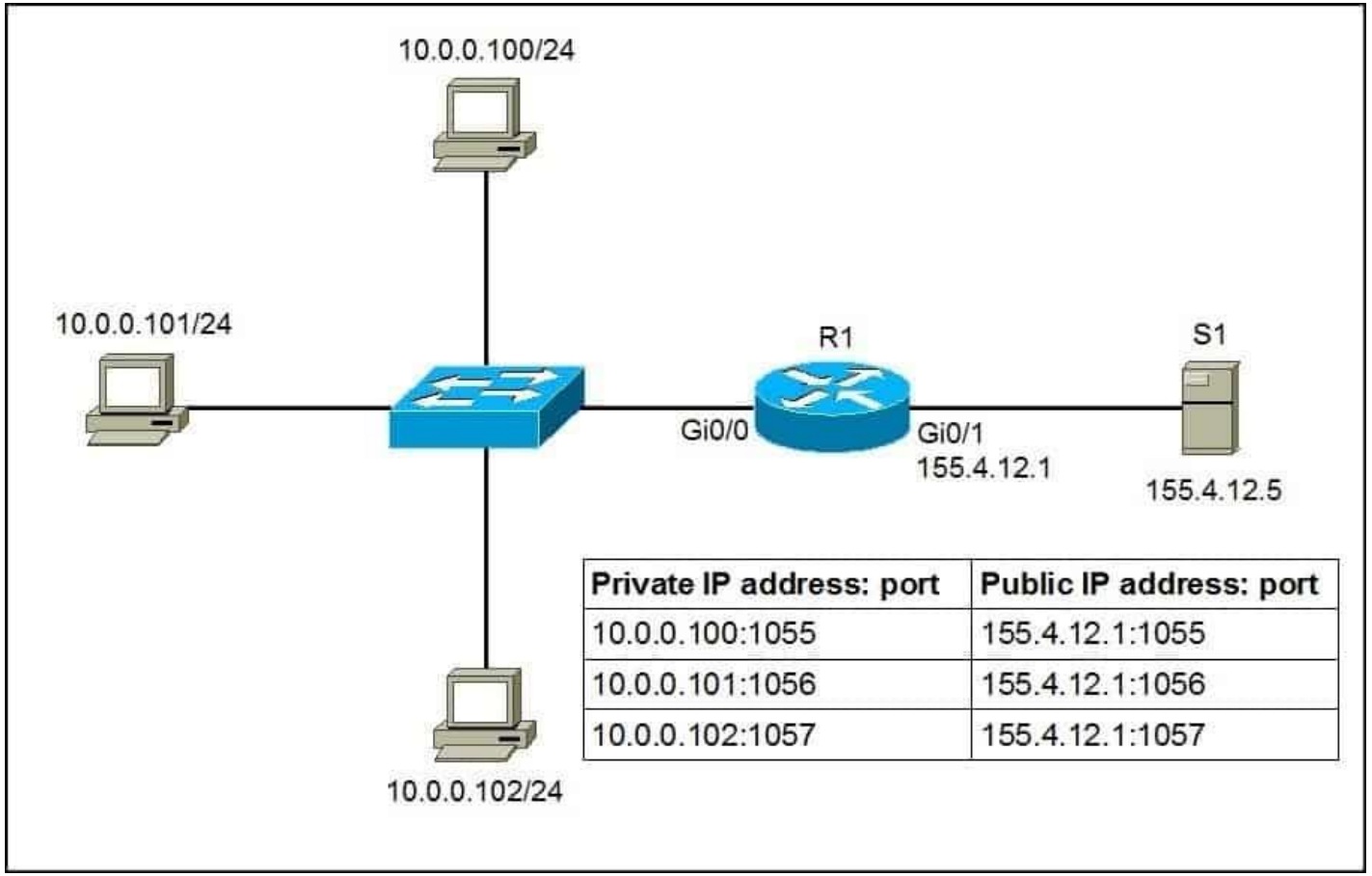
Source: Course Technology / Cengage Learning

DNAT Example



Source: Section (https://www.section.io/engineering-education/implementing-network-address-translation/#configuring-dynamic-nat)

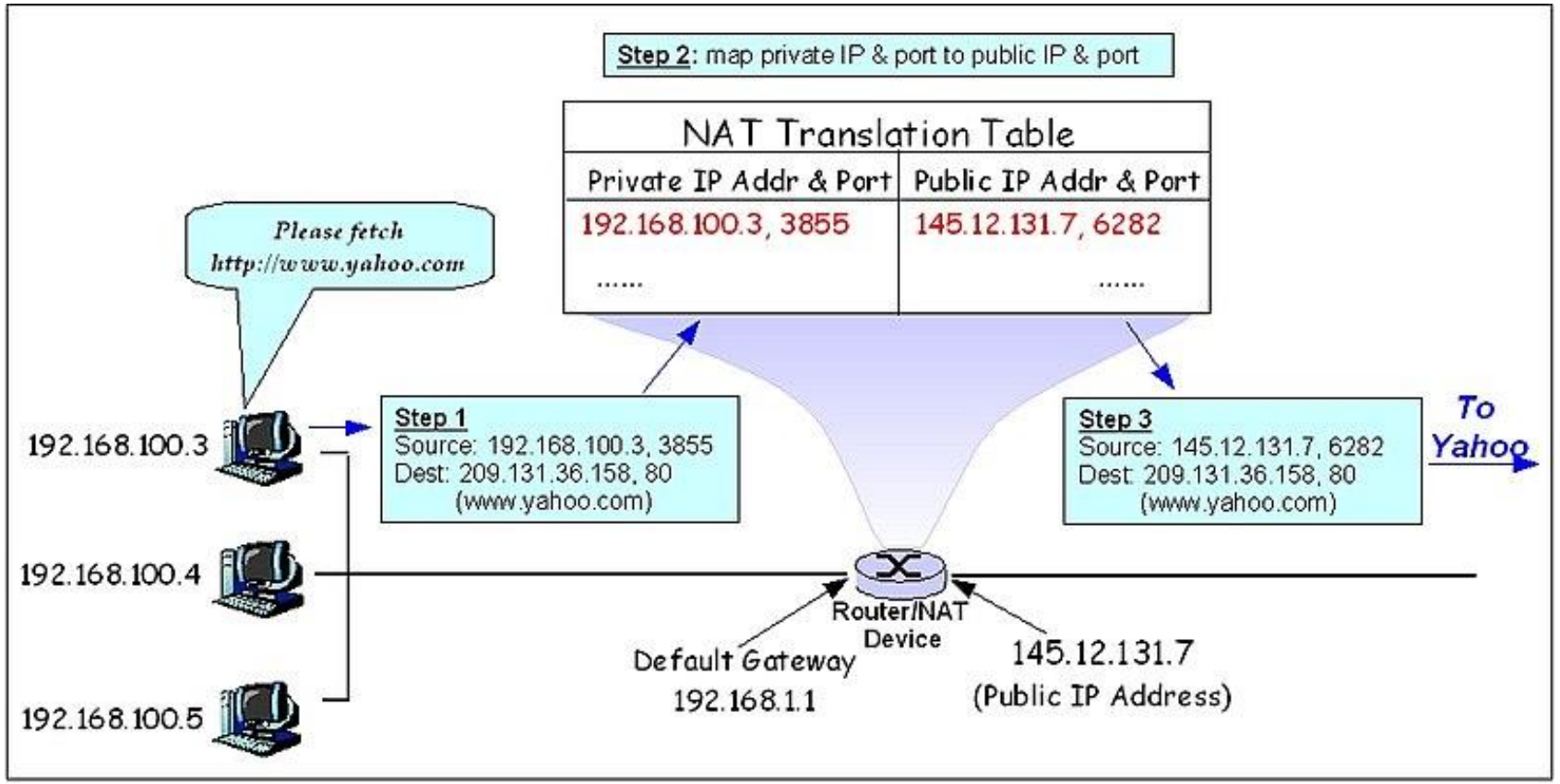
PAT Example (1)



Private IP address: port	Public IP address: port
10.0.0.100:1055	155.4.12.1:1055
10.0.0.101:1056	155.4.12.1:1056
10.0.0.102:1057	155.4.12.1:1057

Source: https://study-ccna.com/port-address-translation-pat-configuration/

PAT Example (2)



Actual Dynamic / Private Ports Range: TCP/49,152 – TCP/65,535

Source: https://en.wikipedia.org/wiki/File:Network_Address_Translation_(file2).jpg

Cisco Command Line Example

CALIFORNIA STATE UNIVERSITY
LONG BEACH
College of Engineering

We will configure the Cisco Router to perform Static NAT on the IP address 10.1.1.200 owned by Web Server and Dynamic NAT to translate the IP addresses of three hosts to dynamically to a pool of addresses.

Router(config)**interface fastethernet 0/0**

Router(config-if)**ip address 10.1.1.1 255.255.255.0**

Router(config-if)**ip nat inside**

Router(config)**interface fastethernet 0/1**

Router(config-if)**ip address 116.100.100.194 255.255.255.248**

Router(config-if)**ip nat outside**

Router(config)**ip nat inside source static 10.1.1.200 116.100.100.195**

— The command above configures static NAT for private IP address 10.1.1.200 to public IP address 116.100.100.195 —

Router(config)**access-list 101 permit ip 10.1.1.10 any**

Router(config)**access-list 101 permit ip 10.1.1.11 any**

Router(config)**access-list 101 permit ip 10.1.1.12 any**

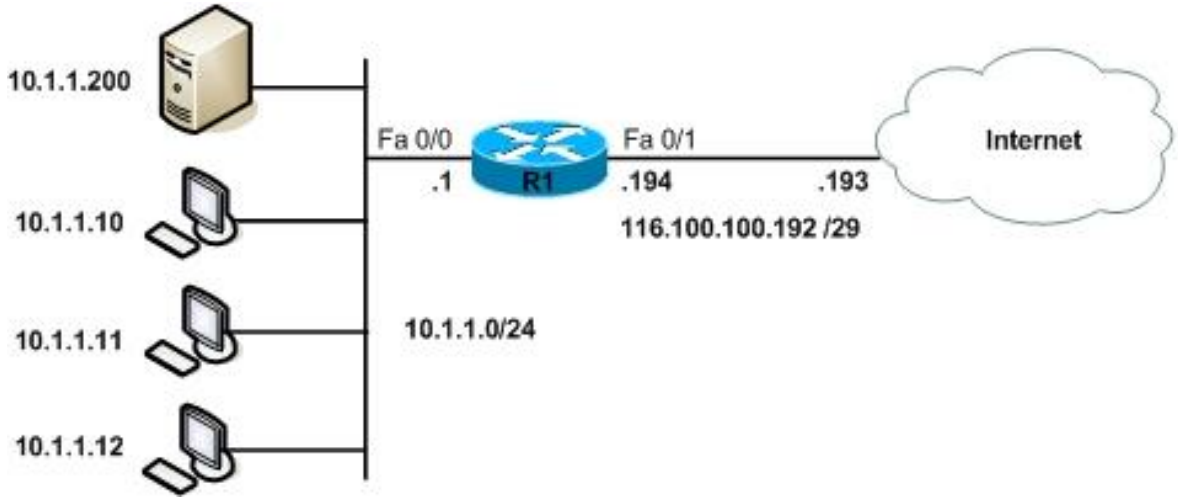
Router(config)**ip nat pool DYN_NAT_POOL 116.100.100.196 116.100.100.198 prefix-length 24**

Router(config)**ip nat inside source list 101 pool DYN_NAT_POOL**

— The commands above configure Dynamic NAT for a group three hosts which are assigned public IP addresses from a pool of three public IP addesses —

We can also configure Port Address Translation for the three hosts such that all three of them will be overloaded to a single IP address. To configure PAT use the following command

Router(config)**ip nat inside source list 101 interface fastethernet 0/1 overload**



Source: certificationkits.com (https://www.certificationkits.com/cisco-certification/ccna-articles/cisco-ccna-network-address-translation-nat/network-address-translation-a-configuration/)

Summary

- Access lists inspects packets but are stateless
- Firewalls are stateful
- Packet filter firewall is the most common type
- iptables is a linux-based stateful firewall
- NAT is how private network communicate with publicly available networks