CALIFORNIA STATE UNIVERSITY
LONG BEACH
College of Engineering

CECS 303: Networks and Network Security

Network Security Principles (cont'd)

Chris Samayoa

Week 5 – 1st Lecture
2/15/2022

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 Network Security fundamentals
- Cryptography introduction
- Authentication basics

Common Issue

- Loosely managed systems
- Security is made even more difficult to implement since today's systems lack a central point of control
  - Home machines unmanaged
  - Networks managed by different organizations.
  - A single function touches machines managed by different parties.
    - Cloud
  - Who is in control?

General Security Concerns

- Buggy code
- Protocol design failures
- Weak crypto
- Social engineering
- Insider threats
- Poor configuration
- Incorrect policy specification
- Stolen keys or identities
- Denial of service

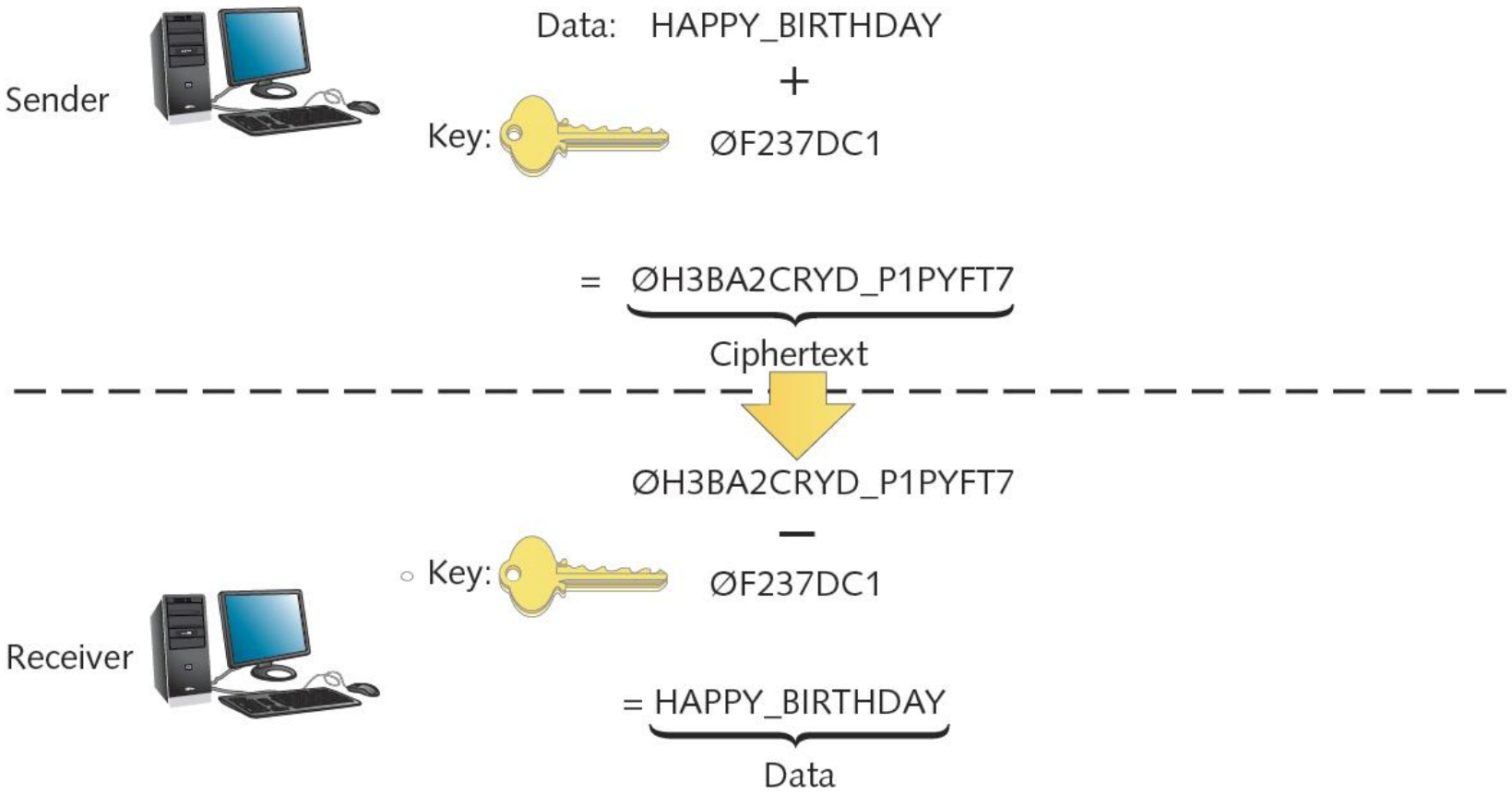
Security Mechanisms

- Encryption
- Checksums
- Key management
- Authentication
- Authorization
- Audit logs
- Firewalls
- Virtual Private Nets (VPNs)
- Intrusion detection
- Intrusion response
- Development tools
- Virus Scanners / (M)EDR
- Policy managers
- Trusted hardware

Cryptography

- Use of algorithm to scramble data
- Cryptography underlies many fundamental security services
  - Confidentiality
    - Data can be viewed only by intended recipient
  - Data integrity
    - Data not modified between being sent and received
    - Data was not forged by an intruder
  - Authentication
- Many encryption forms exist
- Functions as a basic building block for security services

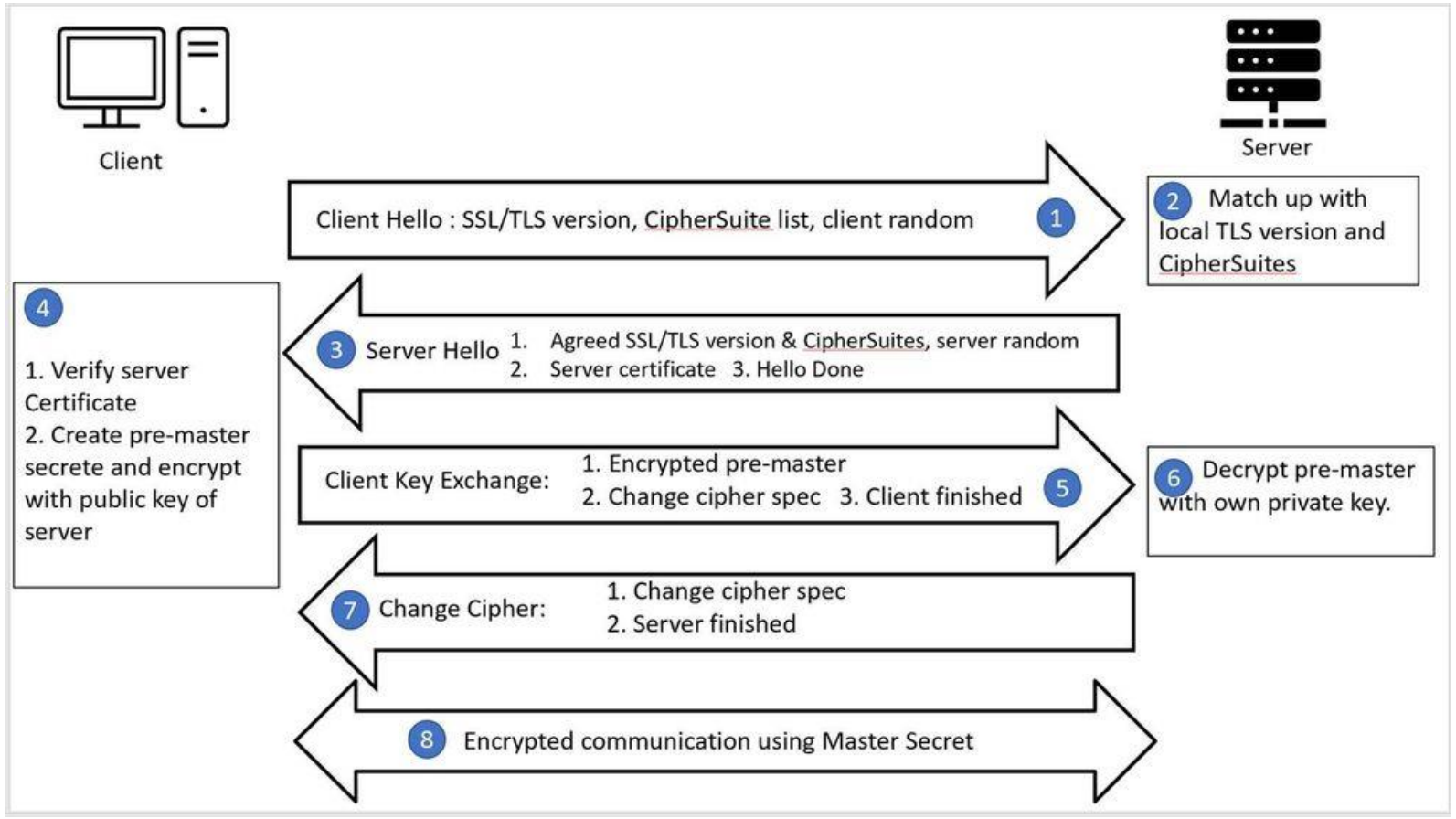
Encryption and Decryption



Common Encryption Types

- Symmetric-key (conventional)
  - Single key used for both encryption and decryption
    - Known only by sender and receiver
  - Keys are typically short, because key space is densely filled
  - Ex: AES, DES, 3DES, RC4, Blowfish, IDEA, etc.
  - Drawback: Sender needs to share private key with recipient
- Asymmetric keys (public-private)
  - Two keys: one for encryption, one for decryption
    - Private key: User knows
    - Public Key: Anyone may request (usually from a publicly accessible host)
  - Keys are typically long, because key space is sparsely filled
  - Ex: RSA, El Gamal, DSA, etc
- Often used in combination
  - e.g. Diffie-Hellman in TLS to exchange AES cipher

TLS



Source: Microsoft

Identification vs Authentication

- Identification
  - Associating an identify with an individual, process, or request
- Authentication
  - Verification of a claimed identity
  - Ideally
    - Who you are
  - Practically
    - Something you know
    - Something you have
    - Something you are

Something You Know

- Password or Algorithm
  - e.g. Encryption key derived from password
- Issues
  - How to keep it secret?
    - Find it, sniff it, social engineer it
  - You need to remember it
  - How is it stored and checked?
- Potential attacks
  - Brute force
  - Dictionary
  - Pre-computed Dictionary
  - Guessing
  - Finding elsewhere

Passwords

- Can have too many password or too few passwords
  - Can lead to reuse of passwords
  - People are lazy
  - Can be mitigated by password vaults
- Passwords need to be presented
  - Relies on uncompromised verifier
- Password recommendations changed
  - Length over special characters at this point

Something You Have

- Cards
  - Mag stripe
  - Smart Card
  - USB Key
  - Time varying password
- Issues
  - How to validate?
    - Verifier can be compromised
  - Need special infrastructure
  - e.g. RSA SecureID (https://www.wired.com/2011/06/rsa-replaces-securid-tokens/)

Something You Are

- Biometrics
  - Iris scan
  - Fingerprint
  - Picture
  - Voice
- Issues
  - Need to prevent spoofing

Summary

- Cryptography is a building block for network security
- Authentication
  - Something you know
  - Something you have
  - Something you are

CALIFORNIA STATE UNIVERSITY
LONG BEACH
College of Engineering

CECS 303: Networks and Network Security

Firewalls

Chris Samayoa

Week 5 – 1st Lecture
2/15/2022

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

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

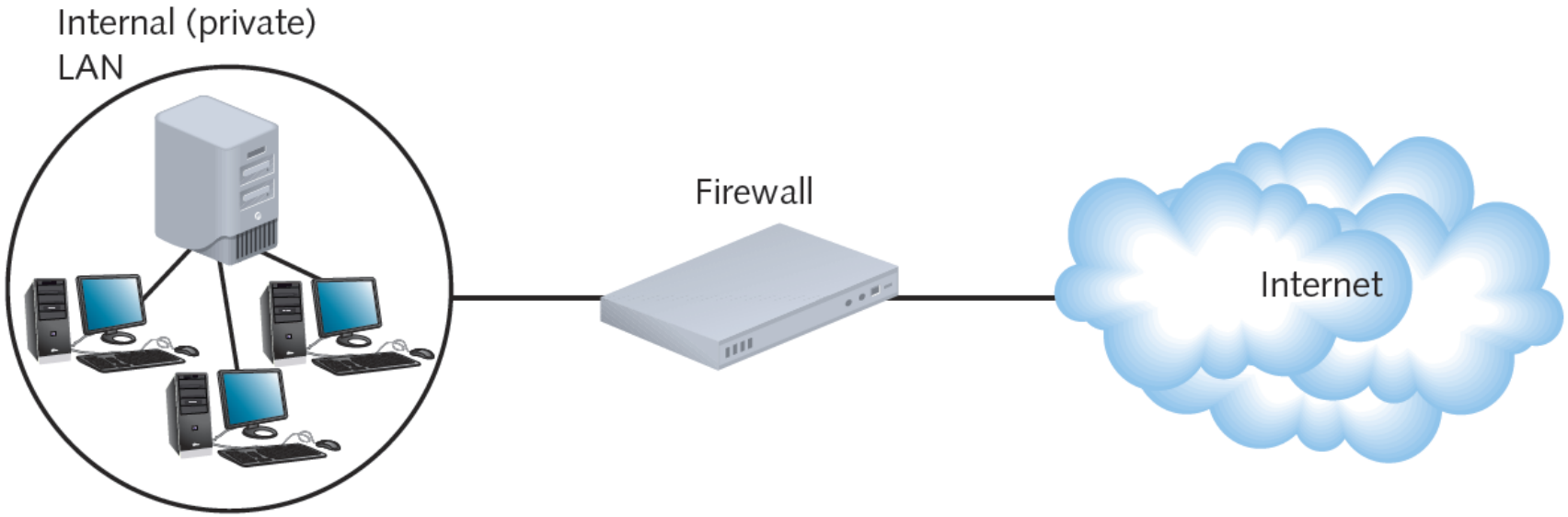
```
R1(config-ext-nacl)#do sh access-list OutBoundAccess
Extended IP access list OutBoundAccess
    10 permit ip 192.168.1.0 0.0.0.255 any
    11 deny tcp 192.168.2.0 0.0.0.127 any eq smtp
    12 deny tcp 192.168.2.0 0.0.0.127 any eq sunrpc
    13 deny tcp 192.168.2.0 0.0.0.127 any eq pop2
    14 deny 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

Firewalls (cont'd)

- Common packet-filtering firewall criteria
  - Source / destination IP addresses and subnet masks
  - Source / destination ports
  - Flags set in the IP header
  - Transmissions using UDP or ICMP protocols
  - Packet's status as first packet in new data stream, subsequent packet
  - Packet's status as inbound to, outbound from private network

Firewalls (cont'd)

- Port blocking
  - Prevents connection to and transmission completion through ports
- Optional firewall functions
  - Encryption
  - User authentication
  - Central management
  - Easy rule establishment
  - Filtering based on data contained in packets
  - Logging, auditing capabilities
  - Protect internal LAN's address identity
  - Monitor data stream from end to end (stateful firewall)

Firewalls (cont'd)

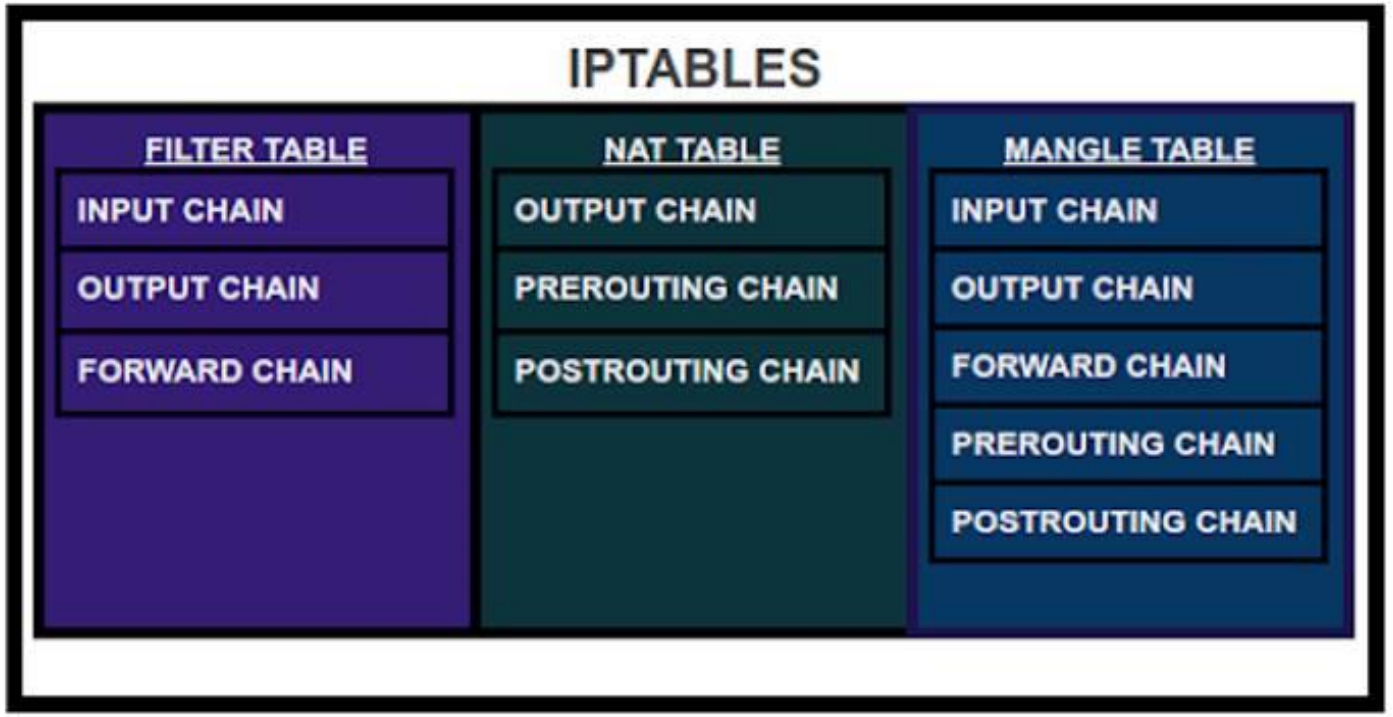
- Tailoring a firewall
  - Consider type of traffic to filter
  - Consider exceptions to rules
- Packet-filtering firewalls
  - Cannot distinguish user trying to breach firewall from authorized user

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$
```

Summary

- Access lists inspects packets but are stateless
- Firewalls are stateful
- iptables is a linux-based stateful firewall