CSCI 4250/6250 – Fall 2013
Computer and Networks Security

Network Security

Goodrich, Chapter 5-6
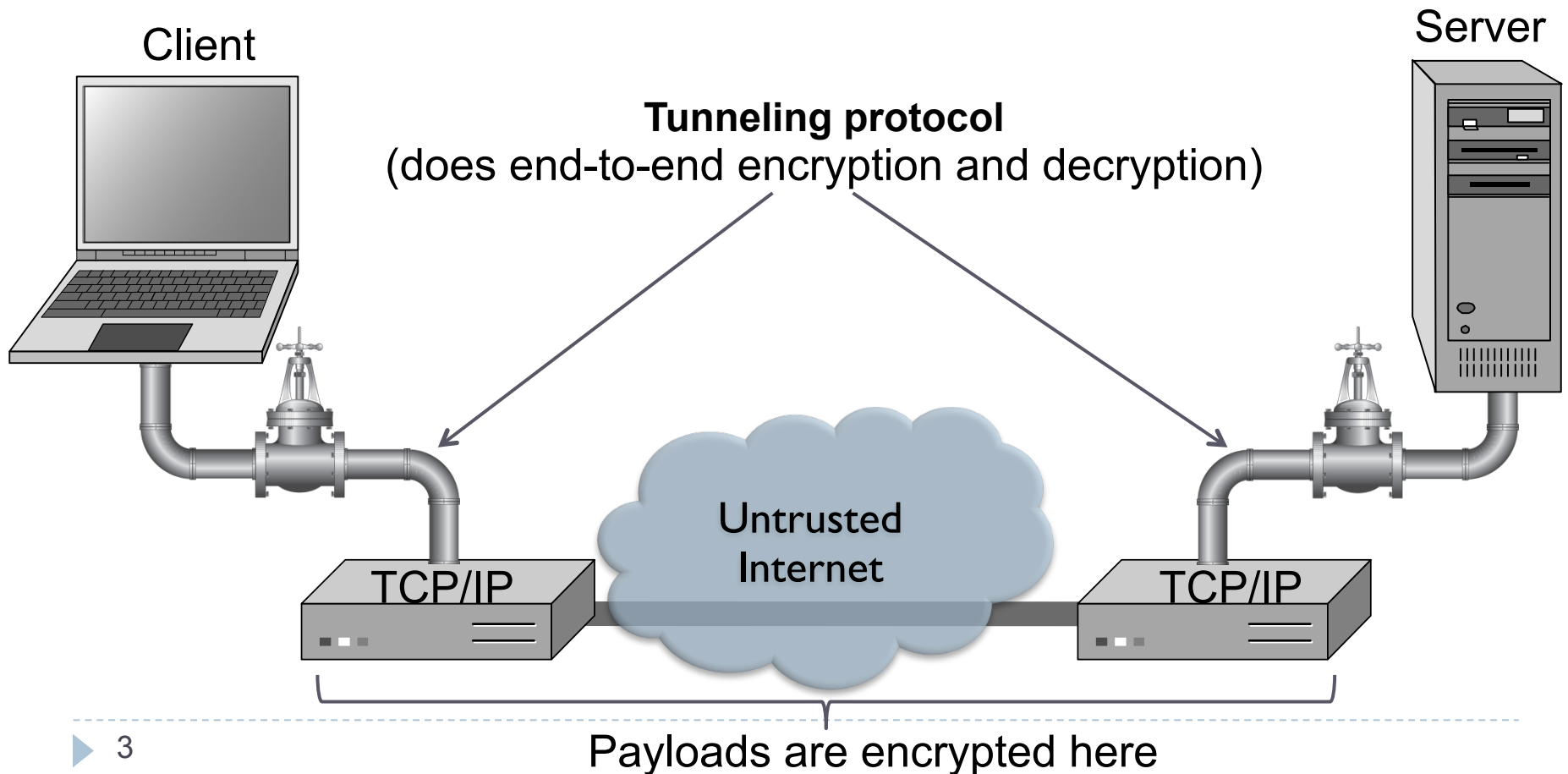
Tunnels

▸ The contents of TCP packets are not normally encrypted, so if someone is eavesdropping on a TCP connection, he can often see the complete contents of the payloads in this session.

▸ One way to prevent such eavesdropping without changing the software performing the communication is to use a **tunneling protocol.**

▸ In such a protocol, the communication between a client and server is automatically encrypted, so that useful eavesdropping is infeasible.

Tunneling Prevents Eavesdropping

▸ Packets sent over the Internet are automatically encrypted.

Client

Server

Tunneling protocol
(does end-to-end encryption and decryption)

Untrusted
Internet

TCP/IP

TCP/IP

Payloads are encrypted here

Secure Shell (SSH)

▸ A secure interactive command session:

1. The client connects to the server via a TCP session.

2. The client and server exchange information on administrative details, such as supported encryption methods and their protocol version, each choosing a set of protocols that the other supports.

3. The client and server initiate a secret-key exchange to establish a shared secret session key, which is used to encrypt their communication (but not for authentication). This session key is used in conjunction with a chosen block cipher (typically AES, 3DES) to encrypt all further communications.

4. The server sends the client a list of acceptable forms of authentication, which the client will try in sequence. The most common mechanism is to use a password or the following public-key authentication method:

   a) If public-key authentication is the selected mechanism, the client sends the server its public key.

   b) The server then checks if this key is stored in its list of authorized keys. If so, the server encrypts a challenge using the client's public key and sends it to the client.

   c) The client decrypts the challenge with its private key and responds to the server, proving its identity.

5. Once authentication has been successfully completed, the server lets the client access appropriate resources, such as a command prompt.

Tunneling with SSH

▸ **Socks proxy demo**

▸ **SOCKS protocol v5**
 ▸ http://tools.ietf.org/html/rfc1928

Firewalls

▸ A **firewall** is an integrated collection of security measures designed to prevent unauthorized electronic access to a networked computer system.

▸ A network firewall is similar to firewalls in building construction, because in both cases they are intended to isolate one "network" or "compartment" from another.

Private Network   Public Network
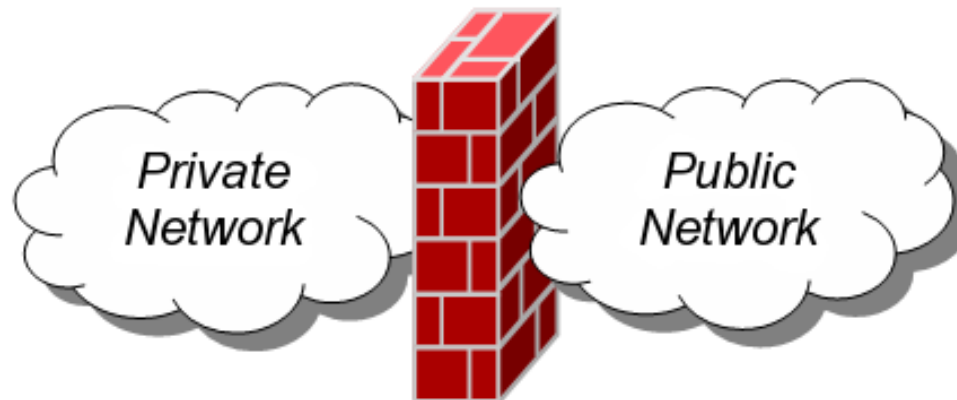
Firewall Policies

▸ To protect private networks and individual machines from the dangers of the greater Internet, a firewall can be employed to filter incoming or outgoing traffic based on a predefined set of rules called **firewall policies**.

Trusted internal network

Firewall policies

Untrusted Internet
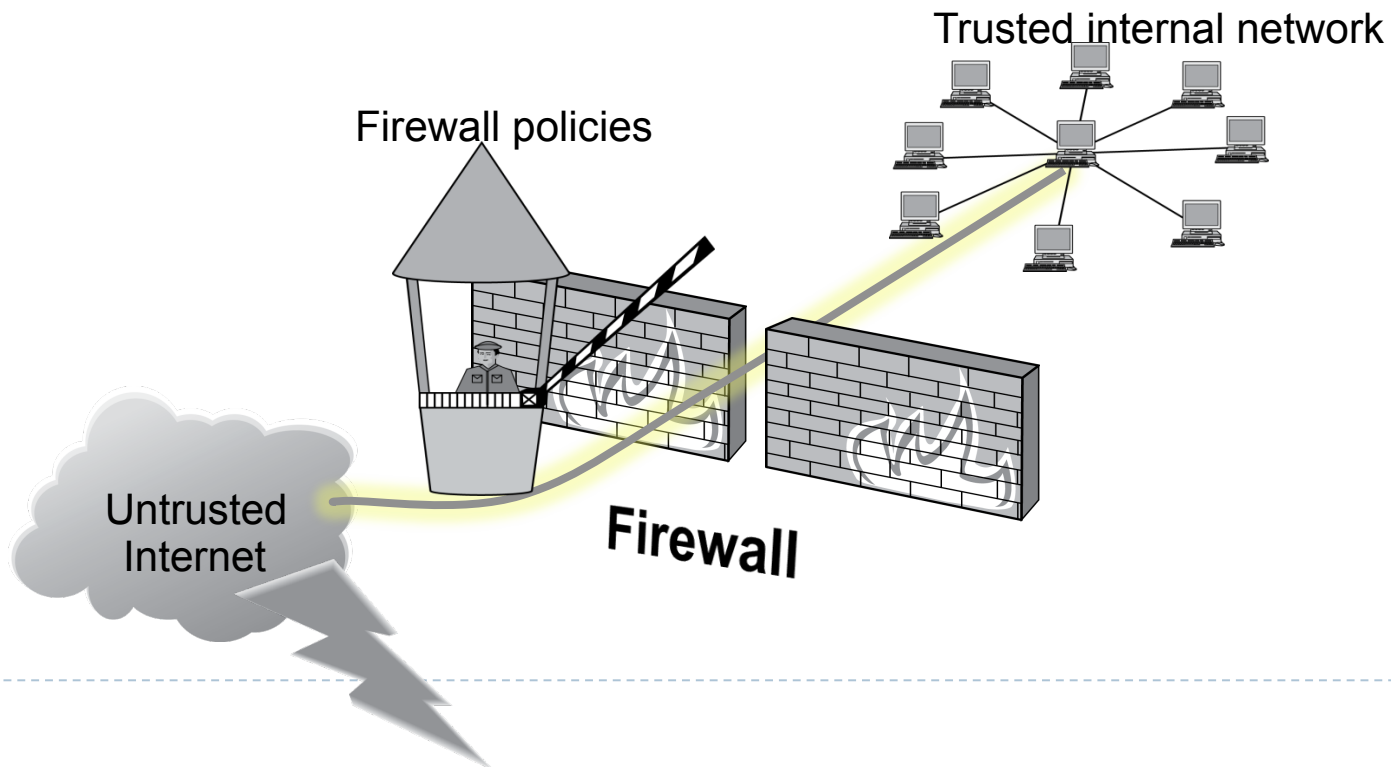
Firewall

Policy Actions

▸ Packets flowing through a firewall can have one of three outcomes:

  ▸ **Accepted:** permitted through the firewall

  ▸ **Dropped:** not allowed through with no indication of failure

  ▸ **Rejected:** not allowed through, accompanied by an attempt to inform the source that the packet was rejected

▸ Policies used by the firewall to handle packets are based on several properties of the packets being inspected, including the protocol used, such as:

  ▸ TCP or UDP

  ▸ the source and destination IP addresses

  ▸ the source and destination ports

  ▸ the application-level payload of the packet (e.g., whether it contains a virus).

Blacklists and White Lists

▸ There are two fundamental approaches to creating firewall policies (or rulesets) to effectively minimize vulnerability to the outside world while maintaining the desired functionality for the machines in the trusted internal network (or individual computer).

▸ **Blacklist** approach

  ▸ All packets are allowed through except those that fit the rules defined specifically in a blacklist.

  ▸ This type of configuration is more flexible in ensuring that service to the internal network is not disrupted by the firewall, but is naïve from a security perspective in that it assumes the network administrator can enumerate all of the properties of malicious traffic.

▸ **Whitelist** approach

  ▸ A safer approach to defining a firewall ruleset is the default-deny policy, in which packets are dropped or rejected unless they are specifically allowed by the firewall.

▸ 9

Firewall Types

- **packet filters (stateless)**
  - If a packet matches the packet filter's set of rules, the packet filter will drop or accept it

- **"stateful" filters**
  - it maintains records of all connections passing through it and can determine if a packet is either the start of a new connection, a part of an existing connection, or is an invalid packet.

- **application layer**
  - It works like a **proxy** it can "understand" certain applications and protocols.
  - It may inspect the contents of the traffic, blocking what it views as inappropriate content (i.e. websites, viruses, vulnerabilities, ...)

Stateless Firewalls

▸ A stateless firewall doesn't maintain any remembered context (or "state") with respect to the packets it is processing. Instead, it treats each packet attempting to travel through it in isolation without considering packets that it has processed previously.
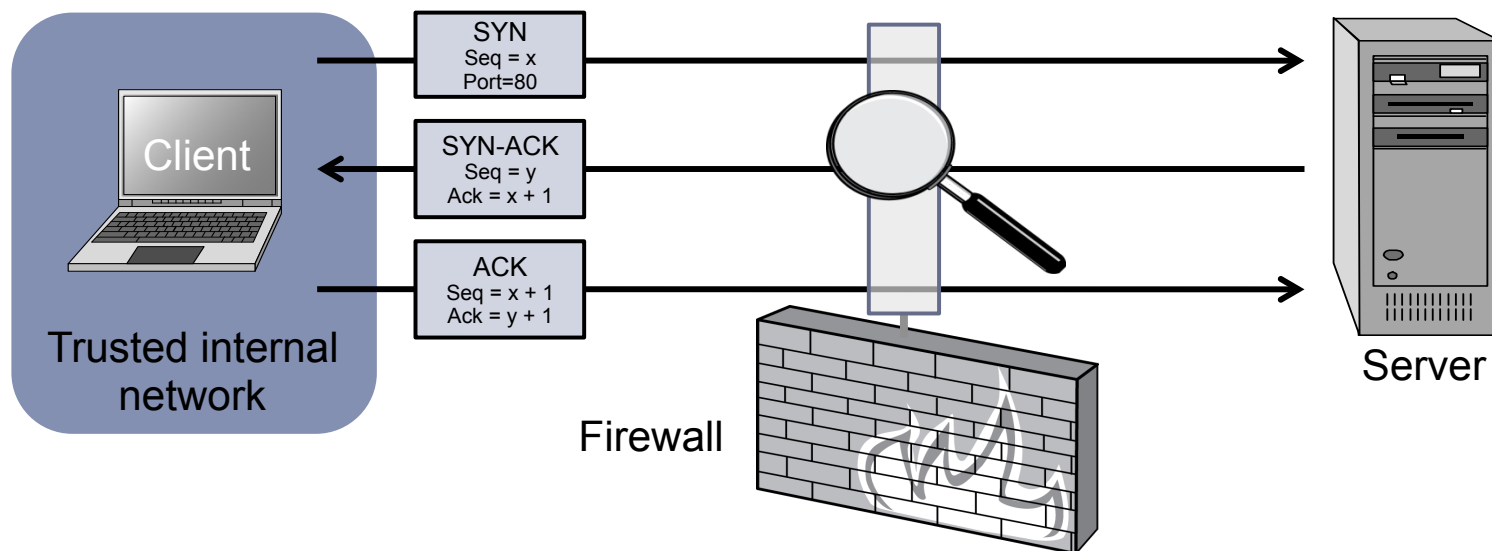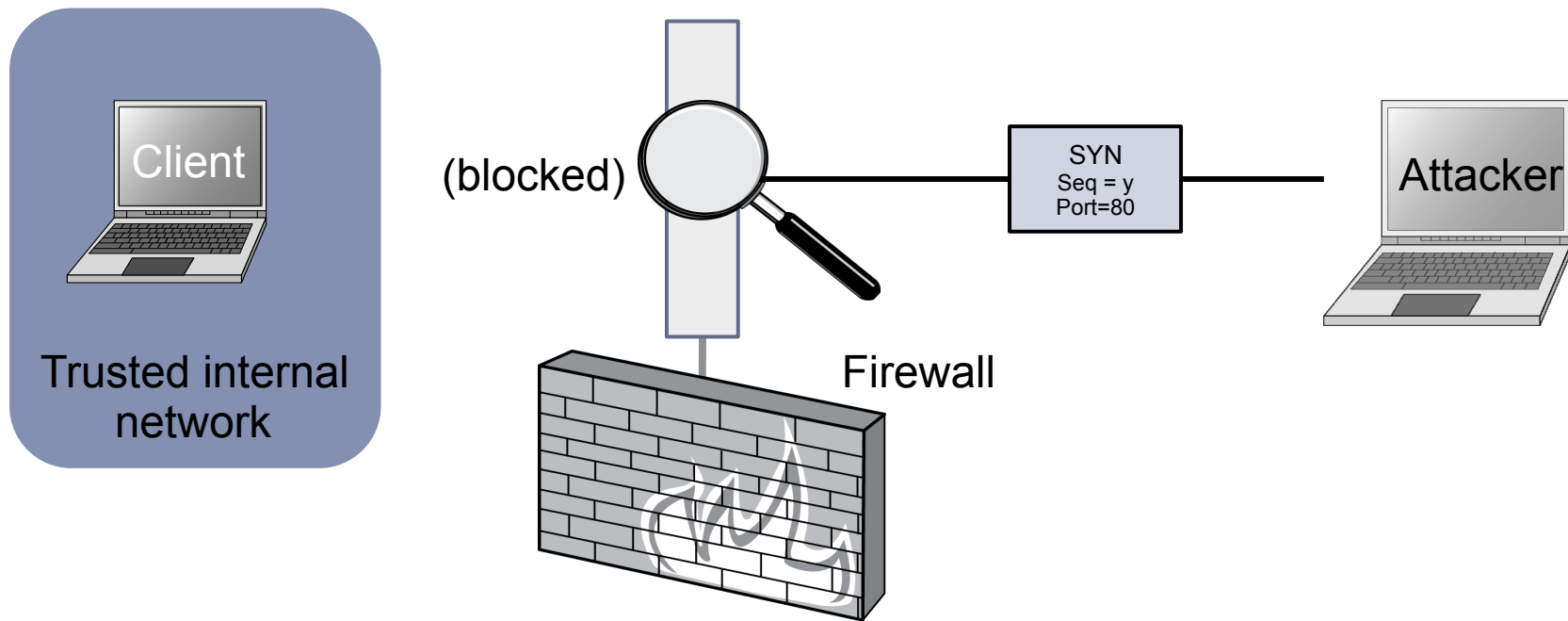

Allow outbound SYN packets, destination port=80
Allow inbound SYN-ACK packets, source port=80

Stateless Restrictions

▶ Stateless firewalls may have to be fairly restrictive in order to prevent most attacks.

Client

Trusted internal network

(blocked)

SYN
Seq = y
Port=80

Attacker

Firewall

Allow outbound SYN packets, destination port=80
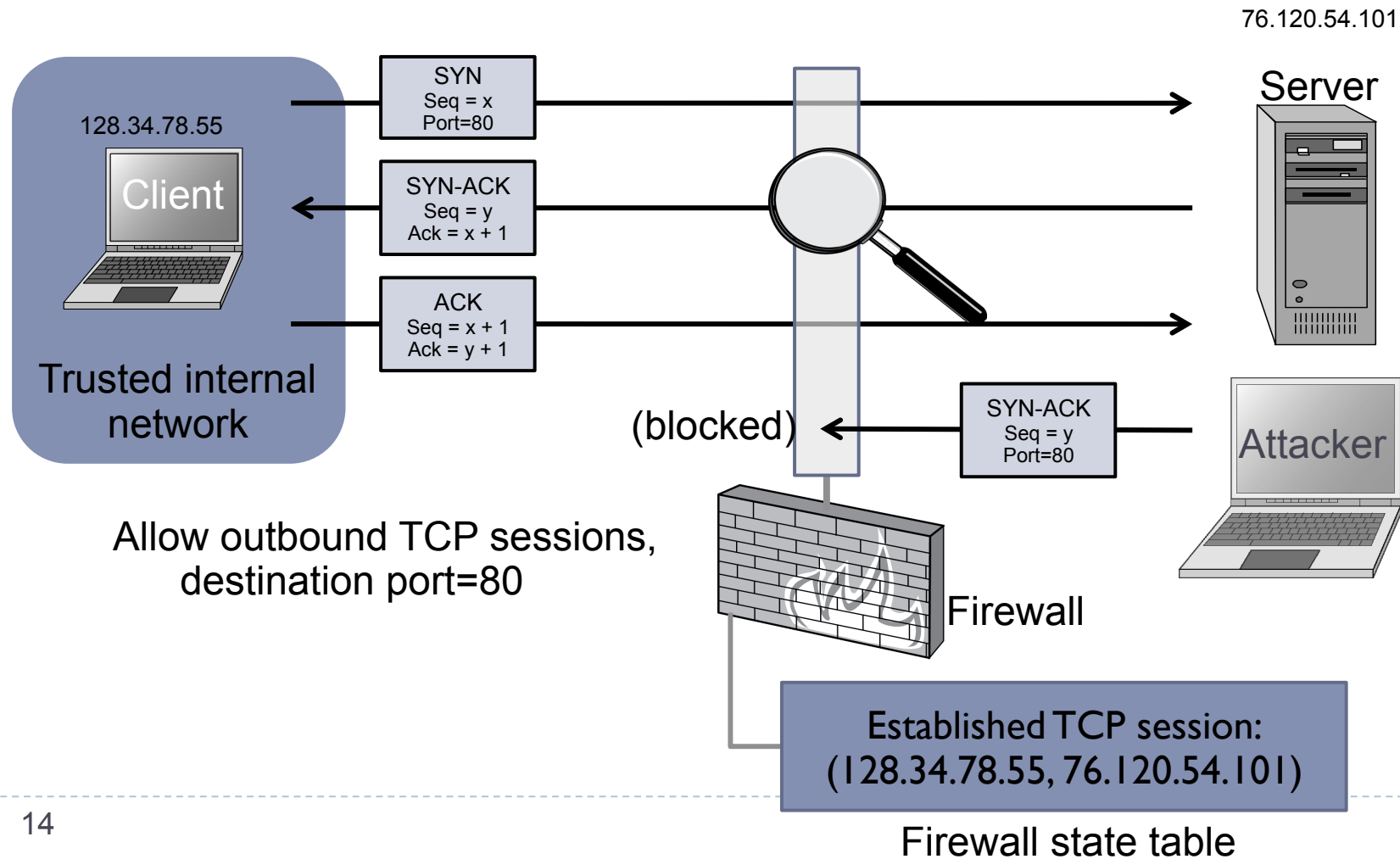Drop inbound SYN packets,
Allow inbound SYN-ACK packets, source port=80

Statefull Firewalls

▸ **Stateful firewalls** can tell when packets are part of legitimate sessions originating within a trusted network.

▸ Stateful firewalls maintain tables containing information on each active connection, including the IP addresses, ports, and sequence numbers of packets.

▸ Using these tables, stateful firewalls can allow only inbound TCP packets that are in response to a connection initiated from within the internal network.

Statefull Firewall Example

▸ **Allow only requested TCP connections:**

76.120.54.101

128.34.78.55

Client

Trusted internal
network

SYN
Seq = x
Port=80

SYN-ACK
Seq = y
Ack = x + 1

ACK
Seq = x + 1
Ack = y + 1

Server

(blocked)

SYN-ACK
Seq = y
Port=80

Attacker

Allow outbound TCP sessions,
destination port=80

Firewall

Established TCP session:
(128.34.78.55, 76.120.54.101)

Firewall state table

Intrusion Detection Systems

- **Intrusion**
  - Actions aimed at compromising the security of the target (confidentiality, integrity, availability of computing/networking resources)
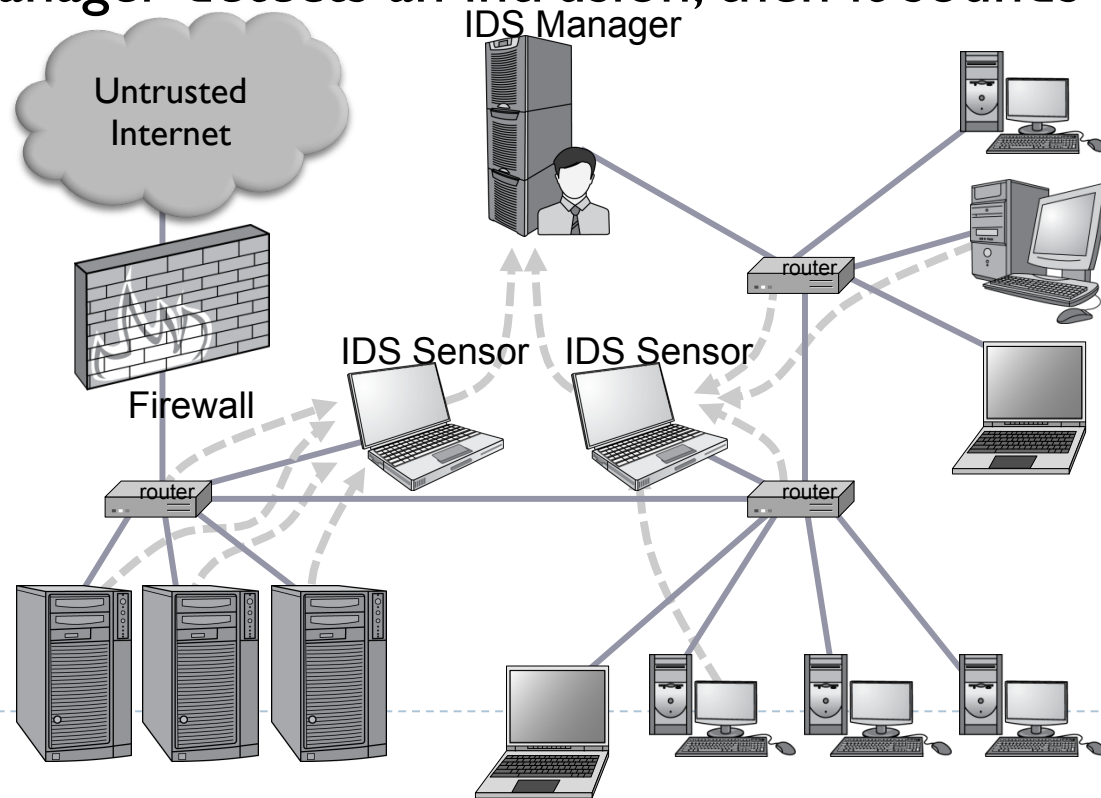
- **Intrusion detection**
  - The identification through intrusion signatures and report of intrusion activities

- **Intrusion prevention**
  - The process of both detecting intrusion activities and managing automatic responsive actions throughout the network

IDS Components

▸ The **IDS manager** compiles data from the IDS sensors to determine if an intrusion has occurred.

▸ This determination is based on a set of **site policies,** which are rules and conditions that define probable intrusions.

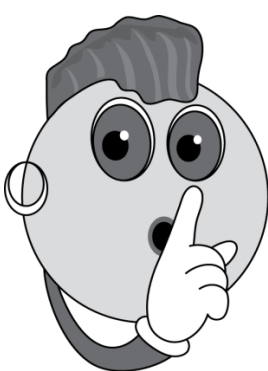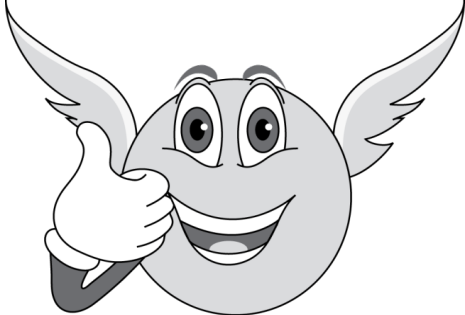▸ If an IDS manager detects an intrusion, then it sounds an **alarm**.

IDS Manager

Untrusted Internet

Firewall

IDS Sensor   IDS Sensor

router

router

router

Intrusions

▸ An IDS is designed to detect a number of threats, including the following:

- ▸ **masquerader:** an attacker who is falsely using the identity and/or credentials of a legitimate user to gain access to a computer system or network
- ▸ **Misfeasor:** a legitimate user who performs actions he is not authorized to do
- ▸ **Clandestine user:** a user who tries to block or cover up his actions by deleting audit files and/or system logs

▸ In addition, an IDS is designed to detect automated attacks and threats, including the following:

- ▸ **port scans:** information gathering intended to determine which ports on a host are open for TCP connections
- ▸ **Denial-of-service attacks:** network attacks meant to overwhelm a host and shut out legitimate accesses
- ▸ **Malware attacks:** replicating malicious software attacks, such as Trojan horses, computer worms, viruses, etc.
- ▸ **ARP spoofing:** an attempt to redirect IP traffic in a local-area network
- ▸ **DNS cache poisoning:** a pharming attack directed at changing a host's DNS cache to create a falsified domain-name/IP-address association

Possible Alarm Outcomes

▸ Alarms can be sounded (positive) or not (negative)

	Intrusion Attack	No Intrusion Attack
Alarm Sounded	True Positive	False Positive
No Alarm Sounded	False Negative	True Negative

The Base-Rate Fallacy

▶ It is difficult to create an intrusion detection system with the desirable properties of having both a high true-positive rate and a low false-negative rate.

▶ If the number of actual intrusions is relatively small compared to the amount of data being analyzed, then the effectiveness of an intrusion detection system can be reduced.

▶ In particular, the effectiveness of some IDSs can be misinterpreted due to a statistical error known as the **base-rate fallacy.**

▶ This type of error occurs when the probability of some conditional event is assessed without considering the "base rate" of that event.

Base-Rate Fallacy Example

▸ Suppose an IDS is 99% accurate, having a 1% chance of false positives or false negatives. Suppose further…

▸ An intrusion detection system generates 1,000,100 log entries.

▸ Only 100 of the 1,000,100 entries correspond to actual malicious events.

▸ Because of the success rate of the IDS, of the 100 malicious events, 99 will be detected as malicious, which means we have **1 false negative.**

▸ Nevertheless, of the 1,000,000 benign events, 10,000 will be mistakenly identified as malicious. That is, we have **10,000 false positives!**

▸ Thus, there will be 10,099 alarms sounded, 10,000 of which are false alarms. That is, roughly 99% of our alarms are false alarms.

Base-Rate Fallacy

▸ Police has a "drunk test" with these properties

 ▸ True positive rate (TPR) = 100%

 ▸ False positive rate = 5%

▸ Assume that about 1/1000 people actually drink and drive. What is the probability P(drunk | Positive) ?

 ▸ P(Positive|drunk) = 1

 ▸ P(Positive|sober) = 0.05

Base-Rate Fallacy

▸ Police has a "drunk test" with these properties

  ▸ True positive rate (TPR) = 100%

  ▸ False positive rate = 5%

▸ Assume that about 1/1000 people actually drink and drive. What is the probability P(drunk | Positive) ?

  ▸ P(Positive|drunk) = 1

  ▸ P(Positive|sober) = 0.05

  ▸ P(drunk | Positive) = P(Positive|drunk)*P(drunk) / P(Positive)

    ▸ =   P(Positive|drunk)*P(drunk) /
          [P(Positive|drunk)*P(drunk) + P(Positive|sober)*P(sober)]

▸ Therefore

  ▸ P(drunk | Positive) = 1*0.001 / [1*0.001 + 0.05*0.999] ~= 0.02

Base-Rate Fallacy

Let I and $\neg I$ denote *intrusive*, and *non-intrusive* behaviour respectively, and A and $\neg A$ denote the presence or absence of an intrusion alarm. Working backwards from the above set of assumptions we can obtain the required values of the:

True Positive rate Or *detection* rate. The probability $P(A|I)$, i.e. that quantity that we can obtain when testing our detector against a set of scenarios we know represent intrusive behaviour.

False Positive rate The probability $P(A|\neg I)$, i.e. the *false alarm* rate, obtained in the same manner as above.

The other two parameters, $P(\neg A|I)$: the *False Negative* rate, and $P(\neg A|\neg I)$: the *True Negative* rate are easily obtained since they are merely:

$$P(\neg A|I) = 1 - P(A|I); P(\neg A|\neg I) = 1 - P(A|\neg I) \tag{6}$$

Base-Rate Fallacy

Applying Bayes' theorem to calculate $P(I|A)$ results in:

$$P(I|A) = \frac{P(I) \cdot P(A|I)}{P(I) \cdot P(A|I) + P(\neg I) \cdot P(A|\neg I)} \qquad (7)$$

Likewise for $P(\neg I|\neg A)$:

$$P(\neg I|\neg A) = \frac{P(\neg I) \cdot P(\neg A|\neg I)}{P(\neg I) \cdot P(\neg A|\neg I) + P(I) \cdot P(\neg A|I)} \qquad (8)$$

The assumptions above gives us a value for the rate of incidence of the actual number of intrusion in our system, and its dual (10 audit records per intrusion, two intrusions per day, and 1000,000 audit records per day). Interpreting these as probabilities:
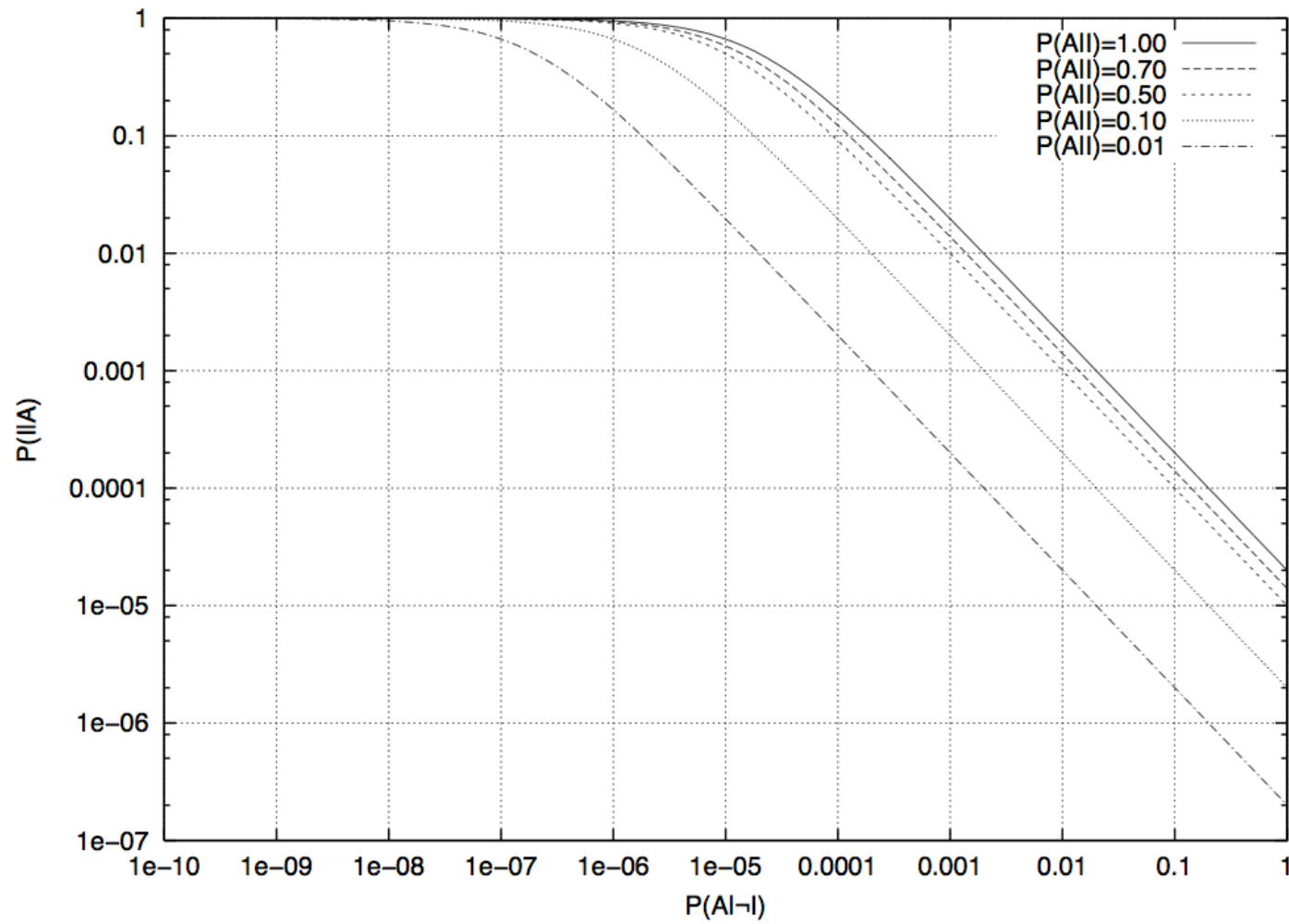
$$P(I) = 1 \bigg/ \frac{1 \cdot 10^6}{2 \cdot 10} = 2 \cdot 10^{-5}; P(\neg I) = 1 - P(I) = 0.99998 \qquad (9)$$

Inserting equation (9) into equation (7):

$$P(I|A) = \frac{2 \cdot 10^{-5} \cdot P(A|I)}{2 \cdot 10^{-5} \cdot P(A|I) + 0.99998 \cdot P(A|\neg I)} \qquad (10)$$

Base-Rate Fallacy

IDS Data

▸ In an influential 1987 paper, Dorothy Denning identified several fields that should be included in IDS event records:

  ▸ **Subject:** the initiator of an action on the target

  ▸ **Object**: the resource being targeted, such as a file, command, device, or network protocol

  ▸ **Action:** the operation being performed by the subject towards the object

  ▸ **Exception-condition:** any error message or exception condition that was raised by this action

  ▸ **Resource-usage:** quantitative items that were expended by the system performing or responding to this action

  ▸ **Time-stamp:** a unique identifier for the moment in time when this action was initiated

Types of Intrusion Detection Systems

▸ **Rule-Based Intrusion Detection**

   ▸ Rules identify the types of actions that match certain known profiles for an intrusion attack, in which case the rule would encode a **signature** for such an attack. Thus, if the IDS manager sees an event that matches the signature for such a rule, it would immediately sound an alarm, possibly even indicating the particular type of attack that is suspected.
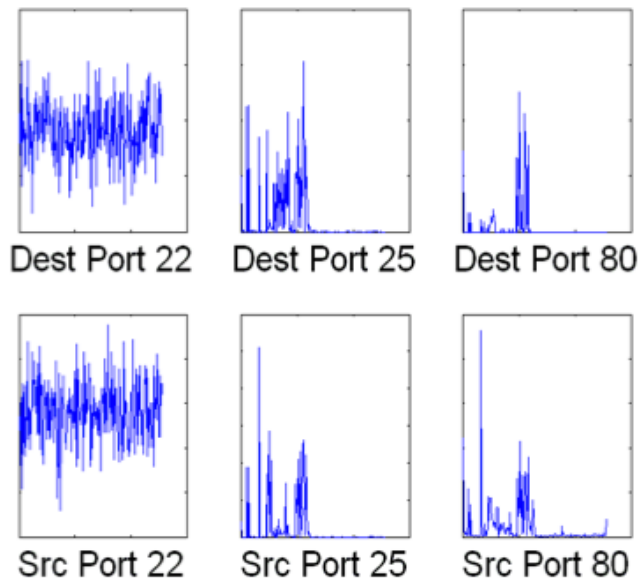
▸ **Statistical Intrusion Detection**

   ▸ A **profile** is built, which is a statistical representation of the typical ways that a user acts or a host is used; hence, it can be used to determine when a user or host is acting in highly unusual, anomalous ways.

   ▸ Once a user profile is in place, the IDS manager can determine thresholds for anomalous behaviors and then sound an alarm any time a user or host deviates significantly from the stored profile for that person or machine.

IDS taxonomy

- Host-based vs. Network-based
- Misuse-based vs. Anomaly-based vs. Specification-based
  - Examples:
    - A sense of self for Unix processes (Forrest et al.)
    - PAYL – payload anomaly detection (Want et al.)

Dest Port 22   Dest Port 25   Dest Port 80

Src Port 22   Src Port 25   Src Port 80

Payload byte distribution

IDS examples

▸ Snort

   ▸ http://snort.org

▸ Suricata

   ▸ http://suricata-ids.org

▸ Bro

   ▸ http://www.bro.org