Computer and Network Security

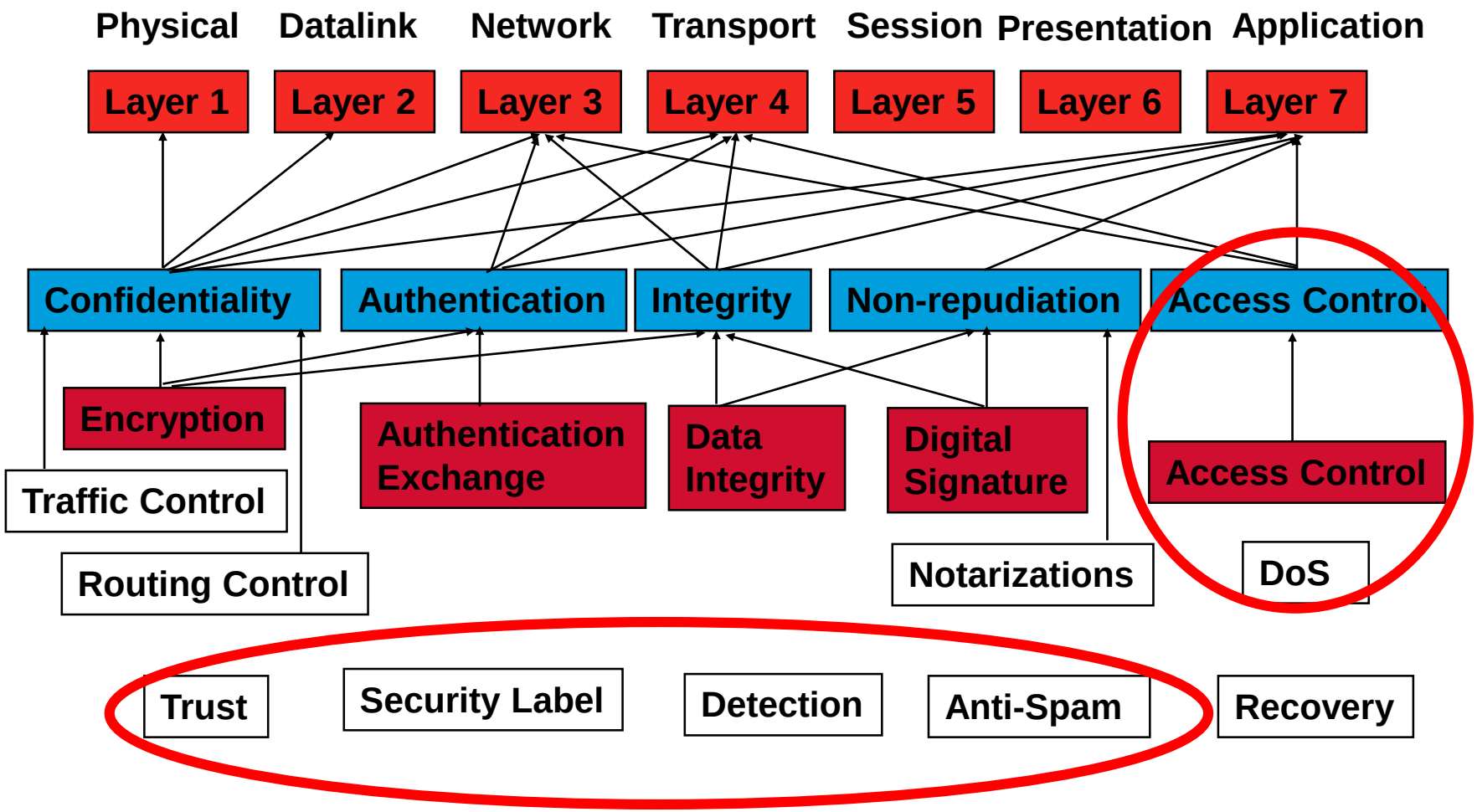
Dr. Chan Yeob Yeun

Week 13-14

Weekly Lecture Plan

Wk	Contents	Cmt	Wk	Contents	Cmt
1	Introduction		9	Foundations of Network Security II	
2	Foundations of Computer Security	Tutorial Assig Plan	10	Network-Based Threats and Attacks	
3	Identification and Authentication I		11	Network Security Protocols I	
4	Identification and Authentication II	Quiz 1	12	Network Security Protocols II	Quiz 3
5	Access Control		13	Firewalls	
6	Modern Computer Attacks		14	IDS / IPS	Assig Submit
7	Malicious Code	Assig Confirm	15	Revision and Presentation	
8	Foundations of Network Security I	Quiz 2	16	Exam	

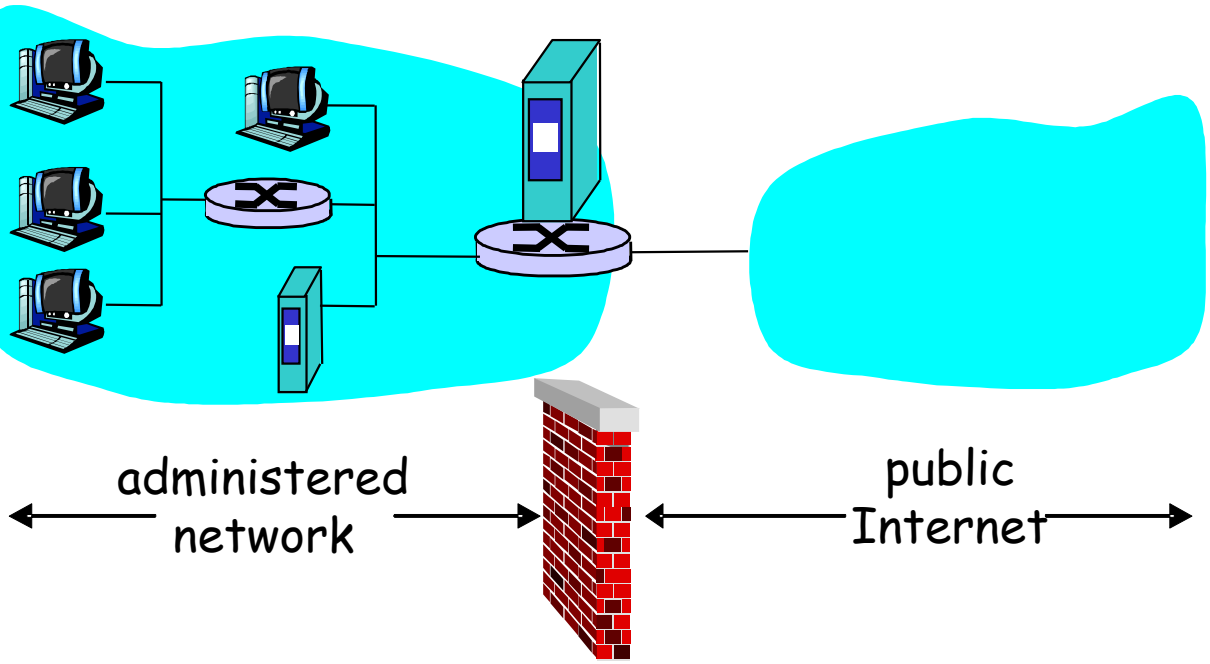
What is Network Security ?



Firewall

A firewall is a machine that monitors all traffic to and from a site

This allows for monitoring, filtering, logging, and proper access to the network



Firewalls

A firewall is a hardware or software solution to enforce security policies

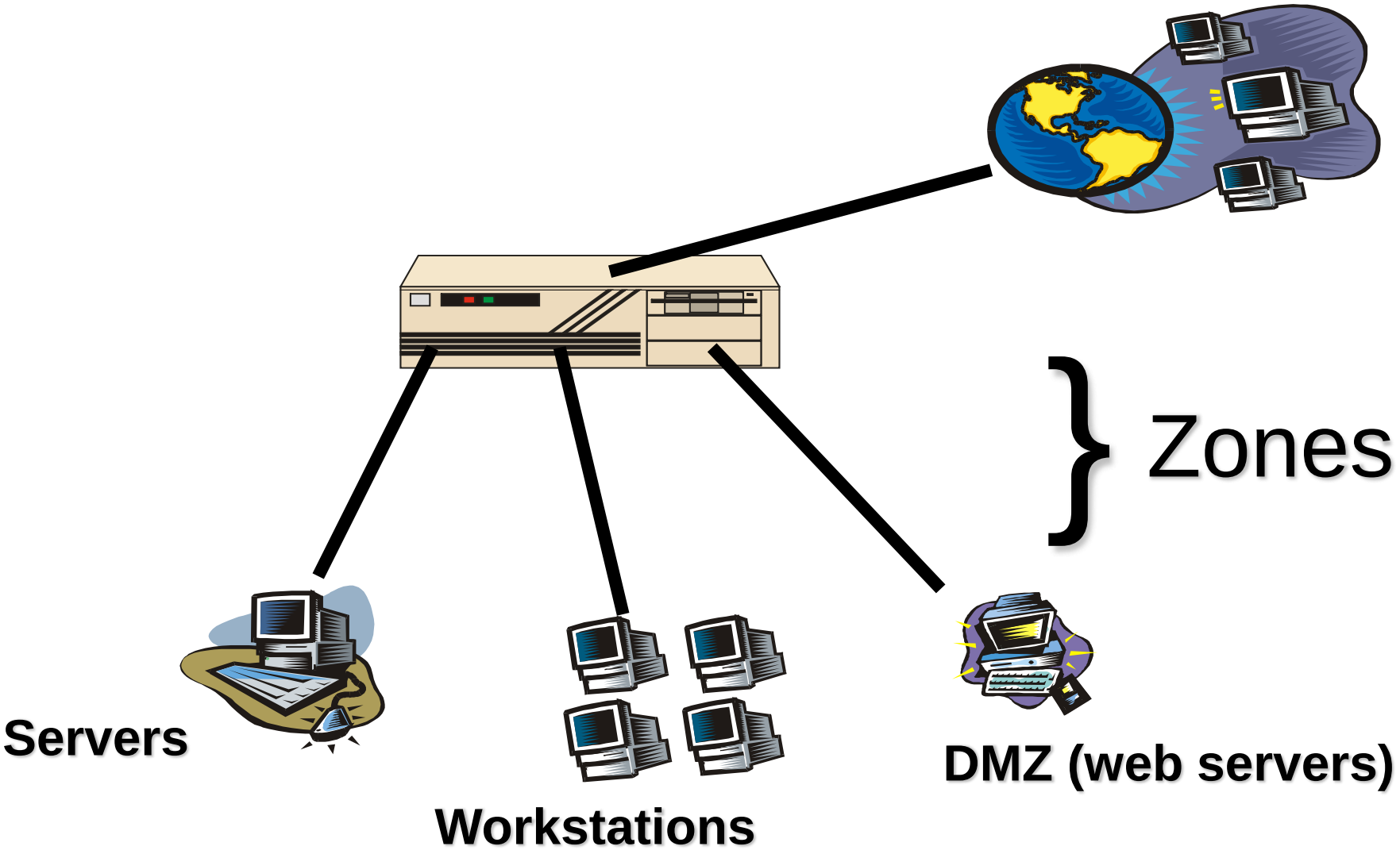
A firewall is a network access control device that is designed to deny all traffic except that which is explicitly allowed.

A router is a network device intended to direct traffic as fast as possible.

A router is not a firewall, but can be configured to do some firewall tasks (e.g deny certain traffic). Most routers nowadays have firewall functions

Firewalls can be configured to log all traffic.

Firewalls Zones:



Why Firewalls?

prevent denial of service attacks:

SYN flooding: attacker establishes many bogus TCP connections, no resources left for "real" connections.

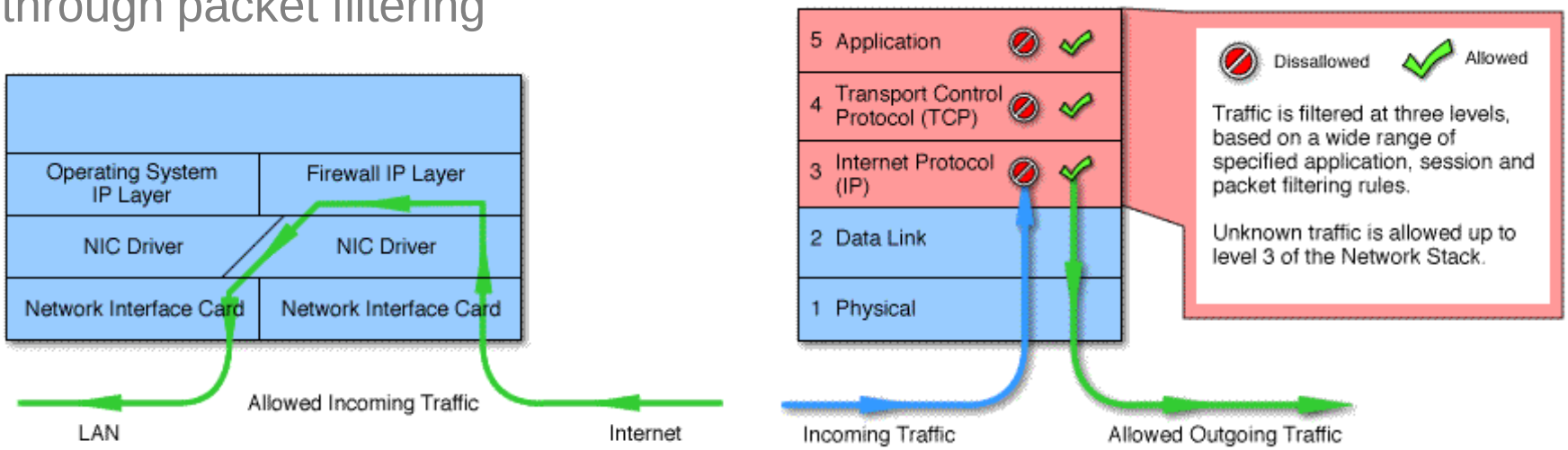
prevent illegal modification/access of internal data.

e.g., attacker replaces CIA's homepage with something else

allow only authorized access to inside network (set of authenticated users/hosts)

How do Firewalls work?

Most firewalls function through packet filtering

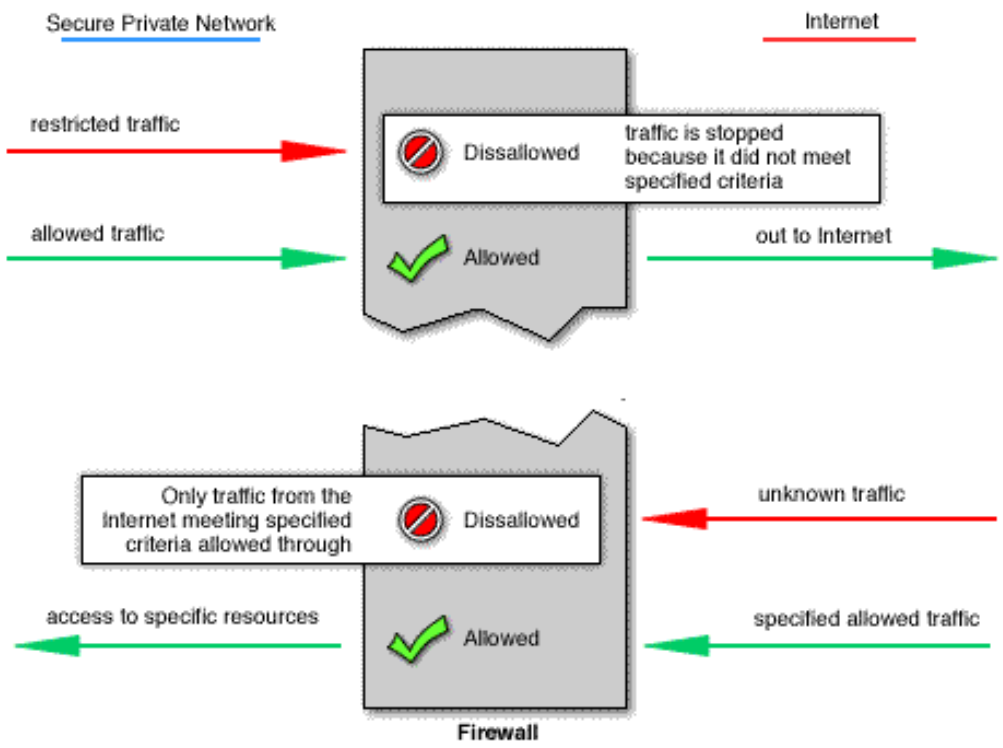


Courtesy http://www.vicomsoft.com/knowledge/reference/firewalls1.html

Filtering based on Port

Filtering based on port occurs by examining the Transport layer

Deny-all Reject all packets except to required services



Courtesy http://www.vicomsoft.com/knowledge/reference/firewalls1.html

Filtering Based on Address

The incoming and outgoing address can be examined to see if the computer is allowed access to the network

However, this can be circumvented by an attacker who fakes the incoming address, making it look as if they are allowed

Firewall Strengths and Weaknesses

Firewall Strengths

- Good to enforce corporation security policies
- Restrict access to specific services such as telnet, ftp and etc.
- Singular in purpose so compromises do not need to be made between security and usability
- Excellent auditors by monitoring all traffic that pass through
- Good to alert appropriate people of specified events

Firewall Weaknesses

- Fail to protect against what has been authorized
- Only as effective as the rules they are configured to enforce
- Cannot stop social engineering attacks
- Cannot fix poor administration practices or poor design security policies
- Cannot stop attacks if the traffic does not pass through them

Firewall Policy Criteria

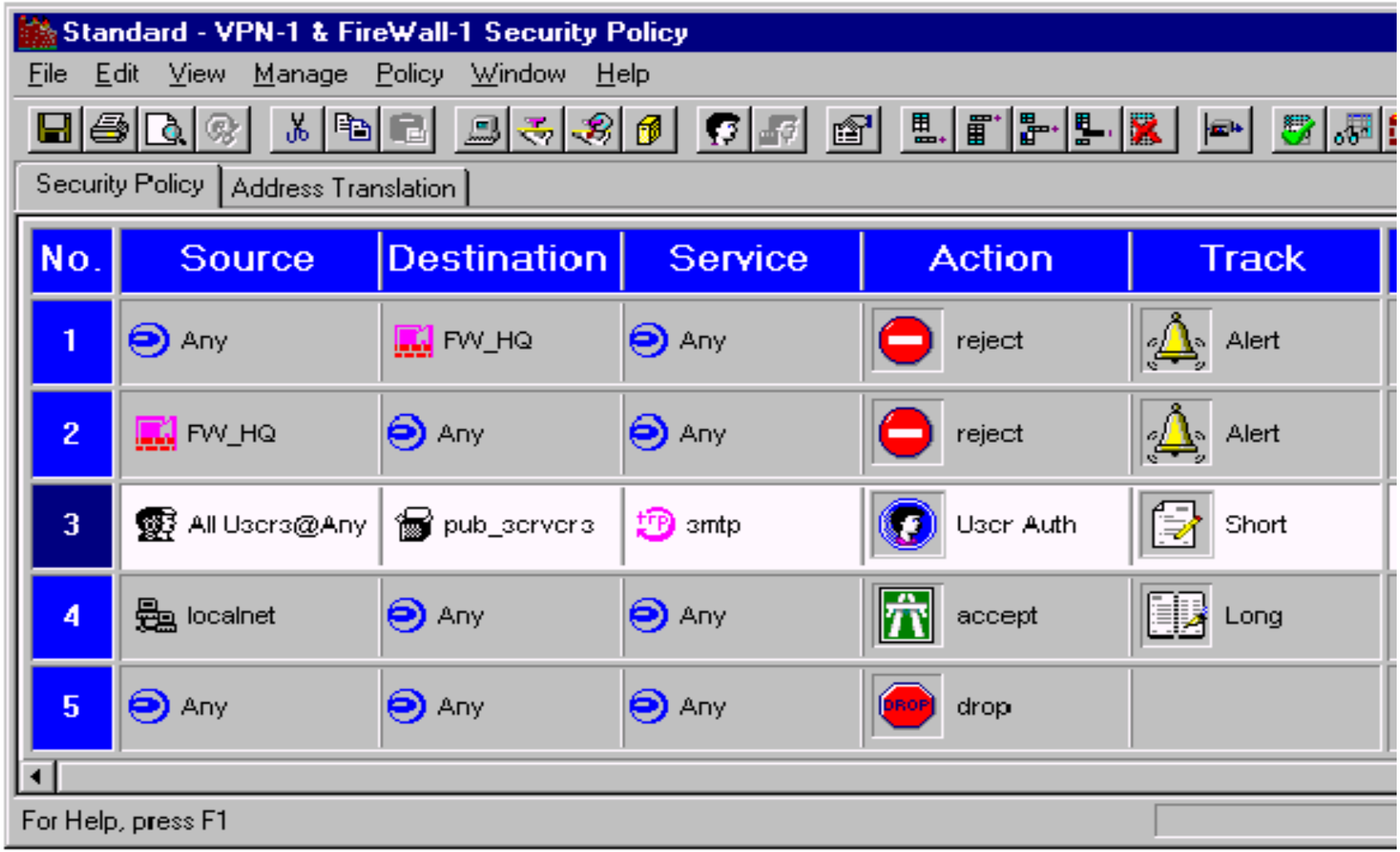
Firewalls can allow/block traffic based on:

- Physical location of packet (inside/outside perimeter)
- Source address
- Destination address
- Type of TCP/IP application (service, port)
- Relationship to other packets (stateful inspection)
- Application function content (e.g., GET, PUT)
- Data payload content (e.g., Java, ActiveX)

Firewall Actions:

- Accept: allow packet to pass through
- Drop: deny access by dropping packet silently - no response
- Reject: deny access - respond with ICMP reject (destination /port unreachable)
- Authenticate: check if connection request is from a valid source

Sample Firewall Policy



Types of Firewalls

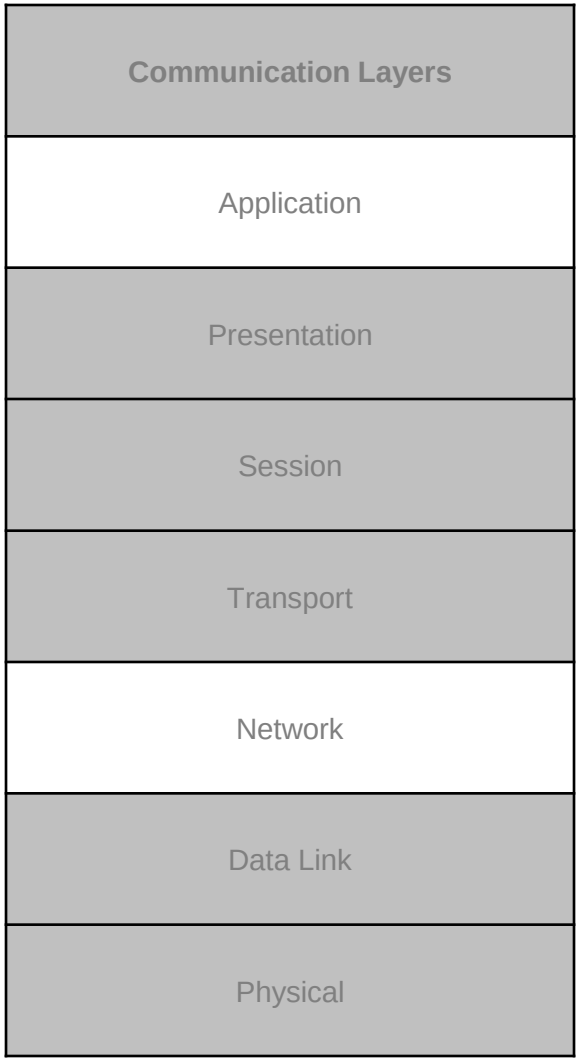
Packet Filtering Firewall

Operates at the network/transport layers and uses packet inspection filters to determine if traffic is allowed or not according to:

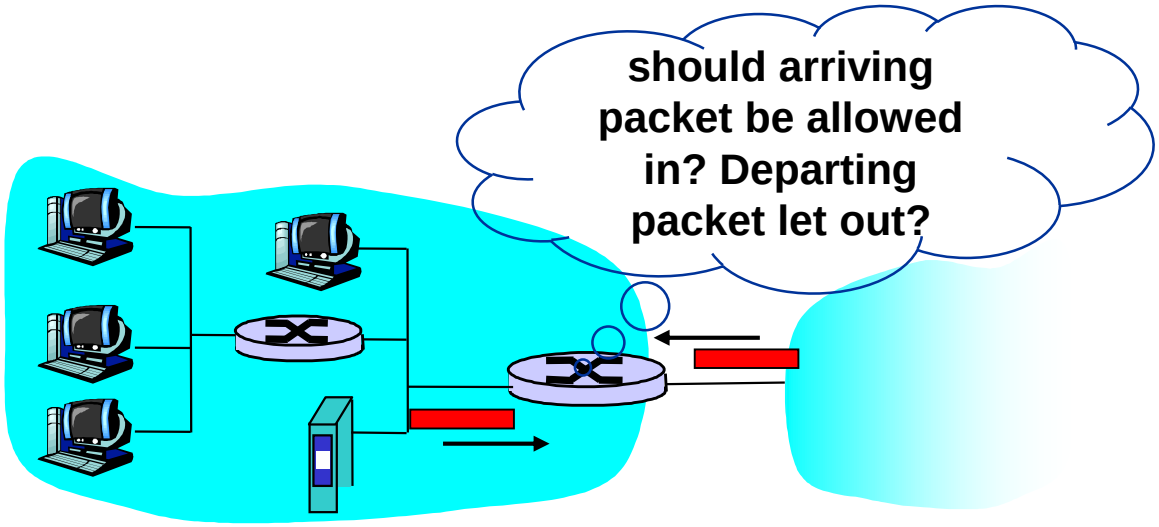
- Policy rules
- Protocol state (stateful inspection)

Application Layer Firewall (Proxy Firewall)

Operates at the application layer of a protocol stack. Generally it is a host using various forms of proxy servers to proxy traffic instead of routing it.



Network Layer – Packet Filtering Firewall



Internal network connected to Internet via router firewall

Router filters packet-by-packet, decision to forward/drop packet based on:

- Source IP address, destination IP address
- TCP/UDP source and destination port numbers
- ICMP message type
- TCP SYN and ACK bits

Network layer - Packet Filtering Firewall

Example 1: block incoming and outgoing datagrams with **IP protocol field = 17** and with either **source or dest port = 23**.

All incoming and outgoing **UDP flows** and **telnet connections** are blocked.

Example 2: Block inbound TCP segments with **ACK=0.**

Prevents external clients from making **TCP connections with internal clients**, but allows internal clients to connect to outside.

Packet Filtering Firewalls

Can be used for filtering any protocol that runs on IP.

They can’t stop attacks going through valid connections.

Generally capable of handling more traffic compared to application layer firewalls.

Internal addressing can be seen from outside since connections don’t terminate at the firewall.

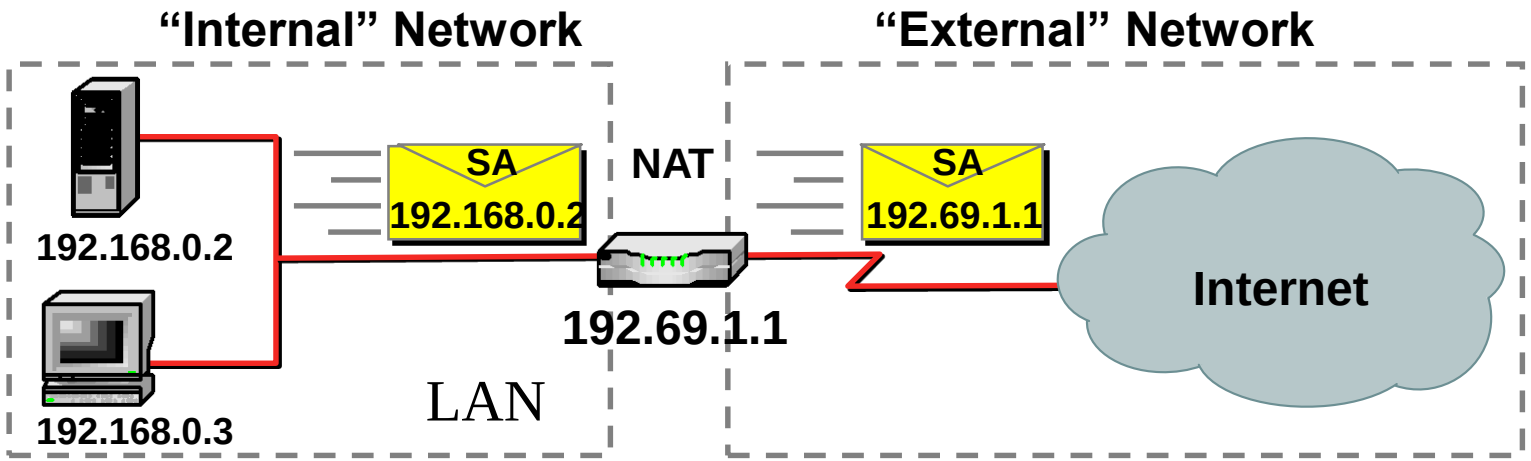
However; most packet filtering firewalls do support Network Address Translation (NAT).

Network Address Translation (NAT)

NAT is a technique in which the source and/or destination addresses of IP packets are rewritten as they pass through a NAT gateway.

It is most commonly used to hide internal IP addresses by enabling multiple hosts on a private network to access the Internet using a single public IP address.

Sometimes referred to as “IP Masquerading”, below is a NAT example:



NAT Operation

A client on the internal network contacts a machine on the Internet. It sends out IP packets (with necessary details) destined for that machine.

NAT is concerned with these pieces of information:

- Source IP address (e.g. 192.168.0.2)
- Source TCP or UDP port (e.g. 2132)

Packets passing through the NAT gateway will be modified so that they appear to be originating from the NAT gateway itself.

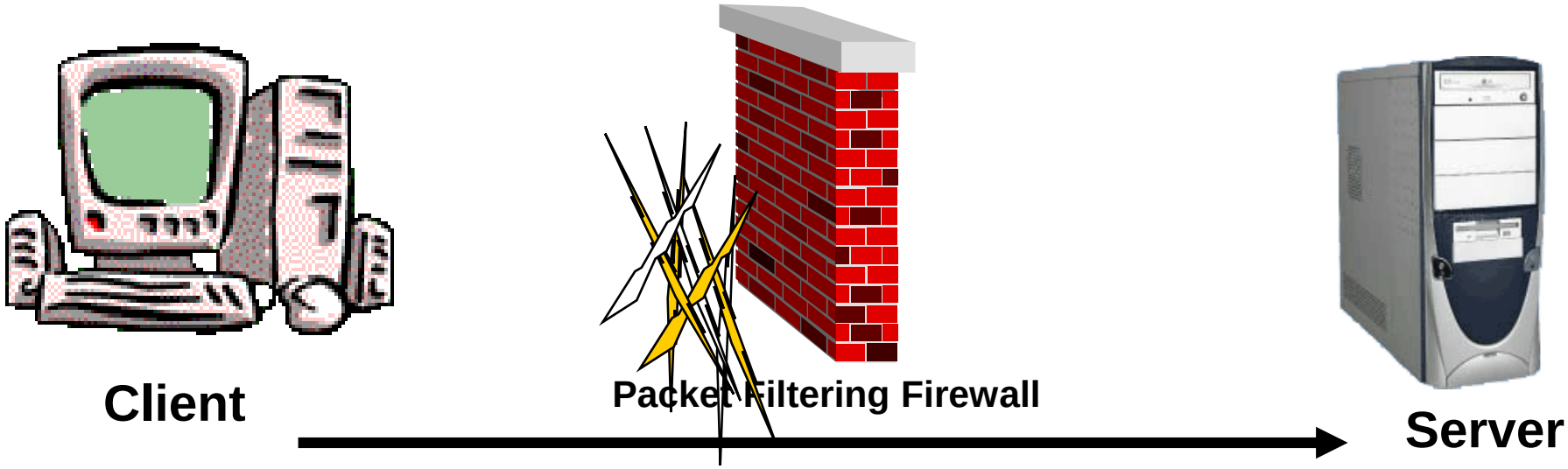
For example, the following changes might be made:

- Source IP: replaced with the external IP of NAT gateway (e.g. 192.69.1.1)
- Source port: replaced with a randomly chosen, unused port on the NAT gateway (e.g 53136)

The NAT gateway records the changes made in its state table in order to reverse them on returning packets

Packet Filtering Firewalls

2. Firewall analyses the packet and the state of the connection against the policy rules.
If the packet is allowed, it is passed directly to the server.



1. Client sends a connection request to the server

Stateful Inspection

Connection state (used as a context for decisions)

- Watch traffic for SYN, ACK, FIN and RST packets
- Check connection state (established, initiating)

More complex than stateless inspection, but gives better security

Example 1: TCP Connection Setup Sequence

- First packet expected is a SYN packet.
- Firewall sees the packet and put connection in the SYN state.
- In this state one of two packets is expected in response, either SYN ACK or RST.
- Anything else the firewall will drop or reject, as it is incorrect to this state

Example 2:

- Deny all ICMP Echo Reply packets not associated with an Echo Request

Application Layer Firewalls

Use proxies for most commonly used protocols (HTTP proxy, SMTP proxy, FTP Proxy)

Since each proxy is designed to inspect its own protocol traffic, it is difficult to pass unwanted traffic (e.g. an HTTP proxy will block a malicious command coming on an HTTP connection)

Can perform content filtering (e.g. websites, viruses, S/W flaws).

Good in hiding internal addresses as connections terminate at the firewall.

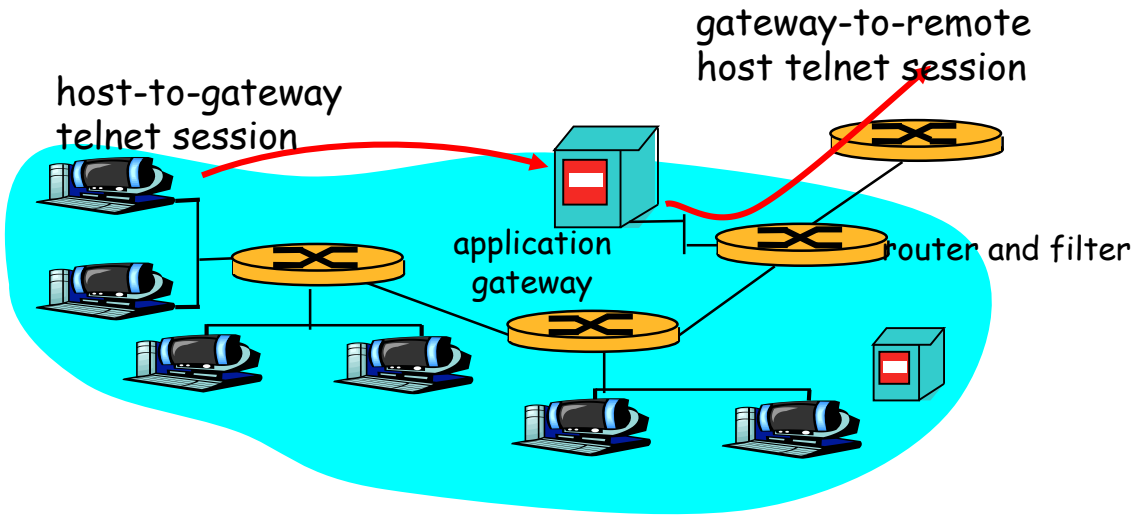
Relatively slower than packet filtering firewalls.

Application layer - Application gateways

Example

allow select internal users to telnet outside.

Users authenticate themselves to create telnet connection



Solution

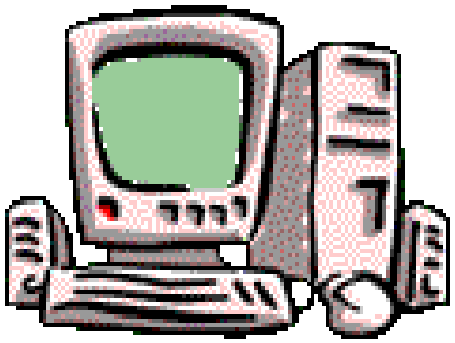
Router filter blocks all telnet connections not originating from gateway.

For authorized users, gateway sets up telnet connection to dest host. Gateway relays data between two connections

Application Layer Firewalls

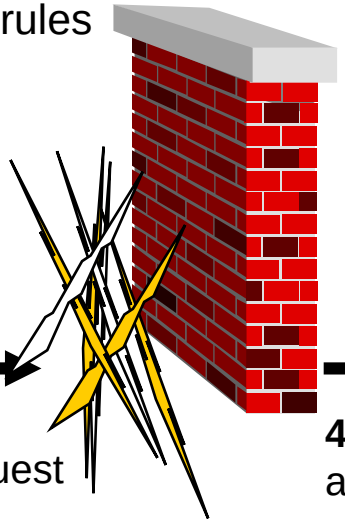
2. Firewall decodes the packet and analyses the protocol according to policy rules

3. If traffic is not allowed, connection is denied. If traffic is allowed, firewall initiates a new connection to server on behalf of the client



Client

1. Client sends a connection request to proxy firewall



4. Firewall initiates a new connection to server

Server

Application Layer Firewall (proxy firewall)

Hybrid Firewalls

Nowadays, difficult to find a pure firewall that performs only packet filtering or application layer filtering.

Hybrid firewalls combine both functionalities, e.g. most packet filtering firewalls come with an SMTP proxy.

Hybrid firewalls allow administrators to tailor the solution to their particular needs.

Case Study: Developing a Firewall Configuration

An organisation that has the following systems:

Web server: that offers HTTP service on port 80 only.

Mail server: that offers SMTP service on port 25 only.

Internal DNS server: that queries Internet DNS systems to resolve domain names.

The Internet policy allows outbound access to the following services:

- HTTP
- HTTPs
- FTP
- Telnet
- SSH

Based on this architecture we can construct firewall policy rules for the various possible Architectures.

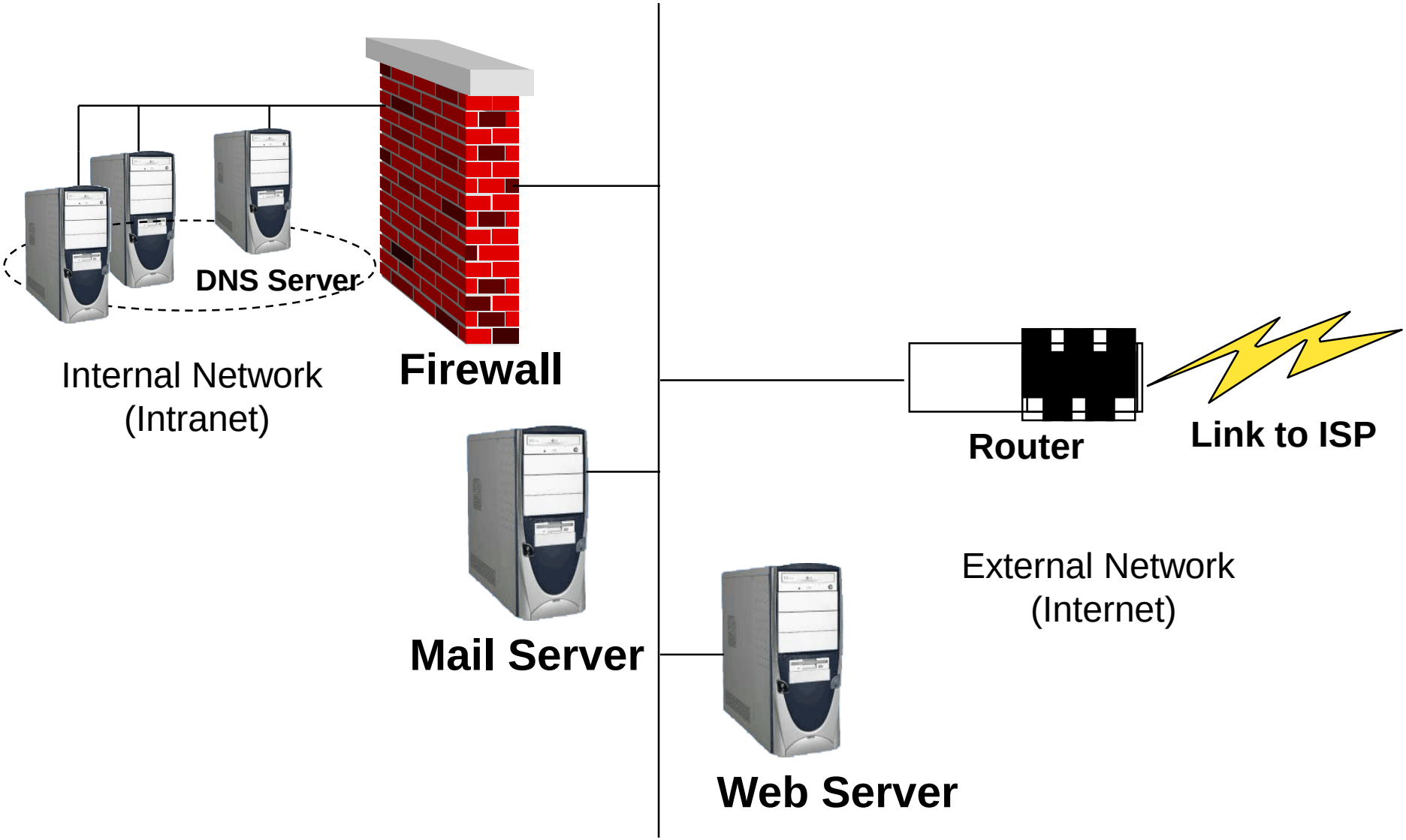
Limitations of Firewalls and Gateways

IP spoofing

- Router can't know if data "really" comes from claimed source
- If multiple app's. need special treatment, each has own app. gateway.
- Client software must know how to contact gateway.
  e.g., must set IP address of proxy in Web browser

- Tradeoff
  - degree of communication with outside world, level of security
  - Performance problem

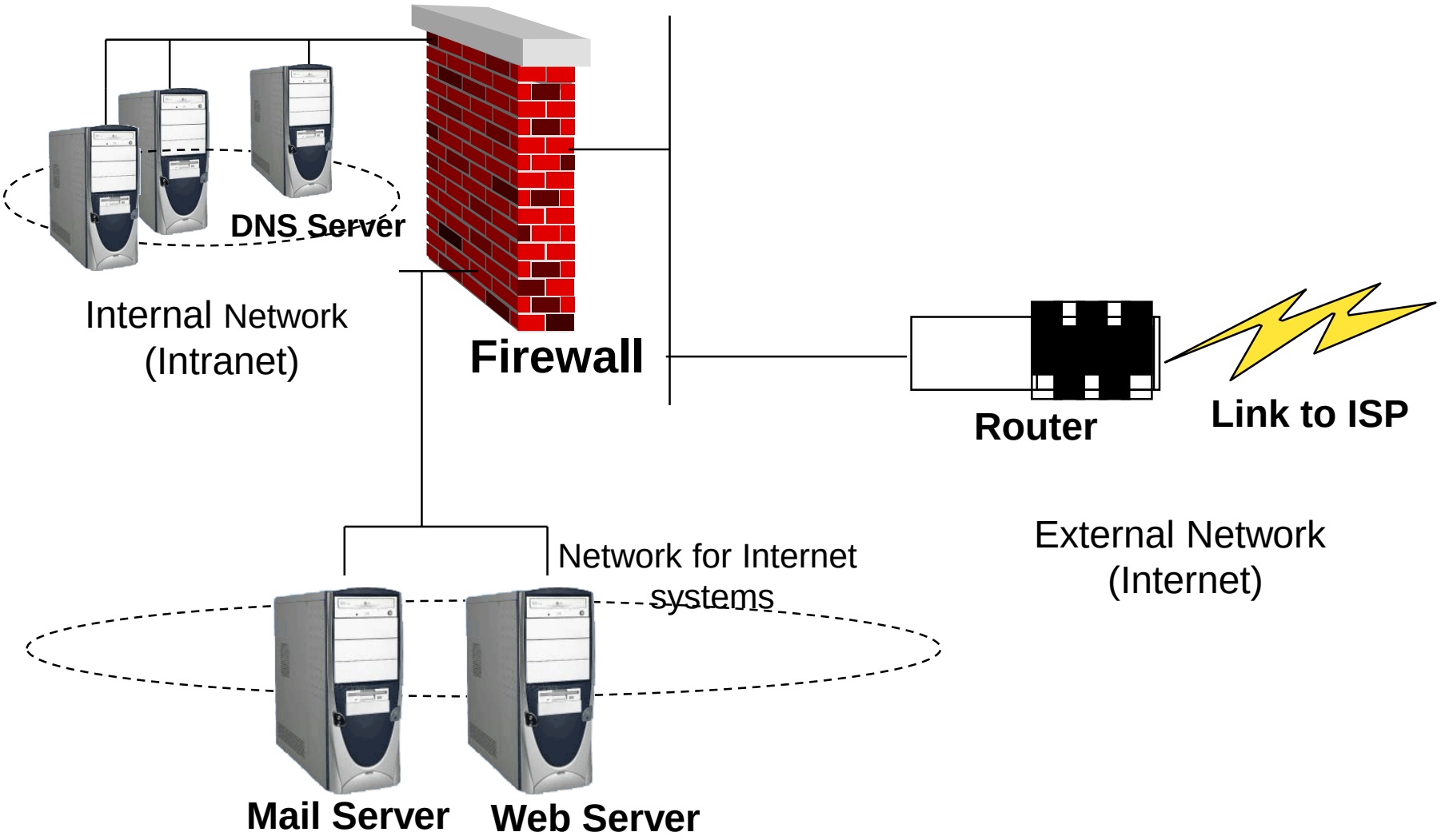
Architecture 1: Internet Accessible Systems Outside the Firewall



Architecture 1: Internet Accessible Systems Outside the Firewall

Rule No.	Source IP	Destination IP	Service	Action
1	**Internal network**	**Mail server**	**SMTP**	**Accept**
2	**Internal network**	**Any**	**HTTP, HTTPS, FTP, Telnet, SSH**	**Accept**
3	**Intenal DNS**	**Any**	**DNS**	**Accept**
4	**Any**	**Any**	**Any**	**Drop**

Architecture 2: Single Firewall **After Authentication to avoid email relaying from outsiders**

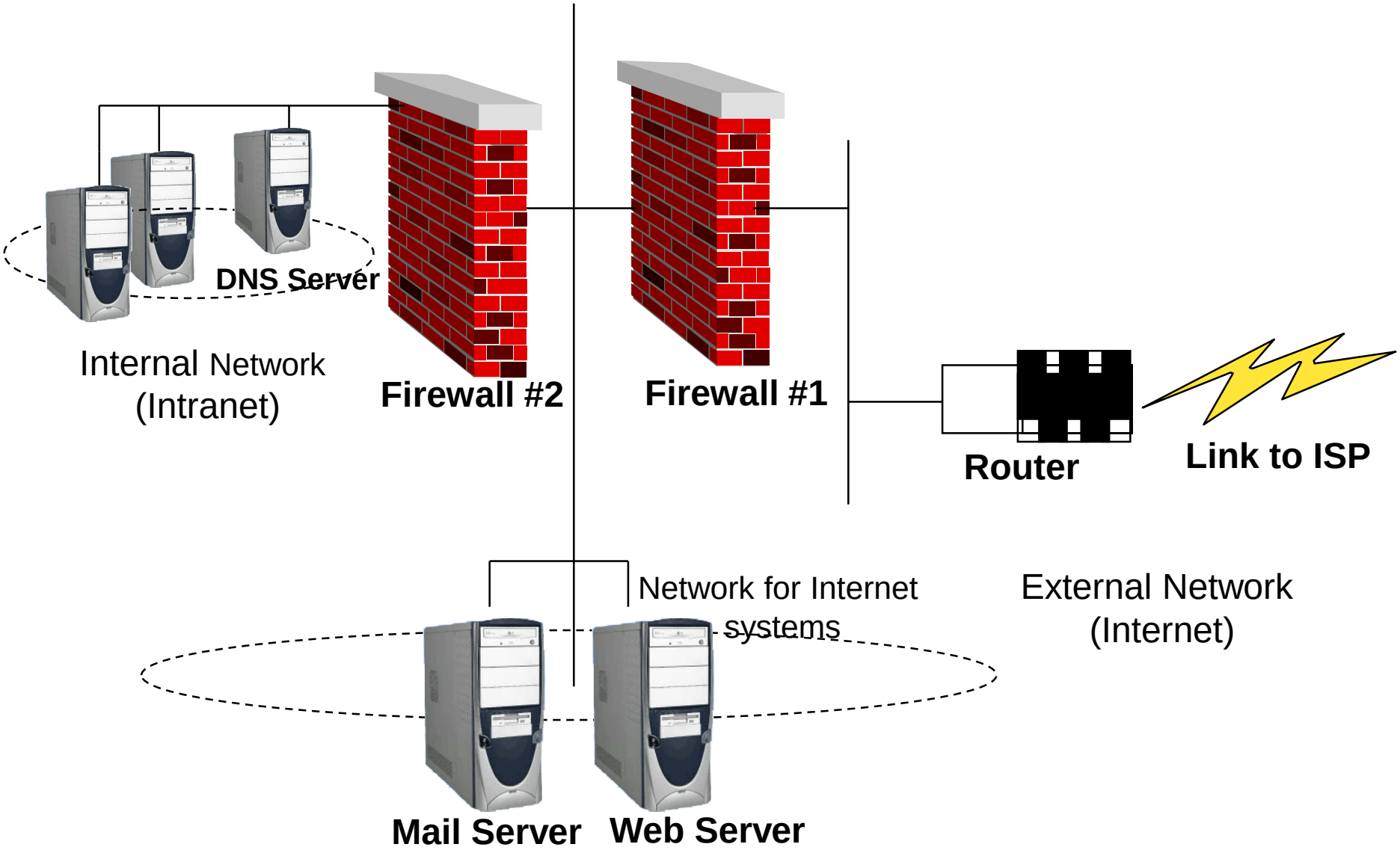


Architecture 2: Single Firewall

Rule No.	Source IP	Destination IP	Service	Action
1	Any	Web Server	HTTP	Accept
2	Any	Mail server	SMTP	Accept*
3	Mail server	Any	SMTP	Accept
4	Internal Network	Any	HTTP, HTTPS, FTP, Telnet, SSH	Accept
5	Internal DNS	Any	DNS	Accept
6	Any	Any	Any	Drop

*** After Authentication to avoid email relaying from outsiders**

Architecture 3: Dual Firewall



Architecture 3: Dual Firewall – Rules for Firewall #1

Similar to rules of a single firewall

Rule No.	Source IP	Destination IP	Service	Action
1	Any	Web Server	HTTP	Accept
2	Any	Mail server	SMTP	Accept*
3	Mail server	Any	SMTP	Accept
4	Internal Network	Any	HTTP, HTTPS, FTP, Telnet, SSH	Accept
5	Internal DNS	Any	DNS	Accept
6	Any	Any	Any	Drop

*** After Authentication to avoid email relaying from outsiders**

Architecture 3: Dual Firewall – Rules for Firewall #2

Rule No.	Source IP	Destination IP	Service	Action
1	**Internal network**	**Mail server**	**SMTP**	**Accept**
2	**Internal Network**	**Any**	**HTTP, HTTPS, FTP, Telnet, SSH**	**Accept**
3	**Internal DNS**	**Any**	**DNS**	**Accept**
4	**Any**	**Any**	**Any**	**Drop**

Firewall Rules – Design Considerations

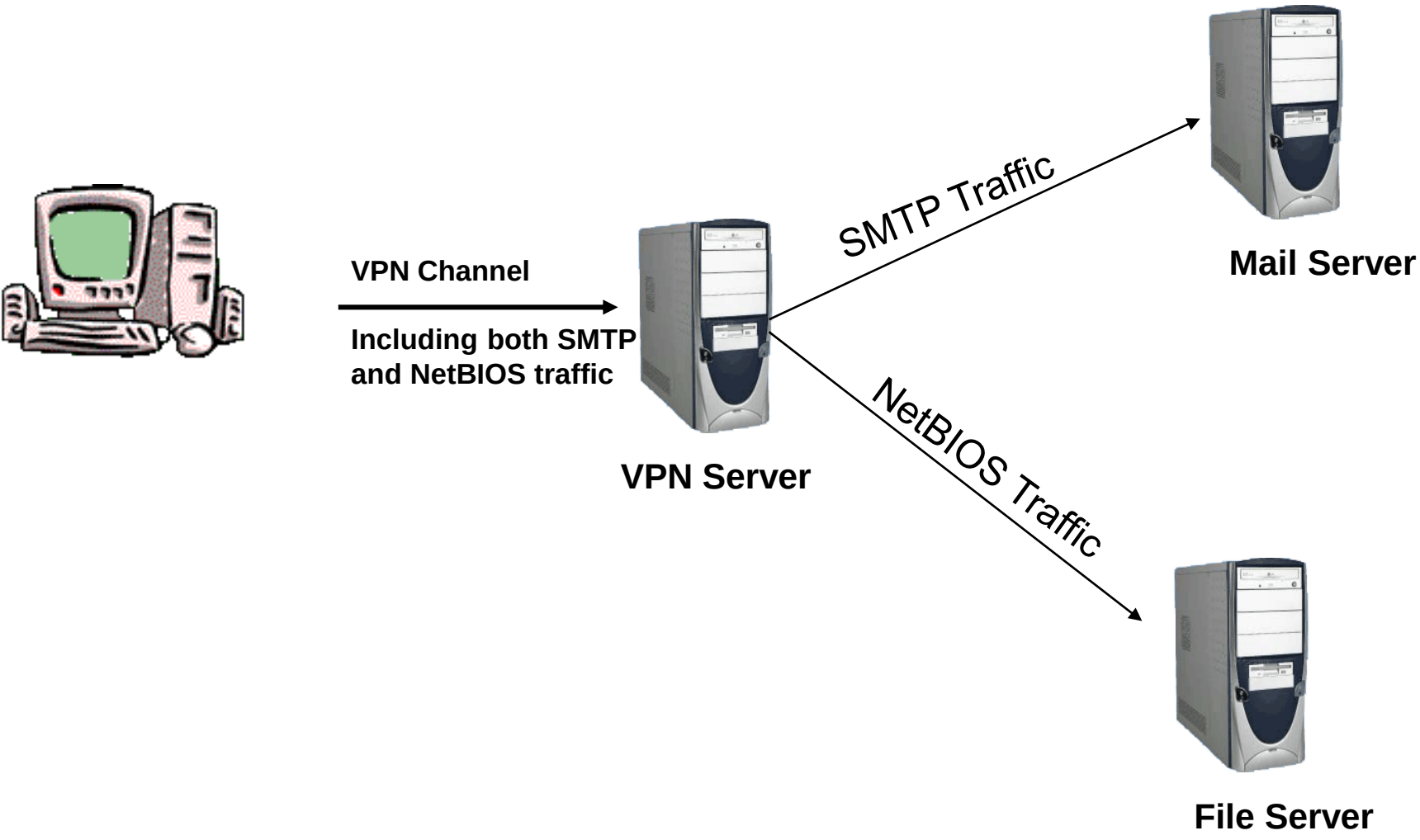
Most firewalls work on “first match” basis, i.e searching rules from top to bottom.

If a rule is matched then ignore the remaining rules

To make firewall more efficient, specific rules should be placed on top, where more generic rules should be at the bottom

For example: HTTP generally has the largest traffic. So HTTP rules should be placed on top of the rule set.

VPN Handles Multiple Protocols



What is IPSec?

IP Security is a set of protocols and standards to support the securing of data at the IP layer. IPSec is a framework, not an implementation.

Supports authentication and encryption of traffic.

- Certifies the originator of the packet.
- Protects the data from interception and tampering while in transit.

Why do we want to use IPSec?

- Secure our network
- Transparent operation
  - IPSec allows us to secure any IP based protocol transparent to the application.
  - Support for legacy software which is inherently insecure (telnet,ftp,SMB).
  - An alternative mechanism to implementing application level security such as using SSL.
- Widest industry support e.g. Cisco, Microsoft, Network Associates, CheckPoint Software, Bay Networks, etc.
- IETF standard – Will be mandatory in IPv6

IP Security (IPSec)

IPSec - A set of standards for virtual private networking

Provides secure data transmission over IP networks

Quite complex, as it includes the following IETF standards:

- RFC 2401: “Security Architecture for the Internet Protocol”
- RFC 2402: “IP Authentication Header (AH)”
- RFC 2406: “IP Encapsulating Security Payload (ESP)”
- RFC 2409: “The Internet Key Exchange (IKE)”

IPSec operates at the network layer

Originally intended for network-to-network VPNs but is now being ported to user-to-network VPNs

IPSec Security Protocols

Authentication Header (AH) provides:

- Session authentication
- Integrity
- Protection against replay attacks

Encapsulating Security Payload (ESP) provides:

- Same security features as in AH
- Confidentiality (encryption)
- Both AH and ESP protocols may be used singly or in combination

IPSec Modes

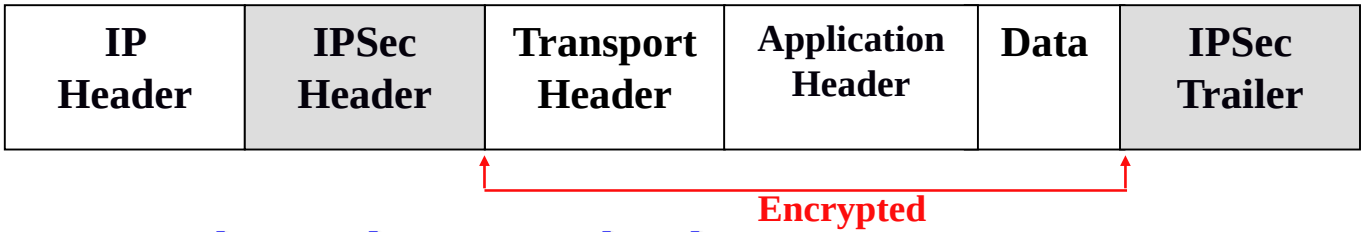
Transport – Secures the payload part of the IP packet, leaves the IP header unsecured. Commonly use for securing traffic on a LAN.

Tunnel – Secures the entire IP packet and encapsulates it within a new IP packet. Commonly used for creating a VPN.

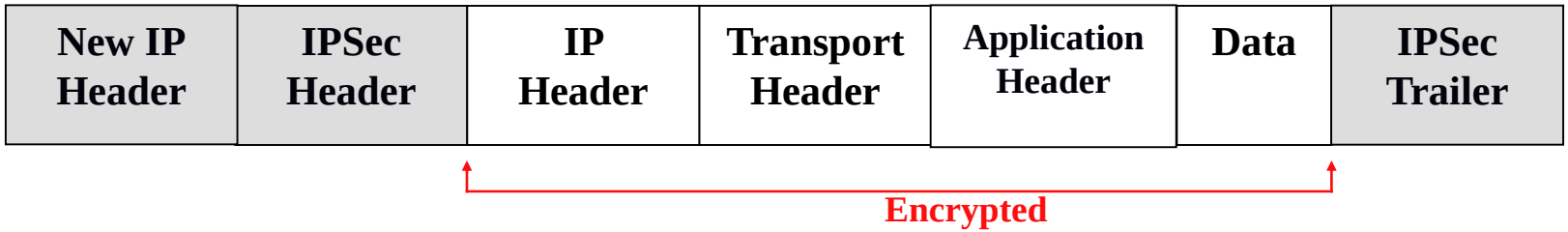
IPSec Modes of Operation

Original Packet →

Transport Mode: protects upper layer protocols only



- **Tunnel Mode: protects the entire IP payload**



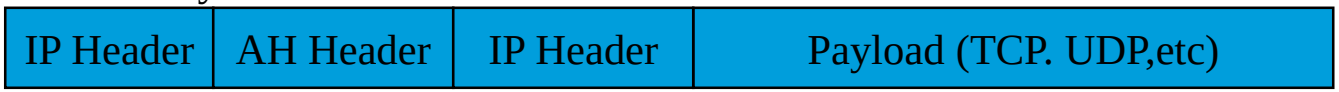
IPSec protocols – AH protocol

AH - Authentication Header

- Defined in RFC 1826
- Integrity: Yes, including IP header
- Authentication: Yes
- Non-repudiation: Depends on cryptography algorithm.
- Encryption: No
- Replay Protection: Yes

Transport Packet layout

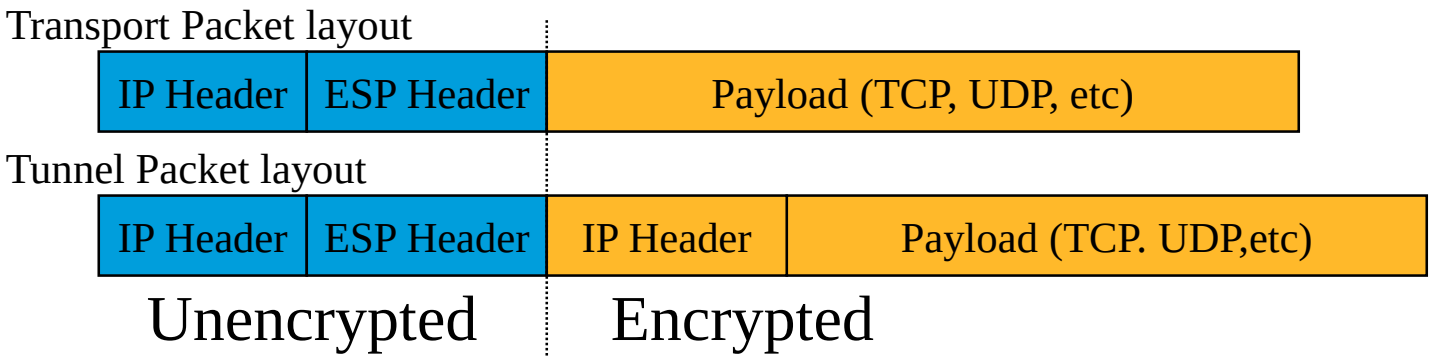
Tunnel Packet layout



IPSec protocols – ESP protocol

ESP – Encapsulating Security Payload

- Defined in RFC 1827
- Integrity: Yes
- Authentication: Depends on cryptography algorithm.
- Non-repudiation: No
- Encryption: Yes
- Replay Protection: Yes



What protocol to use?

Differences between AH and ESP:

- –ESP provides encryption, AH does not.
- –AH provides integrity of the IP header, ESP does not.
- –AH can provide non-repudiation. ESP does not.

However, we don’t have to choose since both protocols can be used in together.

Why have two protocols?

Some countries have strict laws on encryption. If you can’t use encryption in those countries, AH still provides good security mechanisms. Two protocols ensures wide acceptance of IPSec on the Internet.

Security Associations

One of the most important concepts in IPSec is called a Security Association (SA). Defined in RFC 1825.

SAs are the combination of a given Security Parameter Index (SPI) and Destination Address.

SAs are one way. A minimum of two SAs are required for a single IPSec connection.

SAs contain parameters including:

- –Authentication algorithm and algorithm mode
- –Encryption algorithm and algorithm mode
- –Key(s) used with the authentication/encryption algorithm(s)
- –Lifetime of the key
- –Lifetime of the SA
- –Source Address(es) of the SA
- –Sensitivity level (ie Secret or Unclassified)

How IPSec works: Phase 1

Internet Key Exchange (IKE) is used to setup IPSec.

IKE Phase 1:

- Establishes a secure, authenticated channel between the two computers
- Authenticates and protects the identities of the peers
- Negotiates what SA policy to use
- Performs an authenticated shared secret keys exchange
- Sets up a secure tunnel for phase 2
- Two modes: Main mode or Aggressive mode

Main Mode IKE

- Negotiate algorithms & hashes.
- Generate shared secret keys using a Diffie-Hillman exchange.
- Verification of Identities.

Aggressive Mode IKE

- Squeezes all negotiation, key exchange, etc. into less packets.
- Advantage: Less network traffic & faster than main mode.
- Disadvantage: Information exchanged before a secure channel is created.
  Vulnerable to sniffing.

How IPSec works: Phase 2

An AH or ESP packet is then sent using the agreed upon “main” SA during the IKE phase 1.

IKE Phase 2

- Negotiates IPSec SA parameters
- Establishes IPSec security associations for specific connections (like FTP, telnet, etc)
- Renegotiates IPSec SAs periodically
- Optionally performs an additional Diffie-Hellman exchange

How IPSec works: Communication

Once Phase 2 has established an SA for a particular connection, all traffic on that connection is communicated using the SA.

IKE Phase 1 exchange uses UDP Port 500.

AH uses IP protocol 51.

ESP uses IP protocol 50.

IPSEC in details

To establish a secure IPSEC connection two nodes must execute a key agreement protocol.

The sub-protocol of IPSEC that handles key negotiations is called IKE (Internet Key Exchange).

First, assume two nodes have agreed on keys (via IKE) and see how they proceed to protect their communication via IPSEC.

Security Associations in details

An IPsec protected connection is called a security association.

- IPsec is a level-3 protocol (runs on top of IP), and below TCP/UDP
- Security associations may either be end-to-end or link-to-link.

Two modes of encapsulating IPsec data into an IP packet define two modes of operation:

- Transport mode and tunnel mode.

Security Parameter Index (SPI)

The IPsec protocol maintains two databases:

Security association database: Indexed by SPI's, contains the information needed to encapsulate packets for one association: cryptographic algorithms, keys, sequence numbers, etc.

Security policy database: Allows for implementation of packet filtering policies. Defines whether or not to accept non-protected packets, what to require, etc.

Internet Key Exchange (IKE)

Phase I

- Establish a secure channel(ISAKMP SA)
- Authenticate computer identity

Phase II

- Establishes a secure channel between computers intended for the transmission of data (IPSec SA)

IKE Phases

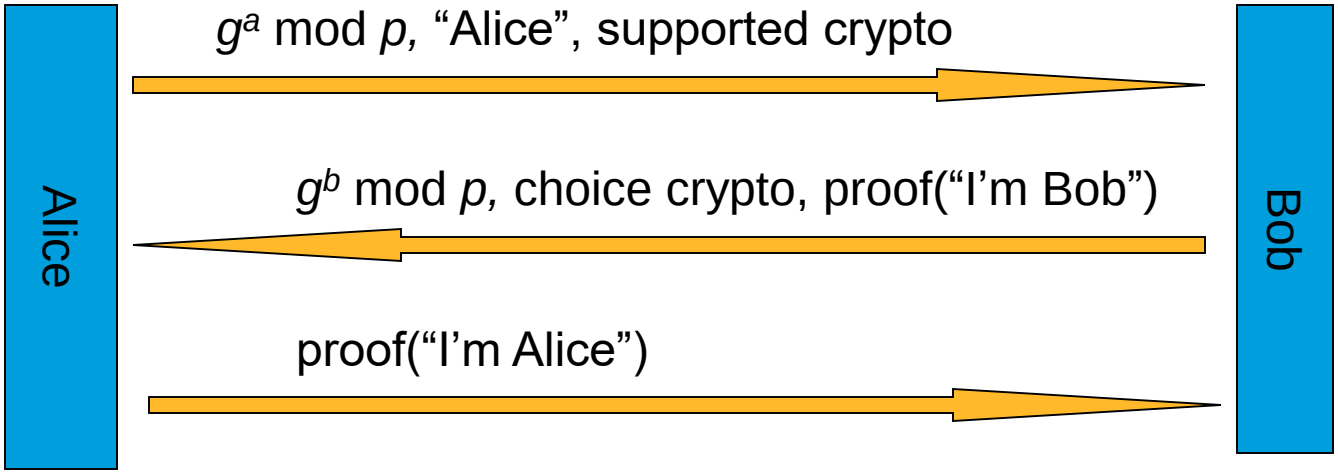
In a design similar to Kerberos, IKE performs a phase 1 mutual authentication based on public keys and phase 2 re-authentication based on shared secrets (from phase 1).

This allows multiple SAs to re-use the same handshake.

Phase 1 has two modes:

- Aggressive mode (3 messages)
- Main mode (6 messages)

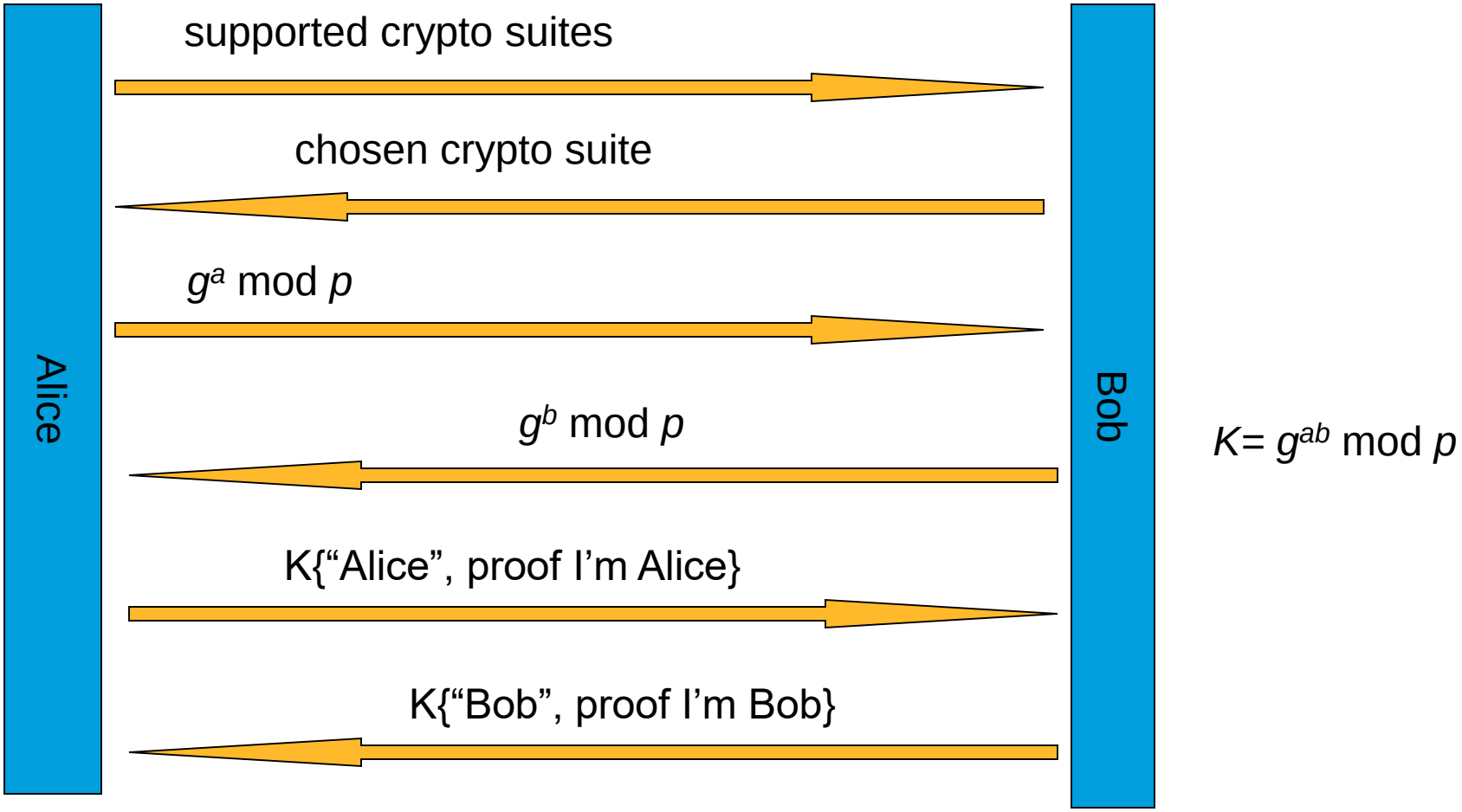
IKE Phase 1: Aggress. Mode



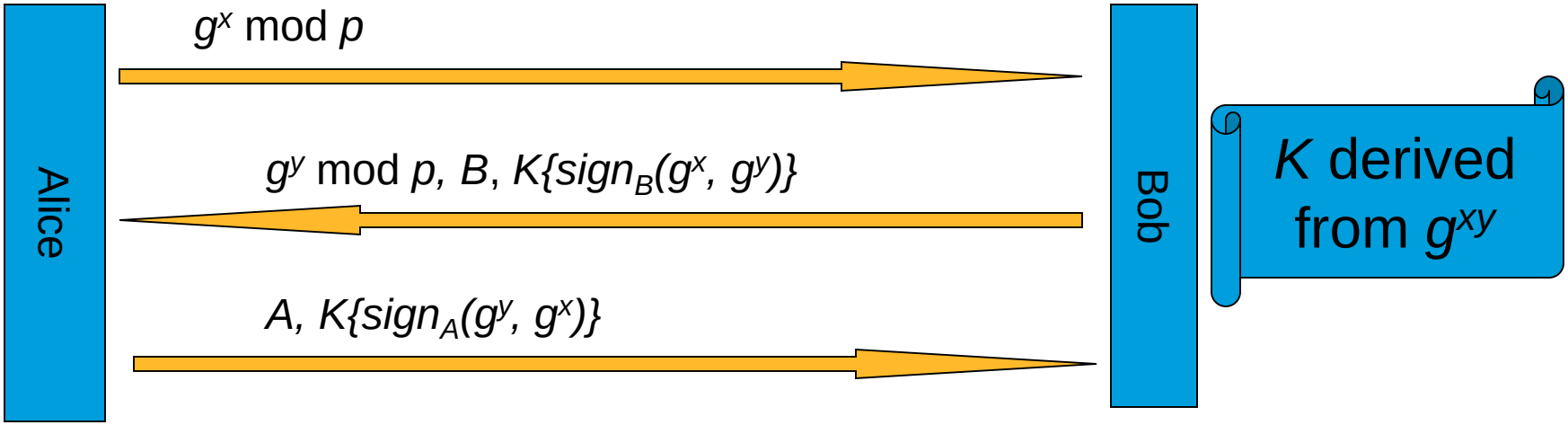
In aggressive mode, Alice chooses some Elgamal context *(p, g).* Bob may not support it, and reject the connection. If that happens, Alice should try and connect to Bob using main mode.

Aggressive mode provides mutual authentication, and a shared secret g^{ab} mod p, which can be used to derive keys for the symmetric crypto protocols.

IKE Phase 1: Main Mode



STS Protocol

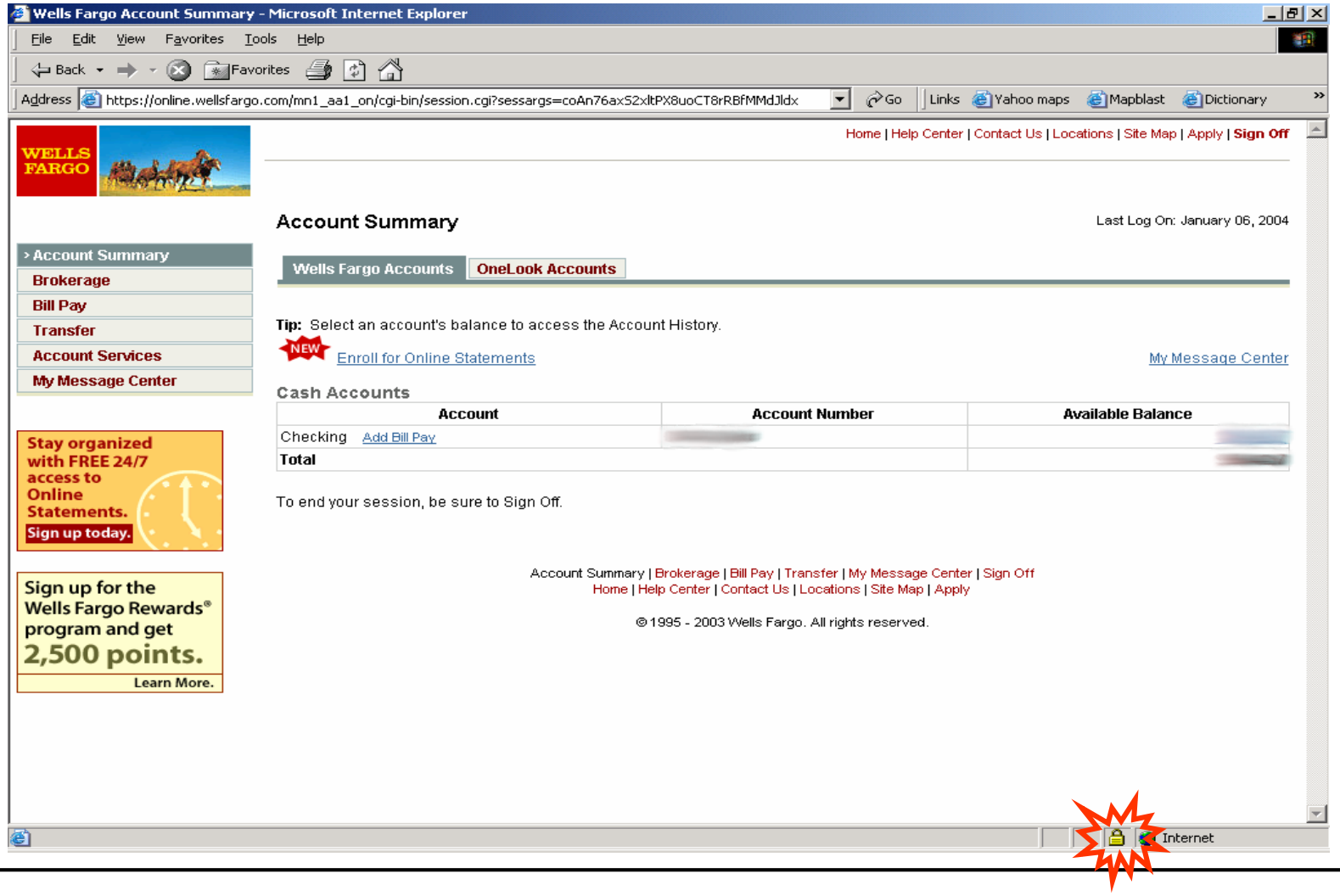


- Intuitively this solves the consistency problem, but no proof exists.
- What if Eve registers Alice's public key on her name?
  - Even if Eve does not know Alice's secret key, she may be able to perform replay attacks to violate consistency of key binding

What is SSL / TLS?

- Transport Layer Security protocol, ver 1.0
  - De facto standard for Internet security
  - “The primary goal of the TLS protocol is to provide privacy and data integrity between two communicating applications”
  - In practice, used to protect information transmitted between browsers and Web servers
- Based on Secure Sockets Layers protocol, ver 3.0
  - Same protocol design, different algorithms
- Deployed in nearly every web browser

SSL / TLS in the Real World



History of the Protocol

SSL 1.0
- Internal Netscape design, early 1994?
- Lost in the mists of time

SSL 2.0
- Published by Netscape, November 1994
- Several problems

SSL 3.0
- Designed by Netscape and Paul Kocher, November 1996

TLS 1.0
- Internet standard based on SSL 3.0, January 1999
- Not interoperable with SSL 3.0

SSL Applications

HTTP – original application

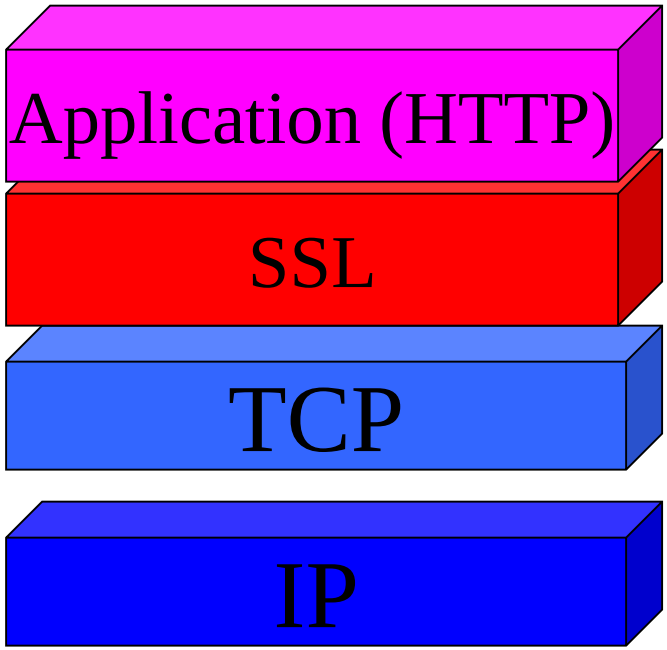
Secure mail

- Server to client connection
- SMTP/SSL?

Telnet, ftp ..

Resources: http://www.openssl.org/related/apps.html

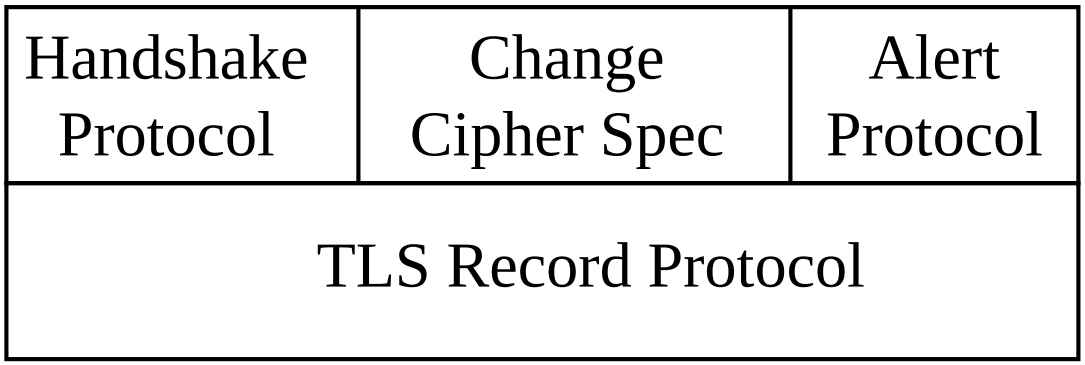
Architecture



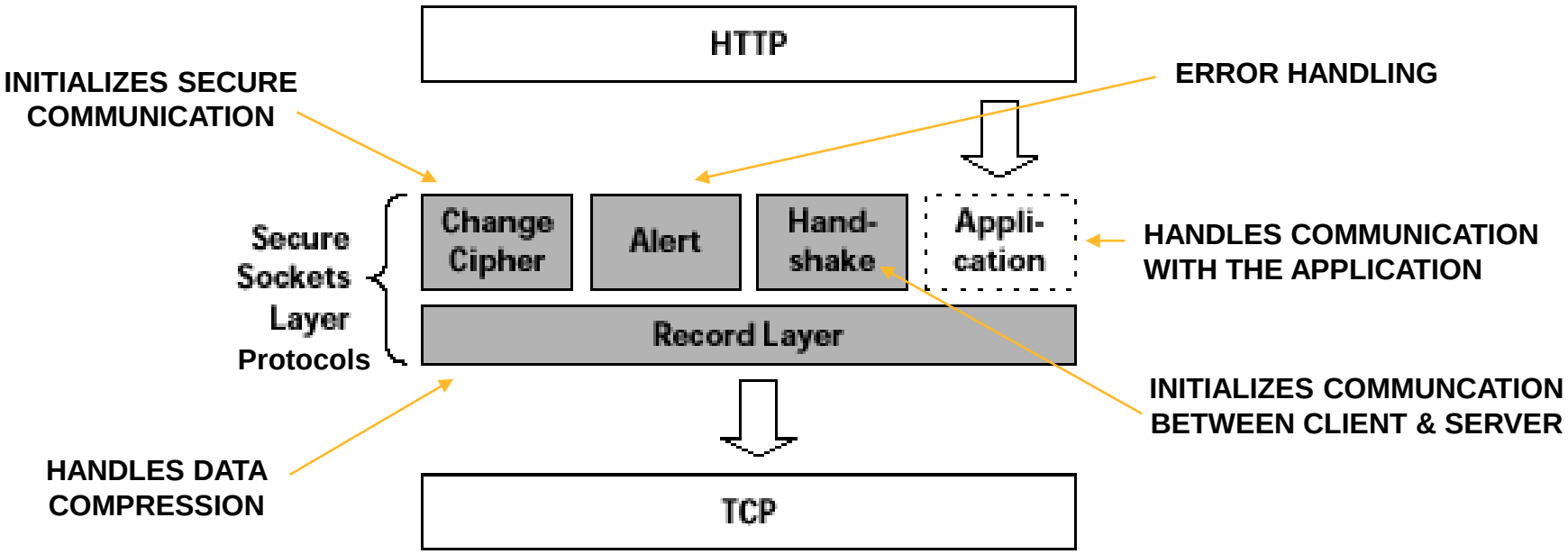
Architecture

Record Protocol to transfer application and TLS information

A session is established using a Handshake Protocol



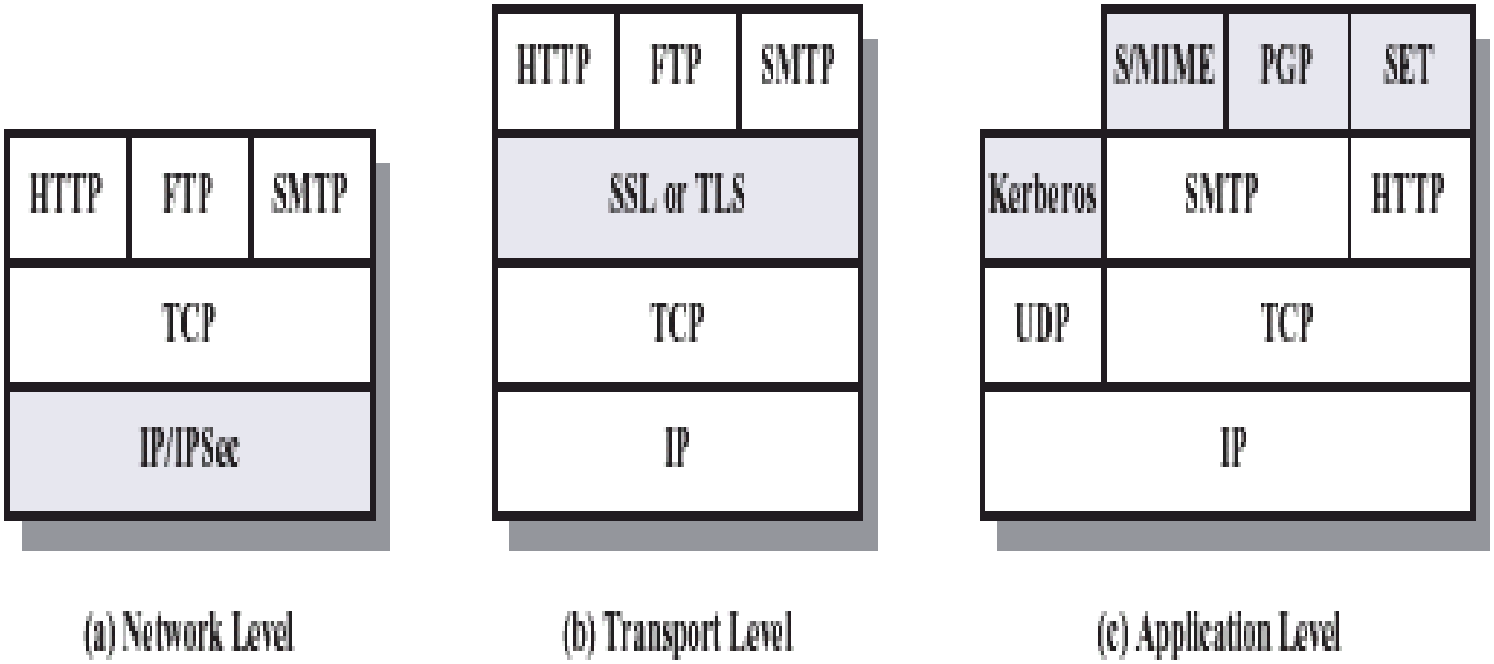
Architecture (cont'd)



Alternative architectures

- Separate Layer
  - Over TCP: SSL
  - Over IP: IPSec
- Application-Specific
  - SHTTP
- Parallel
  - Kerberos; Kerberos with TLS?

Layers of Security



SSL choices

- Connection-oriented
  - SSL, TLS do not support UDP
  - But WTLS does
- No non-repudiation
  - But signatures are used for AKE
- “Only protects the pipe”
  - Attacks *are* mounted on data before and after “the pipe”

SSL security services

Server authentication

Client authentication is optional

Encryption

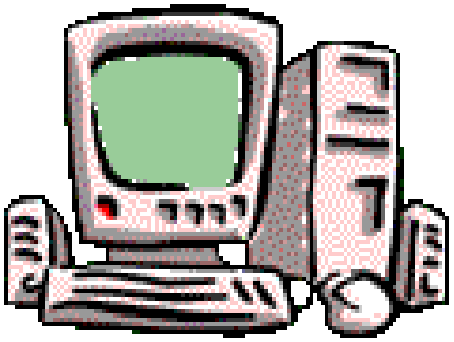
Message integrity

SSL phases

- Handshake
  - Set protocol details
  - Authenticate server
  - Establish keys
- Data transfer

SSL – A Server Authentication Scenario

Client



Server

2. Request a Web Page from Server

3. Send Server certificate to client

8. Send symmetric encryption key to server

1. URL entered in browser

4. Browser looks for certificate in its store of public CA certificates

5. Browser checks the signature on server certificate using Public key of CA

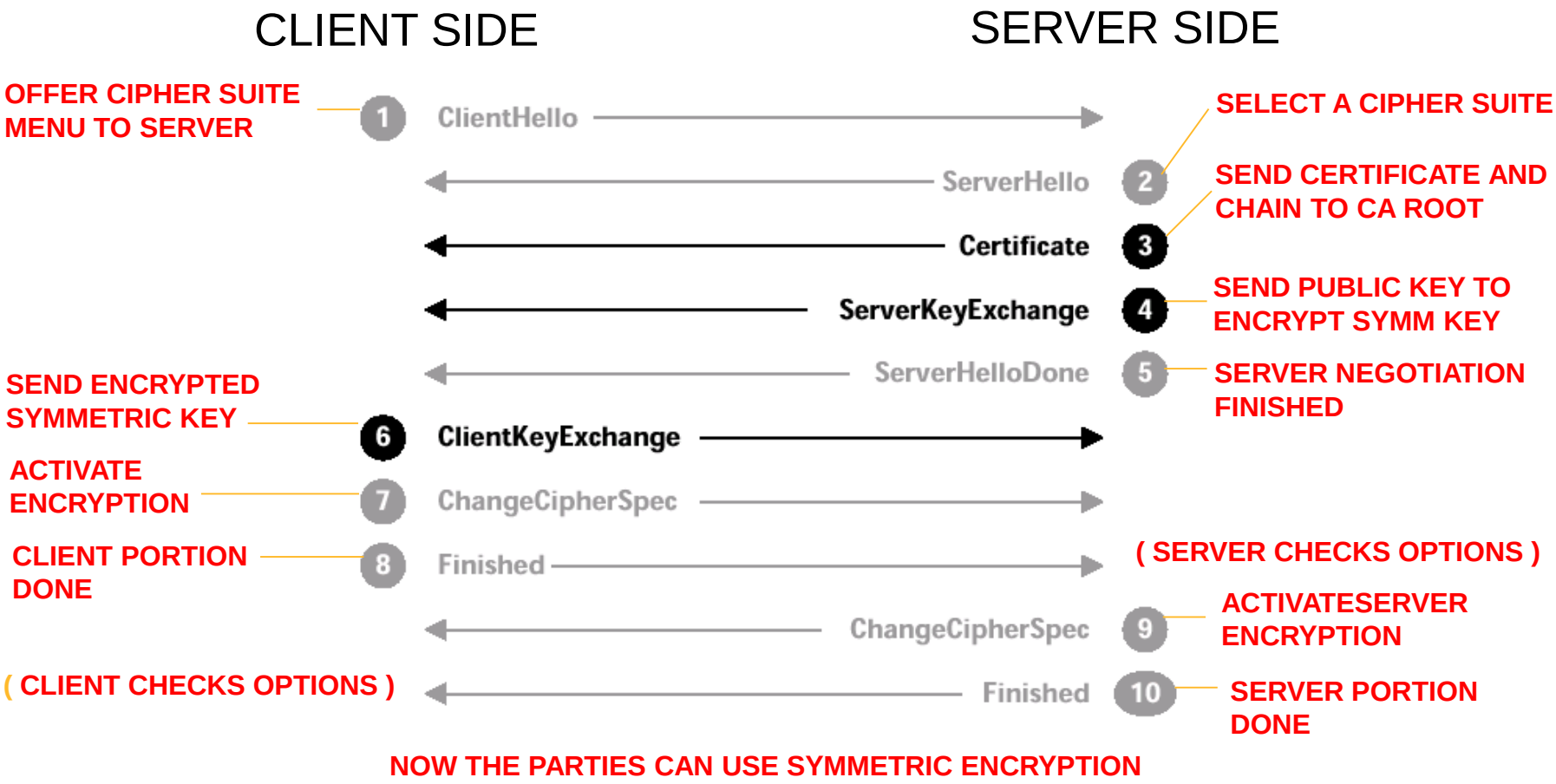
6. If correct signature then Accept

7. Generate a symmetric encryption key using the server's public key

9. Symmetric key is decrypted using server's own private key

10. The symmetric encryption key is now shared between client and server and can be used for secure communication

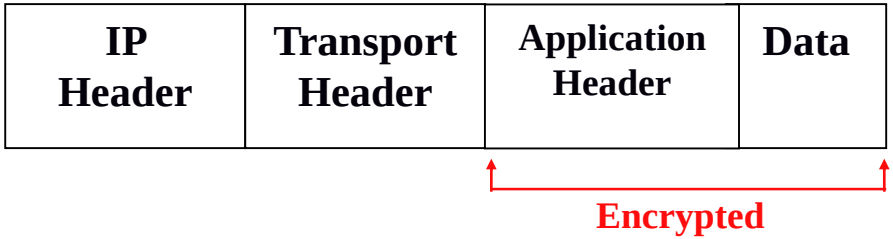
SSL Messages



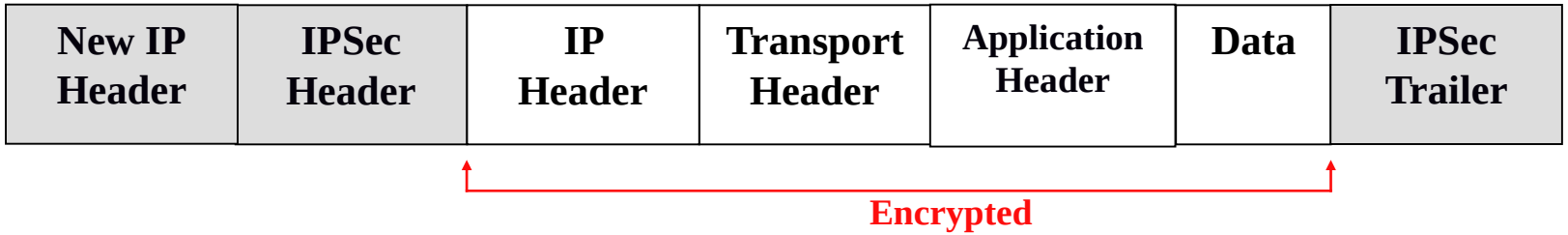
SOURCE: THOMAS, *SSL AND TLS ESSENTIALS*

SSL vs. VPN

- **SSL:**



- **IPSec VPN:**



VPN uses additional header (tunnel) and encapsulation

VPN encrypts original packet (using ESP)

SSL encrypts only the application header and data

Configuring an IPSec Tunnel through a Firewall

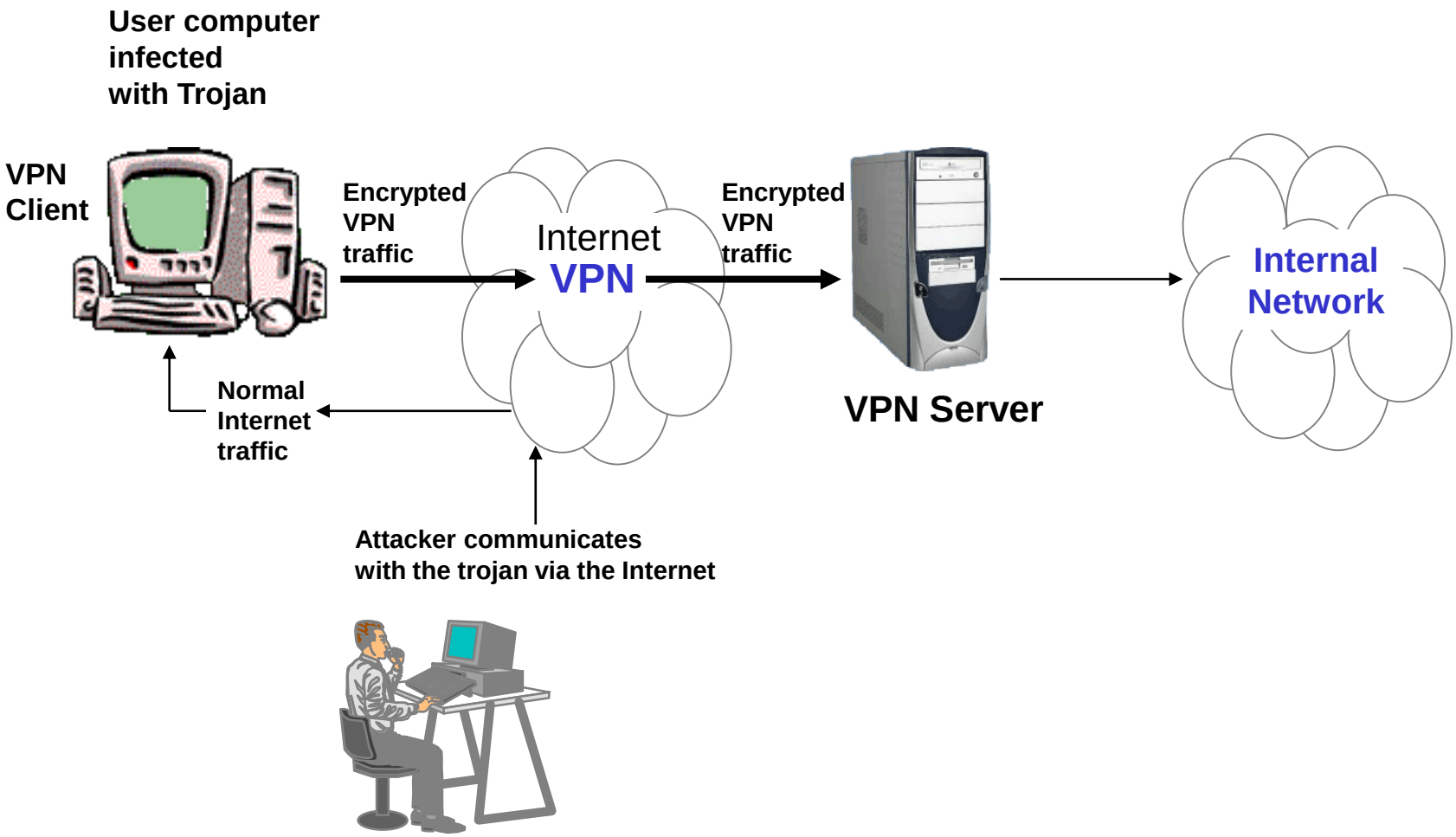
When a firewall or a filtering router exists between IPSec endpoints, it must be configured to forward IPSec traffic as follows:

Need to enable IPSec traffic: through UDP port 500

Also need to enable two additional IP Protocols as:

- IP Encapsulated Security Payload (ESP): through port 50
- IP Authentication Header (AH) protocol: through port 51

VPN User-to-Network: Attack Scenario



VPN Remarks

Select appropriate VPN choice:
- User-to-network
- Network-to-network

Ensure interoperability between VPN endpoints

Use VPN for secure remote access and DMZ component maintenance (e.g., firewalls, web servers, .etc)

Ensure proper VPN configuration through firewall

Benchmark security requirement with VPN complexity

Check if you need a full VPN or other solutions like SSL would be enough

Perform penetration testing of deployed VPN solutions

Keep current with latest security alerts and vendor patches.

Intrusion and Intrusion Detection

Intrusion : Attempting to break into or misuse your system.

Intruders may be from outside the network or legitimate users of the network.

Intrusion can be a physical, system or remote intrusion.

Intrusion Detection Systems (IDS)

Intrusion Detection Systems look for attack signatures, which are specific patterns that usually indicate malicious or suspicious intent.

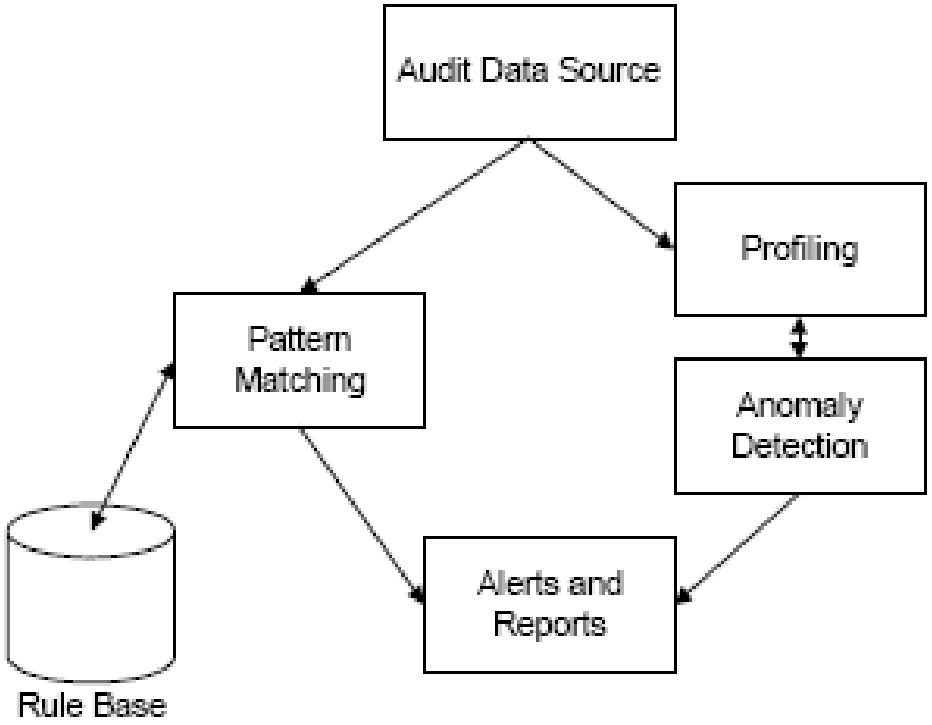
IDS is a host or network security component monitoring activities and identifying patterns of behavior or traffic indicating possible violation of security policy

Intrusion Detection Systems (IDS)

Different ways of classifying a

IDS based on

- Host based
- Network based



Intrusion Detection System (IDS)

An Intrusion Detection System (IDS) is a system for detecting anomaly behaviour within a network or computer resources.

An IDS uses a number of sensors to detect intrusions.

Examples of these sensors are:

- TCP connections monitors
- Log file monitors.
- File integrity checkers.

The IDS system is responsible for collecting data from its sensors and analyzing it to notify administrators of any malicious activity on the network.

Types of IDS

Host-Based IDS (HIDS)

- Resides on a particular host and looks for indications of attacks on that host.

- Sensors can also be loaded on various servers within a network and controlled by a central manager.

Network-Based IDS (NIDS)

- Resides on a separate system that watches network traffic, looking for indications of attacks which travels that portion of the network.

- Basically acting as a packet sniffer by turning the Network Interface Card (NIC) into promiscuous mode (i.e. collect all traffic, not only its own)

- Commonly used on shared media networks (e.g using a HUB), whereas switched networks require special configuration for NIDS to see all traffic.

- Can't examine encrypted traffic (e.g. SSH, SFTP and VPN traffic)

HIDS Detection Models

Log Analyzers

- Check log files to see if there is a match to some criteria
- Log analyzers are reactive by nature (notification is made after an event occurred)
- Good for tracking activities of authorised internal users

Signature-Based Sensors

- Ability to analyze incoming traffic as well as log entries
- Can see attacks as they come into system. However, an attack can finish before taking action
- So they are also reactive systems, and good for tracking activities of authorised internal users

System Call Analyzers

- Analyse calls between applications and the operating system looking for security breaches
- Can prevent an event from happening if it matches a signature (e.g. buffer overflow)

Application Behaviour Analyzers

- Check system calls to see if application is allowed to perform the action, rather than checking if action looks like an attack
- E.g. Can prevent web server from reading/writing files to a location other than Web directory

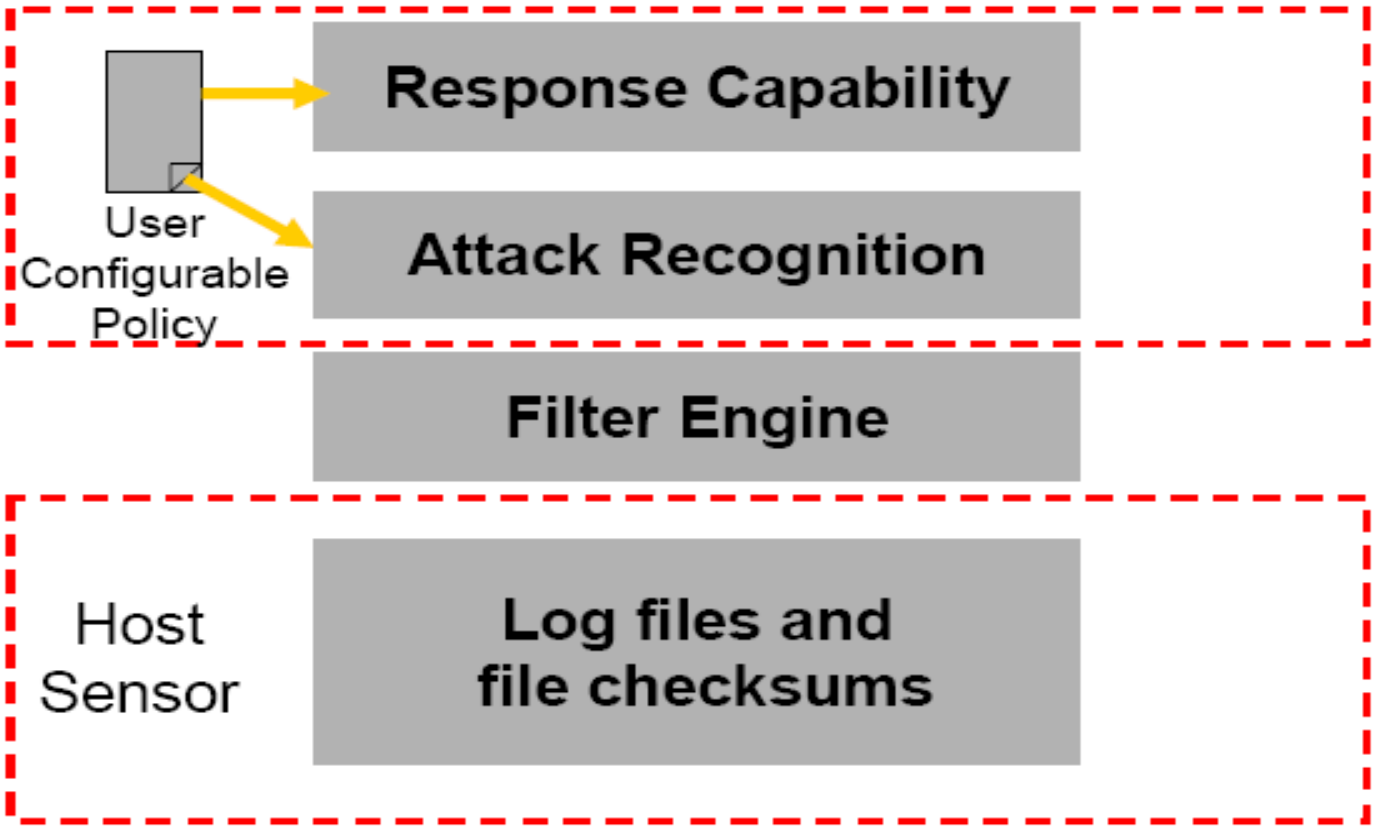
File Integrity Checkers

- Use cryptographic check sum to see if the file content has been changed
- E.g. A web defacement attack can be identified due to changing the content of the homepage

Host Based IDS

- Typically monitors system, event, and security logs on Windows and syslog in Unix environments
  - May use custom sensors (e.g. implemented as kernel modules)
- Checks key system files and executables via checksums at regular intervals for unexpected changes
  - Popularized by the Tripwire utility, now part of Windows Vista
- Some products can use regular-expressions to refine attack signatures
- Some products listen to port activity and alert when specific ports are accessed – resulting in a limited/partial NIDS capability

Host Based IDS



NIDS Detection Models

Signature-based Detection Model

- Database of Malicious Signatures
- Looking for direct matches

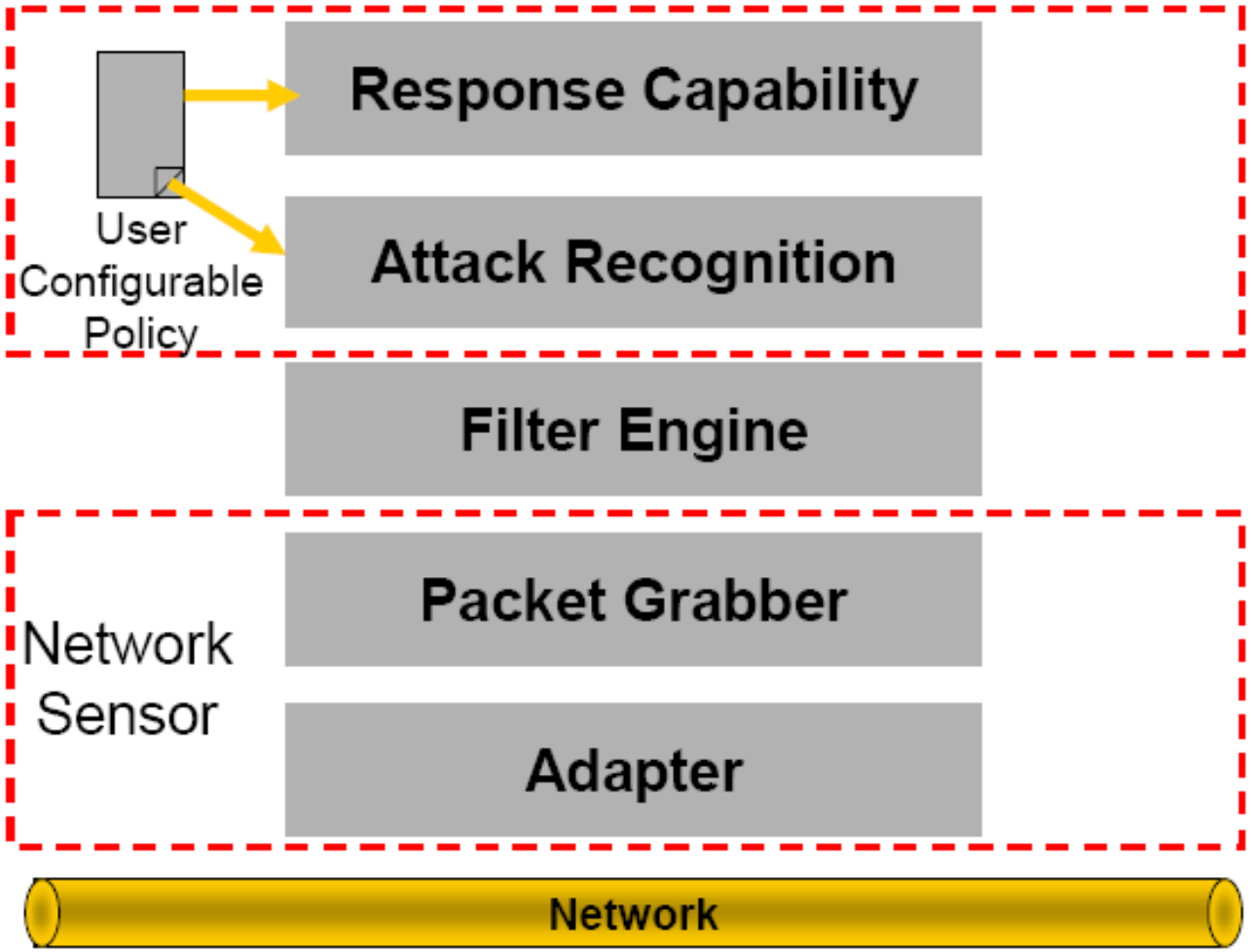
Anomaly-based Detection Model

- Database of Normal Activity
- Looking for deviation from normal
- Still immature area, because of difficulty to define what is normal and what is not.

Network Based IDS

- Uses network packets (from a sniffer or host) as the data source
  - This can simply be a network adapter running in promiscuous mode
  - Objective is to monitor and analyze all traffic on a given network segment in (near) real-time
- Attack recognition can use several techniques to recognize patterns signifying potential attacks, e.g.
  - Pattern, expression or bytecode matching
  - Frequency or threshold crossing (e.g. detection of port scanning activity)
  - Correlation of lesser events (not much of this in commercial systems because of problems with specificity)

Network Based IDS



Signature-Based detection model

Advantages

- No learning curve involved (compared to anomaly detection)
- Works well for known attacks

Disadvantages

- New attacks (not in the database of signatures) cannot be detected
- False positives, no threshold (either true or false)
- Maintenance (difficulty in choosing what to detect)
- Not very hard to bypass (e.g. Packet fragmentation, character encoding)
- Stateless inspection

Anomaly based IDS

This IDS models the normal usage of the network as a noise characterization.

Anything distinct from the noise is assumed to be an intrusion activity.

E.g flooding a host with lots of packet.

The primary strength is its ability to recognize novel attacks.

Drawbacks of Anomaly detection IDS

Assumes that intrusions will be accompanied by manifestations that are sufficiently unusual so as to permit detection.

These generate many false alarms and hence compromise the effectiveness of the IDS.

IDS Placement in a Network

In Front of Firewall (NIDS): P1

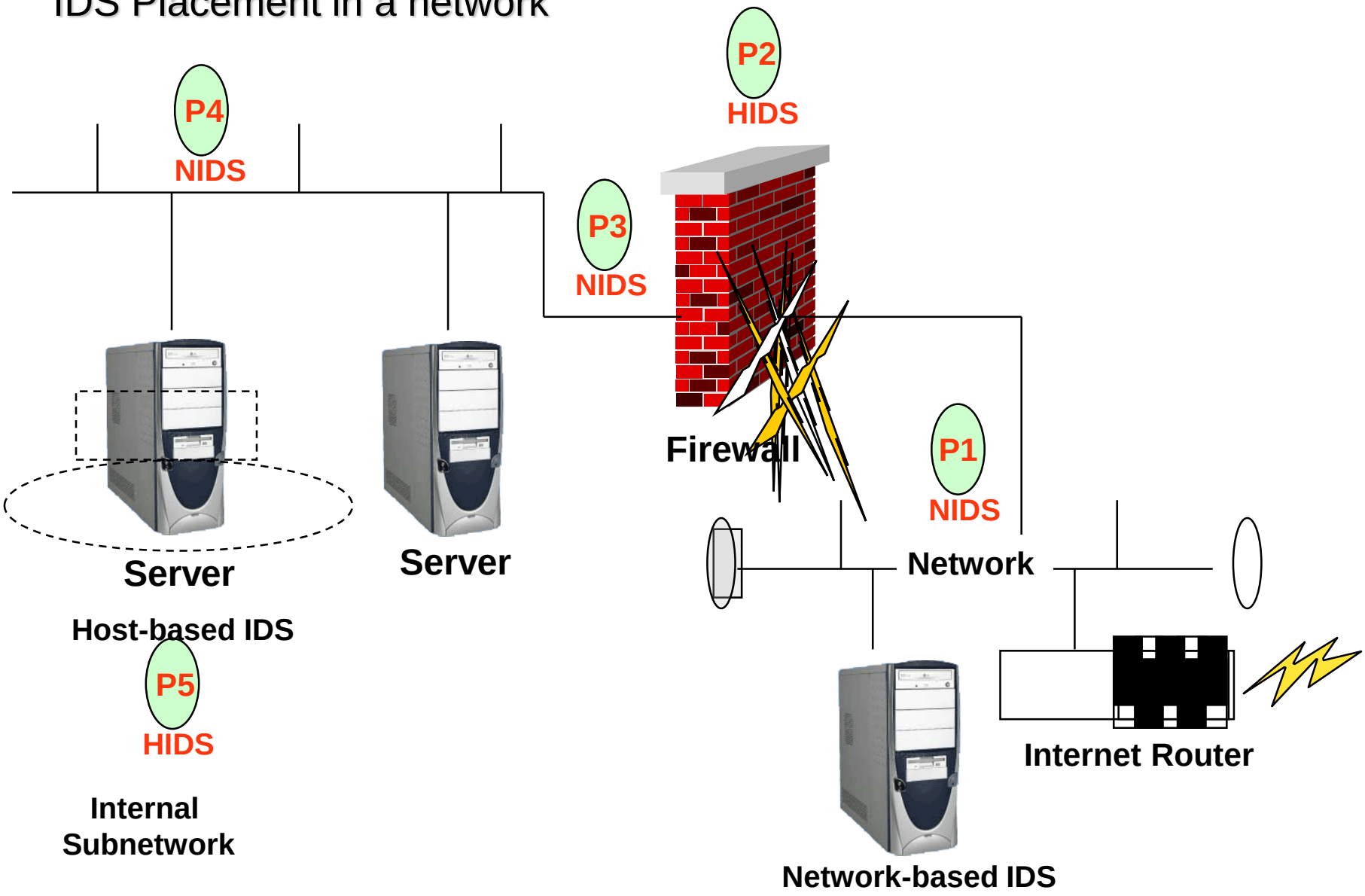
Part of the Firewall (HIDS): P2

Behind the Firewall (NIDS): P3

In Front of Sensitive Internal Subnetwork(s) (NIDS): P4

On Each Sensitive Server (HIDS): P5

IDS Placement in a network



IDS Response Type

Passive Response: Most common response with least disruption of service

Shunning => e.g. ignoring an IIS attack targeted to Apache server
Logging => IP, date and time, Process ID, user ID, .etc
Notification => alert by e.g. by email, pager, console broadcast, .etc.

Active Response: Fast, but needs careful consideration to avoid DOS to legitimate users

Termination => (Connections, Sessions, or Processes)
Reconfiguration => e.g adding attacker IP to firewall blocked list
Deception => E.g. trapping the attacker to a honeypot

A Honeypot: is a system specially designed to attract an attacker into accessing it (thinking of it as a real system) while the attacker is being watched and his actions are being logged?

Passive Vs. Active Response

Policy	Passive Response	Active Response
Detection of Attacks	- Logging - Notification	- None
Prevention of Attacks	- Logging - Notification	- Connection termination - Process termination - Router/firewall reconfiguration
Detection of Policy violation	- Logging - Notification	- None
Enforcement of use policies (e.g. IM)	- Logging - Notification	- Connection termination - Possible proxy reconfiguration
Collection of evidence	- Logging - Notification	- Deception (honeypot) - Possible connection termination

Commercial IDSs

ISS – Real Secure from Internet Security Systems:

Real time IDS.

Contains both host and network based IDS.

Tripwire – File integrity assessment tool.

Bro and Snort – open source public-domain system.

Future of IDS

To integrate the network and host based IDS for better detection.

Developing IDS schemes for detecting novel attacks rather than individual instantiations.

Intrusion Prevention Systems (IPS)

New concept that aims at making IDS preventive i.e. to stop intrusions before happening rather than just being reactive systems.

Preventing attacks using HIDS can be clearly seen, e.g:

- System call analyzers
- Application behaviour analyzers

Preventing attacks using NIDS is slightly more difficult, e.g:

- While NIDS is still analysing traffic, that traffic might have already reached and compromised target system before connection termination or firewall reconfiguration takes place

How a NIDS can be an IPS?

To make a NIDS more preventive it must be placed in-line with traffic (like a firewall is), or be part of a firewall such that the firewall doesn't pass traffic except in consultation with the NIDS.

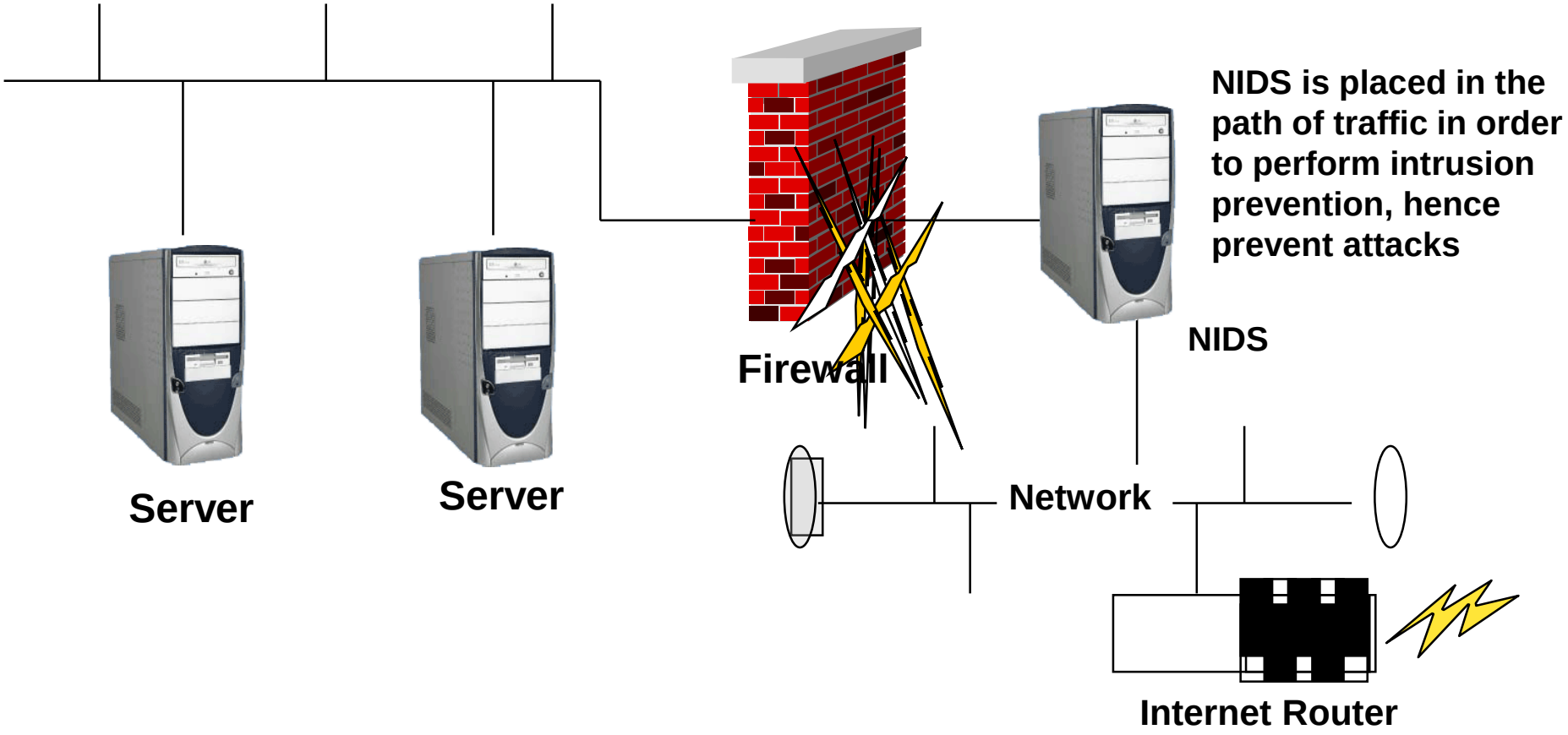
Issues to consider:

DOS – terminating connections to legitimate users

Availability – the NIDS now becomes a single point of failure

Performance - All traffic gets analysed by the NIDS before passing through

How a NIDS can be an IPS?



NIDS is placed in the path of traffic in order to perform intrusion prevention, hence prevent attacks