Internet Security

Internet Security

- Network Layer Security
- Application Layer Security
- System Security

Network Layer Security

- Packets might be
  - Modified in transit
  - May be spoofed
  - May contain bad payload.
- Network layer security provides
  - Authentication and integrity
  - Confidentiality
  - Access control

Application Layer Security

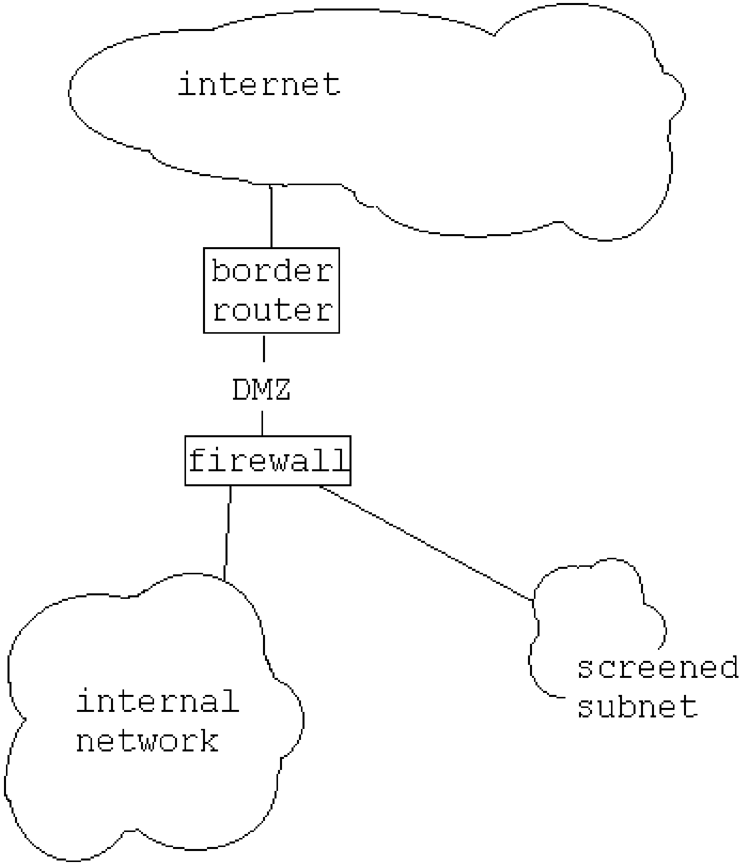
- Safeguards built into a particular application.
- Becomes more important as trust in network layer security diminishes.
- Provides
  - Authentication
  - Access Control
  - Confidentiality
  - Data integrity
  - Non-repudiation

System Security

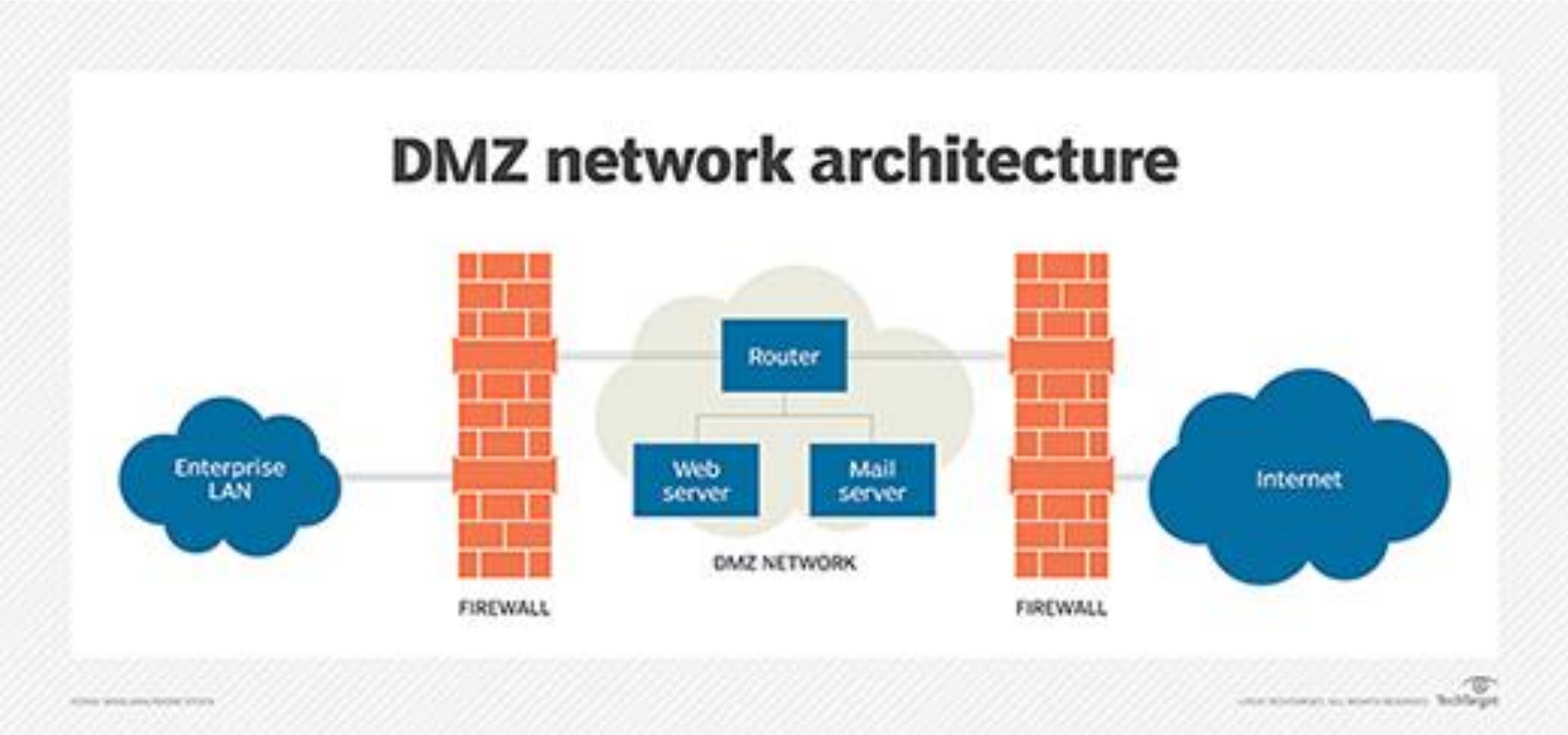
- Protection of a particular end system by
  - Removal of known vulnerabilities (patching)
  - Minimal penetration risk configuration
    - Limits ports on which it listens
    - Limits services that run.
  - Authentication of downloaded software
  - Proper audit mechanisms
  - Up-to-date administration
    - Password changes enforced.
    - Guessable passwords are disallowed.
    - User accounts reflect needs.

Firewalls

- Border Router
  - First / last router under control of system administration.
- DMZ
  - Demilitarized zone.
  - Security is low, since not protected by firewall. Locate webservers and other services there that generate potentially unsafe traffic.
- Firewall
  - Filters packages based on a variety of rules.



Firewalls

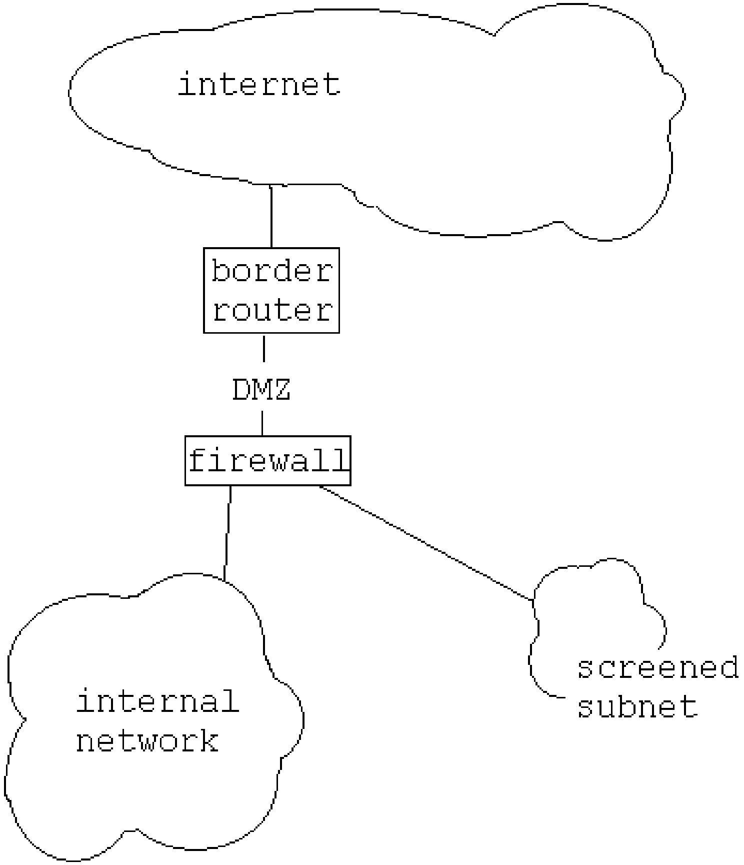

DMZ network architecture
Enterprise LAN
FIREWALL
Router
Web server
Mail server
DMZ NETWORK
FIREWALL
Internet
TechTarget

IDS

- IDS
  - Intrusion Detection System.
    - NIDS: glean intrusion signatures from traffic.
    - HIDS: monitor activity at a host on which they are located.

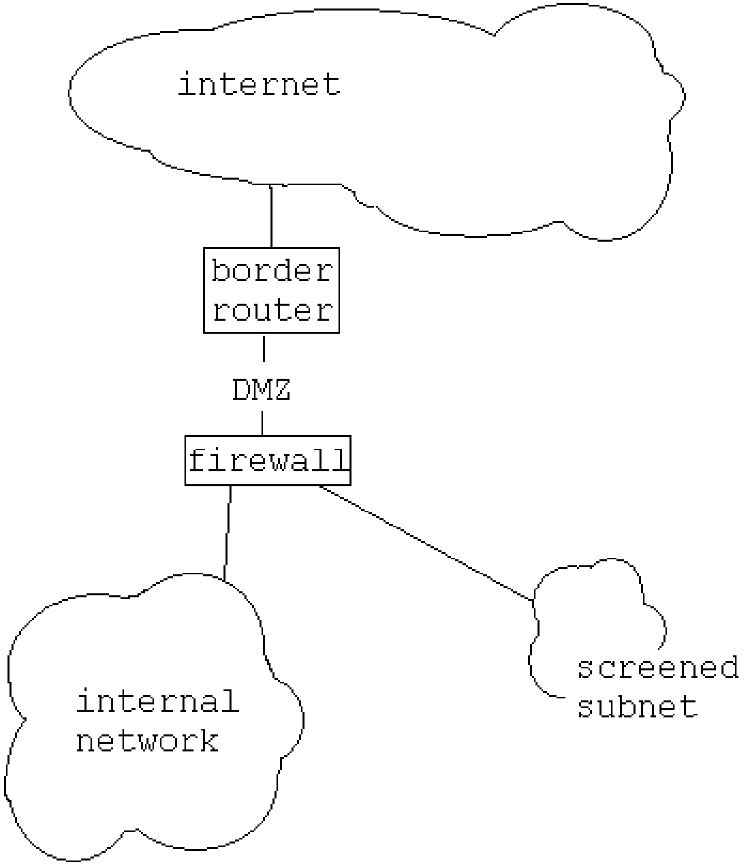
CM

- Configuration Management
  - Known vulnerabilities account for most of actually perpetrated exploits.
  - For most of them, patches were available, but not installed.
  - CM tries to enforce uniform security policies.
- Backdoors
  - An entrance into the system that avoids perimeter defenses.



CM

- Configuration Management
  - Planning
  - Identifying and implementing
  - Controlling
  - Monitoring (automated)
  - Remediation



Firewall Packet Filtering

- Static Packet Filtering
  - Allow or deny access to packets based on internal characteristics.

```
access list 111 deny ip host 205.205.205.205.1 any
access list 111 permit tcp host 205.205.205.205.1 any
access list 111 deny icmp any any echo-request
access list 111 permit icmp any any packet-to-big
access list 111 deny icmp any any
```

Cisco extended ACL

Firewall Static Packet Filtering

Difficult to design efficient rules.

- Easy to get the rules tables wrong and allow bad traffic.

- Security risks
  - People can piggy-back bad messages in harmless ones.
    - http traffic is known to be used as a backdoor.

Firewall Static Packet Filtering

- Configuring a packet filter:
  - Security Policy: what is allowed, what is not allowed.
  - Allowable types of packets must be specified logically, in terms of logical expression on packet fields.
  - Expressions need to be rewritten in the firewall vendor's language.

Firewall Static Packet Filtering

- Example
  - Security Policy:
    - Allow inbound mail messages (SMTP, port 25), but only to gateway.
    - Block host faucet.

action	Our host	port	Their host	port	comment
block	*	*	faucet	*	We don't trust these people.
allow	OUR-GW	25	*	*	Connection to our SMTP server

Firewall Static Packet Filtering

- Example
  - If no rule applies, then the packet is dropped.
    - Without additional rules, our rule set would drop all non-mail packets. There would also be no replies.
  - Beware of a rule like this (intended to allow acks)

action	Our host	port	Their host	port	comment
allow	*	*	*	25	Connection to their SMTP port

  - Based solely on outside host's port number.
    - Port 25 is *usually* the mail port.
    - But there is no guarantee.

Firewall Static Packet Filtering

- Example
  - Expand rule set to allow connection with the outside:

action	Our host	port	Their host	port	Flag	comment
block	*	*	faucet	*		
allow	OUR-GW	25	*	*		
allow	(our host)	*	*	25		Our packets to their port
allow	*	25	*	*	ACK	Their replies

Specify the names of all machines allowed to send mail to the outside here.

Firewall Static Packet Filtering

- Address Spoofing
  - At a minimum:
    - Don’t allow inside source addresses coming in.
    - Don’t allow outside source addresses going out.
    - Block source routing at the border routers.

Firewall Static Packet Filtering

- Routing Information
  - If a node is unreachable from the outside then the node is almost (but not quite) as safe as a node disconnected from the net.
  - Internal routers should not advertise paths to such nodes to the outside.
  - Filter routes learned from the outside:
    - Subversion by route confusion.
    - Route squatting:
      - Use internal addresses that belong to a different domain.
      - The nodes are de facto unreachable from the outside.
      - Use non-announced addresses. (e.g. 10.x.x.x)
        - But beware, when companies merge, these addresses tend to be incompatible.
        - So pick addresses in unpopular address ranges.

Firewall Static Packet Filtering

- Performance
  - Packet filtering is done at the border.
    - No degradation for the internal network.
  - Typically, connection to ISP is the bottleneck.
  - However:
    - Degradation depends on the number of rules applied.
    - Can be mitigated by careful ordering of rules.

Firewall Application Level Filtering

- Packet filters only look at
    - The source address
    - The destination address
    - TCP / UDP port numbers
    - TCP / UDP flags.
- Application filters deals with the details of the service they are checking.
    - E.g. a mail application filter looks at
        - RFC 822 headers.
        - MIME attachments.
        - Might identify virus infected attachments.

Firewall Application Level Filtering

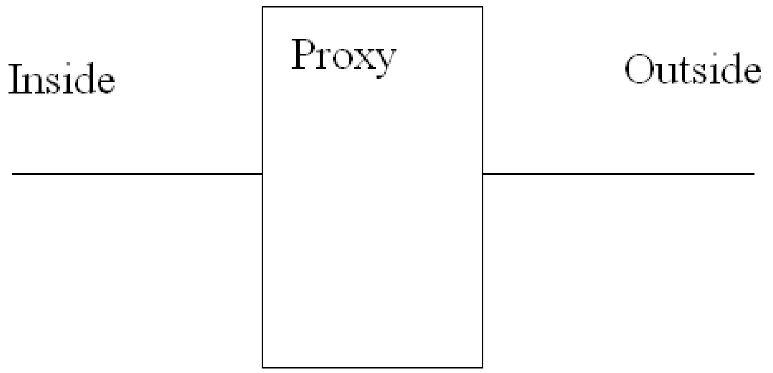
- Snort:
  - Allows to set up rules that pass a packet on to another service.
- Commercial firewalls
  - Include application level filters for many products.
  - Use non-disclosure agreement to obtain proprietary protocols

Firewall Dynamic Packet Filtering

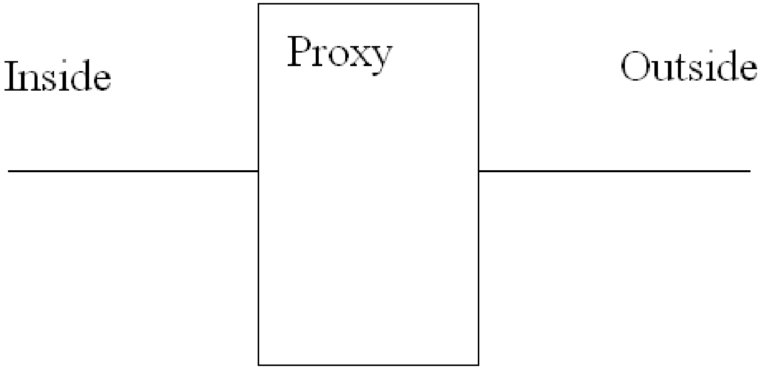
- Stateful Firewall
- Still look at each packet.
- Maintains a state of each connection.
  - Implements connection filtering.
  - Dynamically adjust a filtering table of current connections.
  - Implementation
    - Adjust the filtering rules dynamically.
      - E.g.: We started an HTTP connection to a given host.
      - Now HTTP packages from that host are allowed.
    - OR: Terminate the connection at the firewall and then have the firewall call the ultimate destination (proxying).

Proxy Firewalls

- Proxies act on behalf of a client.
- Proxy firewall
  - Reverse Proxy
    - Receives packages on one card.
    - Processes requests.
    - Translates them into internal requests on other card.
    - Receives answers from inside and translates to the outside.



Proxy Firewalls

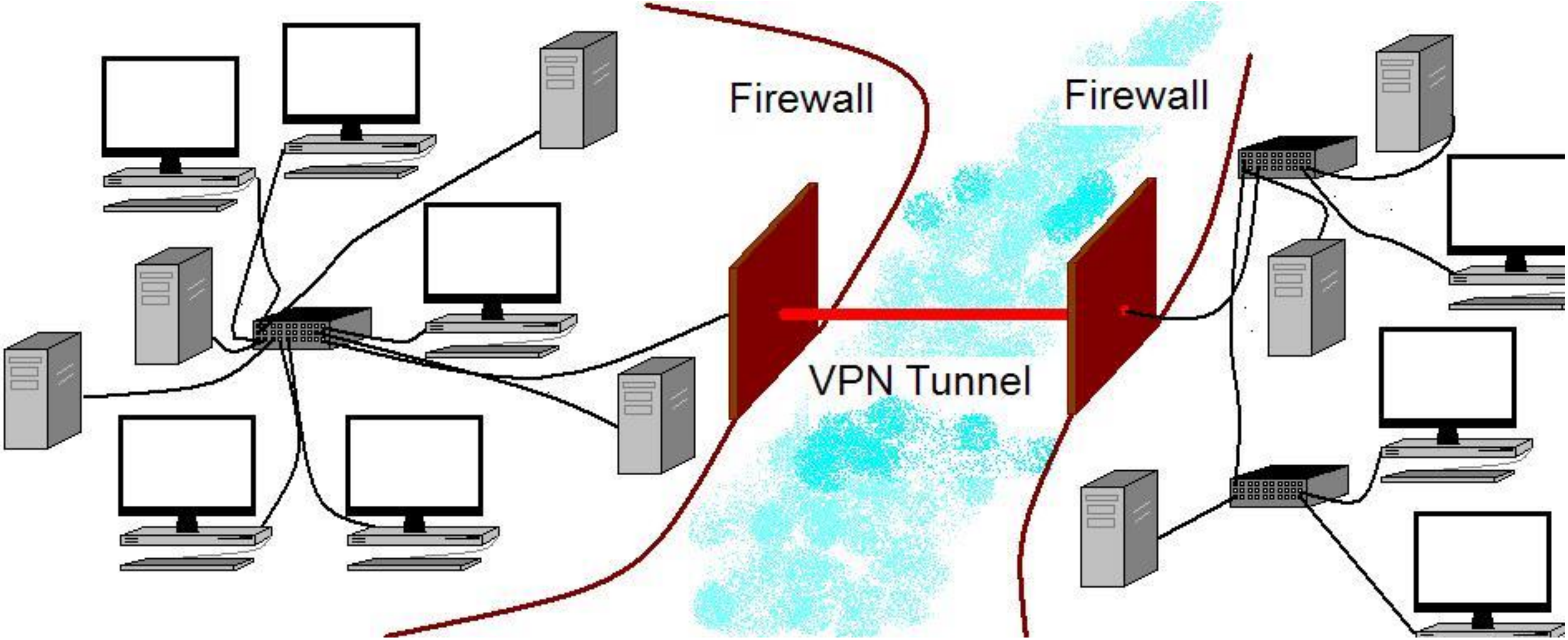


- Proxy firewall
  - Forward Proxy
    - Receives requests from the inside.
    - Processes requests.
    - Translates them into requests to the outside on other card.
    - Receives answers from outside and translates to the inside.
  - Acts on behalf of inside machine that is protected from the vagaries of the internet.

Proxy Firewalls

- Application level proxies work at the level of application.
- Circuit-level proxies
  - does not *understand* the application
  - makes filtering decisions by validating and monitoring sessions.

Virtual Private Networks

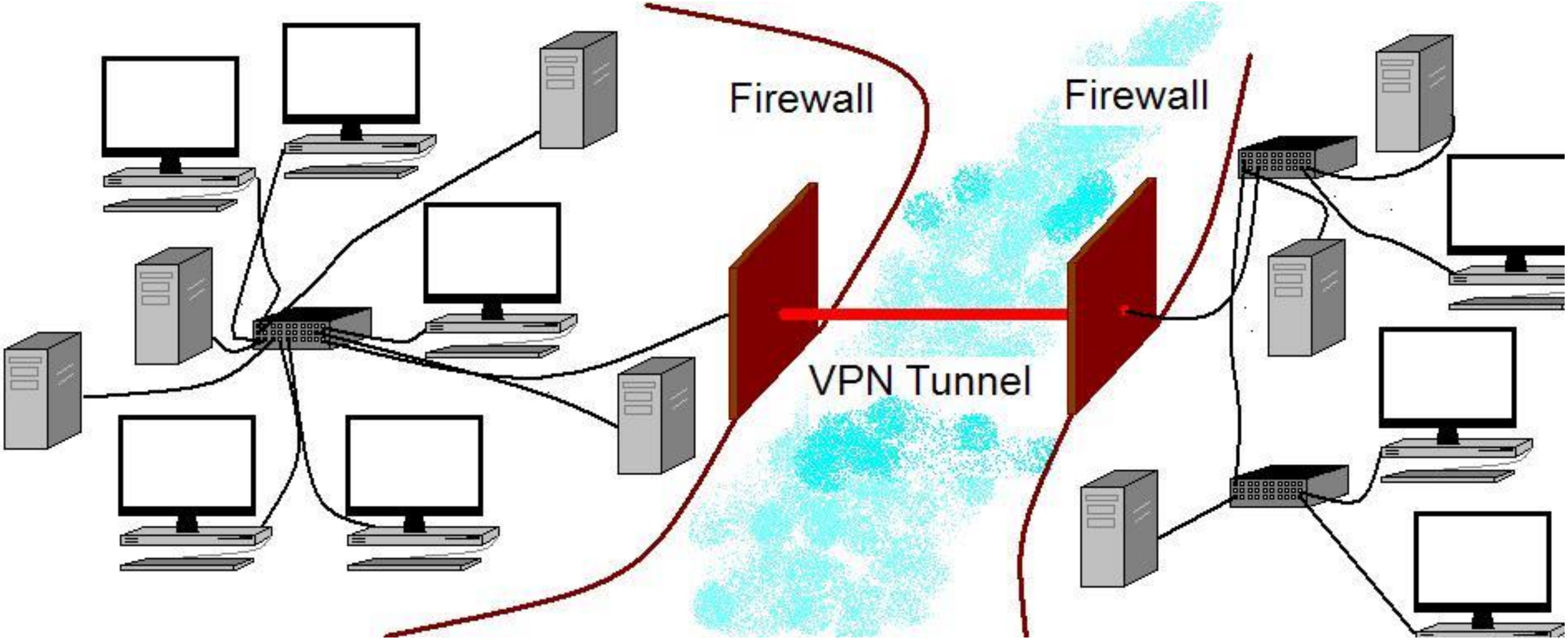


VPN

Virtual Private Networks

- VPN uses connections over an existing public network
- Connection secured with encryption
  - Host to Host
  - Host to Gateway
  - Gateway to Gateway

Virtual Private Networks



Virtual Private Networks

- Encryption can be done at
  - Application level.
  - Transport level.
  - Network level.
  - Data link level.

Virtual Private Networks

- Application Level
    - Pretty Good Privacy
    - Secure Shell (SSH)
- Transport Level
    - Secure Socket Layer
        - Does not protect the package, but its content.
        - Typically runs at the application level of the OS, so **OS does not need to be changed**.
- Network Level
    - IPSec
        - Encrypts package itself.
        - Encrypted package receives a new package header.
            - IPSec protects port address, but not destination address.
        - **OS need to be changed** (but only once: Win2000, WinXP)
- Data Link
    - Layer 2 Tunneling Protocol addition to Point-to-Point protocol (PPP)
        - Encrypts packets on the data layer.

Virtual Private Networks

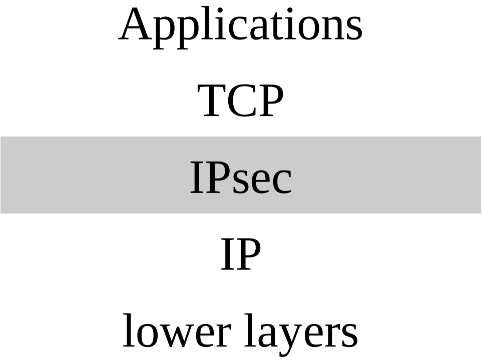
- Alternatives are dedicated point-to-point connections such as a private T1 line.
  - Most secure.
  - Most expensive.
  - Takes time to set-up.

IPSec Overview

- Changes the IP layer to provide security.
  - Transport mode
    - Protects the upper-layer protocol (TCP) data in each packet and provides end-to-end protection
  - Tunnel mode
    - Protects an entire IP packet by enveloping it in a new packet with its own plaintext IP header.

IPSec

- Implemented below the transport layer.
- No application needs to be rewritten.
- Is part of the OS.



Applications

TCP

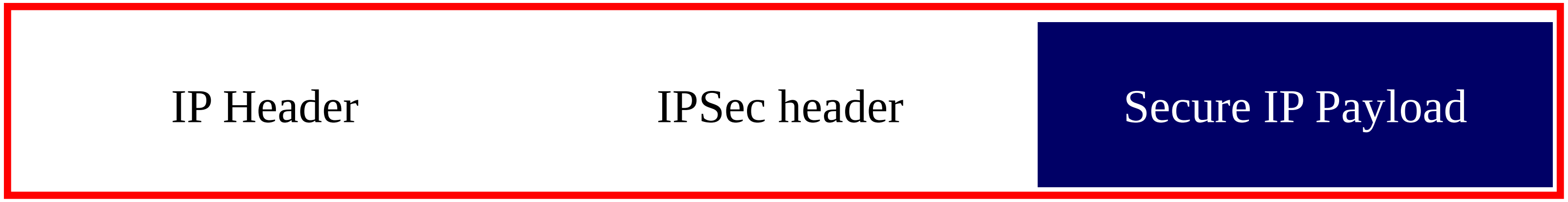
IPsec

IP

lower layers

IPSec

- An IPSec packet in tunnel mode completely encapsulates the payload.
- IP Header is either an
    - Authentication Header
    - ESP Encapsulating Security Payload that tells the user which Security Association to use.



IP Header

IPSec header

Secure IP Payload

IPSec

- Security Association
  - Cryptographically protected connection.
  - Paradigm to manage authentication and confidentiality between sender and receiver.
  - Unidirectional.
  - IPSec header contains **SPI** (Security Parameter Index) that identifies the security association.
    - Allows partner to look up the necessary data such as the key in SA database.

IPSec

- Security Association Database
  - When X transmits to Y in IPSec, X looks up Y in the SA database.
    - Provides key
    - Provides SPI
    - Provides algorithms to be used
    - Provides sequence number
  - When Y receives a transmission, Y uses the SPI and the destination address to find the SA.

IPSec

- Security Policy Database
  - Specifies what to do with packets:
    - Dropping
    - Forwarded and accepted without IPSec protection
    - Forwarded and protected by IPSec
  - Decision based on fields in the IPsec packet.

IPSec

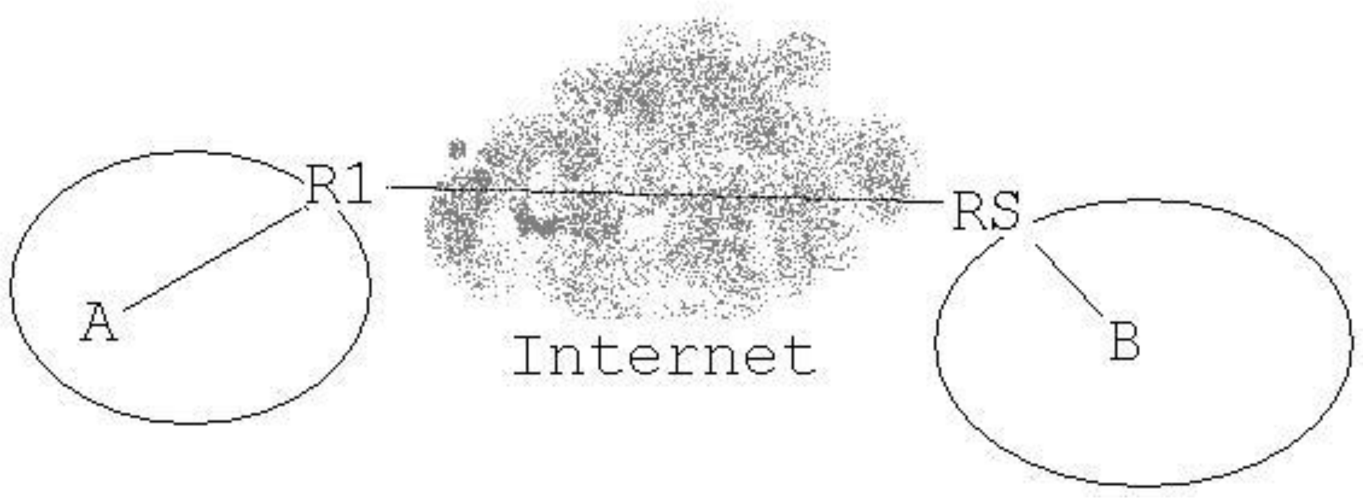
- Two types of IPsec headers.
- AH
  - Authentication header.
  - Provides integrity protection only.
  - Allows firewalls to peek at TCP ports.
- ESP
  - Encapsulating Security Payload
    - Optional integrity protection
    - Optional encryption

IPSec

- Transport mode versus Tunnel mode

Original Packet	IPsec Package in Transport Mode	IPSec Package in Tunnel Mode
IP header | rest	IP header | IPsec header | rest	new IP hdr | IPSec | IP header | rest

IPSec



IPsec in tunnel mode for a VPN:

IP: src=R1, dst=R2 | ESP | IP: src=A, dst=B | packet

Secure Socket Layer

- 1995: deployed in Netscape Navigator as SSLv2.
- 1995: Microsoft fixes SSLv2 and introduces a similar protocol
  - Private Communication Technology (PCT)
- 1996: Netscape introduces SSLv3
- 1999: IETF introduces Transport Layer Security.

- SSLv3 remains the most implemented protocol.

Secure Socket Layer

- SSL is built on top of TCP.
  - TCP provides reliable packet delivery.
  - Rogue packet problem:
    - Maliciously introduced TCP packet.
      - Easy to do, since it only needs to satisfy the non-cryptographic TCP checksum.
    - SSL disregards the package.
    - TCP however will not accept the true packet, because it looks like a double to it.
    - SSL will have to start over.

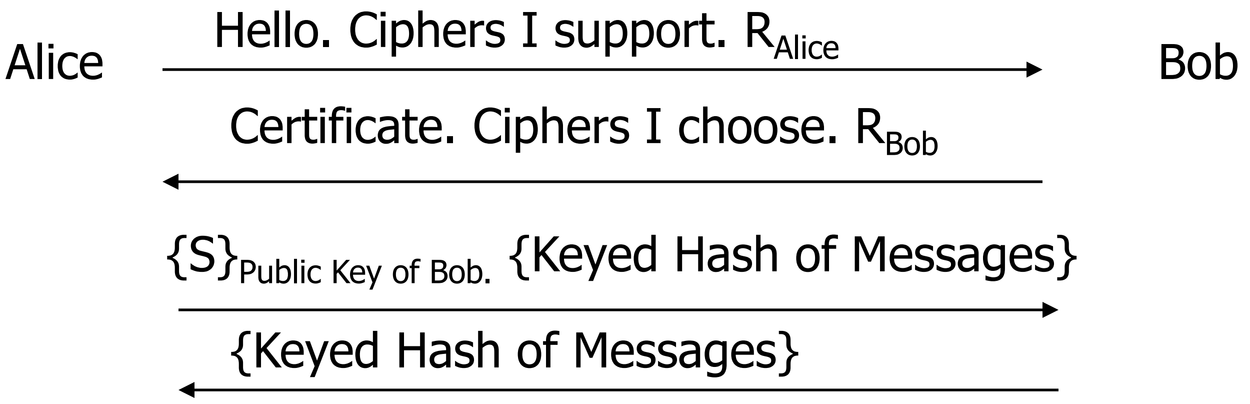
Secure Socket Layer

- Various keys are formed from various random numbers exchanged during the protocol.
- Negotiate crypto-protocols.

Secure Socket Layer

- SSL sessions are long-lived.
- Many SSL connections can be derived from an SSL session.

Secure Socket Layer: Session Connection



Alice

Bob

Hello. Ciphers I support. R_{Alice}

Certificate. Ciphers I choose. R_{Bob}

$\{S\}_{\text{Public Key of Bob.}}$ {Keyed Hash of Messages}

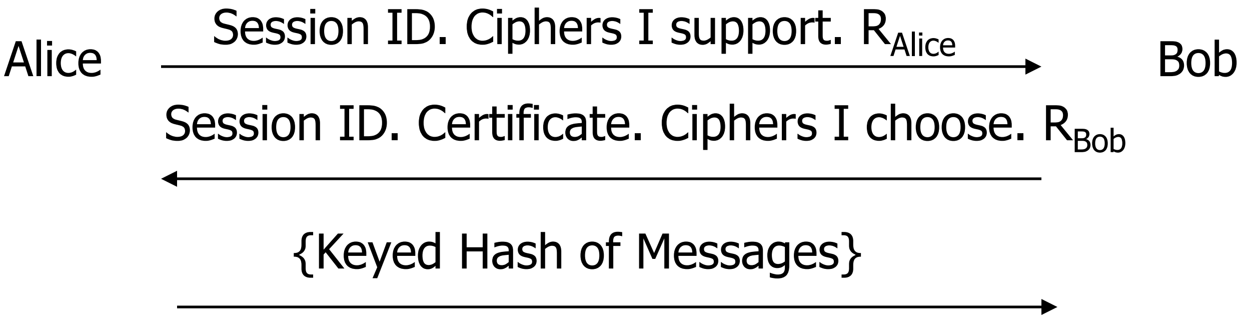
{Keyed Hash of Messages}

S is a random number, the pre-master secret.

K is the master secret, calculated from R_{Alice}, R_{Bob}, S

Secure Socket Layer: Session Resumption

- If Bob wants to have multiple connections per session, he sends in Message 2 a session id.
- If Alice presents in Message 1 a session id, they skip the handshake.
- Alice can still negotiate ciphers with Bob who might have changed policies.


Alice
Session ID. Ciphers I support. R_{Alice}
Bob
Session ID. Certificate. Ciphers I choose. R_{Bob}
{Keyed Hash of Messages}

Secure Socket Layer

- SSL comes deployed with public keys of various trusted organizations.
- User can modify this list.
- User verifies public keys by sending certificate requests to the organizations in the list.

Secure Socket Layer

- SSLv3 upgrades:
  - Protects against the “downgrade attack”
    - Active attacker replaces the initial messages with ones containing weak crypto.
  - Protects against the “truncation attack”
    - Active attacker sends a TCP close (FIN) message.
      - TCP is not protected, so the connection is abnormally terminated without SSL being aware of it.

Secure Shell: SSH

- SSH client and server are applications (running on top of OS).
- SSH consists of a bunch of applications.
- But SSH is not a UNIX shell.

Secure Shell: SSH

- Client contacts server.
- Client and server disclose the SSH versions they support.
- Client and server switch to a packet based protocol.
  - Packet consists of
    - 4B length,
    - 1-8B of random padding,
    - one-byte packet type code,
    - packet payload data,
    - four-byte integrity check field.

Secure Shell: SSH

- Server identifies itself by sending
  - Host key
  - Server key
  - 8 random bytes (use as cookie)
  - List of encryption, compression, authentication methods.
- Both sides compute a 128b session identifier.

Secure Shell: SSH

- When the client receives the host key, the client looks into the **known host database.**
- If the host key matches the one in the database then the client proceeds.
- If the host is in the database but with a different key, then the client queries the user.
- Otherwise, the client warns the user and proposes to add host and key to the known host database.

Secure Shell: SSH

- Client randomly generates a session key.
  - Clients sends the session key encrypted with the server key and then with the host’s public key.
  - Together with the choice of crypto-suites.
- Both sides now use the session key for encryption.
  - Server sends confirmation message encrypted with the session key.
  - This proves the server’s authenticity to the client.

WAP:
Wireless Application Protocol

- Wireless information and telephony services on wireless phones