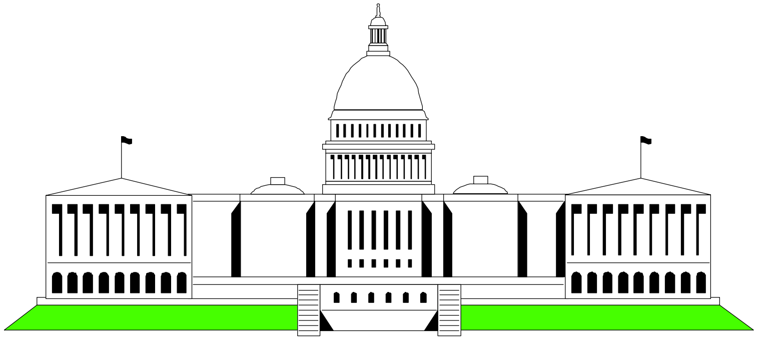
OS/390 Firewall Technology Overview



Mary Sweat
E - Mail: sweatm@us.ibm.com

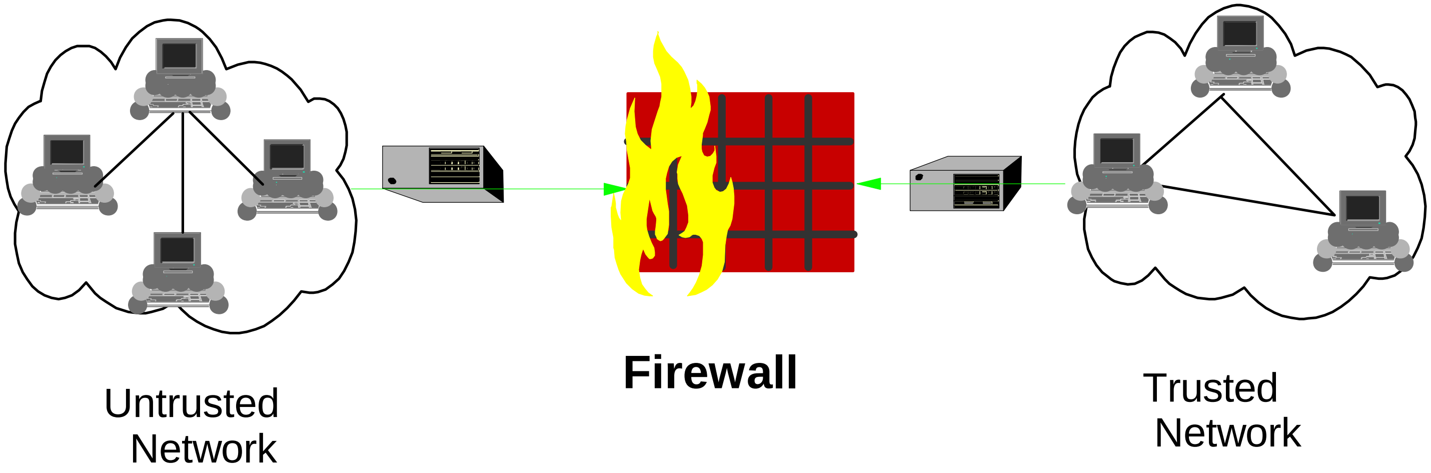
Washington System Center

Agenda

- Introduction
- OS/390 Firewall
  - Hardware requirements
  - Software requirements
  - OS/390 Firewall Features
- IP Filters
- Virtual Private Networks (tunnels)
  - IPSec
    - Authentication
    - Encryption
  - tunnel types and modes
  - IPSec vs SSL
  - benefits

What is a Firewall

- A solution that provides controlled access between a private (trusted) network, and an untrusted network such as the Internet
- A tool for enforcing your network security policy



Why Use a Firewall?

??????????????????????????????????

? ?

- Limit access by persons within the secure network to selected resources in the non-secure network

? ?

- Reduce network traffic outside the secure network

? ?

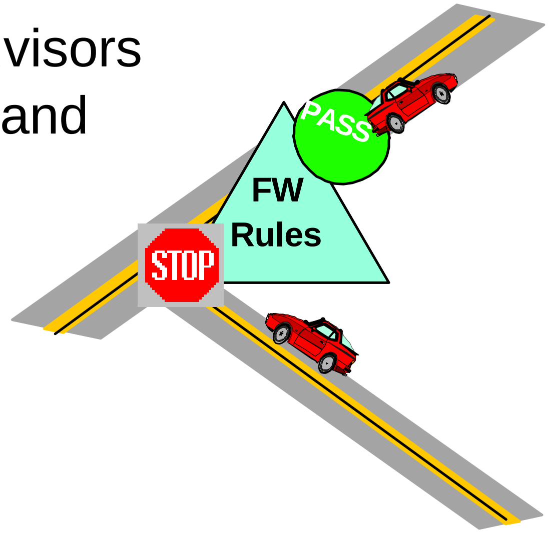
- Improve performance within the secure network

? ?

??????????????????????????????????

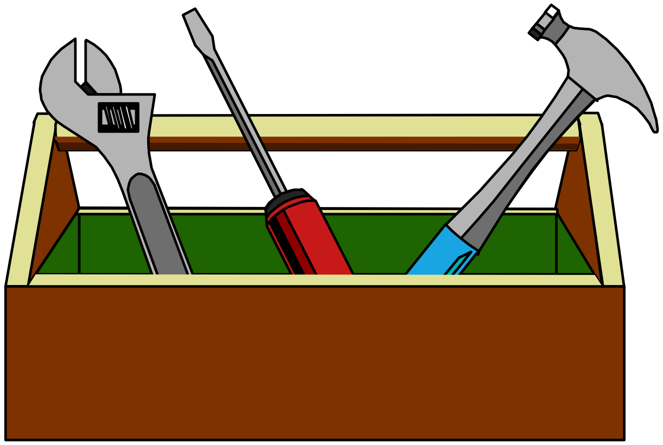
Basic Design Decisions in a Firewall

- Ensure physical security
- Configure the firewall by disallowing everything and then proceed by enabling those services defined in the security policy
  - support only required applications and remove or disable others
- Security policy that defines how a firewall should function
  - created in cooperation with the security group/advisors
  - what type of traffic is allowed through the firewall and under what conditions
- Audibility



Firewall Technologies Tools

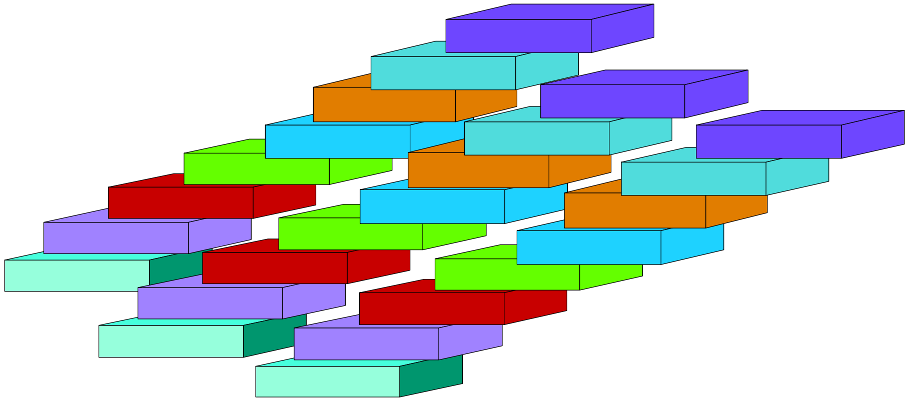
- Included with the OS/390 Security Server
  - Configuration Client (GUI)
  - Configuration Commands
  - Logging Server
  - Proxy FTP server
  - Socks Server
  - Real Audio Support
  - Internet Security Association Key Management Protocol (ISAKMP) Server



- Included with the eNetwork Communications Server for OS/390
  - Network Address Translation (NAT)
  - IP Filters
  - IP Tunnels (IPSec or Virtual Private Network)

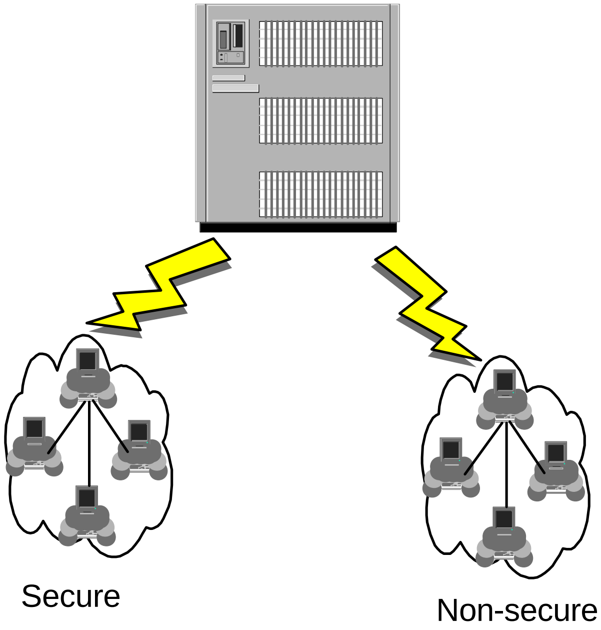
Multi-Stack Support

- 8 Firewalls can now run simultaneously within an LPAR
- Ability to associate firewall functions with particular stack
- Each firewall could have a potentially different configuration



Firewall Hardware Requirements

- Any communication hardware interface supported by the TCP/IP protocol stack to make the network connections
  - OSA, 3172, CTC, XCF, etc.

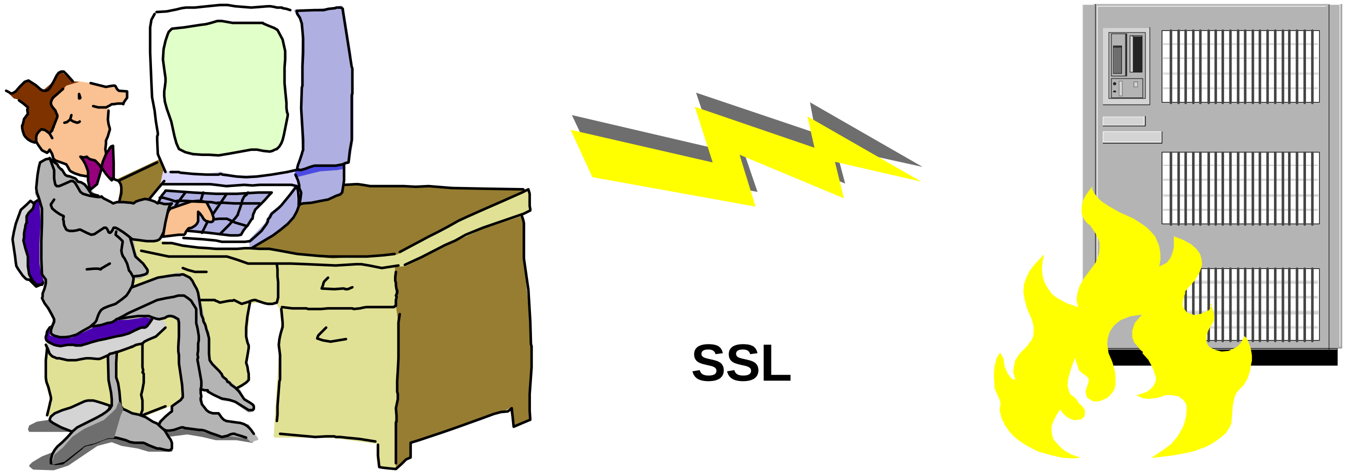


- At least two network interfaces;
  - one network interface connects the secure, internal network that the firewall protects
  - the other network interface connects to the nonsecure, outside network or internet
- Crypto Coprocessor
  - this is optional requirement as the OS/390 firewall can use software encryption (RSA BSAFE)
  - used with Integrated Cryptographic Service Facility (ICSF)

Software Requirements

- OS/390 Security Server (RACF)
- OS/390 eNetwork Communications Server
- OS/390 Unix services (OpenEdition)
- OS/390 C/C++ Collection Cl. Lib.
- OS/390 System Secure Socket Layer (System SSL)
- Open Cryptographic Services Facility (OCSF)
- Security Server Open Cryptographic Enhanced Plug-ins (OCEP)

Graphical User Interface



GUI Client

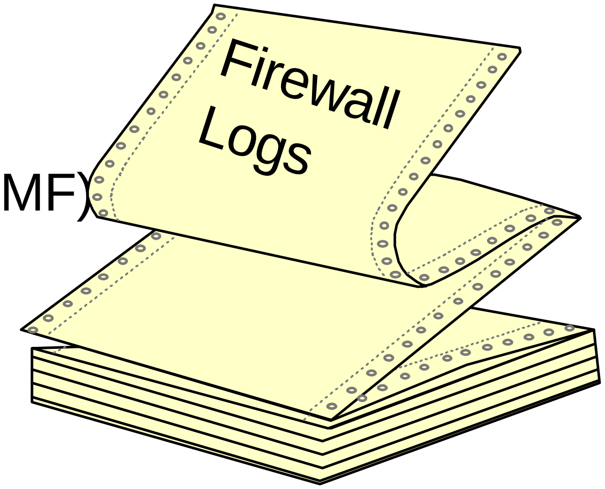
SSL

Configuration Server (OS/390)

- Written in JAVA
- Installs / runs on Windows 95/NT & AIX
  - AIX
    - Java 1.1.4 or higher
    - AIX 4.2 or higher
    - Netscape 3.0.1
  - Windows 95 or Windows NT
    - web browser with Java and frames support
    - zip tool that handles long file names

Logging Server

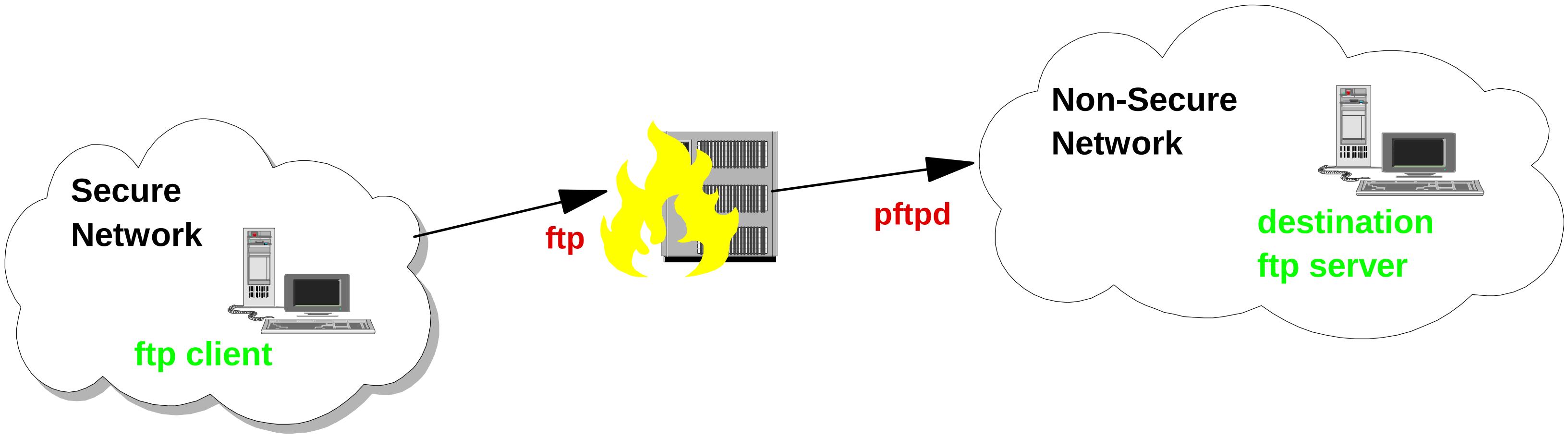
- Captures activity and provides options for handling activity based on origin and type of event
- Log events based on three factors:
  - facility (origin)
  - priority (severity)
  - action to be taken with the event
- Records events;
  - in HFS log files
  - send to other machines
  - send to other users on same machine
  - record in OS/390 System Management Facility (SMF)



FTP Proxy Support

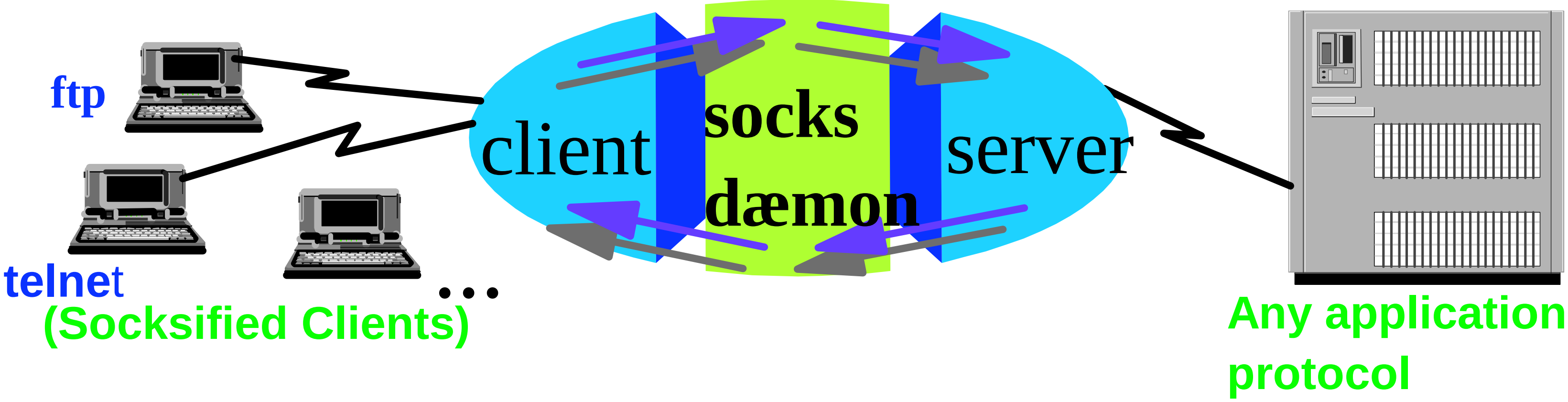
- OS/390 Firewall Technologies supply an FTP proxy server (**pftpd**)
  - access controlled on a user-by-user basis
    - to go out of the secure network
    - to come in from the non-secure world
  - local ***ftp*** commands disabled on the firewall

- Users **ftp** to the firewall and with valid authorizations, **pftpd** contacts FTP server outside the secure network



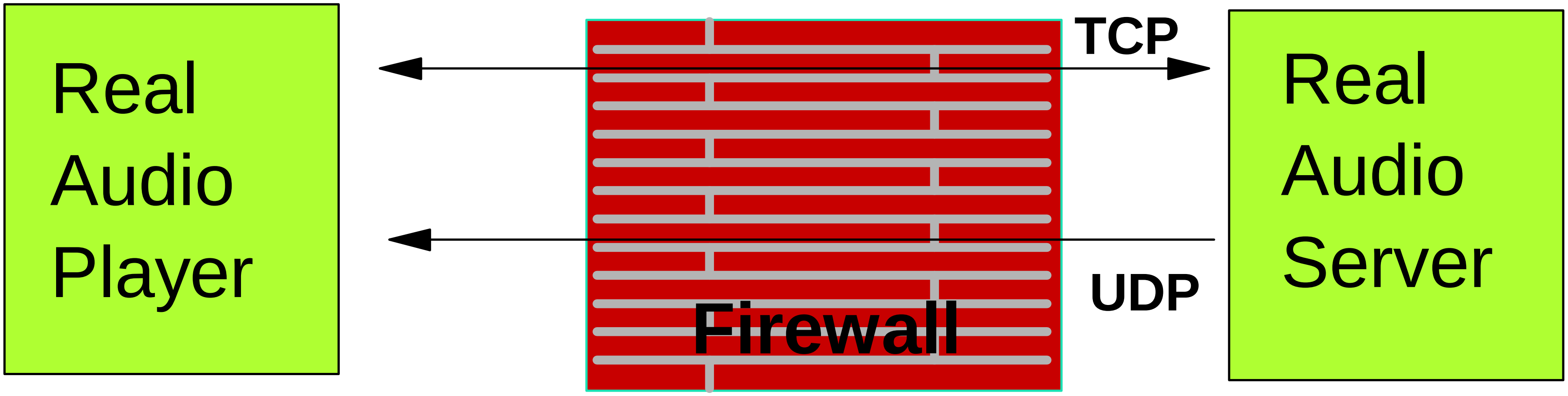
Socks

- A socks dæmon sits between the client and destination server
  - socks dæmon is generic
    - can handle traffic for multiple, different applications
- Socks replaces the IP address of the user with the address of the firewall



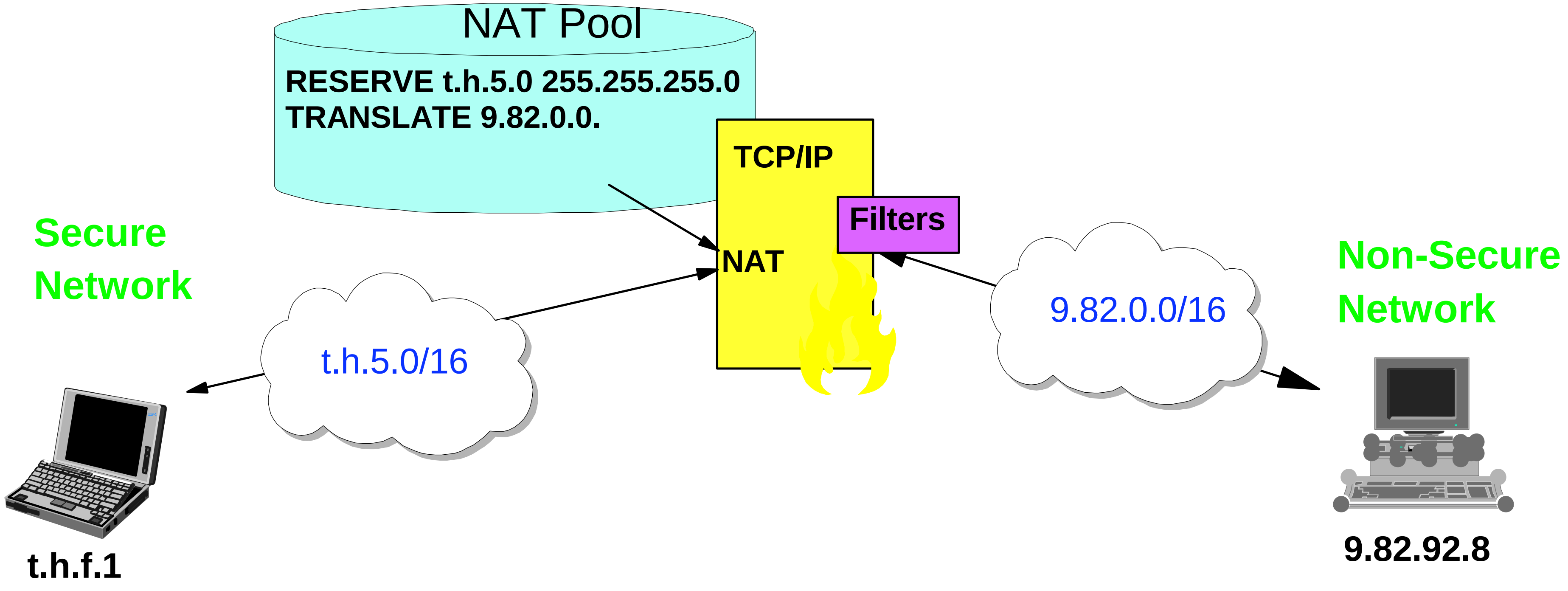
Real Audio Support

- Supports live and on-demand audio from the Internet
  - Special protocol developed by Progressive Networks
- OS/390 Firewall monitors and identifies RealAudio TCP connections
  - dynamic filter rule for a UDP packet is defined when a RealAudio connection is identified
  - rule is removed when the RealAudio TCP connection is closed



Network Address Translation (NAT)

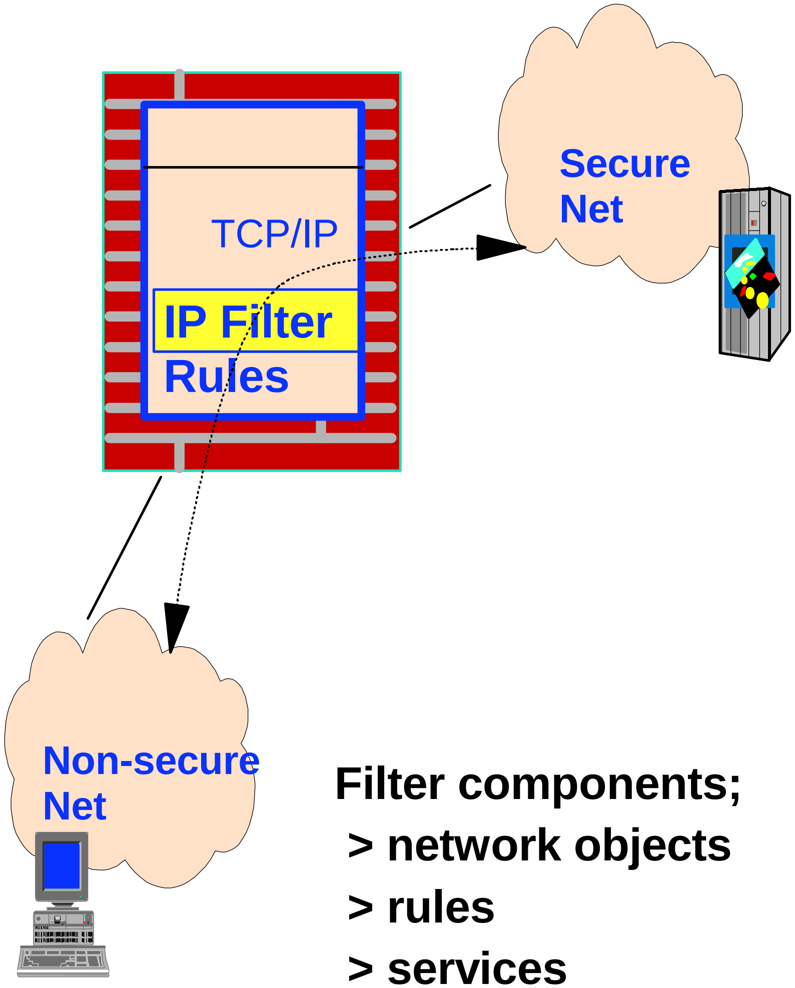
- Network Address Translation provides a translation from an internal (secure) IP address to an temporary external registered address



IP Filters

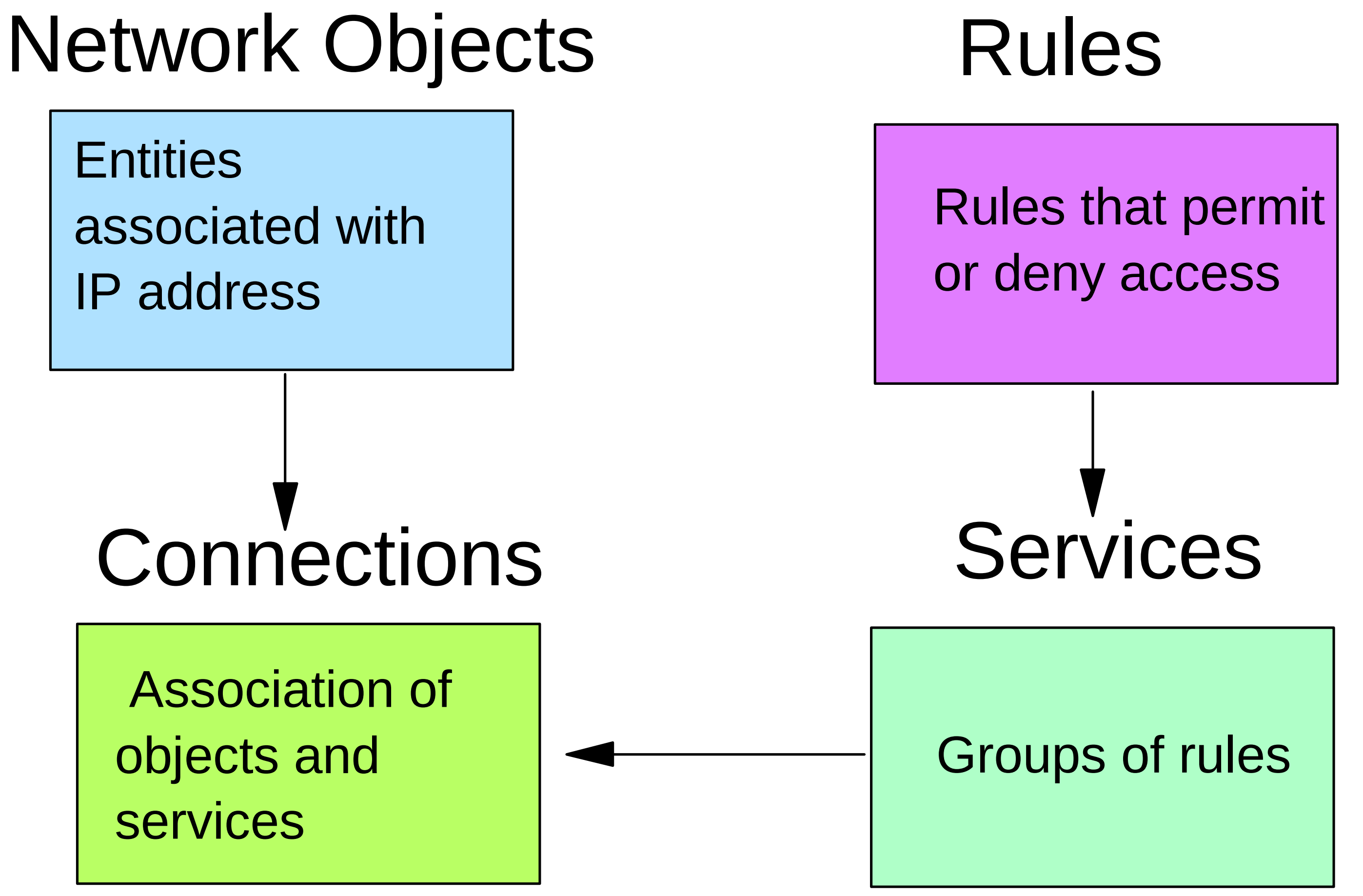
- Basic control feature in firewalls
- Works at the IP layer of TCP/IP
- Determines what traffic is allowed to flow through
- Filters on;

-- source and destination IP address & mask
-- source and destination port
-- direction of the data flow
-- IP protocol
-- type of interface (secure or nonsecure)
-- date/time



Filter components;
> network objects
> rules
> services
> connections

IP Filter Components



Network Objects

Entities associated with IP address

Rules

Rules that permit or deny access

Connections

Association of objects and services

Services

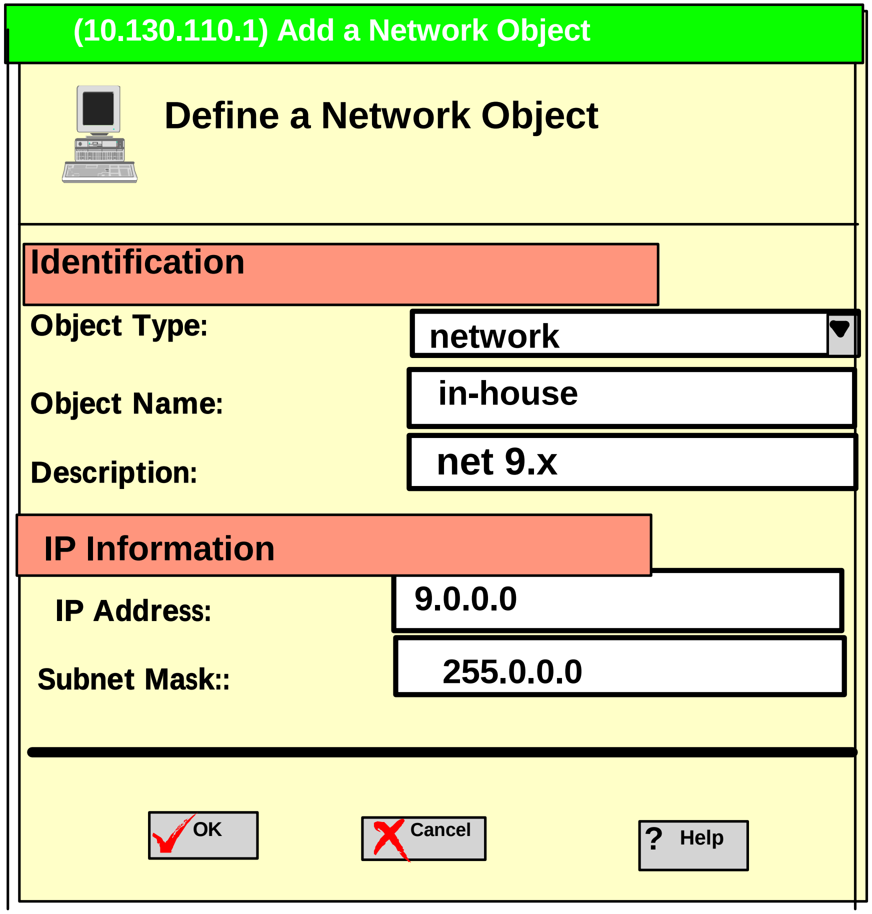
Groups of rules

Network Objects

- Represent various hosts and entities
- Defined with "**fwnwobj**" command or via client GUI

```
fwnwobj cmd=add name=in-house
               type= network desc='net 9.x'
               addr=9.0.0.0
               mask=255.0.0.0

fwnwobj cmd=add name=G-fw
               type= host desc='fw nonsec'
               addr=9.82.94.10
               mask=255.255.255.255
```



(10.130.110.1) Add a Network Object

Define a Network Object

Identification

Object Type: network

Object Name: in-house

Description: net 9.x

IP Information

IP Address: 9.0.0.0

Subnet Mask:: 255.0.0.0

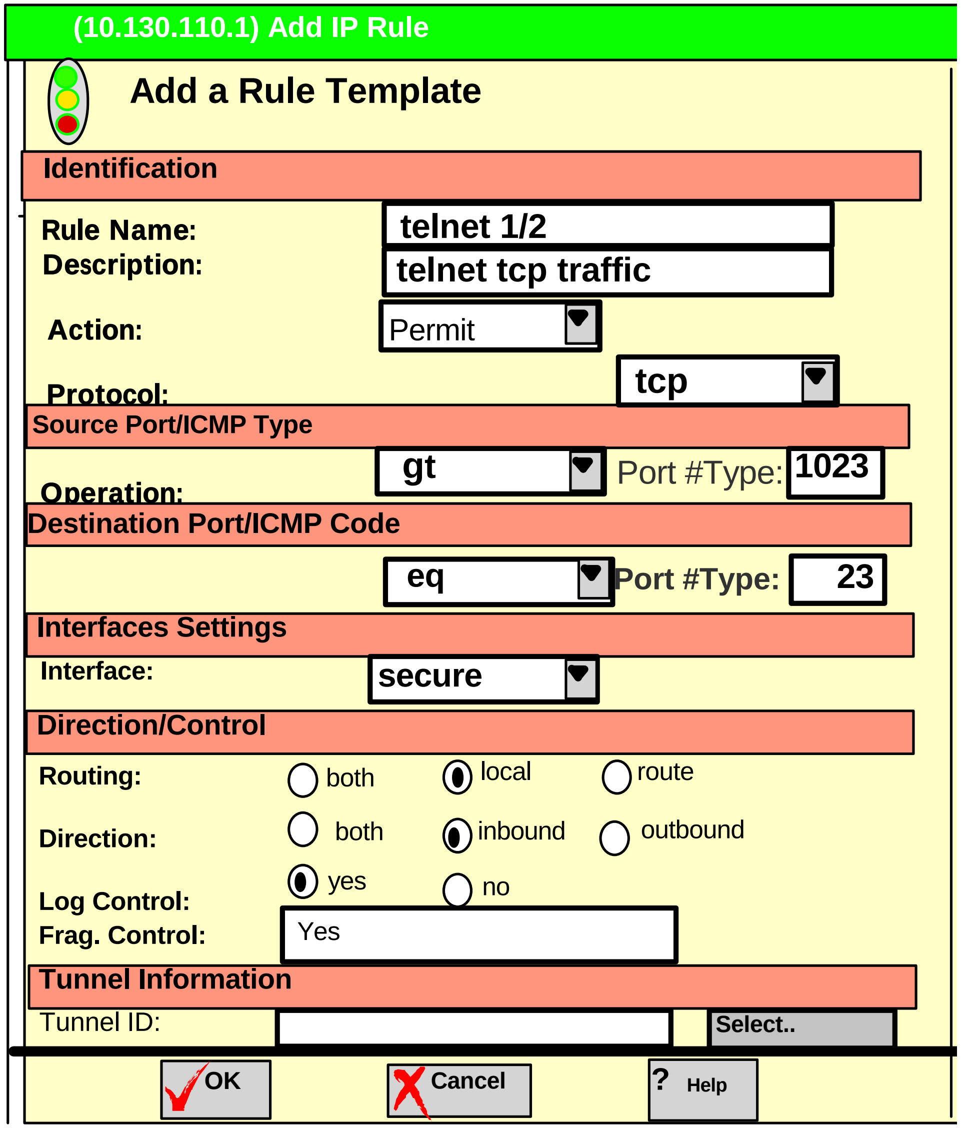
1.Enter object types
2.Enter object name
3.Fill in the description
4.Enter a dotted-decimal IP address for this object
5.Enter a subnet mask for this address
6.Click OK

Rules

- Instructions to permit or deny packets
- Defined with "**fwfrule**" command or via GUI

```
fwfrule cmd=add name='telnet 1/2'
          desc='telnet tcp traffic'
          type=permit protocol=tcp
          srcopcode=gt srcport=1023
          destopcode=eq destpor=23
          interface=secure routing=local
          direction=inbound log=yes

fwfrule cmd=add name='telnet 2/2'
          desc='telnet tcp/ack traffic'
          type=permit protocol=tcp/ack
          srcopcode=eq srcport=23
          destopcode=gt destport=1023
          interface=secure routing=local
          direction=outbound log=yes
```



(10.130.110.1) Add IP Rule

Add a Rule Template

Identification

Rule Name: telnet 1/2

Description: telnet tcp traffic

Action: Permit

Protocol: tcp

Source Port/ICMP Type

Operation: gt Port #Type: 1023

Destination Port/ICMP Code

eq Port #Type: 23

Interfaces Settings

Interface: secure

Direction/Control

Routing: ○ both ● local ○ route

Direction: ○ both ● inbound ○ outbound

Log Control: ● yes ○ no

Frag. Control: Yes

Tunnel Information

Tunnel ID:

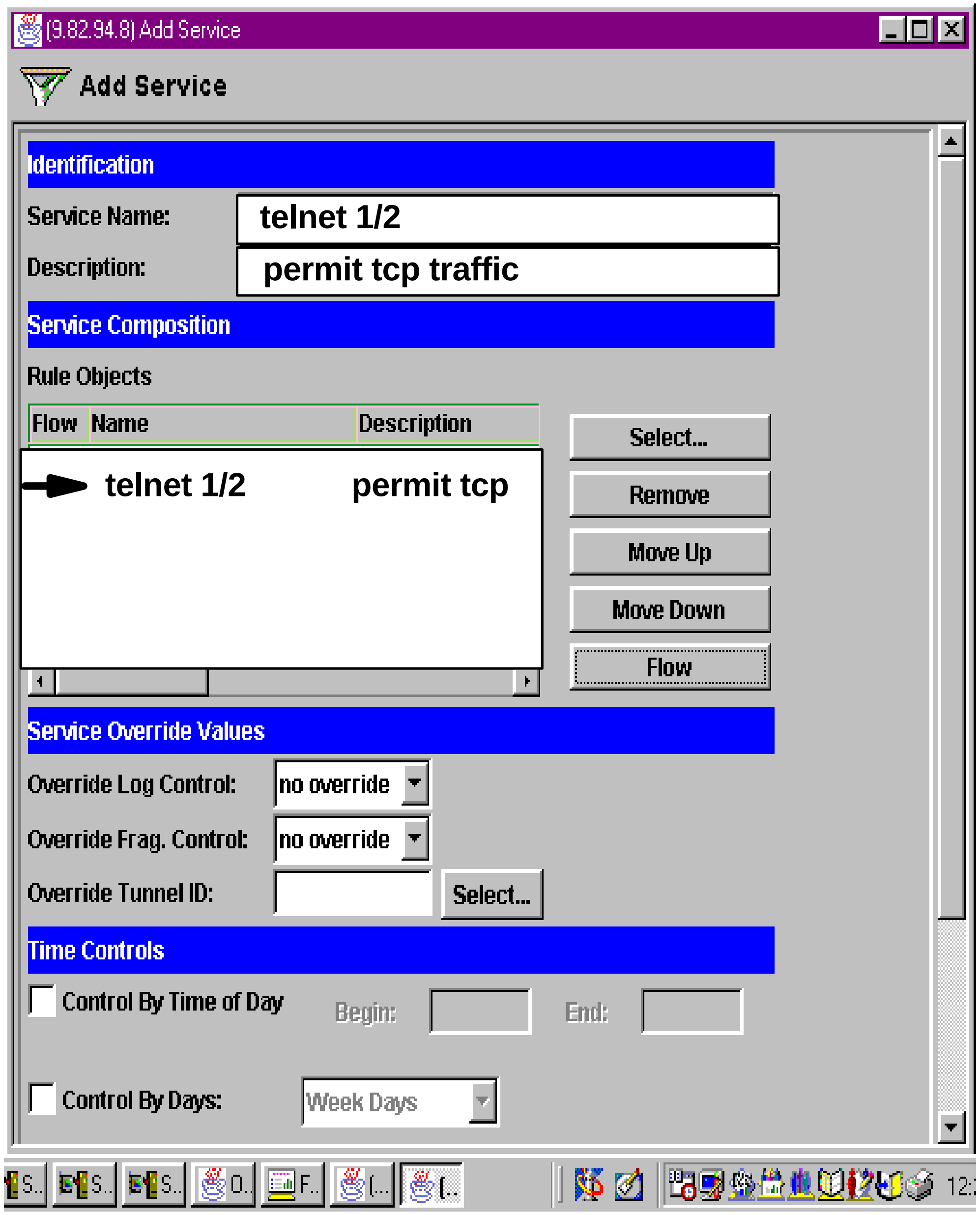
Services

- Groups of rules which instructs the firewall to permit or deny access
- Defined with "**fwservice**" command

```
fwservice cmd=create
             name=telnet 1/2
             desc='permit tcp traffic'
             rulelist=13/f

  fwservice cmd=create
             name=telnet 2/2
             desc='tcp response'
             rulelist=12/f
```

name = name you assign to this service
desc = description that you give this service rule
rulelist = list of rules and direction to add to
this service (forward (f) or backward (b)



(9.82.94.8) Add Service

Add Service

Identification

Service Name: telnet 1/2

Description: permit tcp traffic

Service Composition

Rule Objects

Flow	Name	Description
→	telnet 1/2	permit tcp

Service Override Values

Override Log Control: no override

Override Frag. Control: no override

Override Tunnel ID:

Time Controls

□ Control By Time of Day Begin: End:

□ Control By Days: Week Days

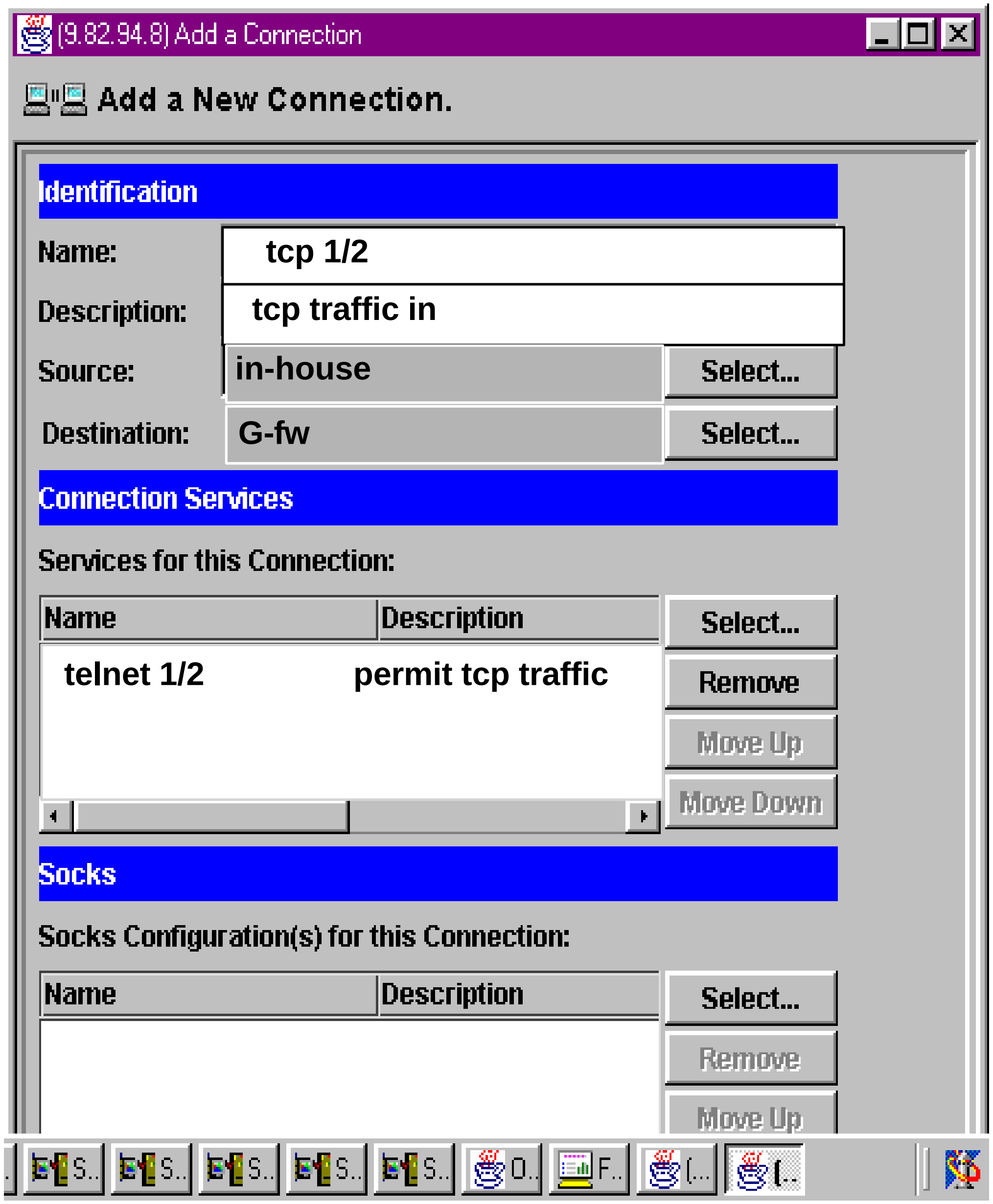
Connections

- Associate network objects with services to define types of communications allowed between endpoints

- Defined with "**fwconns**" command

```
fwconns cmd=create
                name='tcp 1/2'
                desc='tcp traffic in'
                source=in-house
                destination=G-fw
                servicelist=18

fwconns cmd=create
               name='tcp 2/2'
               desc='tcp traffic out'
               source=G-fw
               destination=in-house
               servicelist=19
```



(9.82.94.8) Add a Connection

Add a New Connection.

Identification

Name: tcp 1/2

Description: tcp traffic in

Source: in-house

Destination: G-fw

Connection Services

Services for this Connection:

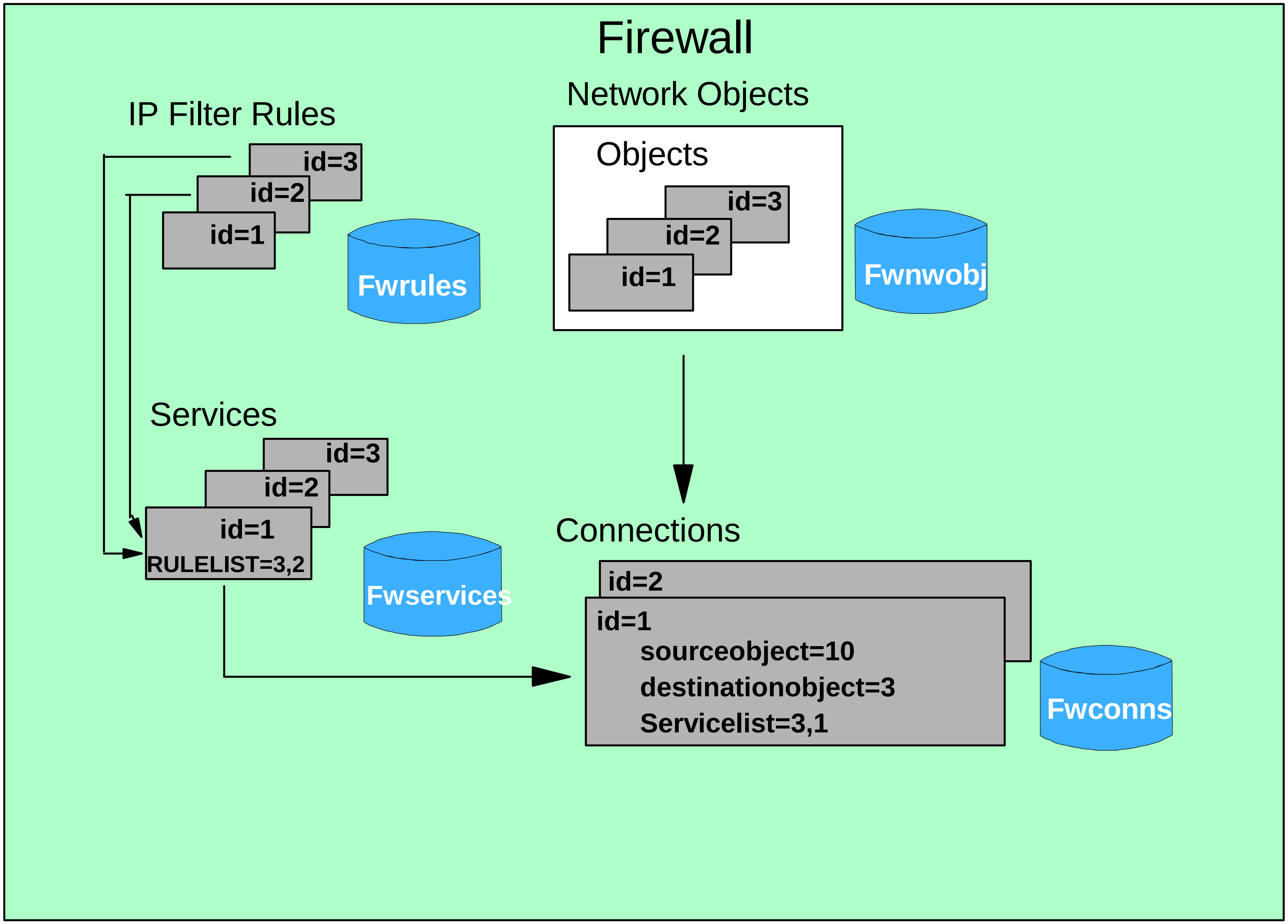
Name	Description
telnet 1/2	permit tcp traffic

Socks

Socks Configuration(s) for this Connection:

Name	Description

Configuration Overview



Refresh

```
fwnwobj cmd=add name='in-house'
              type=network desc='net 9.x'
              addr=9.0.0.0  mask=255..0.0.0
```

```
fwfrule cmd=add name='telnet 1/2'
           desc='telnet tcp traffic'
           type=permit protocol=tcp
           srcopcode=gt srcport=1023
           destopcode=eq destport=23
            interface=secure routing=local
           direction=inbound log=yes
```

```
fwservice cmd=create name='telnet 1/2'
              desc='permit tcp traffic'
              rulelist=13/f,12/b
```

```
fwnwobj cmd=add name=G-fw  type=host
             desc='fw nonsecure'
             addr=9.82.94.10 mask=255.255.255.255
```

```
fwfrule cmd=add name='telnet 2/2'
           desc='telnet tcp/ack traffic'
           type=permit protocol=tcp/ack
           srcopcode=eq srcport=23
           destopcode=gt destport=1023
           interface=secure routing=local
           direction=outbound log=yes
```

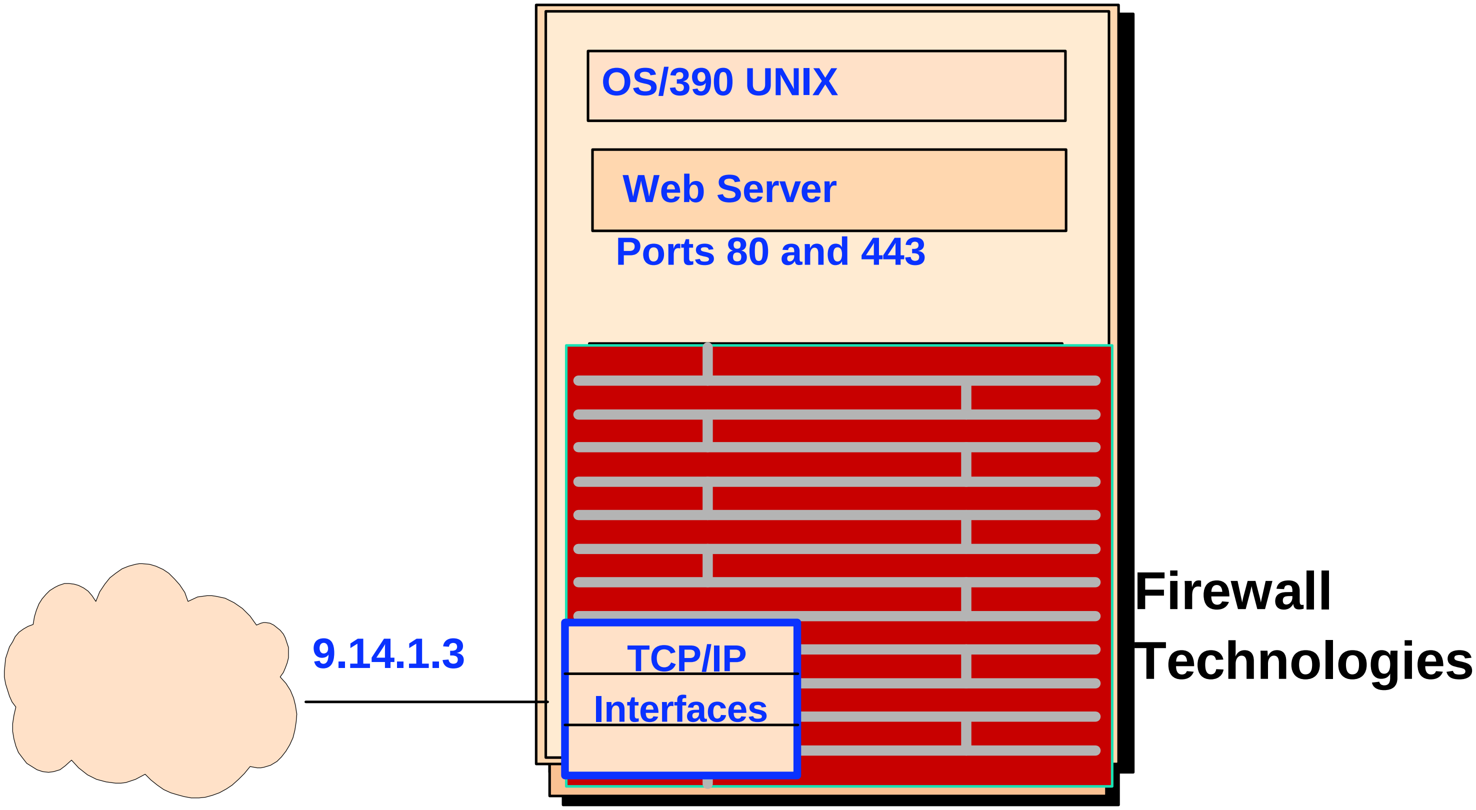
```
fwconns  cmd=create name='tcp 1/2'
           desc='tcp traffic in'
           source=in-house destination=G-fw
           servicelist=18
```

FWFILTER cmd=update

RESULTS: **fwfilter cmd=list**

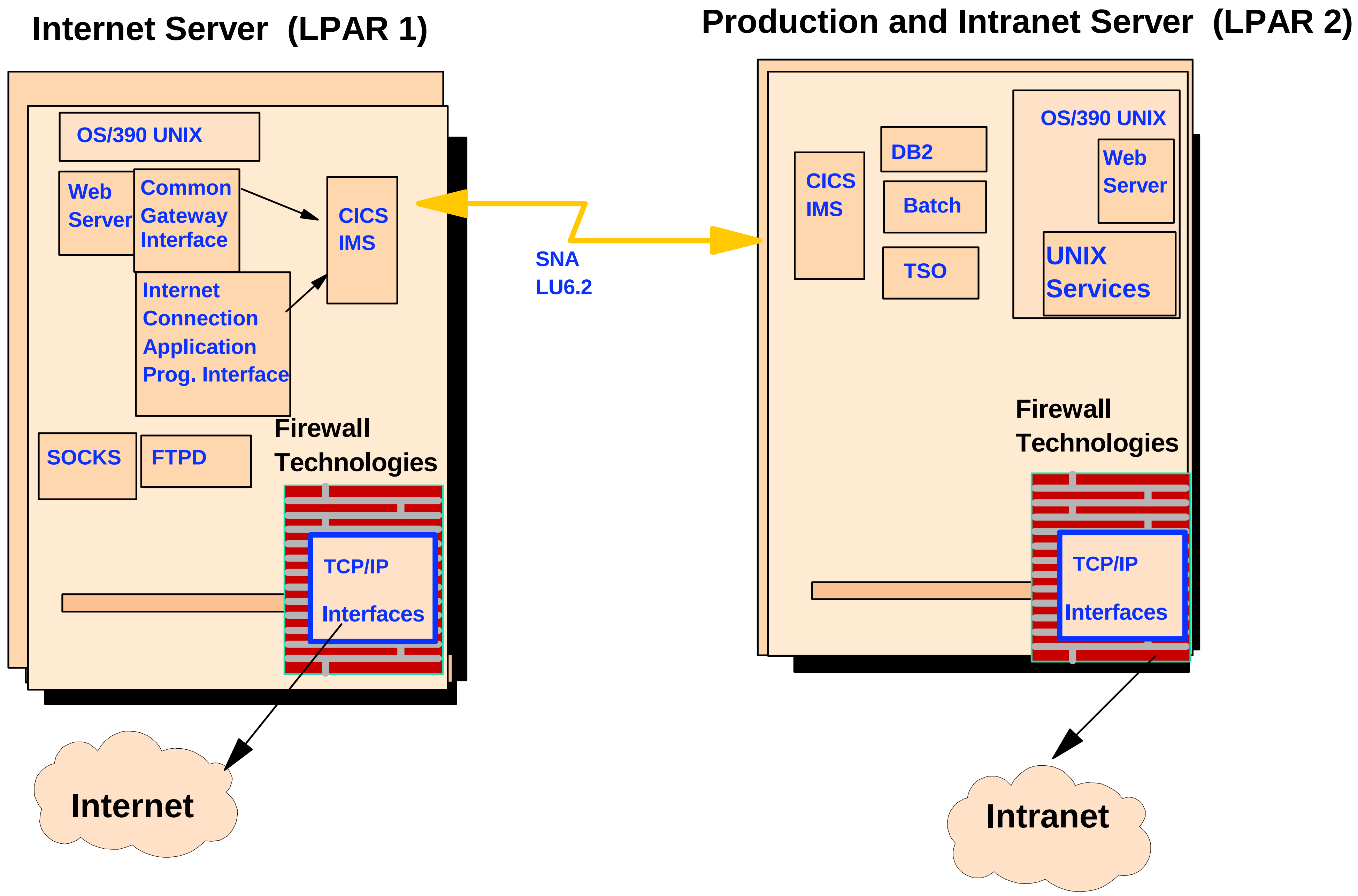
```
#Service: Telnet 1/2
#Description: Permit tcp traffic
 permit 9.0.0.0  255.0.0.0  9.82.94.10  255.255.255.255 tcp gt 1023 eq 23 secure local inbound  l=y
permit 9.82.94.10 255.255.255.255 9.0.0.0 255.0.0.0 tcp/ack eq  23 gt 1023 secure local outbound l=y
```

Firewall Example (one)

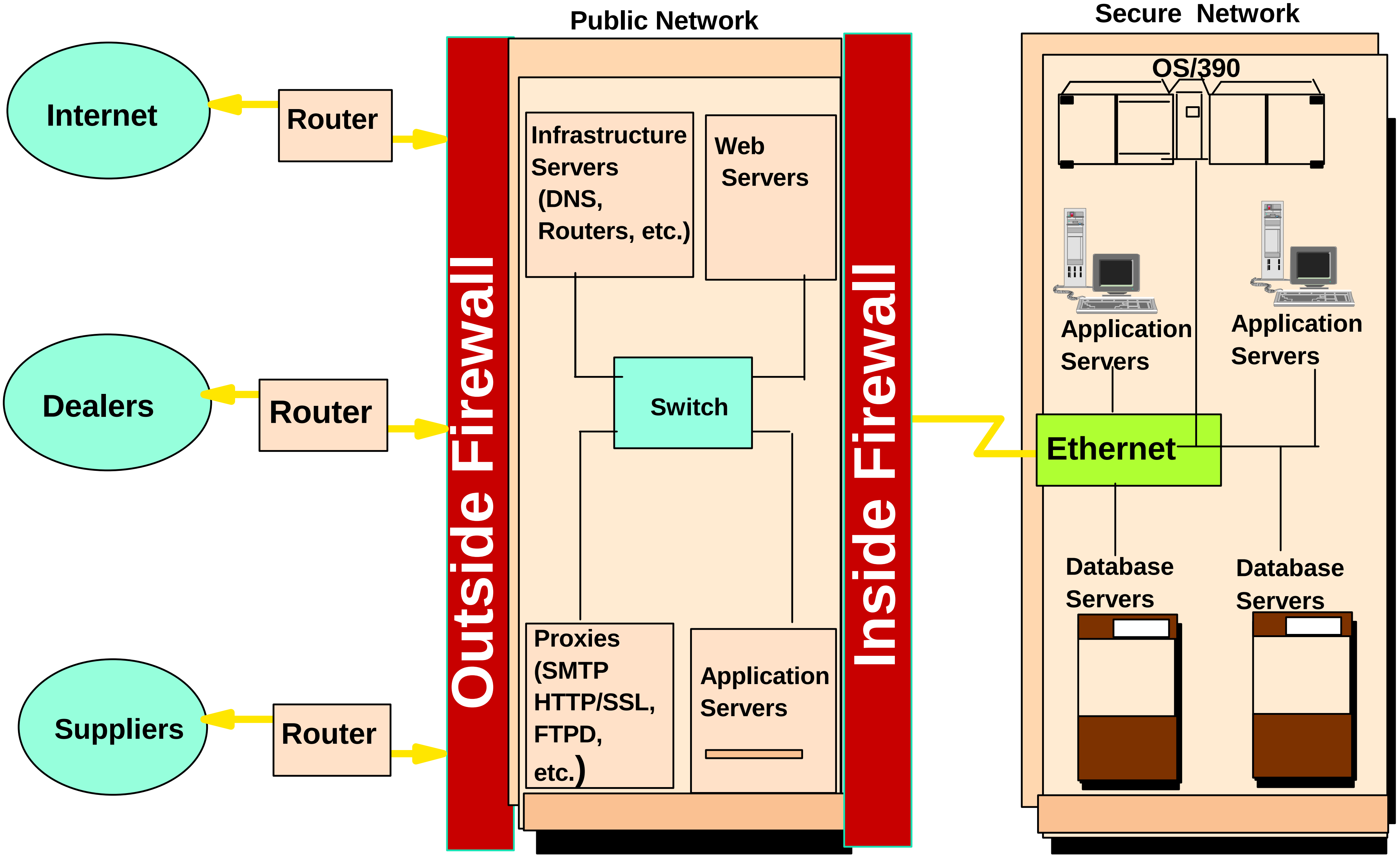


Non-secure Domain:
wscfw.ibm.com

Firewall Example (two)

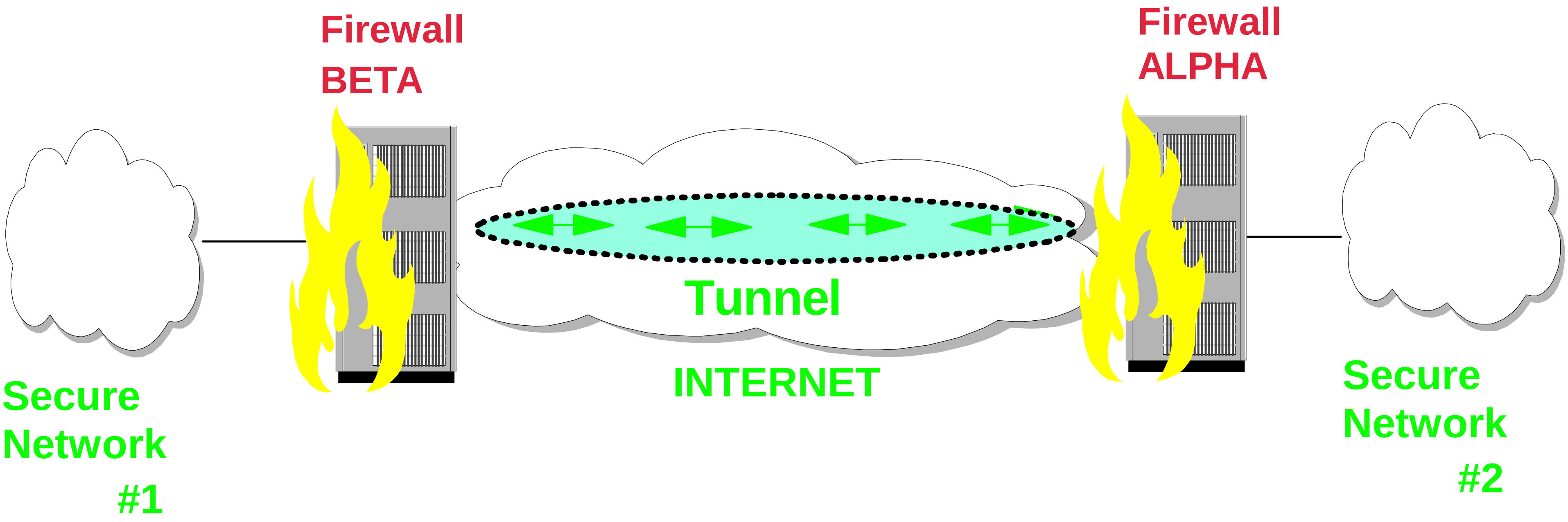


Firewall Example (three)



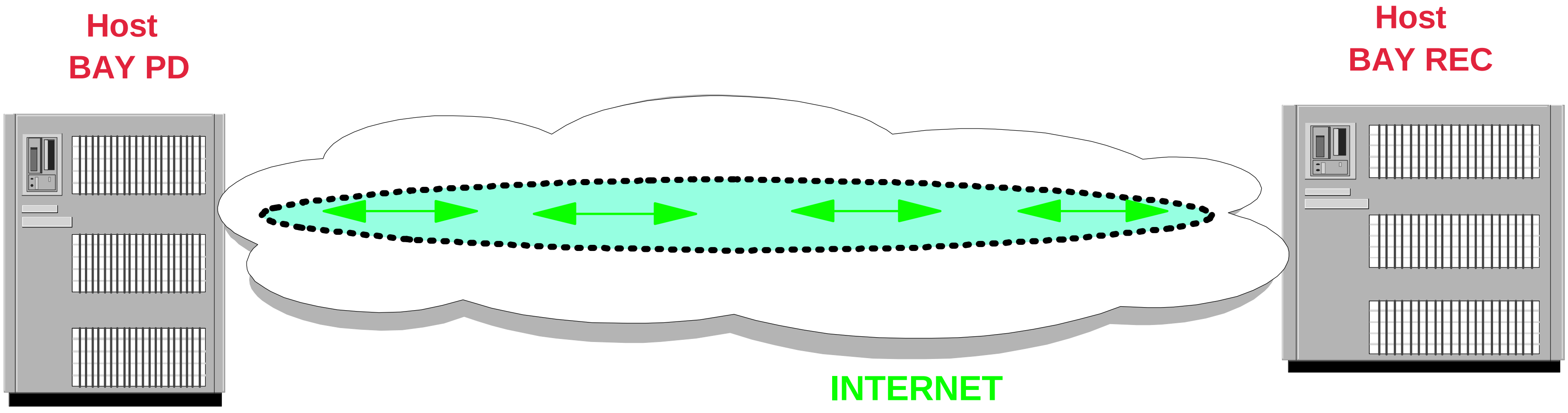
Virtual Private Networks

- Virtual Private Networking (VPN) allows secure communications between remote sites over a public network like the internet
- Secures data traffic at the IP layer
  - secure traffic for all applications, without modifications to applications



Secure Tunnels

- Virtual tunnels created between two hosts
  - uses IPSec protocol not TCP or UDP
    - referred to as a Virtual Private Network
    - user specifies method of encapsulation for IP traffic
    - provides integrity, privacy and authentication



Tunnel Types

- Manual, keys are static
  - encryption & authentication keys are the same for the life of the tunnel
  - must be manually updated
  - has the widest choice of header and encryption options
- Dynamic tunnels (ISAKMP), keys are dynamic
  - based on Internet Security Association and Key Management Protocol (ISAKMP)
  - defines message formats and flows that will allow two devices to dynamically agree to the information shared between them
  - negotiate and refresh security parameters and exchange keys securely

IP Security

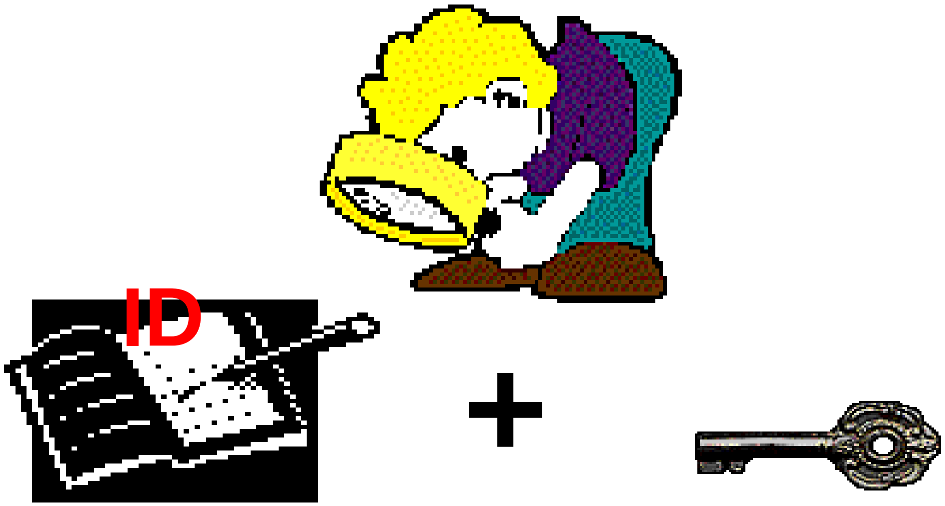
- IPSec is a security protocol used as a industry standard in the area of VPNs
  - defined by Internet Engineering Task Force (IETF)
    - multiple Internet drafts and RFCs
- Basic rules to apply to attributes and encryption keys used by IPSec known as Security Association (SA)
- Uses protocols to secure data
  - Authentication Header (AH) - verifies identity of a host or tunnel end point
  - Encapsulating Security Payload (ESP) - process to ensure data can not be viewed by unauthorized personnel
- Provides specific operation modes
- Uses other protocols to dynamically generate cryptograhic keys

Security Association (SA)

- Defines basic concepts required to agree to attributes and encryption keys used by IPSec
  - information shared between two devices that enables them to protect IP traffic
    - identifies parameters/functions needed to create IPSec packets
    - destination ID/IP address
    - type of security service used (AH or ESP)
    - keys used by cryptographic operations
    - tunnel mode
    - Security Parameter Index (value used in identifying an SA)

IPSec Authentication & Integrity

- Uses IP Authentication Header (AH) protocol
  - proof of the sender's identity and data integrity
    - uses cryptographic hash function with a secret key
    - produces unique digest
    - receiver de-capsulates using same function and key
    - verifies data and sender's key
    - discards data if key is not valid or data has been altered



IPSec Encryption

- Uses IP Encapsulating Security Payload (ESP) protocol
  - provides integrity, authentication and encryption to IP packets
    - uses certain algorithms and keys to produce cyphertext
    - same algorithms and keys used by sender and receiver
    - knows as symmetric encryption algorithms

Tunnel Modes

- Operational Modes
  - transport - only protects the transport-layer packet (such as TCP or a UDP) inside an IP packet
    - data is protected, source and destination addresses remain unchanged
  - tunnel - protects entire IP packet
    - data as well as source and destination addresses are protected

How IPSec Compares to SSL

- Both are similar:
  - provides client and server authentication
  - provides data authentication and secrecy (encryption)
- SSL is implemented at the transport level, IPSec is implemented at the Internet Layer
- SSL does not protect IP headers, IPSec does
- SSL does not protect UDP traffic, IPSec does
- Applications require modification to be made SSL aware, IPSec is transparent to applications
- SSL provides application to application security, IPSec provides device to device security

Internet Security Association Key Management Protocol Server

- Server uses ISAKMP/OAKLEY protocol
  - supports automatic generation of tunnel definitions
- Provides a more automated alternative to manual Virtual Private Networks (VPNs)
  - dynamically establish VPNs
  - negotiate VPN attributes
  - dynamically manage VPN encryption keys
- Offers a method of exchanging encryption keys in a secure manner

Internet Security Association Key Management Protocol

- Enables dynamic SAs and key management
  - enables two devices to dynamically agree to the setup of a tunnel
- Creates common framework for handling SAs
  - definition
  - negotiating
  - modifying
  - deleting
  - authenticating peers
  - exchanging keys
- Key management protocol
- Implemented at the application layer
  - communicates useing UDP port 500

Tunnel Benefits

- Creates a secure private connection through what is basically a private tunnel
- VPNs securely convey information across the Internet connecting remote users, branch offices, and business partners/suppliers into an extended corporate network
- Access to the Internet via service providers is more cost effective
- Eliminate need for
  - expensive leased lines
  - long-distance calls
  - toll-free telephone numbers

Why Dynamic Tunnels

- Ensure interoperability
  - ensure businesses can communicate regardless of vendors VPN
- Address security concerns with key management
  - offers secure manner for exchanging keys
- Ease of use for environments managing numerous VPNs

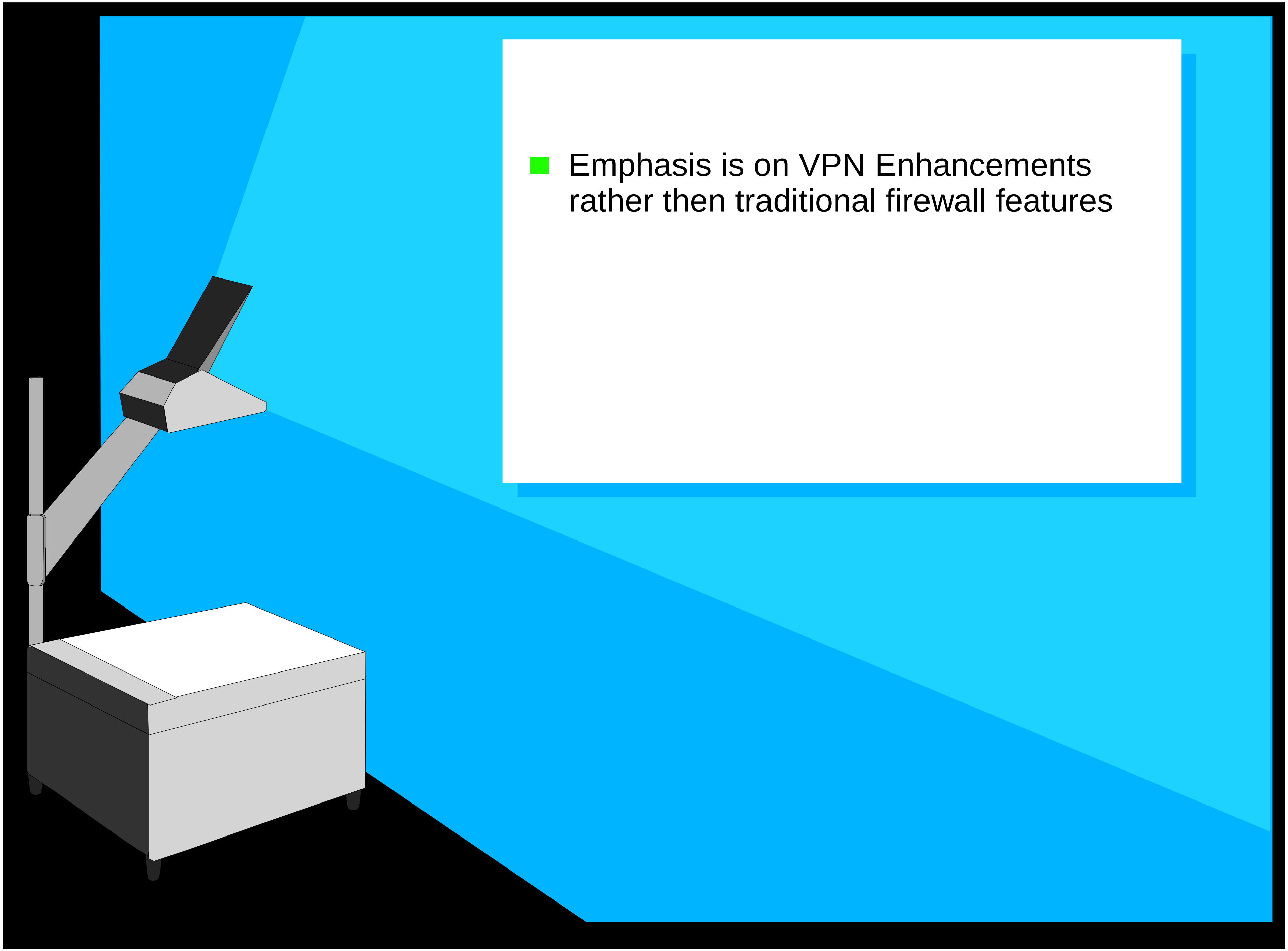
IPSec Standard References

- Request for Commends (RFCs)
  - located at www.ietf.org
    - 1825 Security Architecture for Internet Protocol
    - 1826 IP Authentication Header
    - 1827 IP Encapsulating Security Payload
    - 1828 IP Authentication Using Keyed MD5
    - 1829 The ESP DES_CBC Transform
    - 2401 Security Architecture for Internet Protocol
    - 2402 IP Authentication Header
    - 2403 HMAC-MD5-96 within ESP and AH
    - 2404 HMAC-SHA-1-96 within ESP and AH
    - 2405 The ESP DES-CBC Cipher Algorithm With Explicit IV
    - 2406 IP Encapsulating Security Payload
    - 2407 Internet IP Domain of Interpretation for ISAKMP
    - 2408 Internet Security Association and Key Management Protocol (ISAKMP)
    - 2409 Internet Key Exchange
    - 2410 NULL Encryption Algorithm and Its Use With IPSec

References

- OS/390 Security Server 1999 Updates Technical Presentation Guide (SG24-5627-00)
  - located at www.redbooks.ibm.com
- Stay Cool on OS/390; Installing Firewall Technology (SG24-2046)
- Security in OS/390-based TCP/IP Network (SG24-5383)

Future Releases



- Emphasis is on VPN Enhancements rather then traditional firewall features

QUESTIONS