OS/390 Security Server: Getting Started Using the Firewall

SHARE Session 1745
July 27, 2000
Boston, Massachusetts

Dave Wierbowski
OS/390 Firewall Technologies Development
Endicott, New York

(607) 752-6739
wierbows@us.ibm.com

Trademarks

The following are trademarks of International Business Machines Corporation:

- CICS
- DB2
- IBM
- IMS
- OS/390
- AIX

The following are trademarks or registered trademarks of other companies or institutions:

Windows NT, 95, 98

Abstract

OS/390 Security Server - Getting Started Using the Firewall Session 1744 introduced you to the Firewall Technologies available on OS/390. This session will provide you with a high level understanding of what you need to do in order to use these technologies. It will start off with a brief overview of the steps required to configure the basic functions of the firewall on OS/390. The management of the Firewall Technologies' servers will then be discussed. You will be introduced to the OS/390 Firewall Technologies' GUI and commands. You will be walked through an example of defining a set of filter rules using both the GUIand the command line. Time permitting, the configuration of other technologies will be described at a high level.

Session Objectives

- Provide a brief overview of post-installation configurations steps
- Discuss the management of the OS/390 Firewall daemons
- Introduce the OS/390 Firewall Configuration GUI and Server
- Introduce the OS/390 Firewall Commands
- Walk through an example of using IP packet filtering

★ This presentation is based on R8

General Configuration Considerations

Unix System Services (USS) Considerations

- Check BPXPRMxx in SYS1.PARMLIB
  - **Examine the following values:**
    - **MAXPROCSYS**
    - **MAXPROCUSER**
    - **MAXFILEPROC**
    - **MAXTHREADTASKS**
    - **MAXTHREAD**
    - **MAXSOCKETS**
  - **Verify that the AF_UNIX domain is defined**
  - **Verify that the AF_INET domain is defined**

External Security Manager (ESM) Considerations

- Required Groups
  - **SYS1 (or a logically equivalent group to contain UID=0 users)**
  - **FWGRP**
- Required Users
  - **FWKERN**
- Authorizations Considerations
  - **Define the FACILITY class profile FWKERN.START.REQUEST**
    - **Permit FWKERN update access to this facility**
  - **Permit FWKERN access to start**
    - **FWKERN (JCL)**
    - **Each of the Firewall Daemons (JCL)**
  - **Permit FWKERN access to**
    - **READ the TCP/IP data sets**
    - **BPX.SMF facility**
    - **BPX.DAEMON facility**

Program Control Considerations

- If program control is on
  - **Need to mark the firewall programs as program controlled**
    - **ICA.SICALMOD**
  - **Need to mark the SSL library as program controlled**
    - **hlq.SGSLOAD**

TCP/IP Considerations

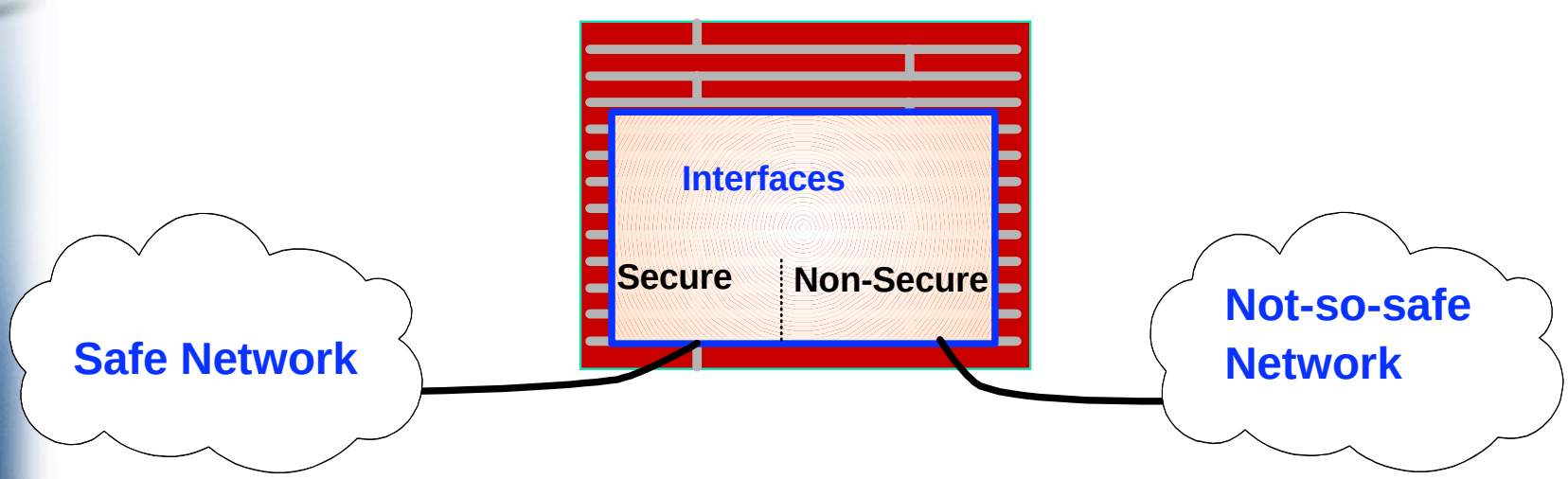
- TCP/IP Profile
  - **Make sure adapters are defined**
  - **Examine AUTOLOG statements**
    - **Remove standards TCP/IP server to harden firewall**
    - **Add FWKERN**
  - **Port reserves**
    - **FTP proxy (TCP 20/21)**
    - **DNS (TCP/53 and UDP/53)**
    - **IKE (UDP/500)**
    - **SYSLOG (UDP/514)**
    - **SOCKS (TCP/1080)**
    - **Config server (TCP/1014)**
  - **IPCONFIG FIREWALL DATAGRAMFWD**
- /etc/services
  - **SYSLOG - UDP/ 514**
  - **IKE - UDP/500**

Additional Considerations

- Sample configuration files
  - **If migrating from a previous release**
    - **Use fwmigrate command to update files**
  - **If not migrating**
    - **Copy with -p option**
- Convert shell scripts to non-IBM-1047 code page
  - **Use iconv command to convert:**
    - **fwlogmgt**
    - **getmsg**
- Define stacks to the Firewall
  - **fwstack command**
- Define secure interfaces
  - **fwadapter command**

Secure Interfaces

- A firewall is device used to separate a "safe" network from a "not-so-safe" network
  - **The "safe" network is referred to as the secure network**
    - **Secure interfaces are those connected to the "safe" network**
  - **The "not-so-safe" network is referred to as the non-secure network**
    - **Non-secure interfaces are those connected to the "not-so-safe" network**



Management of the Firewall Daemons

Introducing the Firewall Kernel Address Space

- All Firewall daemons run in their own address space
- Firewall address spaces are created and managed by the firewall kernel, known as FWKERN
- All requests request to start and stop a daemon goes through FWKERN
  - **FWKERN must be started before any firewall daemon can be started**
- FWKERN is started via
  - **AUTOSTART entry in the TCP/IP profile**
  - **Start operator command**

Introducing the Firewall Daemons

- SYSLOGD
  - **Firewall logging server**
- SOCKS
  - **Firewall SOCKS server**
- PFTPD
  - **Firewall Proxy FTP server**
- CFGSRV
  - **Firewall Configuration server**
- ISAKMPD
  - **Firewall ISAKMP server**
    - **Supports the Internet Key Exchange (IKE) protocol used by dynamic VPNs**
- FWSTACKD
  - **Monitors and communicates with firewall TCP/IP stacks**

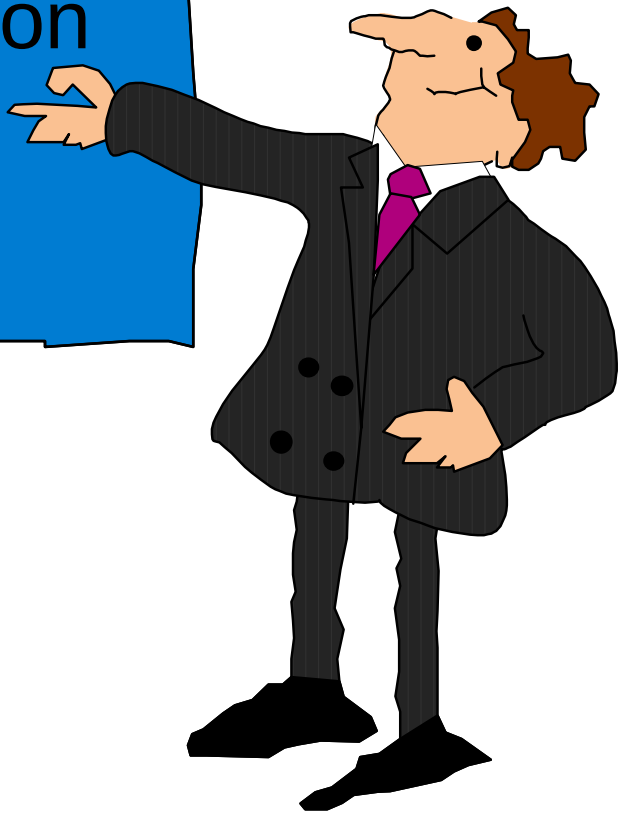
Basic Daemon Configuration

- Accomplished through the fwdaemon command:
  - **Options for defining configuration information:**
    - **daemon (SYSLOGD|SOCKSD|PFTPD|CFGSRV|ISAKMPD|FWSATCKD)**
    - **started (yes|no)**
    - **timeout (seconds)**
    - **restart (seconds)**
    - **maxconns**
    - **runopts**
    - **daemonopts**
    - **outputfile (dddef of filename)**

Starting and Stopping Individual Firewall Daemons

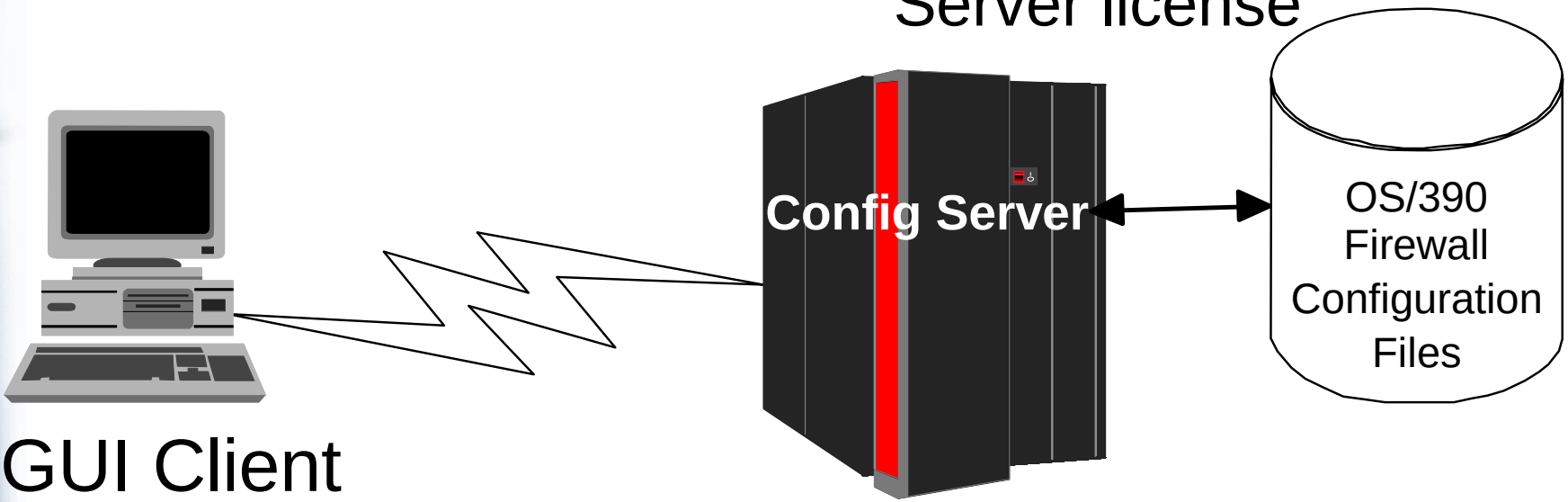
- FWKERN modify command
  - **Runs from the operator console**
  - **Format:**
    - **modify fwkern,[start|stop] daemon**
- fwdaemon command
  - **fwdaemon cmd=start**
  - **fwdaemon cmd=stop**

Intro to the Configuration Server and GUI



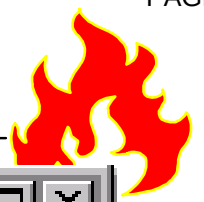
Configuration Server and GUI

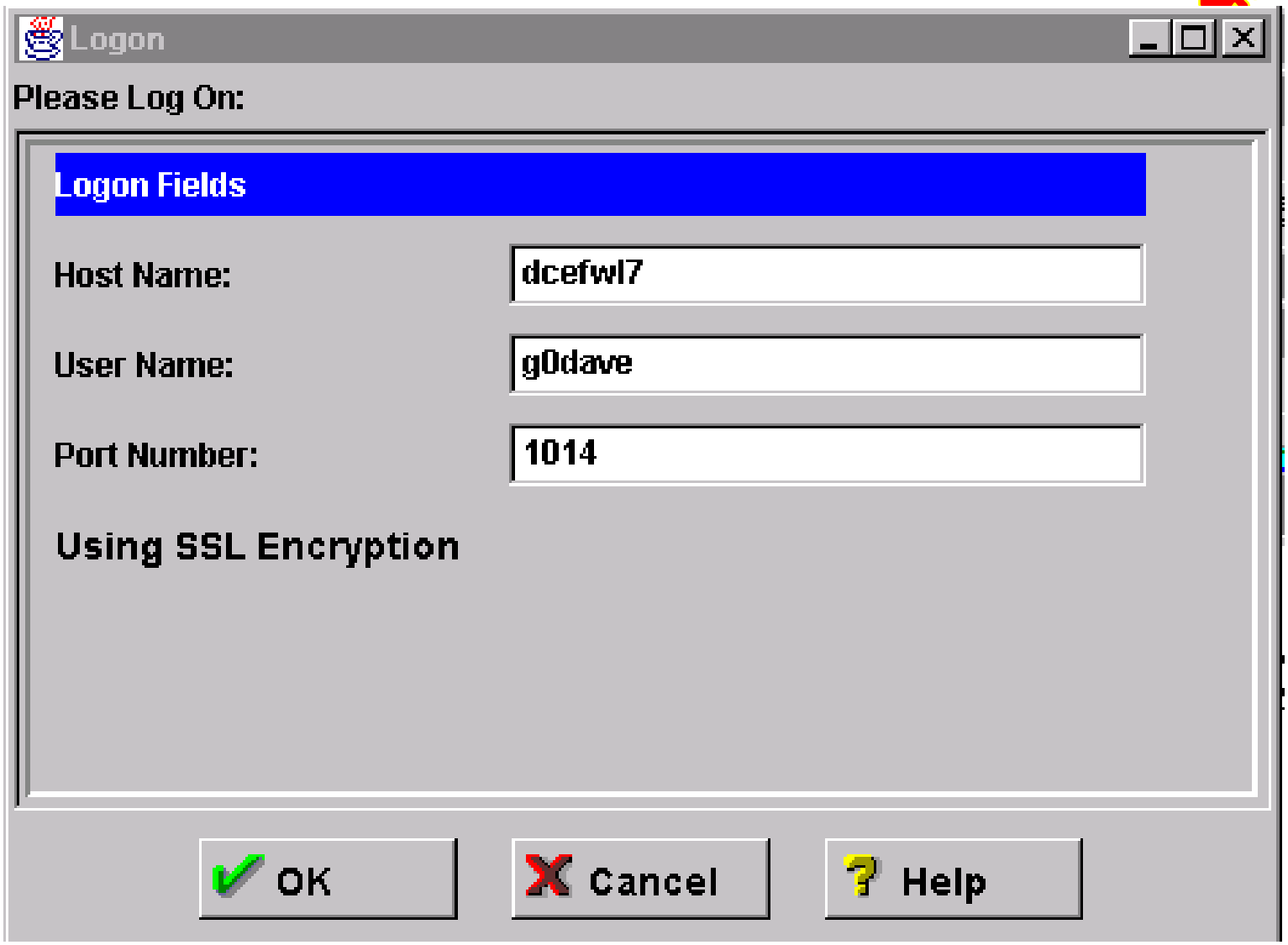
- GUI
  - Introduced in R7
  - JAVA Based
  - Supported on AIX and Windows 95/98/NT
- Config Server
  - Runs on OS/390
  - Controlled by FWKERN
  - Issues "commands" on behalf of the GUI
  - Requires Security Server license



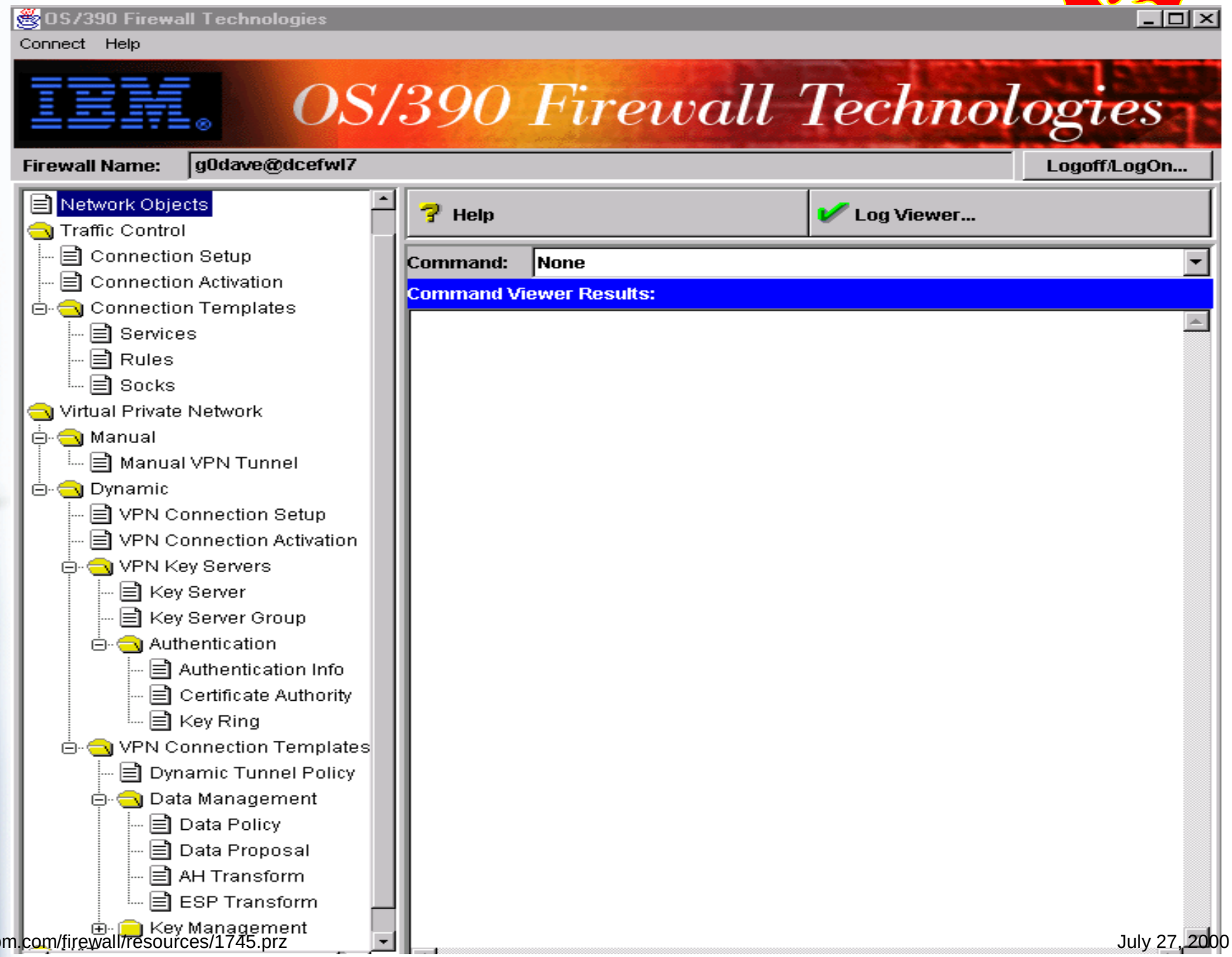
Firewall

GUI Logon Screen





Firewall GUI Main Menu



Sample
Data Entry Panel

(204.146.134.53) Add ESP Transform

Add ESP Transform (fwesptran command)

Identification

ESP Transform Name:	Demo ESP Tran
Description:	

ESP Transform Composition

Encapsulation Mode:	Tunnel
Authentication Algorithm:	HMAC_MD5
Encryption Algorithm:	DES_CBC_8

Initiator Session Expiration

Maximum Data Lifetime:	60
Maximum Size Limit:	0

Responder Session Expiration

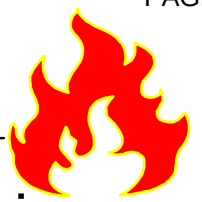
Data Lifetime Range:	60 -	480
Size Limit Range:	0 -	0

OK Cancel Help

Firewall Configuration Server Considerations

- External Security Manager (ESM) Considerations
  - **Define a facility class profile (ICA.CFGSRV)**
  - **Identify users that could use the configuration GUI**
    - **Permit update access to the ICA.CFGSRV facility**
    - **Must be either:**
      - **Super User (UID=0)**
      - **Member of the FWGRP**
- Configuration Server Configuration Options
  - **TCP Port to listen on (-p)**
    - **Default 1014**
      - **Need to have filter rules that allow traffic to flow between client and server!**
  - **Location of an SSL key database with a valid certificate (-f)**
    - **The GUI and configuration server use SSL only to encrypt/decrypt data**
      - **Certificate not use to authenticate configuration server**
      - **Can use a self-signed certificate**
    - **Need to use the SSL option: "Store the encrypted database password"**
  - **Options specified via the fwdaemon command**
    - **fwdaemon cmd=change daemon=CFGSRV daemonopts="-f /mydir/mykey.kdb" -p 1014"**

Firewall Configuration GUI Considerations



- AIX system requirements:
  - **AIX 4.2 or higher**
  - **Netscape nav.rte 3.0.0.0**
  - **Java.rte 1.1.6**
- Download GUI to client
  - **/usr/lpp/fw/bin/fwtech.obj**
- Run SMIT
  - **Select latest install option**
- Windows system requirements:
  - **Windows NT 4.0/95/98**
  - **Browser with Java and frames support**
  - **Zip tool that handles long file names**
- Download GUI to client
  - **/usr/lpp/fw/bin/fwtech.zip**
- Unzip fwtech.zip
- Run setup

Intro to the Firewall Commands

Firewall Commands

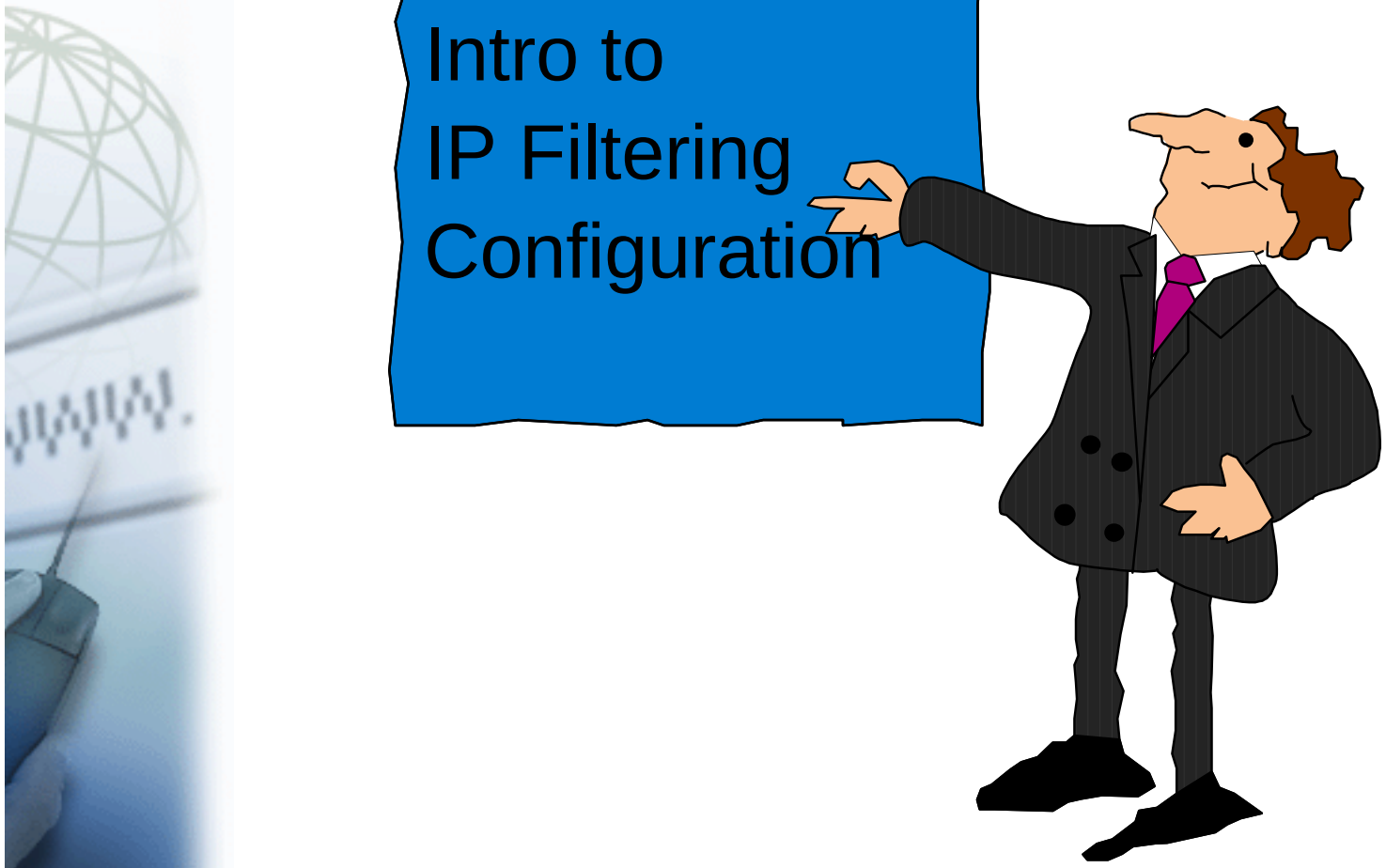
- Shipped as part of Security Server
  - **Can be installed and used WITHOUT a Security Server license**
- Require Unix System Services (e.g. OMVS)
  - **Only exception**
    - **fwkern**
      - **Issued from operators console**
- Must be issued by a superuser or a user in the fwgrp group
- Commands:

• fwadapter	• fwdatapol	• fwfrule	• fwkeytran	• fwsecpolicy
• fwahtran	• fwdataprop	• fwkern	• fwlog	• fwservice
• fwaudio	• fwdns	• fwkeypol	• fwlogmgmt	• fwsrule
• fwauthinfo	• fwdynconns	• fwkeyprop	• fwlogtxt	• fwstack
• fwcertauth	• fwdyntun	• fwkeyring	• fwmigrate	• fwtrace
• fwconns	• fwesptran	• fwkeysrvgrp	• fwnat	• fwtunnl
• fwdaemon	• fwfilter	• fwkeysrv	• fwnwgrp	

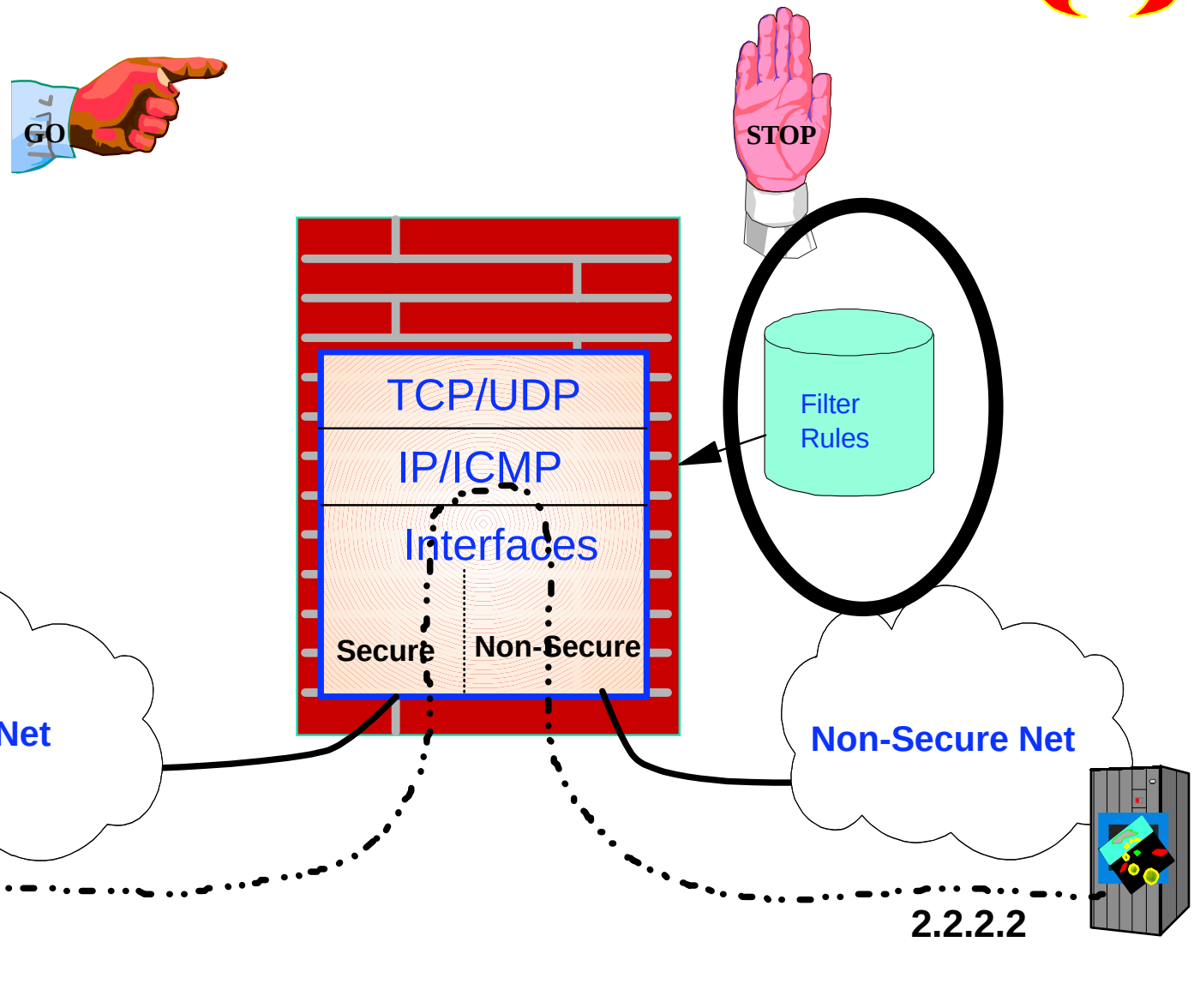
Sample Firewall Command

```
fwesptran   cmd=add
            name="Demo ESP Tran"
            mode=tunnel
            authalg=hmac_md5
            encralg=des_cbc_8
            itime=60
            isize=0
            rtime=60-480
            rsize=0
```

Note: mode, itime, isize, rtime, and rsize could have been defaulted



IP Packet Filtering Review

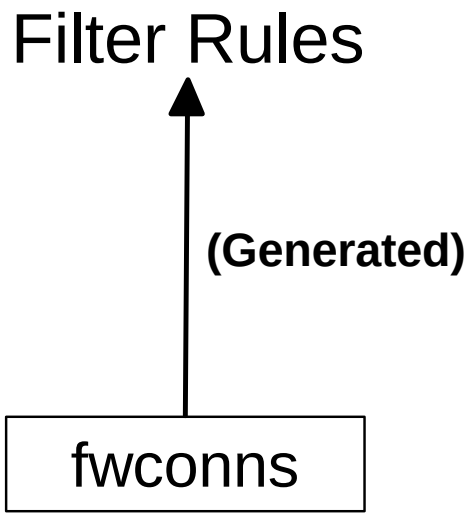


Packet Filter Rule Contents

- Selector Values
  - **Source:**
    - IP Address Specification
    - Port
  - **Destination:**
    - IP Address Specification
    - Port
  - **Protocol**
  - **Interface**
    - Secure/Non-secure/Both
  - **Direction:**
    - Inbound/Outbound/Both
  - **Routing:**
    - Local/Route/Both
- Actions Types
  - **Deny**
  - **Permit**
  - **Anchor**
- Control Information
  - **Logging**
  - **Time filters**
  - **Tunnel (VPN Information)**

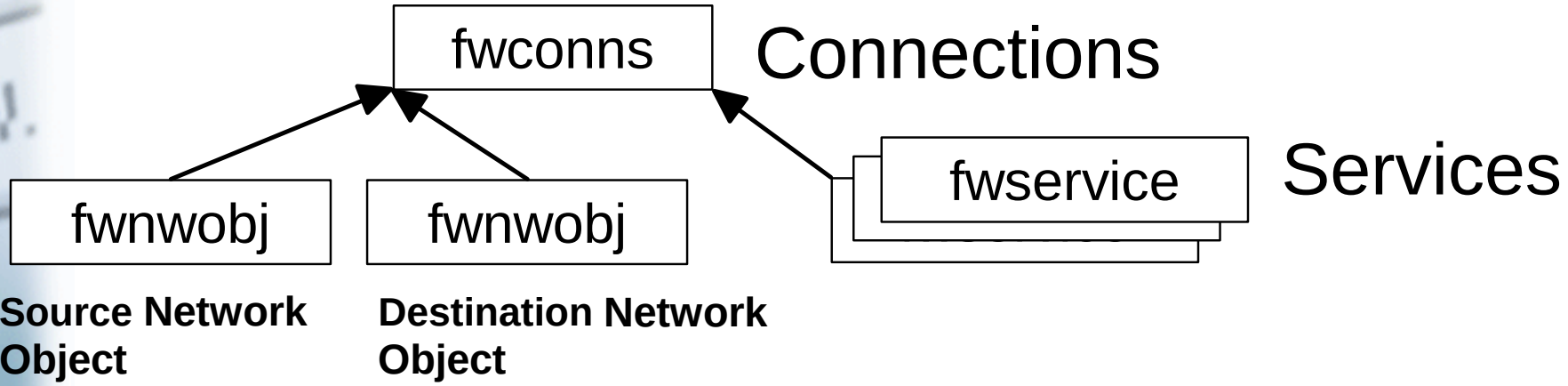
Filter Rules on OS/390

- Generated from connection objects (fwconns command)
  - **Order of connection object determines the order of generated rules**
  - **IP filtering will only consult the first rule an IP packet matches**
    - **It makes sense to position more rules before general rules**



Connection Objects

- Groups two network objects with one or more service objects
  - **Order of service objects within a connection object determines the order of generated rules**



Network Objects

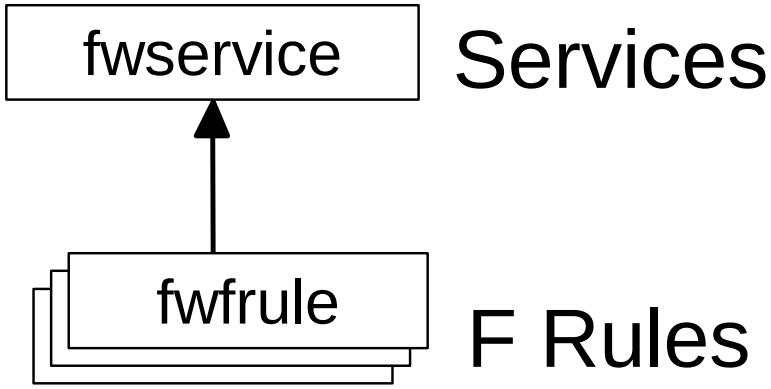
- Represent the various hosts and entities the firewall interacts with
  - **Single IP address**
  - **IP address and mask**
  - **Range of IP addresses**
- Used in the generation of IP filter rules, SOCKS rules, and dynamic connections
- Can be grouped into network object groups

Network Object: Relationship to Filter Rules

- Selector Values
  - **Source:**
    - **IP Address Specification**
    - **Port**
  - **Destination:**
    - **IP Address Specification**
    - **Port**
  - **Protocol**
- Actions Types
  - **Deny**
  - **Permit**
  - **Anchor**
- Control Information
  - **Logging**
  - **Time filters**
  - **Tunnel (VPN Information)**
- **Interface**
  - **Secure/Non-secure/Both**
- **Direction:**
  - **Inbound/Outbound/Both**
- **Routing:**
  - **Local/Route/Both**

Service Objects

- Groups one or more rule objects together to instruct the firewall to filter some sort of meaningful traffic
  - **Examples**
    - **telnet**
    - **ftp**
    - **http**
  - **Order of F rule objects within a service object determines the order of generated rules**
- Defines/overrides control information



Service Object: Relationship to Filter Rules

- Selector Values
  - **Source:**
    - IP Address Specification
    - Port
  - **Destination:**
    - IP Address Specification
    - Port
  - **Protocol**
  - **Interface**
    - Secure/Non-secure/Both
  - **Direction:**
    - Inbound/Outbound/Both
  - **Routing:**
    - Local/Route/Both
- Actions Types
  - **Deny**
  - **Permit**
  - **Anchor**
- Control Information
  - **Logging**
  - **Time filters**
  - **Tunnel (VPN Information)**

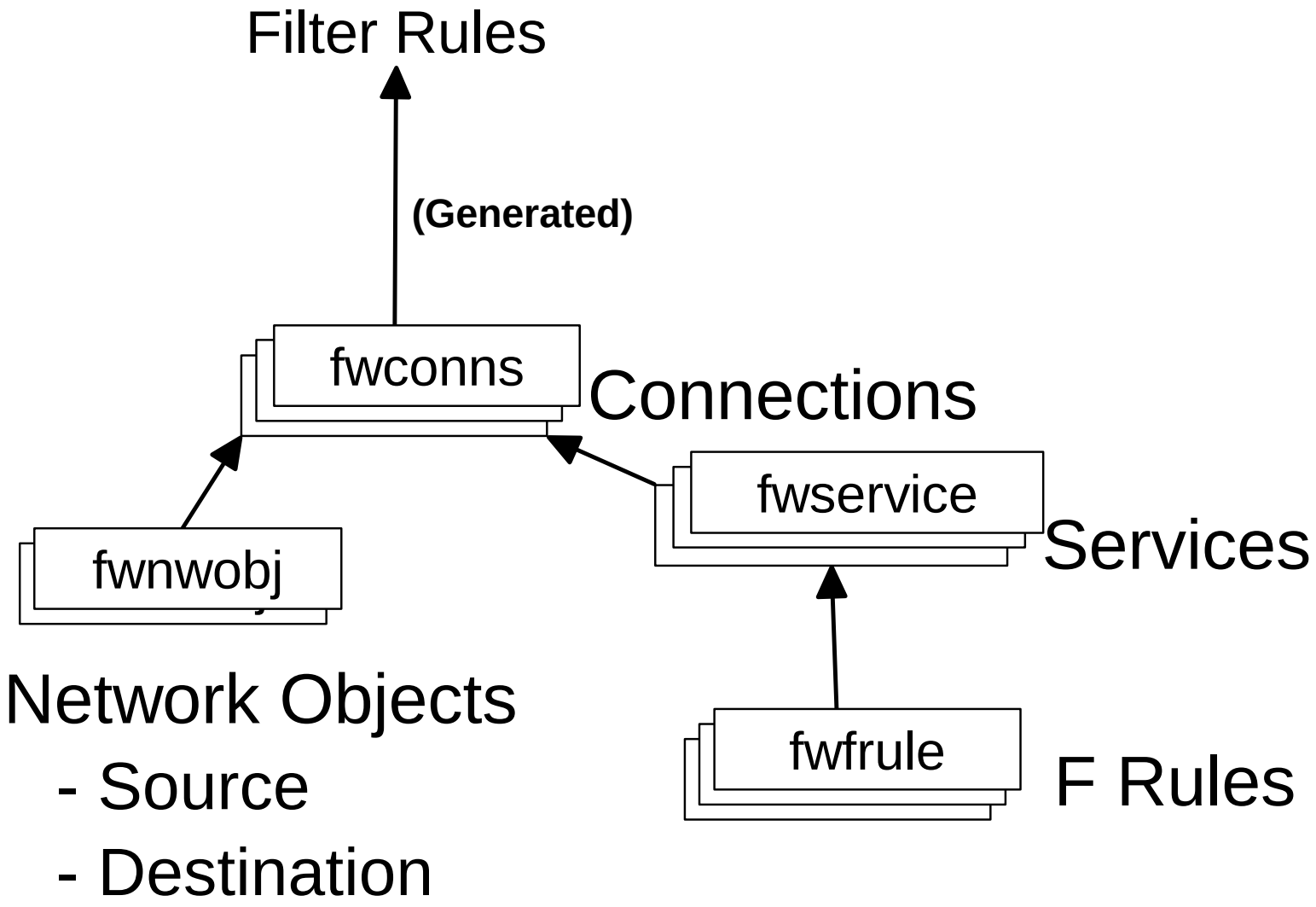
F Rule Objects

- Defines
  - **Characteristics of an IP packet to filter on**
    - source port
    - destination port
    - protocol
  - **Defines environment conditions**
    - interface
    - direction
    - routing
  - **A filter type**
    - permit
    - deny
    - anchor
  - **Control information**
    - logging
    - tunnel (VPN information)

F Rule Objects: Relationship to Filter Rules

- Selector Values
  - **Source:**
    - IP Address Specification
    - Port
  - **Destination:**
    - IP Address Specification
    - Port
  - **Protocol**
  - **Interface**
    - Secure/Non-secure/Both
  - **Direction:**
    - Inbound/Outbound/Both
  - **Routing:**
    - Local/Route/Both
- Actions Types
  - **Deny**
  - **Permit**
  - **Anchor**
- Control Information
  - **Logging**
  - **Time filters**
  - **Tunnel (VPN Information)**

Summary of Firewall Objects Related to Filtering

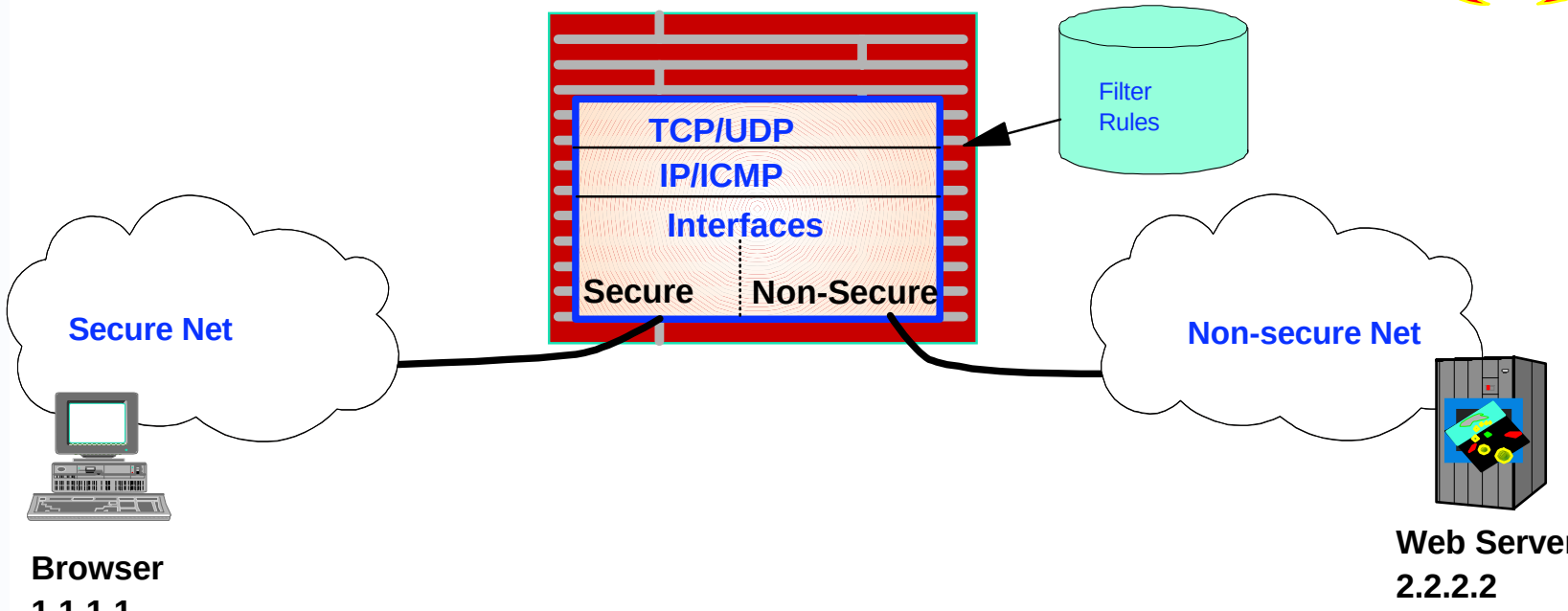


Activation

- Filter rules are generated from connection objects
- Modifications to any of the following objects DO NOT result in an automatic regeneration of filter rules
  - **Connections**
  - **Services**
  - **F Rules**
  - **Network Objects**
- The regeneration of filter rules must be explicitly requested
- If no rules are defined or active then
  - **Local access is permitted from the secure and non-secure interface**
  - **All other access is denied (i.e. all routed traffic)**

IP Filtering Configuration Example

Web Server Example: Overview



Source IP	Source Port	Destination IP	Destination Port	Protocol	Direction	Interface	Routing
1.1.1.1	> 1023	2.2.2.2	80	TCP	Inbound	Secure	Route
1.1.1.1	> 1023	2.2.2.2	80	TCP	Outbound	Non-Secure	Route
2.2.2.2	80	1.1.1.1	> 1023	TCP/ACK	Inbound	Non-Secure	Route
2.2..2.2	80	1.1.1.1	> 1023	TCP/ACK	Outbound	Secure	Route

Web Server Example: Overview (Continued)

F Rule Objects

Service Object

Source IP	Destination IP	Source Port	Destination Port	Protocol	Direction	Interface	Routing
1.1.1.1	2.2.2.2	> 1023	80	TCP	Inbound	Secure	Route
1.1.1.1	2.2.2.2	> 1023	80	TCP	Outbound	Non-Secure	Route
2.2.2.2	1.1.1.1	80	> 1023	TCP/ACK	Inbound	Non-Secure	Route
2.2.2.2	1.1.1.1	80	> 1023	TCP/ACK	Outbound	Secure	Route

Network Objects

Connection Object

Web Server Example:
Using Commands to Define F Rules

```
fwfrule   cmd=add name="http rule1"
          desc="Permit TCP src > 1023 dest 80 into secure adapter and routed"
          type=permit  protocol=tcp srcopcode=ge srcport=1023   destopcode=eq
          destport=80 interface=secure routing=route direction=inbound log=no

fwfrule   cmd=add name="http rule2"
          desc="Permit TCP src > 1023 dest 80 out nonsecure adapter and routed"
          type=permit  protocol=tcp srcopcode=ge srcport=1023   destopcode=eq
          destport=80 interface=nonsecure routing=route direction=outbound log=no

fwfrule   cmd=add name="http rule3"
          desc="Permit TCP src 80 dest > 1023 into nonsecure adapter and routed"
          type=permit  protocol=tcp/ack srcopcode=eq srcport=80 destopcode=gt
          destport=1023 interface=nonsecure routing=route direction=inbound log=no

fwfrule   cmd=add name="http rule4"
          desc="Permit TCP src 80 dest > 1023 out secure adapter and routed"
          type=permit  protocol=tcp/ack srcopcode=eq srcport=80 destopcode=gt
          destport=1023 interface=secure routing=route direction=outbound log=no
```

Web Server Example:
Using Commands to Define a Service

- Find ids of F rules to be added to the service

```
fwfrule   cmd=list
              .
              .
              .
511   permit      http rule1      Permit TCP src > 1023 dest 80 into secure a
512   permit      http rule2      Permit TCP src > 1023 dest 80 out nonsecure
513   permit      http rule3      Permit TCP src 80 dest > 1023 into nonsecur
514   permit      http rule4      Permit TCP src 80 dest > 1023 out secure ad
              .
              .
              .
```

- Define the service

```
fwservice     cmd=create name=http desc="Rules for http"
              rulelist=511/f,512/f,513/b,514/b
```

Web Server Example:
Using Commands to Define Network Objects

fwnwobj cmd=add name="Secure Browser" desc="Dave's Browser"
type=host addr=1.1.1.1 mask=255.255.255.255

fwnwobj cmd=add name="NonSecure HTTP Server" desc="Some HTTP Server"
type=host addr=2.2.2.2 mask=255.255.255.255

Web Server Example:
Using Commands to Define a Connection

- Find ids of service to be added to the connection

```
fwservice      cmd=list
        .
        .
        .
509   http                  Rules for http
        .
        .
        .
```

- Define the connection

```
fwconns        cmd=create name="Dave to Server"
               source="Secure Browser"
               destination="NonSecure HTTP Server"
               servicelist=509
```

Web Server Example:
Using Commands to Activate Filter Rules

fwfilter cmd=update

- Could just activate rules for a particular stack:

fwfilter cmd=update
stack=mystack

Predefined IP Filter Objects

- Pre-loaded set of objects shipped in sample configuration files
  - **Network Objects**
    - **The World**
  - **Services**
  - **Rules**
    - **The http rules defined in this example are pre-loaded**
- Can't be modified by commands or GUI
  - **Can be used by commands and GUI**

IP Filtering Sample GUI Configuration Screens

F Rule Objects

(dcefwl7) Add IP Rule

Add a Rule Template. **(fwfrule command)**

Identification

Rule Name:	Demo Inbound Rule for Config GUI
Description:	
Action:	Permit
Protocol:	tcp

Source Port / ICMP Type

Operation:	Any	Port #/Type:	0

Destination Port/ ICMP Code

Operation:	Equal to	Port #/Code:	1014

Interfaces Settings

Interface	Secure

Direction/Control

Routing:	○ both	◉ local	○ route	
Direction:	○ both	◉ inbound	○ outbound	
Log Control:	○ yes	◉ no	○ permit	○ deny

Tunnel Information

Manual VPN Tunnel ID:		Select...
Dynamic Tunnel Policy Name:		Select...

OK Cancel Help

Service Objects

(fwservice command)

[dcelwl7] Add Service

Add Service

Identification

Service Name: Demo Service

Description:

Service Composition

Rule Objects:

Flow	Name	Description
→	Demo Inbound Rule for Cor	
←	Demo Outbound Rule for C	

Select... Remove Move Up Move Down Flow

Service Override Values

Override Log Control: no override

Override Manual VPN Tunnel ID: Select...

Time Controls

Control By Time of Day Begin: End:

Control By Days: Week Days

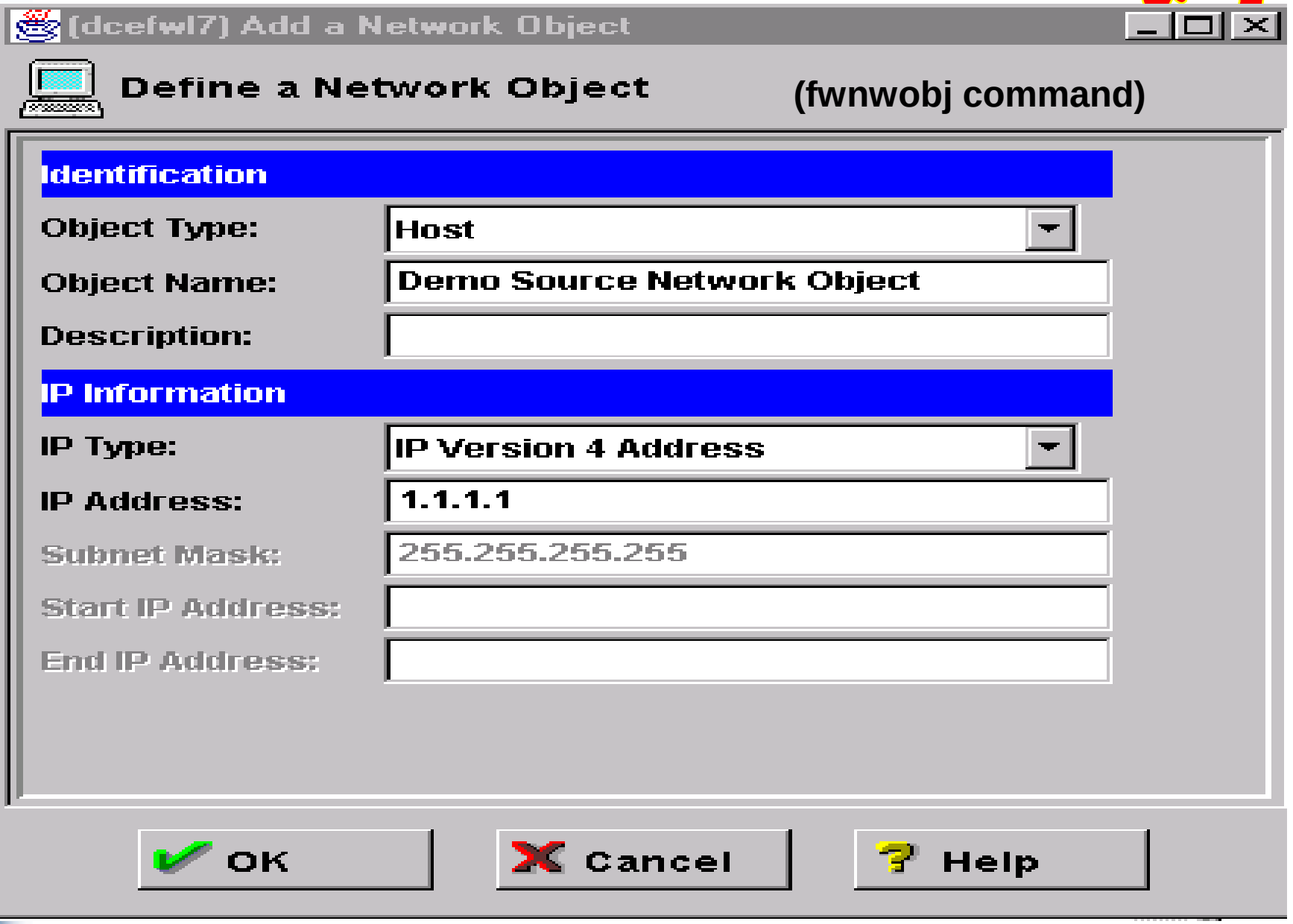
Begin: Sun End: Sun

Time Control Action: Activate Service During Specified Times

Deactivate Service During Specified Times

OK Cancel Help

Network Objects



Connection Objects

(fwconns command)

[dcefwl7] Add a Connection

Add a New Connection.

Identification

Name: Demo Connection

Description:

Source: Demo Source Network Object — Select...

Destination: Demo Source Network Object — Select...

Connection Services

Service Objects:

Name	Description
Demo Service	

Select... Remove Move Up Move Down

Socks

Sock Objects:

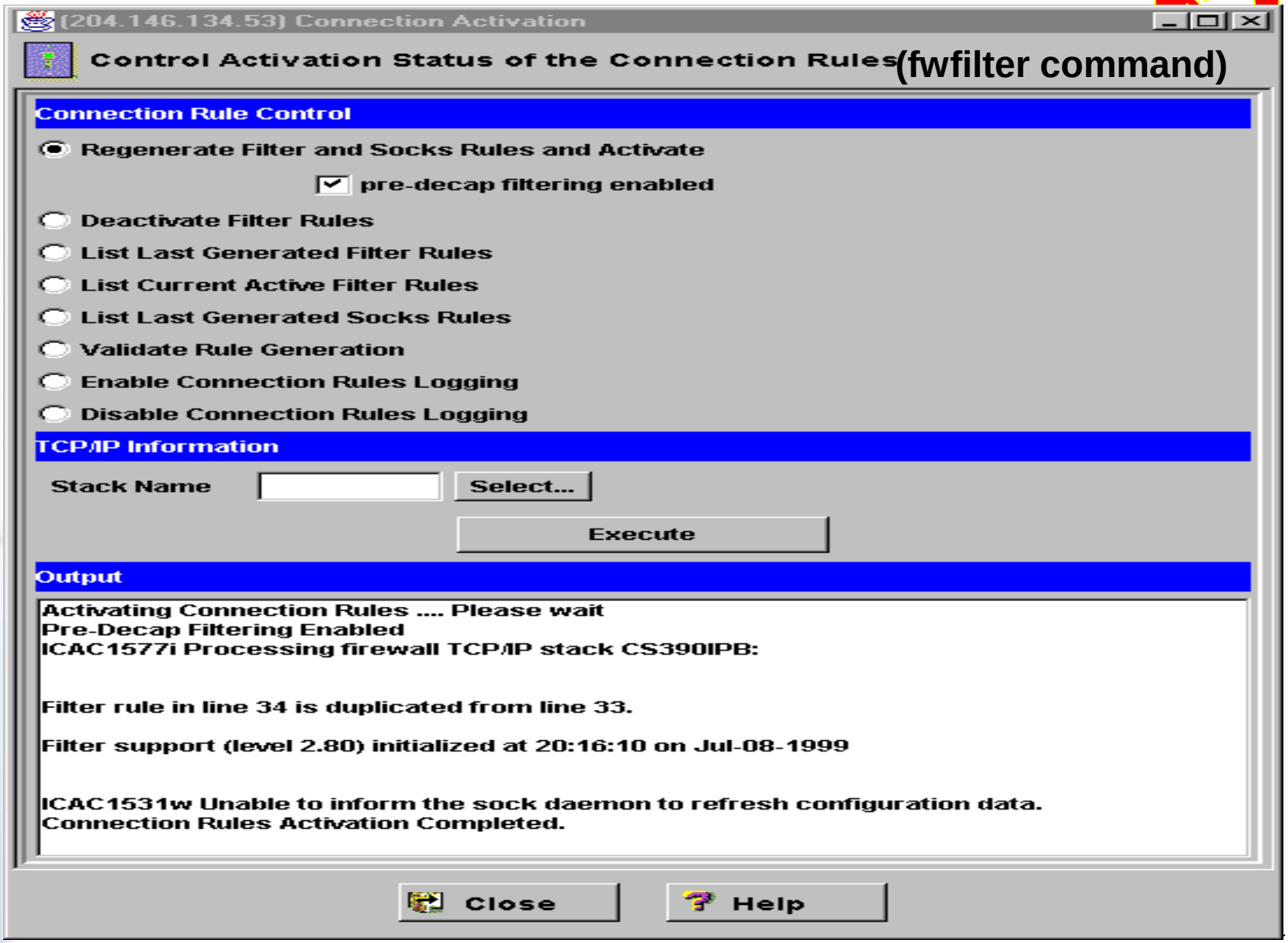
Name	Description

Select... Remove Move Up Move Down

OK Cancel Help

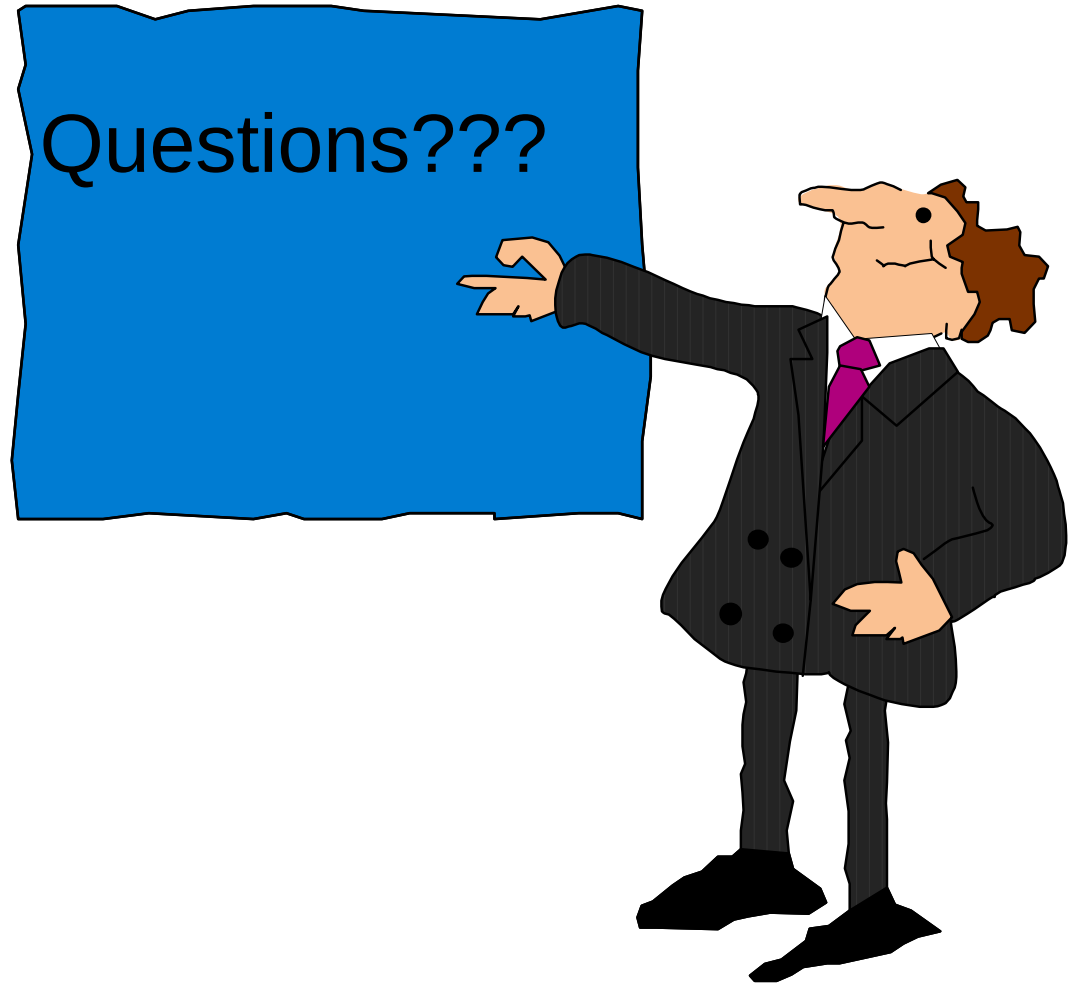
Activation of Filter Rules

(fwfilter command)



Where to Find More Information

- The OS/390 Firewall Technologies Resource Web page
  - **http://www.s390.ibm.com/products/mvs/firewall/resources.html**
    - **See our OS/390 FIREWALL TECHNOLOGIES GUIDE AND REFERENCE**
      - **R4, R5, R6, R7, and R8 versions available**
        - html format
        - pdf format
    - **See the following Freelance presentations:**
      - **OS/390 CONFIGURING VPNS ON OS/390**
      - **GETTING STARTED: IPSEC WITH CS FOR OS/390**
        - Concentrates on actual configuration
      - **GETTING STARTED: IPSEC WITH CS FOR OS/390 (Boston)**
        - Concentrates on gathering information for configuration
      - **FIREWALL OVERVIEW AND DIRECTIONS**
      - **GETTING STARTED USING THE FIREWALL**
        - This presentation



Appendix

SOCKS Configuration Considerations

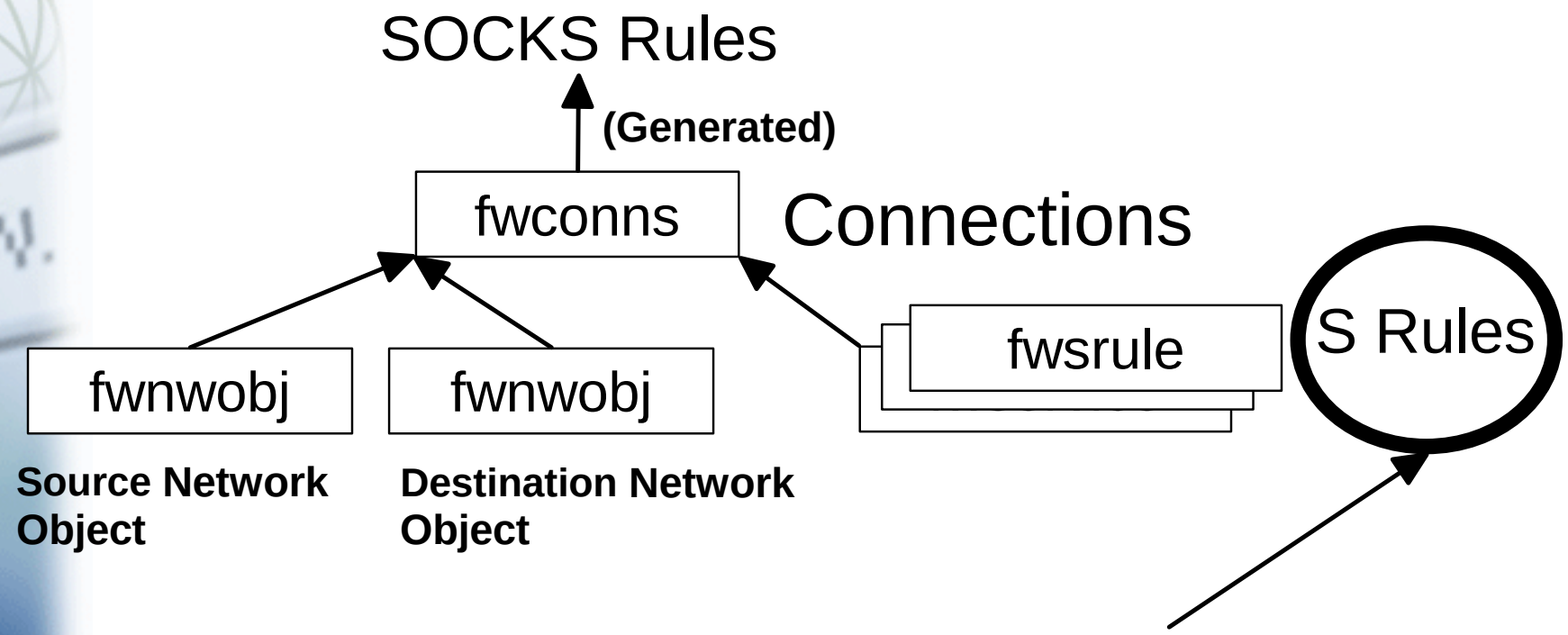
- Options on the daemonopts parameter of the fwdaemon command
  - **-ver**
    - **Displays version information and quit**
  - **-I**
    - **Requires client's identd to be running and the returned ID must match what was sent in on the socks request**
  - **-i**
    - **Like I, except is does not fail if the client is not running identd**
  - **-L**
    - **Echoes all sockd log message to the operator's console**
  - **-N**
    - **Records DNS names in log file**
    - **Requires a DNS lookup**
    - **May degrade performance**
  - **If neither -I or -i is specified**
    - **no identd verification occurs**

SOCKS Configuration Considerations (Continued)

- maxconns options on the daemopts parameter of the fwdaemon command
  - **One address space is created for every 300 connections supported**
  - **An attempt is made to load balance requests across address spaces**
- Need to define appropriate filter rules to allow:
  - **Traffic from client to sockd**
  - **Traffic from sockd to client's final destination**
- Also need to define SOCKS rules

SOCKS Rules

- Generated from connection objects



These are different from F Rules which are used to create filter rules

S Rule Objects

- Define:
  - **Type (action)**
    - permit
    - deny
  - **Type of identd processing**
    - Options
      - Do what was set in the via the daemonopts option of fwdaemon
      - Never perform identd verification
      - Always require identd verification
      - Require only if the client has an identd daemon
    - Overrides daemonopts option of fwdaemon
  - **User IDs this rule applies to**
  - **Port specifications**
  - **A rule command**
    - A shell command that gets executed when the sockd matches the rule
    - May degrade performance

Proxy FTP Configuration Considerations

- Options on the daemonopts parameter of the fwdaemon command
  - **-ns**
    - **Allow connections from a non-secure network**
    - **Possible security risk**
  - **-t**
    - **Timeout value for inactive connections**
  - **-l**
    - **Timeout value for logon requests**
- /etc/fwftp.deniedusers file
  - **Contains a list of user IDs not allowed to use the FTP proxy**
- To use the ftpd proxy you must have:
  - **An ID authorized to the ESM**
  - **A Unix home directory**
  - **A password**
  - **Not be listed in the /etc/fwftp.deniedusers file**

Proxy FTP Configuration Considerations (Continued)

- maxconns options on the daemopts parameter of the fwdaemon command
  - **One address space is created for every 300 connections supported**
  - **An attempt is made to load balance requests across address spaces**
- Need to define appropriate filter rules to allow:
  - **Traffic from client to proxy ftp daemon**
  - **Traffic from proxy ftp daemon to client's final destination**

ISAKMP Daemon Configuration Considerations

- Options on the daemonopts parameter of the fwdaemon command
  - **-keyretry**
    - **Number of times to retransmit an unanswered key negotiation message**
  - **-keywait**
    - **How long to wait before retransmitting**
  - **-dataretry**
    - **Number of times to retransmit an unanswered key negotiation message**
  - **-datawait**
    - **How long to wait before retransmitting**
  - **-L**
    - **Echo log messages to the job output file**
- Need to define appropriate filter rules to allow:
  - **ISAKMP daemons to talk (UDP port 500)**

ISAKMP Daemon Configuration Considerations (Continued)

- Need to have the Open Cryptographic Service Facility (OCSF) installed and configured
  - **Must be enabled for program control**
  - **DLLs must have their APF authorized attributes turned on**
- Need to have the Open Cryptographic Enhanced Plug-ins (OCEP) installed and configured
  - **Must be enabled for program control**
  - **DLLs must have their APF authorized attributes turned on**

ISAKMP Daemon Configuration Considerations (Continued)

- RACF keyring considerations
  - **FWKERN must have READ access the following facilities**
    - **IRR.DIGTCERT.LIST**
    - **IRR.DIGTCERT.LISTRING**
  - **The owner of the ISAKMP daemon's RACF keyring must have**
    - **CONTROL access to the following facilities**
      - **IRR.DIGTCERT.ADDRING**
      - **IRR.DIGTCERT.LISTRING**
    - **UPDATE access to the following facilities**
      - **IRR.DIGTCERT.ADD**
      - **IRR.DIGTCERT.CONNECT**
      - **IRR.DIGTCERT.GENCERT**
      - **IRR.DIGTCERT.GENREQ**
      - **IRR.DIGTCERT.LIST**

ISAKMP Daemon Configuration Considerations (Continued)

- Configuration of a dynamic VPN
  - **Beyond the scope of this appendix**
  - **For dynamic VPN configuration information see the OS/390 Firewall Technologies Resource Web page**
    - **http://www.s390.ibm.com/products/mvs/firewall/resources.html**
    - **See the following Freelance presentations:**
      - **OS/390 CONFIGURING VPNS ON OS/390**
      - **GETTING STARTED: IPSEC WITH CS FOR OS/390**
        - Concentrates on actual configuration
      - **GETTING STARTED: IPSEC WITH CS FOR OS/390 (Boston)**
        - Concentrates on gathering information for configuration