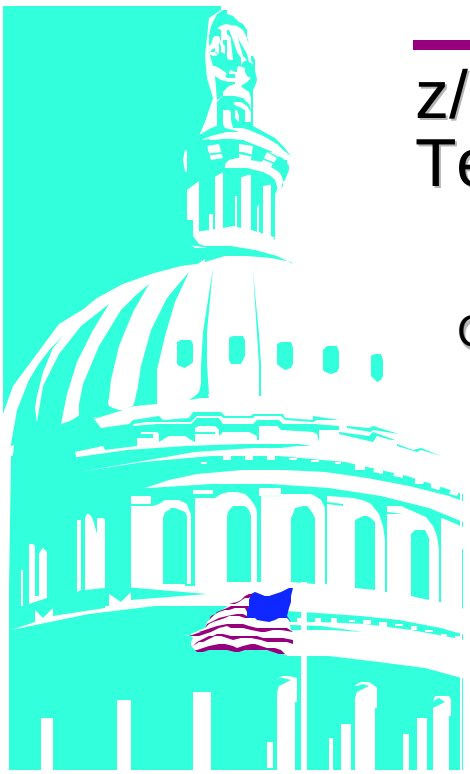


z/OS Firewall Technologies

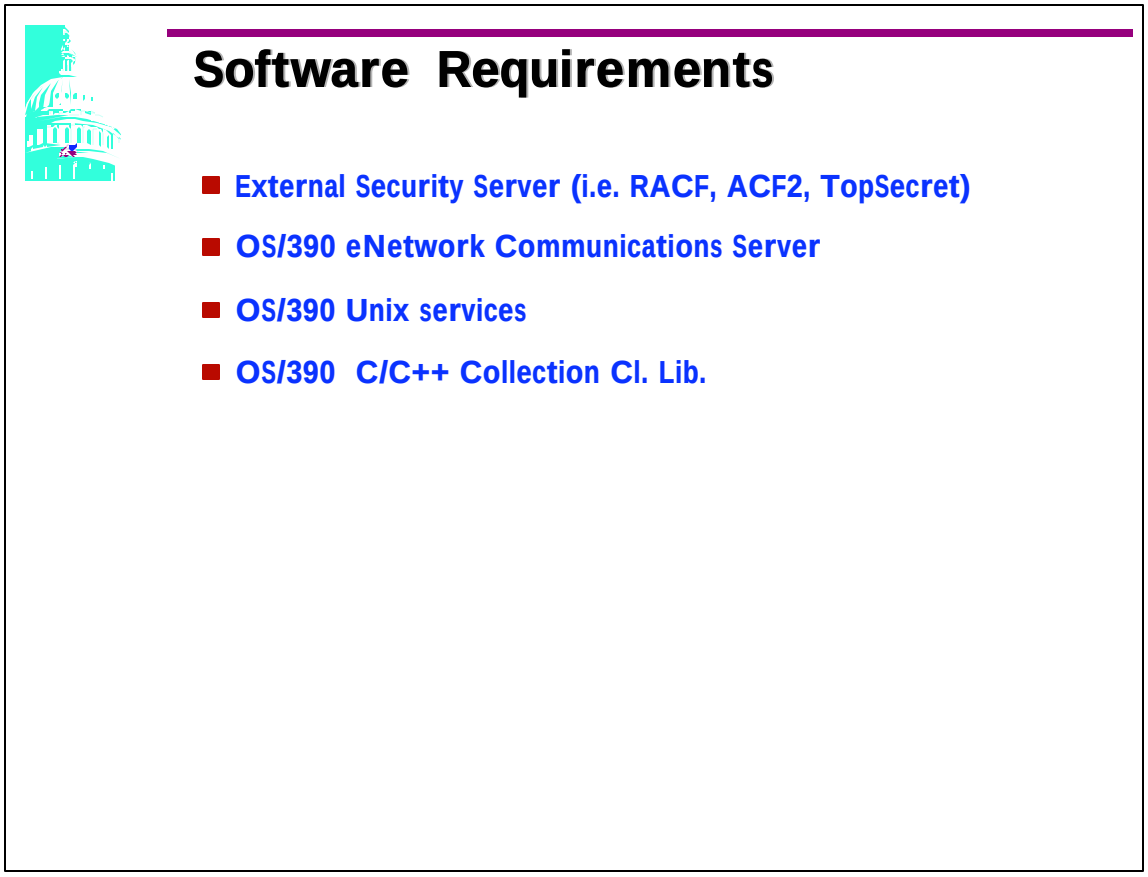
Configuration and Setup

Mary B. Sweat
sweatm@us.ibm.com

IBM S/390 Security
Advanced Technical Support
Washington Systems Center
Gaithersburg, MD

Unit Expectations

- **Purpose**
  - **To teach the students how to install and setup the OS/390 Firewall**
  - **Setup the base for the firewall**



- System Secure Sockets Layer is required for the usage of the OS/390 Firewall GUI.
- OCSF and OCEP is required if dynamic tunnels are used.

Software for Configuration Client

- **AIX**
  - **Java.rte 1.1.4 or 1.1.6**
  - **AIX 4.2 or higher (as long as Java.rte level is supported)**
  - **Netscape nav.rte 3.0.0.1**
- **Windows 95 or Windows NT**
  - **Web browser with Java and frames support**
  - **Zip tool that handles long file names**
    - WinZip32 tool in WinZip

- Java.rte is available on the AIX bonus pack and it is a key part of this requirement. This files is part of JAVA code.
- Nav.rte is part of Netscape.
- Information about WinZip can be found at http://www.winzip.com

SYS1.PARMLIB Member Updates

- **PROGxx**
  - **Add SYS1.SICALMOD (for APF authorization)**
- **LNKLSTxx**
  - **Add SYS1.SICALMOD**
- **IKJTSOxx - AUTHPGM**
  - **add firewall commands & pgms**
    - ICADCT - fwkern process, controls servers
    - ICADDCT - control process for each server started
    - ICADCFGS - configuration server
    - ICADFTPD - FTP Proxy
    - ICADPFTP - FTP Proxy
    - ICADIKED - dynamic tunnels
    - ICADSOCK - SOCKs Server
    - ICADSOXD - SOCKs Server
    - ICADSTAK - Firewall stack

- Some environments may have a LNKLSTxx member and a PROGxx member in SYS1.PARMLIB. Determine the correct member to use to avoid unnecessary debugging of possible errors.

- Add SYS1.SICALMOD to the Authorized Program Facility (APF) member and to the Linklist member in SYS1.PARMLIB.

- Add these firewall authorized programs to member IKJTSOxx. These programs are involved with the startup of the firewall kernel and it's related servers.
  - ICADPFTP & ICADFTPD, ICADSOCK and ICADSOXD are used for multi thread capabilities

PROCLIB Members

- **SYS1.PROCLIB**
  - **Copy or concatenate all members in the firewall dataset xxxx.SICAPROC**
  - **If concatenated updated member JES2 (or JES3)**
    - update SYS1.PARMLIB(MSTJCLxx) with dataset name

```
******************************** Top of Data ******************
//JES2    PROC DSN1='SYS1.PROCLIB',    * STANDARD PROCLIB.
//           DSN2='SYS1.PROCLIB.SYSTEM',  * IBM SUPPLIED PROCS
//          DSN6='SYS1.FIREWALL.SICAPROC', * FIREWALL PROCLIB
//PROC00  DD DSN=&DSN1,DISP=SHR
//              DD DSN=&DSN2,DISP=SHR
//                DD DSN=&DSN6,DISP=SHR
//******** INDIVIDUAL PROCXX CARDS
//PROC01    DD DSN=&DSN1,DISP=SHR
//PROC02    DD DSN=&DSN2,DISP=SHR
//PROC6     DD  DSN=&DSN6,DISP=SHR   *
//HASPPARM  DD DSN=SYS1.PARMLIB(&TYPE.PA
```

- 1.If you concatenate to SYS1.PROCLIB, update member JES2 or JES3 in SYS1.PROCLIB and add the dataset(s). Ensure you also add the dataset name to the PROC statement and the PROC library list.

2.Print out a copy of the JES2/JES3 member before you IPL your system.

3.If error is generated in this member JES will not start.

4.If dasd volume that SYS1.PROCLIB resides on can be access from another system you can correct any errors in the JES member.

SCEERUN

- Firewall requires the run-time library provided by Language Environment, hlq.SCEERUN
  - recommend that this dataset be added to hlq.LNKLIST

OR

  - add hlq.SCEERUN to the procedures in 'hlq.SICAPROC'.
  - under UNIX add "export STEPLIB=hlq.SCEERUN" to /etc/profile file

UNIX Services System Parameters

- **SYS1.PARMLIB (BPXPRMxx)**
  - **check or update as necessary the paramters that control resources for threads, files, sockets**
    - MAXPROCSYS (200) (max. # of processes UNIX allows active at one time)
    - MAXPROCUSER (25) (max. # of processes a single UNIX UID can have active at the same time)
    - MAXFILEPROC (64) (max # of files that a single UNIX UID can concurrently have active or open)
    - MAXTHREADTASKS (50) (# of MVS tasks created with pthread_create that a single user may have concurrently active in a process)
    - MAXTHREADS (200) (max. # of threads created with pthread_create that a single server can have currently active; includes those that are running, queued and exited but not detached)
    - MAXSOCKETS (64) (max. number of sockets that can be obtained for the given file system type)

- The values for threads, files and sockets are used by the Firewall technologies. Default values are provided, however, they should be reviewed to ensure they are sufficient for the environment.

- Reference: "Firewall Technologies Guide and Reference" for the number of processes and threads the firewall uses for each parameter.

- NOTE: A **PTHREAD_CREATE** is used by Unix Services. Instead of processes, UNIX uses threads. Threads run in the same address space as the program that calls it.

UNIX Services System Parameters ....

- **AF_UNIX and AF_INET**
  - **ensure theses file systems are defined in BPXPRMxx**
    - NETWORK DOMAINNAME(AF_UNIX)
      DOMAINNUMBER(1)
      MAXSOCKETS(100)
      TYPE(UDS)
    - NETWORK DOMAINNAME(AF_INET)
      DOMAINNUMBER(2)
      MAXSOCKETS(**n**)
      TYPE(CINET)

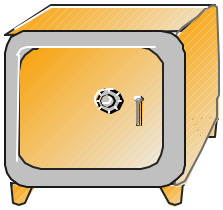
    note: **n** = MAXSOCKETS (see previous foil)

- Network Domain is a socket physical file system domain that should be initialized. When a UNIX user wants to write to a file UNIX sometimes uses a socket to perform the function. This socket physical file system contains the code necessary to perform this write.

- The names used in TYPE field can be assigned or you can use the TYPE listed in the MVS Initialization manual under BPXPRMxx.

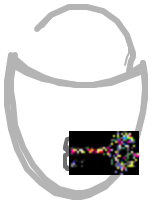
Security Requirements

- **Groups**
  - **Define firewall group with OMVS authority**
    - ag fwgrp sup(sys1) ow(sys1) omvs(gid(100))
  - **Verify existence of group SYS1 or equivalent to contain UID=0 users**
    - if one does not exist, create it AG newgrp omvs(GID(xxx)
- **Define firewall user ID**
  - mkdir '/u/fwkern' mode(7,5,5)
  - au fwkern DFLTGRP(fwgrp) auth(create) uacc(alter) password(xxxx) ow(sys1) omvs(home(/u/fwkern/) uid(0))
  - **activate STARTED class in RACF**
    - setr classact(started)
  - **define firewall ID to started class**
    - rdef started fwkern stdata(user(fwkern))
    - setr raclist (started) refresh



- This foil used RACF commands in the example, yuu may use any external Security Manager.
- The same user ID can be used for all Firewall servers and the kernel (e.g. *fwkern*).
- Group - the GID number, superior group (SUP) and owner (OW) is chosen by the customer to fit their environment. The name of the group can be other then *fwgrp.*
- Add User - the firewall kernel name can be anything this customer chooses, the group must match the name of firewall group defined above
- Define an ID to the STARTED class that will be used with the firewall started task.
- The Firewall administrator ID can be any ID that has SUPERUSER authority or a user ID that has been connected to the Firewall group (*fwgrp*).

Grant Authority to Firewall Objects

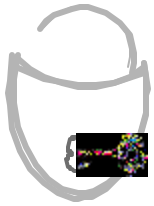


- **Create FWKERN.START.REQUEST profile**
  - **activate FACILITY class**
    - **setr classact(facility)**
  - **define FWKERN.START.REQUEST**
    - **def facility fwkern.start.request uacc(none)**
  - **permit firewall access to profile**
    - **pe fwkern.start.request cl(facility) id(fwkern) ac(update)**
    - setr raclist(facility) refresh
- **Allow Firewall id access to Firewall daemons**
  - **rdef started fwkern.** stdata(user(fwkern) group(fwgrp))**
    icapsock.**
    icappftp.**
    icapcfgs.**
    icapstak.**
    icapiked.**
  - **setr raclist(started) refresh**

- The firewall needs access to the daemons so the firewall kernel ID can start them.

Access Authority

- **Grant Firewall kernel access to TCP datasets**
  - **pe tcpip.** id(fwkern) ac(read)**
- **Grant PFTP server to BPX.DAEMON**
  - **verify BPX.DAEMON facility exist**
  - **create if it does not exist**
    - **rdef facility bpx.daemon uacc(none)**
  - **set the permissions**
    - **pe bpx.daemon cl(facility) id(fwkern) ac(read)**
    - **setr raclist(facility) refresh**



- Since the Firewall kernel ID can be used to run the Firewall daemons as well as the Firewall itself, FWKERN was entered in the example as the userid.
- FTP proxy is the only OS/390 feature that uses a SETUID command, which is why it requires access to BPX.DAEMON. If an installation uses different user IDs to run the daemons then the only ID that needs access to BPX.DAEMON is the userid that runs the FTP PROXY daemon.
- If the FTP proc contains a STEPLIB, all the programs that reside in that dataset must be program controlled and whatever the proc executes must be programmed controlled. If the dataset is defined in the LINKLST then they do not have to be programmed controlled.

SMPE POST JOB

- **Edit ICAPOST to set permission bits for various firewall files**
  - **located in firewall *hlq*.SICASAMP**
    - modify DD statement SYSEXEC, ensure dataset name for hlq. SICASAMP is specified
    - change ICAPOST statement to include
      - pathprefix
      - firewall group ID
      - SYS1 group ID

Example: ICAPOSTR / 100 2

- This is the only SMPE job that the customer must tailor and run. This step does not apply to OS/390 2.5 or 2.6.

Define Firewall Adapters in TCP/IP

- **Configure TCP/IP profile (xxx.profile.tcpip)**
  - **add DEVICE and LINK statements for the system adapters**
    - **EXAMPLE**: DEVice----device name--- **LCS** ---device number
      CTC
      LINK ----- link_name---- **IBMTR** ---- link_number----device_name
      CTC

      Example:

      DEVICE ***OSA5510*** **LCS** ***5510***
      LINK ***OSTR5510*** **IBMTR** ***0*** ***OSA5510***
      ;
      DEVICE ***CTC1*** **CTC** ***5530***
      LINK ***LINKMVS*** **CTC** ***1*** ***CTC1***

- The TCP/IP profile must know about any supported devices used for communication. Each **DEVICE** statement will have a corresponding **LINK** statement. This example reflects a definition for a Open System Adapter device (a token ring card for a mainframe) with a LINK statement for a Token-Ring Network and a CTC (Channel to Channel) device with a link statement for the network interface link associated with the CTC device.

- DEVICE
  > Device_name: name of the device
  > Device type: (LCS, CTC, etc)
  > Device_number: hexadecimal device number of device type (ref. TCP/IP manuals)

- LINK
  > Link_name: name of the link_
  > Link type:(IBMTR, CTC, etc.)
  > Link_number: 0 is for the 1st token ring in the LCS, 1 for the second and so on
  > Device_name: same name specified in the DEVICE statement

- There are numerous types of adapters, reference the IP Configuration manual, SC31-8513-00 for details on different adapter types.

Firewall Adapters .....

- **Internet (IP) addresses of each link in the host**
  - **Example:** HOME---internet_addr--- link_name

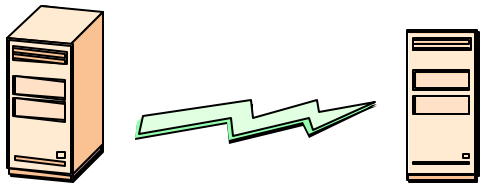
    HOME

    9.81.10.5 OSTR5510

    192.168.16.5 LINKMVS
- **Start all the defined devices**
  - **Example: START---device_name**

    START *OSA5510*

    START *LINKMVS*



- For each device defined a entry must be made in the TCP/IP profile to identify the home address and the associated link names.
- Each device defined must also be started.

TCP/IP Updates

- **Autolog statements**
  - **comment out any AUTOLOG statements for standard TCP/IP daemons (FTPSERVE, TELNET, etc)**
  - **add AUTOLOG statements for the firewall kernel**
    example: AUTOLOG
    ***FWKERN* ; OS/390 Firewall Kernel**
    ENDAUTOLOG
- **Identify TCP/IP profile as a firewall**
  - **IPCONFIG FIREWALL**
- **Enable transfer of data between networks**
  - **IPCONFIG DATAGRAMFWD**

- When implementing multiple firewall stacks, the FWKERN autolog appears only in one profile

- The IPCONFIG statement is used to update the IP layer of TCP/IP. Adding FIREWALL to this statement, identifies this TCP/IP host is to be used as a network firewall.

  Reference: IBM Redbook, OS/390 Firewall Technology manual and the "IP Configuration Guide", SC31-8513-00.

TCP/IP Ports

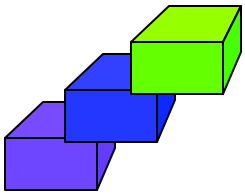
- **Comment out any PORT statements for the standard TCP/IP daemons**
  - **define port reserves for Firewall Technologies daemons**
    - Example:

```
PORT
   20  TCP OMVS   NOAUTOLOG  ; Firewall FTP Proxy
                               server
   21  TCP OMVS               ; Firewall FTP Proxy
                               server
 1080  TCP OMVS               ; Firewall Socks
                               Server
 1014  TCP OMVS               ; Config Server
```

- Ports need to be defined for the Firewall daemons. If other standard TCP/IP daemons are also running there may be other ports defined.
- You need to ensure that a TCP/IP daemon does not start before the firewall daemons because they could use the same port that the firewall daemons are expecting to use.

Firewall Stack

- **Firewall stack**
  - **FWSTACK - define firewall stacks for each one configured**
    - Example: **fwstack cmd=add stack=*stackname* force=yes**



- FWSTACK is used to add, delete,rename or list the stacks.

- The parameter "force" has a default of NO, however, YES must be used when;
  - adding a new stack
  - deleting a stack that is currently active
  - renaming an active stack

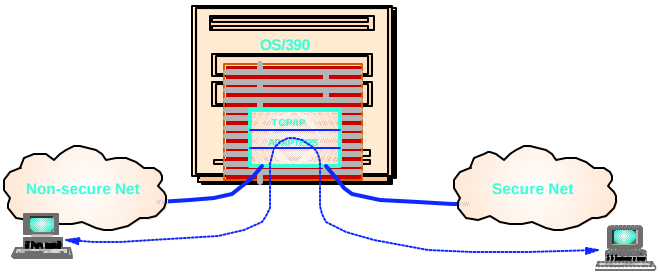
  - This stack name is usually the same as the TCPIP procedure name but not always. The firewall actually checks what is defined in BPXPRMxx. In this member there is a definition for INET or CINET, depending on which one you are using. If you are using CINET, there is a parameter called SUBFILESYSTYPE (xxxxx). In this field is what the FW actually looks at to verify if the TCP/IP stack is up and running. You may have the same parameter (SUBFILESYSTYPE) defined if you are using INET, however, this parameter is not required with INET.

Configuration Files

- **Copy default configuration files from /usr/lpp/fw/etc/security to /etc/security**
  - fwaudio.cfg - real audio
  - fwdaemon.cfg - firewall daemons
  - fwobjects.cfg - object definitions
  - fwservices.cfg - services
  - fwsocks.cfg - socks rules
  - fwrules.cfg - default filter rules
  - fwahtran.cfg - authentication
  - fwesptran.cfg - encryption
  - fwkeypol.cfg - key policy
  - fwkeyprop - key proposals
  - fwkeyring - key ring
  - fwkeytran.cfg - key definition
  - fwdatapol.cfg - data policy
  - fwdataprop.cfg - data proposals
  - fwdyntun.cfg - dynamic tunnel
  - fwguicmds.En_US or fwguicmds.Ja_JP

Adapters

- Firewall Technologies for OS/390 assumes a minimum of two adapters;
  - one secure and one non-secure
  - additional adapters may be either secure or non-secure
  - any adapter not specifically identified as secure is assumed non-secure
  - differentiates customer's internal (secure) network from all external (non-secure) networks



Identification of Secure Adapters

- **To list the adapters attached to the Firewall machine**
  - **fwadapter cmd=list [addr=x.x.x.x]**

    9.82.10.5 Non-Secure Interface OSTR5510

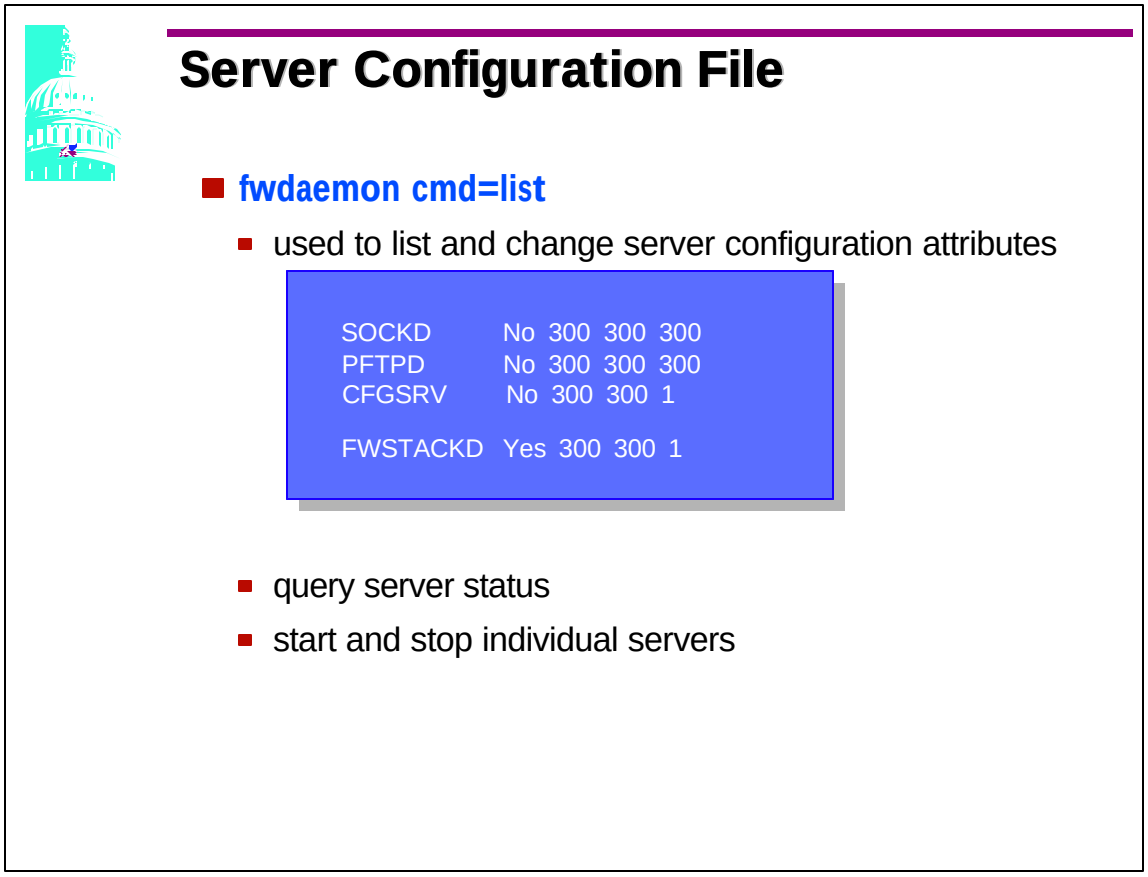
    192.168.16.5 Non-Secure Interface LINKMVS
- **To set the secure/nonsecure state of the adapter:**
  - **fwadapter cmd=change addr=192.168.16.5 state=secure**
  - **fwadapter cmd=list**

    9.82.10.5 Non-Secure Interface OSTR5510

    192.168.16.5 Secure Interface LINKMVS

- fwadapter cmd=list [addr=x.x.x.x]
  >cmd=list , list all adapters attached to this machine and identifies each as being a secure or nonsecure
  >addr=list only the indicated adapter address

- fwadapter cmd=change addr=192.168.16.5 state=secure
  >cmd=change, sets the secure/nonsecure state of the adapter answering to the specified IP address
  >addr= address of the adapter to change
  >state=specifies whether the adapter is to be tagged as being secure or nonsecure



- Firewall servers run in their own address spaces. They are controlled by the Control Task running in the firewall kernel (fwkern) address space.
- FWKERN address space must be started (through operator console) before other servers are started.
- To change or list the server configuration attributes use the FWDAEMON command
- Server Configuration;
  1.name of the server
  2.whether server starts when FWKERN is started
  3.maximum time in seconds for the server to complete initialization (60 - 1800)
  4.number of seconds between restart attemps for the server (60 - 1800)

Enable Firewall Services

- IPL system to activate Firewall changes
- From operator console start the Firewall kernel and selected daemons
  - S FWKERN
- View started servers
  f fwkern,query all

FIR1 STC00298 ICAM1001i Firewall daemon FWSTACKD status is READY and process id is 6710887

- FWKERN command is used to start, stop and query servers. The FWKERN command can also be used to start the firewall kernel without starting any servers, **F FWKERN,parm='-nofw' .**
- When the firewall is first started only the SYSLOGD and the FWSTACKD are started.
- If there is an AUTOLOG statement for FWKERN in xxx.PROFILE.TCPIP, then there is no need to start FWKERN.
- The SYSLOG daemon will only start successfully if the OS/390 Security Server has been purchased. If the Security Server is not used, the TCP/IP SYSLOG can still be used, however, it will not have the firewall enhancements.
- TCP/IP must be started before starting the Firewall kernel.

Logging

- **Firewall logs use the Communication Server SYSLOG**
  - **log firewall events in the form of system messages**
  - **send results to;**
    - log files in HFS
    - OS/390 System Management Facility (SMF)
- **Log events based on three factors:**
  - facility (or origin) of the event
  - priority (or severity) of the event
  - action to be taken with the event

- The most important step to securing an environment is to make sure you log as much activity as you can. The logs are the only records an installation will have of intrusions or attacks. If they don't exist or they are not reviewed an installation will have no idea if they have unwanted visitor's.