This chapter provides guidelines to help you configure Application Program Interfaces (APIs) for use with OpenEdition (UNIX System Services) and TCPaccess. This chapter includes this section:

- Configuring for OpenEdition (UNIX System Services)
  Describes information on how to configure applications to use OpenEdition (UNIX System Services).
- SYS1.PARMLIB (BPXPRMxx)
  Describes the BPXPRMxx member.
- Using the Cisco IOS for S/390 Stack Only
  Describes configuration changes for Cisco IOS for S/390.
- Common Inet Support
  Describes configuration for Common Inet Support.
- Startup Configuration
  Describes startup considerations.
- Host Name/Address Resolution
  Describes host name considerations.
Configuring for OpenEdition (UNIX System Services)

Some of the configuration information for Cisco IOS for S/390 OpenEdition (UNIX System Services) Converged socket support is provided in this chapter. For complete information on configuration and using Cisco IOS for S/390 UNIX System Services Converged Socket support, refer to the *Cisco IOS for S/390 C/Socket Programmer’s Reference* and IBM’s *MVS/ESA: Planning Open Edition MVS, BPXB2 MO4/5 SC23-3015-01/02*.

SYS1.PARMLIB (BPXPRMxx)

To use Cisco IOS for S/390 OpenEdition (UNIX System Services) sockets, you will need to edit the configuration member for UNIX System Services, BPXPRMxx. This member is in the MVS/ESA system configuration data set, SYS1.PARMLIB.

For detailed information regarding the UNIX System Services MVS PARMLIB Member, refer to UNIX System Services MVS manual SC23-3015. The format of this member changes in version 5.2.2 of OpenEdition.

You can configure your UNIX System Services support in either of two ways. You can use the Cisco IOS for S/390 stack only, or you can configure to use common Inet support (more than one physical file system).

Using the Cisco IOS for S/390 Stack Only

This section describes the statements you will need to change in BPXPRMxx to use only the Cisco IOS for S/390 stack.

FILESYSTYPE Statement

To identify the Cisco IOS for S/390 PFS (Physical File System) to UNIX System Services MVS, you must add the following statement to BPXPRMxx:

```
FILESYSTYPE TYPE (filesysname) ENTRYPPOINT (T010PFS)
[PARM (SYSID (xxxx))]
```

Syntax Description

- **TYPE(filesysname)**
  - Specifies the name of the PFS. This name may be configured and can be 1 to 8 characters long. This same name must also be specified on the NETWORK statement (read the NETWORK Statement).

- **ENTRYPOINT(T010PFS)**
  - Specifies the load module entry point for initialization.

- **PARM (SYSID (xxxx))**
  - Specifies the subsystem id of Cisco IOS for S/390 (xxxx).
  - Default: ACSS
NETWORK Statement

You will also need to change the NETWORK statement to assign socket domains or address families to Cisco IOS for S/390 PFS:

```
NETWORKTYPE (filesysname)
DOMAINNAME (AF_INET)
DOMAINNUMBER (domain_number)
MAXSOCKETS (num_sockets)
```

Syntax Description

- **NETWORKTYPE (filesysname)** Specifies the name of the PFS. This name must match the TYPE operand used on the FILESYSTYPE statement that defined the PFS (see FILESYSTYPE statement).
- **DOMAINNAME (AF_INET)** Specifies the domain, or address family. This can be any name you choose. AF_INET is the default.
- **DOMAINNUMBER (domain_number)** Specifies the numeric value of the domain that is passed by programs that call socket(). A value of 2 should be used, but is not required. Certain name resolution functions will not work if domain 2 is not defined.
- **MAXSOCKETS (num_sockets)** Specifies the maximum number of currently active sockets that are supported.

Common Inet Support

This section describes the changes that you need to make to BPXPRMxx to support more than one AF_INET physical file system.

**Caution**  You should be aware that there are performance considerations when using common Inet support and it is not recommended.

For more information on common Inet support, read MVS/ESA: Planning Open Edition MVS, BPSB2 MO4/5 SC23-3015-01/02.

FILESYSTYPE Statement

To use common Inet support, you must add the following statement to BPXPRMxx:

```
FILESYSTYPE TYPE (filesysname) ENTRYPPOINT (BPXTCINT)
[PARM (SYSID (xxxx))]
Syntax Description

**TYPE (filesysname)** Specifies the name of the PFS. This name may be configured and can be 1 to 8 characters long. For common Inet support, set this to CINET.

**ENTRYPOINT (BPXTCINT)** Specifies the load module entry point for initialization.

**PARM (SYSID (xxxx))** Specifies the subsystem ID of Cisco IOS for S/390 (xxxx). Default: ACSS

**SUBFILESYSTYPE Statement**

To use common Inet support, you must add the following statement to BPXPRMxx for each PFS you want to use:

```
SUBFILESYSTYPE NAME (TCPPFS1) TYPE (CINET) ENTRYPOINT(T010PFS)
[PARM (SYSID (xxxx))]
```

Syntax Description

**TCPPFS1** Specifies the name of the PFS.

**CINET** Specifies to use common Inet support. TYPE names must match on all FILESYSTYPE and SUBFILESYSTYPE statements.

**T010PFS** Specifies the load module entry point for initialization.

**PARM (SYSID (xxxx))** Specifies the subsystem id (xxxx). Default: ACSS

**NETWORK Statement**

The following NETWORK statement is for common Inet support only.

```
NETWORKTYPE (CINET)  DOMAINNAME (AF_INET)
  DOMAINNUMBER (2)
  MAXSOCKETS(4096)
  INADDRANYPORT(4096)
  INADDRANYCOUNT(512)
```

Syntax Description

**NETWORKTYPE (CINET)** Specifies to use common Inet support.

**DOMAINNAME (AF_INET)** Specifies the domain, or address family. This can be any name you choose
Startup Configuration

UNIX System Services must be configured to execute program T010PFS upon start-up. To do this, the PFSLOAD data set must be added to the STEPLIB DD statement in the OMVS cataloged procedure. Read SYS1.PARMLIB (BPXPRMxx) for details on configuring UNIX System Services.

Host Name/Address Resolution

If you will be using Cisco IOS for S/390 UNIX System Services sockets, the method used to resolve host names and addresses is different than that used for the Cisco IOS for S/390 socket API. OpenEdition (UNIX System services) MVS version 1.2 uses the LE/370 version 1.3 or 1.4 runtime libraries to perform certain socket related functions such as gethostname() and getprotobyname(). To perform this functionality, the LE/370 runtime library (RTL) reads specific MVS data sets to map services to names and to obtain domain name resolution configuration information.

Refer to the Cisco IOS for S/390 C/Socket Programmer’s Reference for a complete description of the method and the data set members that you will need.