

CA Gen

CICS Multi Socket Server Listener

November 30, 2010

Table of Contents

Multi Socket Server Listener Overview	3
Sample Flow/Execution Diagram	4
Defining and Installing the Listener.....	5
TIRSLEXT – CICS Socket Server Listener Exit	7
TIRSLTMX – CICS Sockets Server Listener TIMEOUT Exit	11

Multi Socket Server Listener Overview

The new CICS TCP/IP Direct Connect implementation (TISRVMML) differs from the earlier version (TISRVLIS) in the way that it manages multiple socket requests simultaneously while the prior listener managed single thread requests. The flow diagram shows the logic of the listener application.

With each pass through the logic, the TISRVMML will check for a new request, accept that request as well as process any waiting requests for sockets that have already been accepted. The actions that can be done on an accepted socket are:

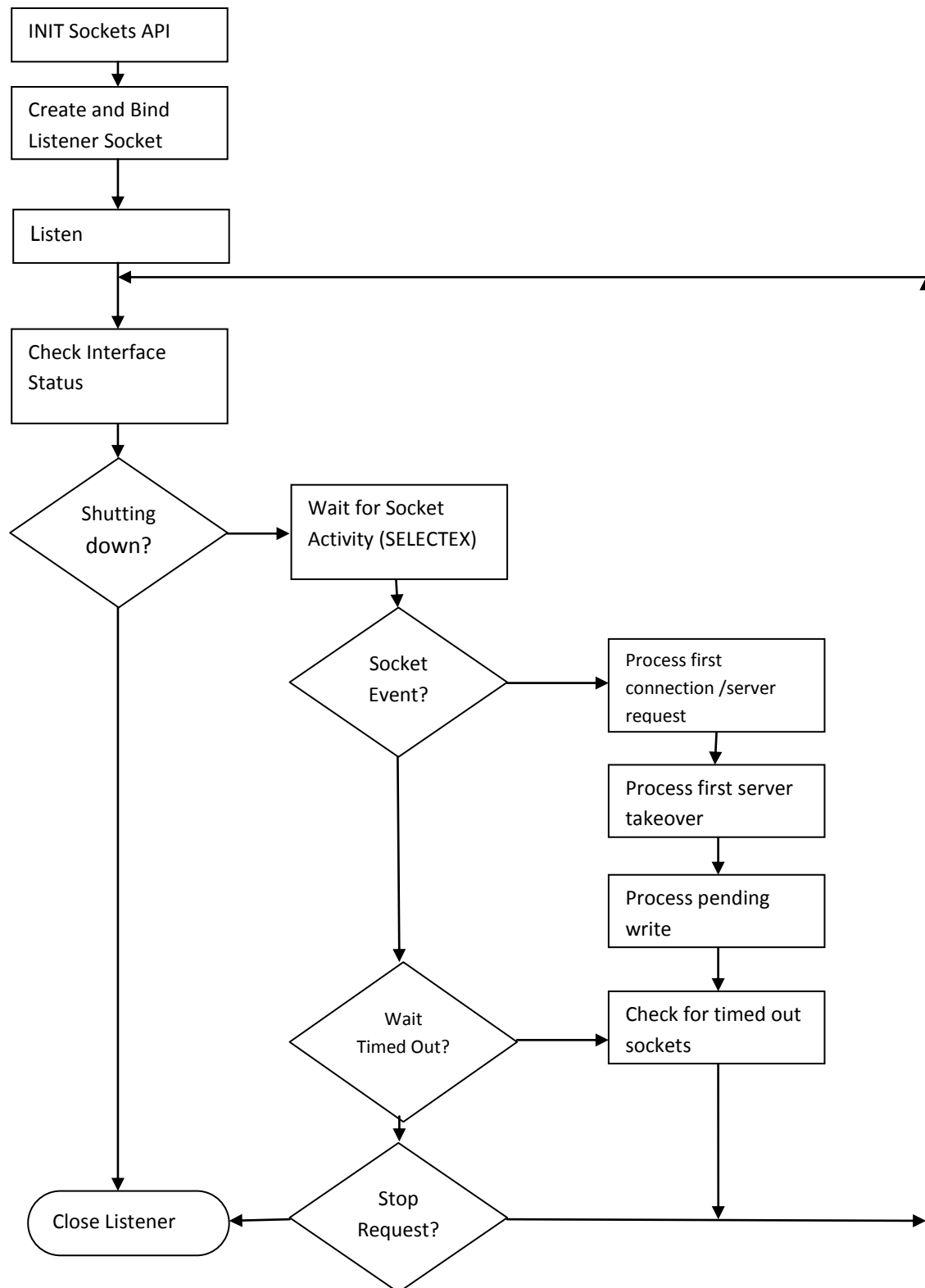
- Receive the common format header data
- Call the TIRSLEXT exit
- Give the socket to the server
- Start the server
- Check that the server took the socket
- Write an error message if needed

Like the TISRVLIS listener, the TISRVMML listener uses the ACCTIME, REATIME and GIVTIME timeouts that are defined in the listener's EZACONFG file definition. It also allows these values to be overridden in the TIRSLEXT and TIRSLTMX exits should microseconds need to be used for more granularity. Other than the ACCTIME, the TISRVMML listener uses timeouts to determine if wait time for a process has expired instead of as a timeout for a socket select function.

The TISRVMML listener allows the server to be routed to a CICS region different from the one where the TISRVMML is executing. Routing may be invoked by using the SYSID where the server is execute or by using Distributed Routing. The SYSID may be supplied as the variable "CICS" in the TIRSLEXT user exit. Distributed Routing may be implemented via the DSRTPGM program or the CICS transaction definition.

The TIRSLEXT exit has a new parameter which allows remote servers to be started with the NOCHECK option. The default value is Y which will cause the server to be started with CHECK. The exit can be modified to set the flag based on the server trancode. Using the CHECK option may impact overall performance so some analysis may need to be performed to determine the tradeoff in notification when servers fail to start weighed against overall performance impact in performing these checks.

Sample Flow/Execution Diagram



Defining and Installing the Listener

To install the TISRVMSL listener into CICS, you will define it as a Standard Listener. The following record is required in the EZACONFG file:

- TYPE=LISTENER
- TRANID=TISM*
- TRANTRN=NO
- TRANUSR=NO

The parameters SECEXIT and WLM group names should be left blank.

CICS definitions are required for:

- The Listener transaction TIML*
- The programs TISRVMSL, TIRSLEXT and TIRSLTMX

The following information can be used to run DFHCSDUP (add a group of your choice):

```
DEFINE TRANSACTION(TIML*)
```

```
DESCRIPTION(CA Gen CICS Multi Sockets Server Listener Transaction)
```

```
PROGRAM(TISRVMSL)
```

```
TASKDATALOC(ANY)
```

```
TASKDATAKEY(CICS)
```

```
DEFINE PROGRAM(TISRVMSL)
```

```
DESCRIPTION(CA Gen CICS Multi Socket Server Listener Program)
```

```
LANGUAGE(ASSEMBLER)
```

```
DATALOCATION(ANY)
```

```
EXECKEY(CICS)
```

```
DEFINE PROGRAM(TIRSLEXT)
```

```
DESCRIPTION(CA Gen CICS Socket Server Exit Program)
```

```
LANGUAGE(ASSEMBLER)
```

```
DATALOCATION(ANY)
```

```
EXECKEY(CICS)
```

```
DEFINE PROGRAM(TIRSLTMX)
```

```
DESCRIPTION(CA Gen CICS Socket Server Timeout Program)
```

```
LANGUAGE(ASSEMBLER)
```

```
DATALOCATION(ANY)
```

```
EXECKEY(CICS)
```

*This transaction identifier can be something other than TIML but the same name must be used in the EZACONFG file and in the CICS definition.

TIRSLEXT – CICS Socket Server Listener Exit

The CA Gen CICS Sockets Server Listener Exit is used by the z/OS CICS user-written Listener program TISRVLIS, one of the CA Gen TCP/IP implementations.

Note: The TIRSLEXT exit routine is invoked by CICS Sockets Server Listener program TISRVLIS only. The TIRSLEXT exit is not invoked by CA Gen code.

Source Code

The source code for this exit is in CA Gen SAMPLIB library, in member TIRSLEXT. The sample exit provided is written in ASSEMBLER and is invoked by TISRVLIS and TISRVMISL via an EXEC CICS LINK API call.

Data used and returned by TIRSLEXT is passed in the COMMAREA as follows:

1. ASCII transaction code
2. EBCDIC transaction code
3. ASCII userid
4. EBCDIC userid
5. ASCII client code page
6. EBCDIC client code page
7. EBCDIC password
8. Enhanced Security Flag
9. Enhanced Type Flag
10. Check Flag
11. Security Flag
12. CICS System Id
13. Client Port
14. Client IP Address
15. Listener GIVESOCKET/TAKESOCKET TIMEOUT - Seconds
16. Listener GIVESOCKET/TAKESOCKET TIMEOUT – Microseconds

Purpose

This exit translates the user ID to the server's EBCDIC code page but it provides the opportunity to:

- modify the user ID supplied by the client application before it is translated
- validate the supplied, optionally modified, and translated security data through an External Security Manager
- verify that an enhanced security buffer exists

- perform customer-specific security authorization
- provide a System ID to use to transaction route the server via the SYSID operand of the EXEC CICS START API call

Arguments

The following table gives a brief description of each of the arguments.

Name	I/O	Description
DFHEIBLK	input/output	CICS DFHEIBLK, automatically included by the CICS translator.
COMMAREA	input/output	CICS communications area with the following contents:
ASCTRAN	input	The server load module ASCII transaction code.
EBDTRAN	input	The server load module EBCDIC transaction code.
ASCUID	input	The ASCII userid as supplied by the client.
EBDUID	output	The EBCDIC translated version of the ASCII userid sent by the client, modified if required.
ASCCOPG	input	The ASCII code page as supplied by the client.
EBDCODPG	input	The EBCDIC code page as supplied by the client.
EBDPSW	input	The EBCDIC password as supplied by the client.
ENHSCFLG	input	A flag indicating the presence of enhanced security data.
ENHTPFLG	input	A flag indicating if the security data is encrypted.
CHECKFLG	input/output	A flag indicating if NOCHECK should be used on CICS START
RESERVED	none	Reserved.
SECFLAG	output	A flag indicating the result of the security validation. The following values are supported: ZERO - SFUIDAUT (USERID-IS-AUTHORIZED) 709 – ECTYPINV (ENCRYPTION-FLAG-NOT-SET) 902 - SFINVUID (INVALID-USER-ID) 903 - SFINVPSW (INVALID-PSWD) 904 - SFNTAUTF (NOT-AUTHORIZED-FORFUNCTION) 905 - SFSYSUNV (SYSTEM-UNAVAILABLE) 906 - SFUIDSUS (USERID-SUSPENDED) 907 - SFPSWEXP (PSWD-EXPIRED) 908 - SFERROR (INTERNAL-ERROR)
CICS	output	CICS System Id to route the server to.
PORT	input	The TCP/IP Port of the client.
IPADDR	input	The TCP/IP Address of the client.
LGVTIMS	input/output	The GIVESOCKET Timeout - seconds
LGVTIMM	output	The GIVESOCKET Timeout - microsec

Return Code

No explicit return code value is provided by the TIRSLEXT exit. CICS provides information about the return from the CICS Program Link that indicates if TIRSLEXT exists and its execution was successful. In addition, the values contained in the parameter list addressed by the COMMAREA are returned.

Default Processing

The default processing of this exit is to translate the ASCII user ID to EBCDIC, set the Security Flag to USERID-IS-AUTHORIZED. In addition the exit will return the value received as input for the CHECKFLG and the Listener GIVESOCKET/TAKESOCKET TIMEOUT – Seconds.

Customizing the Exit

This is an exit for TISRV LIS and TISRVMSL not the server application. The exit is invoked after TISRV LIS or TISRVMSL obtains a copy of the cooperative buffer header sent by the client and extracts the data required to start the server.

The ASCII user ID, as supplied by the client, can be modified by those users that include special characters (excluding A-Z and 0-9) in their user ID fields. This ASCII user ID is then translated to EBCDIC in the exit. If conversion of the security data to uppercase is required, it can be done in this exit.

The variable 'CICS' can be set to the SYSID of the CICS region where the server is to be routed to. If the value of the variable 'CICS' returned to TISRV LIS or TISRVMSL is the same as the SYSID of the CICS region where TISRV LIS or TISRVMSL is executing, the EXEC CICS START API command will not specify the SYSID operand (a local START). If the value of the variable 'CICS' returned is different, the returned value of the variable 'CICS' is used in the START.

The Sockets Server implementation allows the server to be routed to a CICS region different from the one where the TISRV LIS listener is executing. Routing may be invoked by using the SYSID where the server is to execute, as described above, or by using Distributed Routing. Distributed Routing may be implemented via the DSRTPGM program or the CICS transaction definition.

The CICS Sockets Interface must be active in the CICS region the server will be routed to but an active listener is not required in that CICS region.

On input, the Listener GIVESOCKET/TAKESOCKET Timeout – Seconds value is the value entered for the GIVTIME parameter in the EZACONFG file. This value is used to populate the TIMEOUT Seconds parameter of the SELECT Sockets API call used by the Listener to wait for the socket to be taken (using the GIVESOCKET/TAKESOCKET process) by the server. Since the EZACONFG file does not capture the equivalent Microseconds value, the only opportunity to provide a value for this parameter is in this exit.

On input, the Check Flag is set to Y which means to start the server transaction with CHECK. If you wish a remotely started transaction to be started with NOCHECK, set the flag to N. You can also use the EBDTRAN to decide the value of the flag.

This exit includes sample code that can be used to validate the user ID and password, the presence of enhanced security data, the presence of encrypted data, or the presence of encrypted and enhanced security data, to modify the Listener Givesocket/Takesocket Timeout value and the check flag. Modify the exit to execute the desired validation code.

Building on z/OS

For more information about modifying the exit, see the comments in the SAMPLIB member TIRSLEXT.

Build the exit as follows:

1. Copy the default exit from the AllFusion Gen SAMPLIB library to one of your libraries.
2. Add the desired new code to the copied exit. Do not change the CSECT name (entry point).
3. This exit uses EXEC CICS API calls so it must be processed by the CICS translator. The exit does not include RUNTIME-PARM1 and RUNTIME-PARM2. Do not modify the COMMAREA definition (COMMAREA DSECT).
4. Assemble and link-edit the exit. Use link-edit parameters
'RENT,REUS,AMODE(31),RMODE(ANY)' and control statements:
ORDER DFHEAI,TIRSLEXT
INCLUDE ASMOBJ(TIRSLEXT)
INCLUDE SYSLIB(DFHEAI)
INCLUDE SYSLIB(DFHEAIO)
ENTRY TIRSLEXT
MODE AMODE(31),RMODE(ANY)
NAME TIRSLEXT(R)
5. Ensure the new TIRSLEXT exit is made available in the DFHRPL concatenation and if applicable issue a NEW COPY PROG(TIRSLEXT) in CICS. The Multi Sockets Server listener (TIML) does not need to be stopped and restarted for the new TIRSLEXT exit to be used.

TIRSLTMX – CICS Sockets Server Listener TIMEOUT Exit

The AllFusion Gen CICS Sockets Server Listener Timeout Exit is used by the z/OS CICS user-written Listener program TISRVLIS, one of the AllFusion Gen TCP/IP implementations.

Note: The TIRSLTMX exit routine is invoked by CICS Sockets Server Listener program TISRVLIS only. The TIRSLTMX exit is not invoked by AllFusion Gen servers.

Source Code

The source code for this exit is in AllFusion Gen SAMPLIB library, in member TIRSLTMX. The sample exit provided is written in ASSEMBLER and is invoked by TISRVLIS via an EXEC CICS LINK API call.

Data used and returned by TIRSLTMX is passed in the COMMAREA as follows:

1. TISRVLIS EBCDIC transaction code
2. CICS APPLID
3. Listener ACCEPT TIMEOUT – Seconds
4. Listener ACCEPT TIMEOUT – Microseconds
5. Listener READ TIMEOUT – Seconds
6. Listener READ TIMEOUT – Microseconds
7. Listener ERROR TIMEOUT – Seconds
8. Listener ERROR TIMEOUT – Microseconds
9. IP Address Flag

Purpose

This exit can be used to customize the Listener's SELECT API Calls timeout value that controls the wait time involved with accepting new connections (ACCEPT) and reading input data. The corresponding values entered in the EZACONFG file are passed to this exit in the seconds field. These values can remain as they are or can be modified. In addition an extra value may be provided at the microsecond level.

The exit provides the opportunity to:

- modify the value entered for the ACCTIME parameter in the EZACONFIG file. This value is used to populate the TIMEOUT Seconds parameter of the SELECT Sockets API call used by the listener to wait for new connections

- provide the value to populate the TIMEOUT Microseconds parameter of the SELECT Sockets API call used by the listener to wait for new connections
- modify the value entered for the REATIME parameter in the EZACONFIG file. This value is used by the listener to populate the TIMEOUT Seconds parameter of the SELECT Sockets API call used by the listener when reading input data
- provide the value to populate the TIMEOUT Microseconds parameter of the SELECT Sockets API call used by the listener when reading input data
- modify the value defaulted in the listener program. This value is used by the listener to populate the TIMEOUT Seconds parameter of the SELECT Sockets API call used by the listener when reading input data when an error has occurred.
- provide the value to populate the TIMEOUT Microseconds parameter of the SELECT Sockets API call used by the listener when reading input data when an error has occurred.
- provide the flag to turn on writing the IP Address as part of some socket and CICS error messages.

Arguments

The following table gives a brief description of each of the arguments.

Name	I/O	Description
DFHEIBLK	input/output	CICS DFHEIBLK, automatically included by the CICS translator.
COMMAREA	input/output	CICS communications area with the following contents:
SLTRANID	input/output	The Listener EBCDIC transaction code
CICSAPPL	input/output	APPLID of the CICS where the Listener runs
ACCTOUTS	input/output	The Listener ACCEPT TIMEOUT value in seconds
ACCTOUTM	input/output	The Listener ACCEPT TIMEOUT value in microseconds
REATOUTS	input/output	The Listener READ TIMEOUT value in seconds
REATOUTM	input/output	The Listener READ TIMEOUT value in microseconds
ERRTOUTS	input/output	The Listener ERROR TIMEOUT value in seconds
ERRTOUTM	input/output	The Listener ERROR TIMEOUT value in microseconds
IPADRFLG	input/output	The IP address flag

Return Code

No explicit return code value is provided by the TIRSLTMX exit. CICS provides information about the return from the CICS Program Link that indicates if TIRSLTMX exists and its execution was successful. In addition, the values contained in the parameter list addressed by the COMMAREA are returned.

Default Processing

The default processing of this exit is to check if the TISRVLIS trancode matches the EIBTRNID and return.

Customizing the Exit

This is an exit for TISRVLIS, not the server application. The exit is invoked before TISRVLIS accepts a connection from a client and has access to the data sent by the client.

The SLTRANID and the CICSAPPL can be used to decide if the TIMEOUT values provided by the EZACONFG file to be used by the Listener suffice or must be changed.

The Listener Accept TIMEOUT Seconds, as received by this exit contains the value specified in the EZACONFG file for the ACCTIME parameter. This value can be changed, including zeroed out. In addition, if a value smaller than 1 second is required the Listener Accept TIMEOUT Microseconds variable can be used.

The Listener Read TIMEOUT Seconds, as received by this exit contains the value specified in the EZACONFG file for the REATIME parameter. This value can be changed, including zeroed out. In addition, if a value smaller than 1 second is required the Listener Read TIMEOUT Microseconds variable can be used.

The Listener Error TIMEOUT Seconds, as received by this exit contains the value specified in the listener code and is 50 microseconds. This value can be changed, including zeroed out. In addition, if a value smaller than 1 second is required the Listener Error TIMEOUT Microseconds variable can be used.

The IP Address Flag is set in this exit to turn on the writing of the IP Address as part of some of the socket and cics error messages. The default is N – do not write the IP Address.

This exit includes sample code that can be used to set the various TIMEOUT parameters as required. Modify the exit to execute the desired code.

Building on z/OS

For more information about modifying the exit, see the comments in the SAMPLIB member TIRSLTMX.

Build the exit as follows:

1. Copy the default exit from the AllFusion Gen SAMPLIB library to one of your libraries.

2. Add the desired new code to the copied exit. Do not change the CSECT name (entry point).
3. This exit uses EXEC CICS API calls so it must be processed by the CICS translator. The exit does not include RUNTIME-PARM1 and RUNTIME-PARM2. Do not modify the COMMAREA definition (COMMAREA DSECT).
4. Assemble and link-edit the exit. Use link-edit parameters
 'RENT,REUS,AMODE(31),RMODE(ANY)' and control statements:
 ORDER DFHEAI,TIRSLTMX
 INCLUDE ASMOBJ(TIRSLTMX)
 INCLUDE SYSLIB(DFHEAI)
 INCLUDE SYSLIB(DFHEAI0)
 ENTRY TIRSLTMX
 MODE AMODE(31),RMODE(ANY)
 NAME TIRSLTMX(R)
5. Ensure the new TIRSLTMX exit is made available in the DFHRPL concatenation and if applicable issue a NEW COPY PROG(TIRSLTMX) in CICS. The Multi Sockets Server listener (TIML) needs to be stopped and restarted for the new TIRSLTMX exit to be used.