HCL Z Common Components

Customization Guide and User Guide

Version 1.1.2

Note

Before you read this document, look at the general information under Notices on page xxxvi.

Edition notice

This edition (published December 2023) applies to Version 1 Release 1 Modification Level 2 of HCL Z Common Components and to Version 1 Release 1 Modification Level 2 of HCL Z Data Tools (program number HCL19OP1220), and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

PDF documentation

PDF documentation is also available by opening any online topic and clicking the 📖 icon.

HCL Z Common Components Customization Guide and User Guide V1.1.2 (English)

HCL Z Common Components Customization Guide and User Guide V1.1.2 (Japanese)

Preface

This document provides information about HCL Z Common Components.

It is intended for people who are responsible for installing, configuring, and using Z Common Components. You should have a working knowledge of:

- z/OS® operating system
- system programming
- configuration of servers

In these topics, Z Common Components is also referred to as ZCC.

Summary of changes

This edition of the document provides information applicable to HCL Z Common Components Version 1 Release 8. Changes to the latest release are indicated by a "|" change bar in the left margin of the page in the PDF format only.

December 2023

This edition of the documentation contains minor clarifications and corrections from the previous version.

Chapter 1. Introduction to HCL Z Common Components

HCL Z Common Components consists of these major features:

- ZCC server
- Interactive Panel Viewer

ZCC server

The ZCC server is an extensible server program that runs on a z/OS® system to serve clients. Multiple clients can connect to a single instance of the server program and request a service by invoking a specific extension of the server. The server needs to be customized to install various extensions. Without installing the extensions, the ZCC server program alone does not serve any purpose.

For more information about configuring the product-specific extensions to the ZCC server, see the product-specific customization guide.

ZCC server is used by Z Data Tools for CICS® and Z Data Tools Remote Services. For more information about configuring the product-specific extensions to the ZCC server, see the *Z Data Tools Customization Guide*.

Interactive Panel Viewer

The Interactive Panel Viewer feature enables ISPF-based applications to display panels under CICS®.

The Interactive Panel Viewer feature is used by Z Data Tools for CICS®. See the *Z Data Tools Customization Guide* for details on customization.

Chapter 2. ZCC server overview

ZCC server runs a process that identifies a connection request on a specific port. ZCC server can be started manually, or during an IPL, by running a customized procedure. A sample procedure, HFISRV1, is supplied in the sample library hlq.SHFISAM1.

Multiple servers might be simultaneously run, provided different port numbers are used for each server.

For participating products that use the ZCC server, the server negotiates SSL-encrypted communications if configured to do so, then verifies the client user ID, password, or passphrase. If valid, the server creates a new process for that user.

The ZCC server consists of a main program module, HFISRV, and supporting message and API-related modules.

HFISRV requires a parameter string **'port family trace'** where:

port

Describes the port number that is used to bind and accept incoming connections.

family

The addressing family to bind to. For example, `AF_INET`, or `AF_INET6`.

trace

N, T, D, U, or omitted. This parameter specifies the level of tracing to be performed by the server, and is intended only for diagnostic purposes. N is for no tracing, while T or D produce HFITRACE, or STDOUT, outputs of undocumented messages that show flow and processing details for diagnostic purposes. U produces trace entries showing user connections to participating HCL Z products.

Sample server procedure

The ZCC server is recommended to run as a started task, although it might be run as a job.

A sample procedure, HFISRV1, is supplied in the hlq.HFISAM1 data set. Copy the procedure to your procedure library.

```
//HFISRV1  PROC PORT=2800,FAMILY='AF_INET',TRACE=N
//********************************************************************
//* Copyright = Licensed Materials - Property of HCL                 *
//*                                                                  *
//*             19OP1220 HCL Z Common Components                     *
//*                                                                  *
//*             (C) Copyright IBM Corp. 2006, 2017.                  *
//*             All Rights Reserved.                                 *
//*             (C) Copyright HCL Technologies Ltd. 2017, 2023.      *
//*             All Rights Reserved.                                 *
//*                                                                  *
//*             US Government Users Restricted Rights -              *
//*             Use, duplication or disclosure restricted by         *
//*             GSA ADP Schedule Contract with IBM Corp.             *
//*                                                                  *
//* Status = HCL Z Common Components, Version 1 Release 1             *
//*                                                                  *
//********************************************************************
```

```
//* FAMILY=AF_INET|AF_INET6   for TCP/IP V4 or V6 socket and bind
//* TRACE =N|D|U              No server trace, detailed trace or
//*                           user connection trace
//*
//* This is not a complete JCL procedure. It requires customisation
//* steps before running. To customise,
//* 1. Customise the HFICONFG member
//* 2. Customise and run the HFIMKDIR sample job to match
//* 3. replace HFI with your high level qualifier for ZCC product
//* 4. Uncomment and replace CEE for your hlq for the LE C runtime
//*    if SCEERUN is not in the site linklist
//*
//* 5. If wanting to use a specific cipher string, uncomment and
//*    modify the ENV variable setting
//     SET ENV=''
//*    SET ENV='ENVAR(GSK_V3_CIPHER_SPECS=33)'  <==modify to suit
//RUN      EXEC PGM=HFISRV,REGION=40M,
//            PARM=('&ENV/&PORT &FAMILY &TRACE')
// SET HFI=HFI                                  <== Update HLQ
//* Common component authorised library
//STEPLIB  DD DISP=SHR,DSN=&HFI.SHFIMODA        <== ZCC APF LIBRARY
//*        DD DISP=SHR,DSN=CEE.SCEERUN          <== LE C RUNTIME
//SYSPRINT DD SYSOUT=*
//HFITRACE DD SYSOUT=*                          <== OUTPUT if Tracing
//STDOUT   DD SYSOUT=*
//* Server wide, then participating product configurations
//CONFIG   DD DISP=SHR,DSN=&HFI.SHFISAM1(HFICONFG)
**************************** Bottom of Data **************************
```

Startup, shutdown, and activity tracing

The server is controlled using the START (S), STOP (P) and MODIFY (F) z/OS® system commands. These commands are typically issued on a z/OS® system console.

Use `START procname` to start the server.

Use `STOP procname` to stop the server.

To enable activity tracing, usually as an HCL support request, the following modify command can be used:

```
F procname,APPL=TRACEON
```

To disable activity tracing, the following modify command can be used:

```
F procname,APPL=TRACEOFF
```

To display the release and PTF level of the running server, the following modify command can be used:

```
F procname,APPL=VER
```

Configuration file keyword descriptions

The configuration data might contain line comments. Line comments begin with either an asterisk (*) or a hash/pound (#) character, and continue to the end of the line. When configuration involves data set names that include hash/pound (#), such characters must be escaped using a backslash (\) so that they are not interpreted as comments.

CONFIG=*name*

name is the name of the configuration as specified by the client. At least one configuration is expected with a name of DEFAULT. Other configuration keywords apply to the current CONFIG name, in top-down order.

APPLID=*applid* (Optional)

The ZCC server uses C runtime services to switch user context when spawning processes for requesting clients that provide a valid user ID and password. These services are associated with the OMVSAPPL resource (or the HFIAPPL resource if PASSTK is specified) of the APPL class by default, if the APPL class is active. If this is the case, the authenticating user ID must have READ access to the OMVSAPPL or HFIAPPL resource of the APPL class.

Alternatively, your configuration file can specify APPLID=*applid*, where *applid* is a 1- to 8-character resource name defined to the APPL class. When APPLID is configured, the ZCC server will use the specified APPL class *applid* rather than OMVSAPPL or HFIAPPL.

The APPLID parameter must be specified under the CONFIG=DEFAULT configuration.

PASSTK=*nnn* (Optional)

The server can be configured to use PassTickets to start sessions for authenticated clients. If you specify the PASSTK parameter in your configuration, the server will generate and use PassTickets for requesting clients that provide a valid user ID and a valid password or passphrase.

After successfully connecting to the server, a client can start new sessions for the period in minutes specified by *nnn* without having to re-authenticate. Allowable values are 1 to 720 (12 hours). If PASSTK is specified without a value, the default is 480 (8 hours).

If PASSTK is not specified PassTickets will not be generated or used by the server. This feature is primarily to facilitate multifactor authentication (MFA) clients. See Using PassTickets on page 16 for more details.

The PASSTK parameter must be specified under the CONFIG=DEFAULT configuration.

WORKDIR=/path

The CONFIG=DEFAULT set of parameters needs the WORKDIR=path keyword. This keyword specifies where the server can write semi-permanent (existing at least while the server task is running) files. A sample job, HFIMKDIR is supplied in the sample library to create this path.

ATTLS=YES|NO (Optional)

The Application Transparent Transport Layer Security (AT-TLS) feature of z/OS® Communication Server can be used to secure communications between the ZCC server and connecting clients. See Using AT-TLS for encrypted communications on page 18 for more details.

SSL_REQUIRED=YES|TLSV1.2|TLSV1.3|NO (Optional, default is NO)

Determines whether SSL/TLS encrypted communications are mandatory for the server and the desired protocol level. SSL/TLS communications are achieved by using the System SSL APIs. The default protocol level is TLS 1.2 when YES is specified.

If SSL encryption is used, then the server uses a certificate stored in either a RACF® keystore, when specified via the SSL_KEYRING keyword, or a GSKKYMAN managed key database and certificate for this server as specified in the SSL_CERT keyword or, if that keyword is omitted, at the WORKDIR specified location.

SSL_CERT=/path/keyringfile (optional, for use of user created certificate)

The path and name of a key database that contains a stored certificate that is used by the server. This parameter is passed to the gsk toolkit as the GSK_KEYRING_FILE setting. If this parameter is omitted, the server attempts to create a key database and self-signed certificate as it starts up.

SSL_CERTPW=keyringpw (optional, for use of user created certificate)

The password to be used to access the certificate repository. If omitted, the server uses a default password.

SSL_KEYRING=userid/keyring

If SSL is being used for the server, this configuration option provides the userid and keyring name for a certificate being held in a SAF keyring. The userid should match the ID used when creating the keyring.

SSL_LABEL=labelstring (optional, for use of user created certificate)

The label of the certificate from the key database to be used.

SPAWN_ACCT=accountdata

Allows specification of the account data used for the spawned address space. This is as per the _BPX_ACCT_DATA environment variable discussed in the z/OS® UNIX™ System Services Planning manual.

SPAWN_TIME=nn

Allows specification of the CPU time limit, in seconds, used for the spawned address space.

SPAWN_PROGRAM=PROGRAM

Specification of the program that is launched for the client connection. The server checks the existence of the named program. If you want to specify the name of a z/OS® UNIX™ executable file, rather than a load module in a STEPLIB data set, include the path. Otherwise, the server creates a sticky bit file in the WORKDIR specified location. Sticky bit is the mechanism in the z/OS® UNIX™ file system of indicating that this file is a load library member. The program is launched as a UNIX System Services process, but can be a traditional z/OS® program.

SPAWN_STEPLIB=steplib1:steplib2 (optional)

Allows specification of the run libraries that are used for the spawned address space. Support for continuing library specifications is provided by ending a line with the colon character.

If the run libraries are not all APF authorized, you must ensure that the _BPX_SHAREAS environment variable is set to NO to avoid a potential abend S306. The server will then spawn the participating products in their own address space. You can add a STDENV DD statement to set the environment variable in the server procedure. For example:

```
//STDENV *
_BPX_SHAREAS=NO
/*
```

SPAWN_PARMS_SECTION

This entry marks the beginning of extra parameters that are passed to the spawned process. The contents of this area are determined by the products that use the server.

Launching a TSO environment is provided for by the ZCC server when the SPAWN_PROGRAM is set to HFISRVTE. In such a configuration, the launched process deals with these extra keywords that follow the SPAWN_PARMS_SECTION:

SPAWN_DD=ddname=datasetname1:datasetname2

Specification of a data set or data sets that are allocated with DISP=SHR to the supplied DD name.

SPAWN_DD=ddname=SYSOUT=c

Specification of a sysout allocation that is allocated with the specified class c, to the supplied DD name.

Use of SYSOUT=* is not permitted as the spawned address space is not running as a batch job with a JCL MSGCLASS. Use of SYSOUT=* will result in the server terminating until the configuration is corrected.

SOCKETFIONBIO

Specification that the socket communications run in nonblocking mode.

Specify this keyword only when the application for the particular CONFIG allows or expects it.

TSO_CMD=command;

Specification of a command that is run in the TSO environment. This command typically instigates the launch of the participating products main serving function. This parameter can be repeated as needed for multiple TSO commands.

MIXEDCASEPASS=YES|NO (optional, default is NO)

Determines whether uppercase translation is performed for incoming passwords for this system. If this system supports mixed case passwords, set this to YES and specify this keyword in the CONFIG=DEFAULT section.

SPAWN_REGIONSZ=nnn (optional, default is to inherit the region size of the server)

Determines the region size (in MB) for the launched process. Participating products being launched have their own recommendations for this sizing.

Chapter 3. Customizing the ZCC server

This chapter provides you with instructions on how to customize the ZCC server. In brief, this consists of the following general checklist:

- APF authorize the SHFIMODA library
- Add programs in SHFIMODA to program control
- Add user for server started task
- Add task to STARTED class
- Add sample HFISRV1 to system procedure library
- Permit server user/group to BPX.SERVER facility
- Permit server user/group to CSF* profiles (if used)
- Permit connecting users/groups to OMVSAPPL or an equivalent resource (if used).
- Update sample HFICONFG
- Create matching WORKDIR by running job HFIMKDIR
- Review address space timeout settings
- Configure the TCP/IP stack affinity

Required authorizations

The STEPLIB hlq.SHFIMODA must be APF-authorized.

Associate the started task that is used to run the ZCC server with a user ID that has an OMVS segment. If the BPX.SERVER facility is active give the user ID READ access to it, otherwise the user ID requires superuser access. Make sure write access to the z/OS® UNIX™ directory is available, as specified by the WORKDIR= configuration parameter. Edit and run the job HFIMKDIR in the sample library (HFI.SHFISAM1) to create this directory. Furthermore, any users logging in to the ZCC server require read access to this location. Similarly, if you configure the ZCC server to a key database of your own creation, the ZCC server and any users who log into it require read access to the specified key database.

Products that make use of the SPAWN_JOBNAME configuration keyword require the following authorizations. The user ID of the ZCC server must be permitted to the BPX.SUPERUSER resource of the FACILITY class and must have READ access to the BPX.JOBNAME resource, if it is defined.

The ZCC server uses C runtime services to switch user context when spawning processes for requesting clients that provide a valid user ID and password. These services are associated with the OMVSAPPL resource (or the HFIAPPL resource if PASSTK is specified) of the APPL class by default, if the APPL class is active. If this is the case, the authenticating user ID must have READ access to the OMVSAPPL or HFIAPPL resource of the APPL class.

Alternatively, your server configuration can specify APPLID=*applid*, where *applid* is a user-defined resource name defined to the APPL class. When APPLID is configured, the ZCC server will use the specified APPL class *applid* rather than OMVSAPPL orHFIAPPL. If PassTickets are used, the default resource name is HFIAPPL, however this can also be overridden by the APPLID configuration parameter. In all cases, authenticating users must have READ access to the appropriate resource of the APPL class (if it is active).

If enhanced program security is enabled, at a minimum the following programs must be defined to program control, unless BPX.DAEMON.HFSCTL was set up:

- HFISRV
- HFIMSGT
- HFICMENU
- HFICMJPN
- UHFIMSGT
- HFI0LVL

Alternatively, define all ZCC server programs in the library HFI.SHFIMODA to program control, rather than specifying individual programs.

If enhanced program security is enabled, HFISRV must be defined with the MAIN attribute, using the APPLDATA operand on the PROGRAM profile.

Example commands for RACF®

To activate program control if not already active, use the following command:

```
SETROPTS WHEN(PROGRAM)
```

To add all ZCC server programs in a library to program control, use the following command:

```
RDEFINE PROGRAM HFI* ADDMEM('HFI.SHFIMODA'//NOPADCHK) UACC(READ)
```

In addition, the following command is required for alias member UHFIMSGT:

```
RDEFINE PROGRAM UHFIMSGT ADDMEM(’HFI.SHFIMODA’//NOPADCHK) UACC(READ)
```

To add individual programs, use the following command:

```
RDEFINE PROGRAM HFISRV ADDMEM('HFI.SHFIMODA'//NOPADCHK) UACC(READ)
```

To refresh, use the following command:

```
SETROPTS WHEN(PROGRAM) REFRESH
```

Note:

- If you are using Japanese, then include the module IPVCMJPN in program control.

If RACF®, or an equivalent security product is implemented, the ZCC server (HFISRV1) started task must also be defined to the STARTED class. For example, to add HFISRV1 as an STC, the RACF® commands in the example that is shown here could be used, where `HFISRV1` is the name of your ZCC server procedure and *userid* is the userid that the started task runs under:

```
RDEFINE STARTED HFISRV1.* STDATA(USER(userid))

SETROPTS RACLIST(STARTED) REFRESH
```

For more information about started tasks and security, see the z/OS® Security Server RACF® Security Administrator's Guide, or equivalent documentation for your security product.

Multi-Factor Authentication (MFA)

When clients initially connect to the ZCC server they are prompted for a user ID and a password or passphrase. If the credentials are valid, the client can start sessions on the relevant z/OS system as the nominated user.

Rather than prompting for the user ID and password each time a new session is required, the plug-in client reuses the initial user ID and password. This can pose a problem for Multi-Factor Authentication users as their password or passphrase is typically single-use only. Consequently, reusing a credential will likely fail.

To support Multi-Factor Authentication users, the ZCC server provides support for PassTickets. For more information about PassTickets, refer to the documentation for RACF or your equivalent security product.

Using PassTickets

The ZCC server can be configured to use PassTickets for authenticated clients.

To exploit this feature, a client must first authenticate with a valid user ID and password or passphrase. Following a successful authentication, the server generates and use PassTickets for requesting clients. Such requests are valid for the period (in minutes) specified by the PASSTK configuration parameter.

To enable the use of PassTickets, complete the following steps:

1. Specify the PASSTK parameter in your ZCC server configuration file. For a description of the parameter, see Configuration file keyword descriptions on page 10.
2. The ZCC server must run APF-authorized. For more information about APF authorization and PassTickets, refer to the documentation for RACF or your equivalent security product.
3. PassTickets are generated in association with an APPLID. For ZCC, the default APPLID is HFIAPPL.

   If the APPL class is active, connecting users must have READ access to the relevant APPLID resource name in the APPL class. The APPLID resource name can be overridden by the APPLID parameter in the ZCC server configuration file, in which case, authorization checks are performed against the configured APPLID resource name.
4. The server started task user ID must have the following authorizations to generate PassTickets:

   ```
   SETROPTS CLASSACT(PTKTDATA)
   SETROPTS RACLIST(PTKTDATA)
   RDEF PTKTDATA HFIAPPL SSIGNON(KEYMASKED(yourmaskvalue))
   RDEF PTKTDATA IRRPTAUTH.HFIAPPL.* UACC(NONE)
   PERMIT IRRPTAUTH.HFIAPPL.* ID(your.userid) ACCESS(UPDATE) CLASS(PTKTDATA)
   SETR RACLIST(PTKTDATA) REFRESH
   ```

   If the server has the necessary authority, message HFI0052I is generated at startup, otherwise, message HFI0050S is generated.

Note: This feature primarily exists to facilitate multi-factor authentication (MFA) clients. Your MFA environment might require additional authorizations to use PassTickets. Refer to the instructions on using MFA with PassTickets in the documentation for IBM® Z Multi-Factor Authentication or equivalent MFA product.

Setting SSL/TLS encrypted communications

The sample HFICONFG configuration file member has TLS 1.2 encrypted communications active with the following line under the `CONFIG=DEFAULT` section:

```
SSL_REQUIRED=YES
```

If you would like to use other versions of TLS, see Configuration file keyword descriptions on page 10 for other values that can be specified for `SSL_REQUIRED`. If TLS encryption is not required in your environment, comment out this line and uncomment the next line (or alter your existing line to `SSL_REQUIRED=NO`). If TLS is required, replace `SSL_REQUIRED=YES` with `SSL_REQUIRED=TLSVxxx`, where `TLSVxxx` is one of the supported TLS versions listed in the description of the `SSL_REQUIRED` keyword in Configuration file keyword descriptions on page 10.

If using a SAF keyring and not using AT-TLS, uncomment and modify the `SSL_KEYRING` line. The `SSL_LABEL` line should also be uncommented and modified if the certificate you generate does not have a label of 'ZCC Server Certificate'.

For use of a certificate in a keyring, the userid of the server task or job, as well as the userids connecting to the server need to be permitted UPDATE access to the IRR.DIGTCERT.LISTRING facility and CONTROL access to the IRR.DIGCERT.GENCERT facility in order to share the certificate amongst users of the common server.

For RACF® users, a keyring and certificate can be created by the following example commands. Note that the minimum key size when using TLS 1.3 is 2048.

```
RACDCERT ID(HFISRV) ADDRING(RINGA)
RACDCERT GENCERT SITE SIZE(2048)            -
         SUBJECTSDN(                        -
           CN('Common Server')              -
           OU('ADL')                        -
           O('ADL')                         -
           C('AU'))                         -
 WITHLABEL('ZCC Server
      Certificate')
RACDCERT ID(HFISRV)                                        -
         CONNECT(SITE LABEL('ZCC Server
      Certificate')   -
         RING(RINGA) USAGE(PERSONAL)                       -
         DEFAULT)
SETR REFR RACL(DIGTCERT)
```

In this example, HFISRV is used for the user ID of the ZCC server task.

Note that the generated certificate must be a SITE certificate. This is because multiple users will need access to the certificate. An alternative to SITE certificates is to use AT-TLS. See Using AT-TLS for encrypted communications on page 18 for more information.

Updating the server config to include `SSL_KEYRING=HFISRV/RINGA` would use the above generated certificate. These commands serve as a working example only and should be updated as desired to match your needs. RACDCERT commands are documented in the z/OS® Security Server RACF® Command Language Reference.

If you are using ICSF and have protected resources through the CSFSERV facility class, the server user or group id needs to be permitted to the resource, for example:

```
PERMIT  CSF*  CLASS(CSFSERV)
            ID(groupid)  ACCESS(READ)
```

For more details see the *Cryptographic Services ICSF Administrator's Guide*.

If you wish to specify a cipher string or TLS key shares for the System SSL component to use, you can do this by modifying the server JCL to specify environment variables via the STDENV DD statement as required. The sample server JCL member HFISRV1 includes an example of specifying `GSK_V3_CIPHER_SPECS_EXPANDED` and `GSK_SERVER_TLS_KEY_SHARES` via STDENV.

Considerations when using TLS 1.3

There are two ways to configure the usage of TLS 1.3 to communicate between clients and the ZCC server. The first is to specify `SSL_REQUIRED=TLSV1.3` and `ATTLS=NO` in your server configuration parameters. This method causes the server to use its own built-in support for TLS 1.3.

When using this method, you must specify values for `GSK_V3_CIPHER_SPECS_EXPANDED` and `GSK_SERVER_TLS_KEY_SHARES` in your server startup proc. For more information on cipher specs and key share groups that can be used with TLS V1.3, see https://www.ibm.com/docs/en/zos/2.4.0?topic=protocols-required-updates-enable-tls-v13-protocol-support.

The second method is to specify `SSL_REQUIRED=TLSV1.3` and `ATTLS=YES` in your server configuration. This method causes the server to offload the encryption work to **ATTLS**. For more information on using AT-TLS, see Using AT-TLS for encrypted communications on page 18.

Using AT-TLS for encrypted communications

The Application Transparent Transport Layer Security (AT-TLS) feature of z/OS® Communication Server can be used to secure communications between the ZCC server and connecting clients by setting the ATTLS configuration parameter to the value ‘Y’. For example:

```
ATTLS=Y
```

Using AT-TLS requires the configuration of z/OS® Communications Server and policy agent rules to enable TLS protection of inbound connections to the ZCC server and subsequent data flows between the client and server. Your security administrator or system programmer can create this configuration in accordance with your installation standards and ensure that the z/OS® Communication server policy agent is running to provide AT-TLS services.

To establish an AT-TLS environment, take the following steps:

Note: Particulars might vary by installation.

1. Change the z/OS® Communication Server profile TCPCONFIG statement to activate the AT-TLS function. For example:

   ```
   TCPCONFIG  TTLS  ; Required for AT-TLS
   ```

   Optionally, installations might also change the z/OS® Communication Server profile AUTOLOG statement to automate starting the policy agent (PAGENT), which is needed to effect AT-TLS rules. For example:

   ```
   AUTOLOG
         PAGENT  ; POLICY AGENT, required for AT-TLS
   ENDAUTOLOG
   ```

2. Create the z/OS® Communication Server policy agent (PAGENT) configuration to establish AT-TLS rules for inbound connections to the ZCC server. For example:

   ```
   TTLSRule                     rule_ZCC
   {
    LocalPortRange              2800
    Direction                   Inbound
    TTLSGroupActionRef          grp_ZCC
    TTLSEnvironmentActionRef    env_ZCC
   }
   TTLSGroupAction              grp_ZCC
   {
    TTLSEnabled                 On
   }
   TTLSEnvironmentAction        env_ZCC
   {
   HandshakeRole                Server
   TTLSKeyRingParms
   {
    Keyring                     ZCC.KEYRING
   }
   TTLSEnvironmentAdvancedParms
   {
     TLSv1.3                    On
     HandshakeTimeout           30
     ApplicationControlled      On
    }
   TTLSCipherParms
    {
      V3CipherSuites4Char 13021301
    }
   TTLSSignatureParms
    {
      ServerKeyShareGroups 00230024002500290030
    }
   TTLSGskAdvancedParms
    {
      GSK_SESSION_TICKET_SERVER_ENABLE Off
    }
   }
   ```

Note: The **ApplicationControlled** parameter must be on for the ZCC server. In addition, the **SSL_REQUIRED** configuration parameter must be set to a valid protocol value. The protocol that is chosen must match a protocol that is supported by the AT-TLS rules that are specified in the AT-TLS configuration TTLSEnvironmentAdvancedParms statement. For example:

```
SSL_REQUIRED=TLSv1.3
```

A *HandshakeTimeout* value of 30 seconds is recommended. If using a LocalAddr* (LocalAddr, LocalAddrRef, LocalAddrSetRef, LocalAddrGroupRef) statement within your rule to limit the IP addresses on which the ZCC server listens, you must ensure that the statement allows connections to the server on address 127.0.0.1.

In addition, the HFISRV STC user will require access to the keystore that is identified on the **Keyring** parameter of the TTLSKeyRingParms statement. For more information on cipher specifications, key share groups, and certificate types supported for TLS 1.3, see https://www.ibm.com/docs/en/zos/2.4.0?topic=protocols-required-updates-enable-tls-v13-protocol-support.

3. Start the z/OS® Communications Server policy agent.

Note: If your policy agent configuration, or the key ring or keystore that is identified in the policy agent configuration is changed, restart the policy agent.

Clients such as z/OS® Explorer will be prompted to trust the server certificate identified in the AT-TLS configuration if the certificate is not registered as trusted.

Clients such as Z Data Tools Remote Services might require that the remote server CA certificate is imported as a SITE certificate on the client z/OS® system for establishing trust of the remote system.

Update sample HFICONFG

The CONFIG ddname in the ZCC server JCL procedure provides parameters that can be used to configure the ZCC server on startup. A sample configuration member is provided in HFI.SHFISAM1(HFICONFG), and the member can be customized as required.

Update the sample configuration member to suit your site, according to the comments in that member. In general terms, review the following items in the config file:

- Alter ddname=SYSOUT=H to suitable classes for your site. For example, for tracing activity, the CONFIG=DEFAULT section contains a SPAWN_DD=HFITRACE=SYSOUT=H card that other configurations inherit and write trace output (if activated) to. Adjust this class to a class suitable for your site.
- Alter SPAWN_STEPLIB data set names to the installation high-level qualifiers for the relevant libraries. The SPAWN_STEPLIB statement is not required if all of the libraries are already in the linklist for your site.

- If a configuration makes use of the SPAWN_JOBNAME statement, then all address spaces that are launched for that connection type run with that specified jobname (the owner of each job reflects the user that is logged in).
- Do not alter CONFIG=name and SPAWN_PROGRAM=name values unless otherwise detailed in the participating product's documentation.

The configuration file supports the setting and reference of substitution variables in the following form:

```
$VAR=value
```

For setting these variables, specify the above form before the first CONFIG statement, or between the CONFIG and SPAWN_PARMS_SECTION statements. If using concatenations for the CONFIG DD, the first CONFIG refers to the statements in the first of the concatenations.

In following statements in the configuration, occurrences of '$VAR' are replaced by the 'value' specified. This could be used to represent high level qualifiers that are repeated in the configuration file. For example, set the value:

```
$HFIHLQ=SYS1.HFI
```

Then allow a reference in a following statement, such as:

```
SPAWN_STEPLIB=$HFIHLQ.SHFIMODA
```

The sample HFICONFG makes use of this for high level qualifiers but it could also be used for other substitutions as desired.

Create matching WORKDIR by running job HFIMKDIR

The HFIMKDIR job creates a work directory to be used with the server. It is supplied in the sample library hlq.SHFISAM1.

HFIMKDIR creates a directory hierarchy in the following form:

```
/etc/hfi/v11/hfisrv1
```

You can alter this to suit your site. You must update the WORKDIR statement in the server configuration to refer to the created directory. A unique path is recommended.

Tip: Do not use `/tmp` as a directory location.

The files in the work directory must be owned by the user ID of the ZCC server. The HFIMKDIR job issues the chown command to set the owner of the files and any sub-directories within the work directory. It is recommended to use a unique work directory that is not used by other program products. The file system containing the work directory must allow the user ID to be changed through the SETUID attribute. If the file system is mounted with the NOSETUID attribute, the APF extended attribute set by the HFIMKDIR job is ignored, resulting in abend code EC6 when connecting to the ZCC server.

Note: The HFIMKDIR job is expected to run with superuser authority. That is, HFIMKDIR must have READ access in the FACILITY class to BPX.SUPERUSER and BPX.FILEATTR.APF. The job will try to set the sticky bit attribute and the APF extended attribute. If these file attributes are not set correctly, attempts to start sessions using the ZCC server might fail with authorization errors or abend code EC6. After running this job you can check the extended file attribute in the job output or use the ls -E z/OS® UNIX™ command. For more information, see the HFIMKDIR sample member.

As an alternative to running the HFIMKDIR job, you can manually create the working directory and its contents by executing the following steps in z/OS® UNIX™ System Services:

1. Start an OMVS session as superuser.
2. Create the working directory. For example:

   ```
   mkdir /etc/hfi/v11/hfisrv1
   ```

3. Create the set of session files in the working directory using the touch command:

`touch HFMCSEP`	*Z Data Tools*
`touch HFIVRFY`	*Z Common Components*

4. Set the APF file attributes for Z Data Tools session files:

   ```
   extattr +a HFMCSEP
   ```

5. Set the file ownership of the work directory and session files:

   ```
   chown <HFISRV> /etc/hfi/v11/hfisrv1
   chown <HFISRV> /etc/hfi/v11/hfisrv1/HFMCSEP
   chown <HFISRV> /etc/hfi/v11/hfisrv1/HFIVRFY
   ```

   Where <HFISRV> is the user ID that will run the ZCC server started task.
6. Set the file permissions for the work directory and session files:

   ```
   chmod 755 /etc/hfi/v11/hfisrv1
   chmod 755 HFMCSEP
   chmod 755 HFIVRFY
   ```

7. Set the sticky bit file attribute for all session files:

   ```
   chmod +t HFMCSEP
   chmod +t HFIVRFY
   ```

Related information

Mounting file systems

z/OS® UNIX System Services Command Reference

Check address space timeout

When an address space is launched for a client, and it has completed its current function, the address space is waiting for TCP/IP communications from the peer. In line with this, the client address space might be subject to an s522 abend if waiting longer than the active site settings for job wait time. The job wait time is controlled by the `JWT` parameter of the `SMFPRMxx` member, but might also be set to never time out by the site settings for `MAXCPUTIME` in the site's `BPXPRMxx` member. Set these parameters as needed by the site.

Add ports to TCPIP reservation list

Add the ports for the server, or servers, you want to run to the reserved port list in your TCPIP configuration data.

Configuring TCP/IP stack affinity

About this task

When multiple TCP/IP stacks are in use on the system, you can specify the desired ZCC stack on the started task.

1. Create a new PDSE data set with LRECL=80 and RECFM=FB. For example:

   ```
   <hlq>.EXMP.CONFIG
   ```

2. Create a new member in the data set, for example, `<hlq>.EXMP.CONFIG(TCPDATA)`. In the new member, include the following line:

   ```
   TCPIPJOBNAME TCPIPPRD
   ```

   where *TCPIPPRD* is the name of the desired TCP/IP stack.

3. In the IPVCONFIG member , for example, `<hlq>.EXMP.CONFIG(IPVCONFIG)`, add the following line to each product's SPAWN_PARMS_SECTION:

   ```
    SPAWN_DD=SYSTCPD=<hlq>.EXMP.CONFIG(TCPDATA)
   ```

4. Restart the ZCC started task for the changes to take effect.

Appendix A. Messages

The following information is provided for each ZCC server message:

- The message identifier.
- The text of the message.
- An explanation of the message.
- The required user response.

Messages have a unique alphanumeric identifier with the following format:

```
HFInnnns
```

where:

nnnn

Is a 4-digit number.

s

Is a severity level indicator with the following meanings:

- I - Informational
- W - Warning
- S - Severe

ZCC server messages

HFI0001I

Server on port %i exiting

Explanation: The server is finished processing. Either errors occurred during startup, running, or the server is responding to a shutdown command.

System action: The server finishes processing.

User response: If the shutdown was unexpected, examine previous messages for the cause.

HFI0002I

Error establishing SSL environment: %i

Explanation: An error occurred while establishing the SSL environment.

System action: The ZCC server attempts to continue.

User response: Examine previous messages for reasons for environment failure. If previous messages do not help, contact HCL support.

HFI0003S

Console modify/stop interface failed rc=%i, errno=%i error= %s

Explanation: An error occurred while establishing the console interface.

System action: The ZCC server exits.

User response: Examine the provided error for reasons for failure. If previous messages do not help, contact HCL support.

HFI0004I

Number of configurations %i

Explanation: During start or configuration refresh, the CONFIG data was read and the specified number of configurations were recognized.

System action: None.

User response: If the number of configurations is unexpected, check the CONFIG concatenations and contents.

HFI0005I

Config number %i startup %s

Explanation: During start or configuration refresh, the configuration specified an initial program to run.

System action: None.

User response: None.

HFI0006W

System call rc=%i error=%s

Explanation: A call to run a program according to a configuration failed.

System action: None.

User response: None.

HFI0007W

Expected a portnumber integer. Received %s

Explanation: The server expects an integer portnumber as the first parameter.

System action: The server attempts to continue starting, using port 2800.

User response: Check the invocation parameter for the server.

HFI0008W

Expected AF_INET or AF_INET6. Received %s

Explanation: The server expects the address family type as the second parameter.

System action: The server attempts to continue starting, using the AF_INET family.

User response: Check the invocation parameter for the server.

HFI0009I

Using address family %s.

Explanation: The server is using the specified address family.

System action: None.

User response: None.

HFI0010I

Using port %i.

Explanation: The server is using the specified port number.

System action: None.

User response: None.

HFI0011S

listen() error: %s

Explanation: The listen call failed with the specified error.

System action: The server is shut down.

User response: Correct the listed error if possible and restart the server.

HFI0012W

Spawn failure for %s. Error: %s __errno2 = %08x

Explanation: The attempt to spawn the specified program failed with the listed error and error code.

System action: The server continues to run.

User response: Examine the error and possibly examine the CONFIG file ensuring that customization occurred correctly.

HFI0013W

Missing value for keyword '%s'

Explanation: While reading the CONFIG file, an expected value for a keyword was missing.

System action: The server continues to run.

User response: Check the CONFIG file for the specified keyword and specify an appropriate value.

HFI0014W

Failure to acquire storage for configuration instance %i

Explanation: While preparing configurations, a failure to acquire storage occurred.

System action: The server attempts to continue to run.

User response: Check the REGION specification for the server. Increase and restart the server.

HFI0015I

ZCC server Running on port %i.

Explanation: Console message to indicate that the server is now accepting connections.

System action: None.

User response: None.

HFI0016I

Established SSL environment.

Explanation: The call to System SSL to initialize an environment was successful.

System action: None.

User response: None.

HFI0017W

Unable to create temporary file %s. %s

Explanation: The call to create a temporary file for a configuration failed.

System action: The server attempts to continue, however the configuration might be unusable.

User response: Examine the file path and error condition as shown. Correct the configuration file or update the directory permissions and restart or refresh the server.

HFI0018W

Unable to verify dsn %s

Explanation: The existence of data set %s in a STEPLIB= value could not be verified.

System action: The server attempts to continue, however the configuration might be unusable.

User response: Examine the named data set and ensure that it is the correct name. If necessary, update the configuration file and restart or refresh the server.

HFI0019W

Unable to open CONFIG %s

Explanation: During startup, or a refresh command, the DD CONFIG was unable to be opened.

System action: If this occurs during initial start of the server, the server terminates. During a refresh, no new configurations are loaded.

User response: Examine the error and the CONFIG data sets to ensure that they exist. If necessary, update the configuration file and restart or refresh the server.

HFI0020I

REFRESH completed, %i configs processed.

Explanation: A REFRESH console command has now completed. The server has re-read the configurations as specified in the CONFIG DD.

System action: None.

User response: None.

HFI0021W

REFRESH found errors in new configs, not activated.

Explanation: A REFRESH console command was issued, but during reading of the CONFIG DD, some errors occurred.

System action: The server continues with its prior configuration.

User response: Check the server output for possible further information on the problems that are found in the CONFIG file(s)

HFI0022S

Creation of key database at %s failed, error %s

Explanation: The configuration specifies that the server create a certificate to be used, however an error as described occurred when attempting to create the key database.

System action: The server terminates.

User response: f the error is an IO error, check the specified location for enough space (65KB). Otherwise, check that the location is writeable. To specify an alternate location, set the configuration keyword WORKDIR to the directory to be used.

HFI0023S

Creation of self-signed certificate failed, error %s

Explanation: The configuration specifies that the server create a certificate to be used, however an error as described occurred when attempting to create the self-signed certificate in the key database.

System action: The server terminates.

User response: Check the listed error and check documentation for the gsk_create_self_signed_certificate API.

HFI0024I

Traceon received, trace already active.

Explanation: The Server received a modify command to turn on tracing, but it is already on.

System action: None.

User response: None.

HFI0025I

Traceon received, trace turned on.

Explanation: The Server received a modify command to turn on tracing and has done so. Trace output goes to the HFITRACE file(DD) if present, or to the STDOUT file if not.

System action: None.

User response: None.

HFI0026I

Traceoff received, trace already off.

Explanation: The Server received a modify command to turn off tracing but it is already off.

System action: None.

User response: None.

HFI0027I

Traceoff received, trace turned off.

Explanation: The Server received a modify command to turn off tracing and has done so.

System action: None.

User response: None.

HFI0028I

Unrecognized modify command.

Explanation: The Server received a modify command, but did not recognize it.

System action: None.

User response: Check that modify contained one of the valid requests; TRACEON, TRACEOFF, VER or REFRESH.

HFI0029W

Client config name %s not found in CONFIG DD content.

Explanation: The Server received a client connection request for the named config, but no matching CONFIG=name statement was found in the data that was contained in the CONFIG DD concatenation.

System action: The client connection request is refused.

User response: Check that the configurations referenced by the CONFIG DD for the server, contain a CONFIG=name section.

HFI0030I

API start PID=processid

Explanation: A process (processid) launched by the common server has invoked the common server subordinate API to start the environment setup and handshake with client.

System action: None.

User response: None.

HFI0031I

API closure PID=processid

Explanation: A process has invoked the common server subordinate API to close the environment setup and client connection.

System action: None.

User response: None.

HFI0032I

ZCC server Release=%s PTF=%s

Explanation: In response to the VER modify command, the server lists its release and PTF level information.

System action: None.

User response: None.

HFI0033W

Unknown token %s with value %s for CONFIG=%s

Explanation: While processing the configuration file, an unrecognized token/value pair was found.

System action: The invalid token is ignored and processing attempts to continue.

User response: Review the configuration file for the named token. Look for a misspelling or incorrect token or value.

HFI0041W

Maximum user variables (500) reached when processing token %s, value %s in configuration %s

Explanation: The limit of substitution values has been reached.

System action: The server attempts to continue, however the configurations might be unusable.

User response: Examine the number of $token=value pairs present in the configuration file and reduce to less than 500.

HFI0042W

Unable to stat file %s.

Explanation: The server is unable to check the configuration launch file entry.

System action: The server attempts to continue, however this launch configuration is unusable.

User response: Examine the file path and ensure that the setup was completed correctly. Most likely the file or directory path is not owned or correctly permitted in order for this server instance to access the named file. The WORKDIR configuration step of installation needs to be checked and rerun.

HFI0043W

Not owner of launch file %s.

Explanation: The server is not the owner of a configuration launch file entry.

System action: The server attempts to continue, however this launch configuration is unusable.

User response: Examine the file path and ensure that the setup was completed correctly. Correct the condition by ensuring that the file owner is updated to the userid of the server. The file system that the file is mounted on needs to allow SETUID for the owner to be changed with the chmod command.

HFI0044W

Launch file %s is not marked as sticky.

Explanation: A configuration launch file has not been created correctly.

System action: The server attempts to continue, however this launch configuration is unusable.

User response: Examine the file path and WORKDIR location. If the WORKDIR is correct, the installation configuration step for the WORKDIR might need to be rerun.

HFI0045S

Configuration specifies AT-TLS, but AT-TLS rule is missing or invalid.

Explanation: The configuration specifies ATTLS=Y, but an AT-TLS rule for the inbound connection was not found or was not 'ApplicationControlled'.

System action: The ZCC server is shut down.

User response: Contact your security administrator or system programmer to verify the AT-TLS configuration of your installation.

HFI0046S

AT-TLS specified, but no protocol provided by SSL_REQUIRED parameter.

Explanation: The configuration specifies ATTLS=Y, but the SSL_REQUIRED parameter does not specify a protocol value.

System action: The ZCC server is shut down.

User response: Contact your security administrator or system programmer to verify that the ZCC server configuration of your installation specifies a valid protocol that is supported by your AT-TLS configuration.

HFI0047S

Insufficient storage available.

Explanation: An attempt to acquire storage failed because insufficient storage was available.

System action: The ZCC server is shut down.

User response: Check your system for any task using excessive storage. Restart the server when sufficient storage is available.

HFI0048S

SYSOUT=* not permitted in configuration.

Explanation: The configuration file specified by the CONFIG DD statement uses SYSOUT=*, which cannot be resolved by a started session.

System action: The ZCC server is shut down.

User response: Change the configuration member to specify a valid SYSOUT class.

HFI0049S

PASSTK parameter expresses an invalid timeout value

Explanation: The PASSTK parameter in CONFIG DD expresses an invalid value. The PASSTK value should express a positive integer that specifies a timeout period in minutes for a client to use PassTickets following a successful log on. The default is 480 (8 hours).

System action: The ZCC server is shut down.

User response: Change configuration to use a valid PASSTK value.

HFI0050S

Attempt to verify PASSTICKET environment failed, rc=*nn*

Explanation: The attempt to verify PASSTICKET authority failed. The server does not have sufficient authority to generate PASSTICKETs. *nn* is the return code from the PassTicket generation routine. A return code of 16 means the server is not APF authorized.

System action: The ZCC server is shut down.

User response: Ensure that the execution environment meets the prerequisites for PASSTICKET generation and then restart the server.

HFI0051I

Verifying server PASSTICKET authority

Explanation: The server configuration indicates that PASSTK services are required. The server must verify that it has sufficient authority to satisfy client PASSTK requests.

System action: The ZCC server verifies that it has PASSTICKET authority.

User response: None.

HFI0052I

Server PASSTICKET authority verification successful

Explanation: The ZCC server has verified that it has sufficient authority to satisfy client PASSTK requests.

System action: The ZCC server continues to run.

User response: None.

HFI0053S

APPLID parameter has an invalid value length

Explanation: APPLID configuration parameter value has an invalid length. The APPLID value should be between 1 and 8 characters. The value represents a resource name in the APPL class, and clients connecting to the ZCC server must have READ access to this resource if the APPL class is active.

System action: The ZCC server is shut down.

User response: Change the configuration to use a valid APPLID value.

Appendix B. Troubleshooting

Error scenarios and tracing

If the installed library has not been added to program control, this message appears in the JESMSGLG for the server task:

```
ICH420I PROGRAM HFISRV FROM LIBRARY HFI.V1R1.SHFIMODA CAUSED
THE ENVIRONMENT TO BECOME UNCONTROLLED.  BPXP014I ENVIRONMENT MUST
BE CONTROLLED FOR SERVER (BPX.SERVER) PROCESSING.
```

Messages similar to the following might be generated if the user connecting to the server does not have read access to the SHFIMODA library:

```
ICH408I USER(BILLMAN ) GROUP(USERCOD ) NAME(MANDELLA, BILL ) 218
HFI.V1R1.SHFIMODA CL(DATASET ) VOL(COD035)
INSUFFICIENT ACCESS AUTHORITY
FROM HFI.V1R1.* (G)
ACCESS INTENT(READ ) ACCESS ALLOWED(NONE )
IEC150I 913-38,IFG0194E,BILLMAN,OS390,ISP19502,8E10,COD035,HFI.V1R1.SHFIMODA
```

Messages on SYSLOG at the time of attempted connection, like the ones that are shown here, are generated when the relevant CONFIG contains an invalid library, or is missing a library from the SPAWN_STEPLIB statement:

```
IEA995I SYMPTOM DUMP OUTPUT
SYSTEM COMPLETION CODE=EC6  REASON CODE=0B26C032
 TIME=11.37.04  SEQ=38113  CPU=0000  ASID=00ED
 PSW AT TIME OF ERROR  070C3000   82C44CE8  ILC 2  INTC 0D
   NO ACTIVE MODULE FOUND
   NAME=UNKNOWN
   DATA AT PSW  02C44CE2 - C06C18F2  0A0D41B0  D4D0180B
   AR/GR 0: 00000000/00000026_00000648   1: 00000000/00000000_04EC6000
         2: 01FF000C/00000000_0B26C032   3: 00000000/00000000_8286F5B8
         4: 00000000/00000000_00000000   5: 00000000/00000000_00000000
         6: 01FF000C/00000000_00000700   7: 01FF000C/00000000_09BFC3F8
         8: 00000000/00000000_11F4B610   9: 00000000/00000000_163031FF
         A: 00000000/00000000_11F4B610   B: 01FF000C/00000000_7FFC3A00
         C: 00000000/00000000_02C47AC0   D: 00000000/00000000_16302200
         E: 00000000/00000000_82C44CB0   F: 00000000/00000000_0B26C032
 END OF SYMPTOM DUMP
```

If the above are not occurring, but connections are still not successful, shutdown the server and start it again with tracing active. If using the supplied sample, this can be done on the start command. For example, S HFISRV1,TRACE=D. This produces trace entries in the server task on the HFITRACE DD.

A typical trace, with SSL active, before connections are made, looks similar to the one shown here. The main entries of interest confirming start up was successful are highlighted:

```
2018-11-29-10:54:39.442 [HFISRV:266] Server built at: Nov 29 2018 10:54:03
2018-11-29-10:54:39.601 [HFISRV:952] Record in length:1903
2018-11-29-10:54:39.601 [HFISRV:969] Token: CONFIG Value: DEFAULT
2018-11-29-10:54:39.601 [HFISRV:989] Config DEFAULT allocated.
2018-11-29-10:54:39.601 [HFISRV:969] Token: SSL_REQUIRED Value: YES
2018-11-29-10:54:39.601 [HFISRV:969] Token: WORKDIR Value: /etc/hfi/v11/hfisrv1
2018-11-29-10:54:39.602 [HFISRV:1070] Confirmed temporary write access ok dir=/etc/hfi/v11/hfisrv1.
```

```
2018-11-29-10:54:39.602 [HFISRV:969] Token: SPAWN_STEPLIB Value: HFI11SVC.BUILD.LOAD ...
2018-11-29-10:54:39.602 [HFISRV:969] Token: CONFIG Value: FM
2018-11-29-10:54:39.602 [HFISRV:989] Config FM allocated.
2018-11-29-10:54:39.602 [HFISRV:969] Token: SPAWN_PROGRAM Value: HFMCSEP
2018-11-29-10:54:39.602 [HFISRV:1089] Creating temp filename.
2018-11-29-10:54:39.602 [HFISRV:1106] Created temporary spawn image file ok.
2018-11-29-10:54:39.602 [HFISRV:1116] spawn_program /etc/hfi/v11/hfisrv1/HFMCSEP
2018-11-29-10:54:39.602 [HFISRV:1117] spawn_fn HFMCSEP
2018-11-29-10:54:39.602 [HFISRV:969] Token: SPAWN_JOBNAME Value: FMCLIENT
2018-11-29-10:54:39.602 [HFISRV:969] Token: SPAWN_STEPLIB Value: HFM.V1R1M2.OPTIONS...
2018-11-29-10:54:39.602 [HFISRV:969] Token: SPAWN_PARMS_SECTION Value:
2018-11-29-10:54:40.495 [HFISRV:1956] Environment open rc=0 Handle=16AB09A8 Ha=16AA6490
2018-11-29-10:54:40.495 [HFISRV:1965] Set SSLV2 off rc=0
2018-11-29-10:54:40.495 [HFISRV:1973] Set SSLV3 off rc=0
2018-11-29-10:54:40.495 [HFISRV:1982] Set TLSV1 off rc=0
2018-11-29-10:54:40.495 [HFISRV:1997] Certfile=/etc/hfi/v11/hfisrv1/HFISRVC3-HFICERT.kdb
2018-11-29-10:54:40.495 [HFISRV:1998] Set keyring rc=0
2018-11-29-10:54:40.495 [HFISRV:2006] Set pw rc=0
2018-11-29-10:54:40.511 [HFISRV:2014] Environment init rc=0 Handle=16AB09A8
2018-11-29-10:54:40.511 [HFISRV:281] Mixed case password support is off
2018-11-29-10:54:40.512 [HFISRV:1902] Set socket linger rc=0
2018-11-29-10:54:40.512 [HFISRV:1906] Set socket reuseaddr rc=0
2018-11-29-10:54:40.512 [HFISRV:1910] Set socket keepalive rc=0
2018-11-29-10:54:40.512 [HFISRV:301] Launching accept thread socket 0, listen code 0
2018-11-29-10:54:40.512 [HFISRV:513] Acceptor thread running.
2018-11-29-10:54:40.512 [HFISRV:527] About to accept.
```

If the highlighted statements are similar to the example that is shown here, all rc=0, then try to connect.

Notices

This information was developed for products and services offered in the U.S.A.

HCL may not offer the products, services, or features discussed in this document in other countries. Consult your local HCL representative for information on the products and services currently available in your area. Any reference to an HCL product, program, or service is not intended to state or imply that only that HCL product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any HCL intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-HCL product, program, or service.

For license inquiries regarding double-byte (DBCS) information, contact the HCL Intellectual Property Department in your country or send inquiries, in writing, to:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

HCL TECHNOLOGIES LTD. PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. HCL may make improvements and/or changes in the product(s) and/or the program(s) described in this document at any time without notice.

Any references in this information to non-HCL websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this HCL product and use of those websites is at your own risk.

HCL may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who want to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085

USA
Attention: Office of the General Counsel

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by HCL under terms of the HCL Customer Agreement, HCL International Program License Agreement, or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-HCL products was obtained from the suppliers of those products, their published announcements or other publicly available sources. HCL has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-HCL products. Questions on the capabilities of non-HCL products should be addressed to the suppliers of those products.

Statements regarding future direction or intent of HCL Z Data Tools are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to HCL, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. HCL, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. HCL shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).
Portions of this code are derived from IBM Corp. and/or
HCL Ltd. sample programs.

Programming interface information

This documentation describes intended Programming Interfaces that allow the customer to write programs to obtain the services of Z Data Tools.

Trademarks

HCL, the HCL logo, and hcl.com® are trademarks or registered trademarks of HCL Technologies Ltd., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM® or other companies.

Index