Installing EntireX Security under z/OS

This chapter covers the following topics:

- Installing EntireX Security for Broker Kernel
- Setting up EntireX Security for Broker Stubs

Notes:

1. If you are using EntireX Security, and if your application(s) use ACI version 7 or below, you must install EntireX Security for Broker stubs. For ACI version 8 and above, see *Writing Applications using EntireX Security* to ensure that your application(s) will perform as expected when using EntireX Security.
2. Before installing EntireX Security, make sure that all prerequisites for EntireX components have been met. See *z/OS Prerequisites*.

Installing EntireX Security for Broker Kernel

This section describes the steps for installing EntireX Security for Broker kernel under z/OS. The installation procedure has the following steps:

- Modify Broker Attribute File
- Define RACF Resource Profiles
- Perform Client Authorization Checks (Optional)
- Enable Trusted User ID (Optional)
- Build Language-specific Messages (Optional)
- Start (Restart) Broker Kernel

Modify Broker Attribute File

To modify the Broker attribute file

1. Insert the following parameter in the section `DEFAULTS=BROKER` of the Broker attribute file:

   ```
   SECURITY=YES
   ```

2. Modify the Security-specific attributes section of the Broker attribute file according to your requirements. These parameters are used to determine whether you will use SAF Security or LDAP-based authentication. See *Security-specific Attributes* under *Broker Attributes*. If you are using LDAP-based authentication, authorization checks are not available to you.

Note:
Setting `SECURITY=YES` will load the provided LOAD module USRSEC from the EXX970.LOAD library. This module will perform privileged operations, such as execute the `RACROUTE`, requiring APF authorization.

Define RACF Resource Profiles

EntireX Security performs checks against user profiles and resource profiles represented in RACF, CA ACF2, and CA Top Secret. See *Resource Profiles in EntireX Security*.

Perform Client Authorization Checks (Optional)

For services supporting Natural RPC or other applications that use RPC, you can perform authorization checks on the client by defining the "per service" attribute `CLIENT-RPC-AUTHORIZATION=YES` in the Broker attribute file. Setting this parameter to `YES` will cause the RPC library and program names to be appended to the profile associated with the authorization check. The resource profile would then appear as follows:

```
Class.server.service.rpc-library.rpc-program
```

If the total length of the resource profile exceeds 80 bytes, increase the parameter `MAX-SAF-PROF-LENGTH`.

This check applies only to the client and not the server. `CLIENT-RPC-AUTHORIZATION=YES` should not be set for any services which do not utilize RPC protocol.

Note:
Natural Security performs its resource authorization checks as follows:
`<prefix-character>.rpc-library.rpc-program`
To allow conformity with Natural Security, the `CLIENT-RPC-AUTHORIZATION` parameter can optionally be defined with a prefix character as follows:
`CLIENT-RPC-AUTHORIZATION=(YES,<prefix-character>).`

Enable Trusted User ID (Optional)

If you use the trusted user ID option, set the parameter `TRUSTED-USERID=YES` in the `DEFAULTS=SECURITY` section of the attribute file.

- The trusted user ID feature automatically acquires the identity of the logged-on user or batch job. It must therefore only be used with TP monitors running under the control of RACF, CA ACF2 or CA Top Secret. Batch jobs must run under an identifiable user ID, as inherited by the job submitter, scheduler, or other means.

- Applications using the trusted user ID feature must execute under z/OS and on the same machine, or another z/OS machine connected to Broker through Entire Net-Work. Communication is through the Adabas SVC mechanism.

- Applications must not assign a password to the ACI control block if they intend to use trusted user ID. This applies to all applications, including EntireX RPC Server. If the application cannot avoid supplying a password, it is permissible to assign a password value of `NOPASSWORD`.

- EntireX Security trusted user ID functionality is relevant only for determining the z/OS user ID associated with applications executing on z/OS which communicate with EntireX Broker, which are also executing on z/OS via the Adabas SVC mechanism. It cannot be used in configurations which include application components executing on separate, non-z/OS computers that communicate with EntireX Broker through Entire Net-Work. Such configurations invalidate the usage of trusted user ID.

- The SVCSAF module is supplied with EntireX. If your Adabas version is lower than 8.2, the resulting Adabas SVC must be linked into an APF authorized library. Since Adabas 8.2, the SAF component is linked to ADASVC by default. Linkage example:

```
//*------------------------------------------------------
//* CREATE A NEW ADASVC MODULE THAT INCLUDES SVCSAF MODULE
//*------------------------------------------------------
//LNKSVC EXEC PGM=IEWL,PARM='XREF,LIST,LET,NCAL,RENT,REUS'
//SYSPRINT DD SYSOUT=*
//SYSUT1  DD SPACE=(CYL,(1,1)),UNIT=VIO
//WALLIB  DD DISP=SHR,DSN=WAL826.LOAD      ADASVC and SVCSAF
//SYSLMOD DD DISP=SHR,DSN=WAL826.NEW.LOAD  NEW ADASVC OUTPUT
//SYSLIN  DD *
MODE AMODE(31),RMODE(24)
INCLUDE WALLIB(ADASVC)
SETCODE AC(1)
INCLUDE WALLIB(SVCSAF)
NAME ADASVC(R)
/*
//*
```

- Implementing the SAF trusted user ID option in EntireX Security under CICS TS version 1.2 and above requires the installation of the Adabas task related user exit (ADATRUE) and setting either the ADAGSET or LGBLSET (depending upon the Adabas version) parameter SAF=YES. See *Installing Adabas with TP Monitors* in the Adabas installation documentation for complete details on installing ADATRUE. Samples of the ADAGSET and LGBLSET parameter modules can be found in the library WAL826.SRCE.

For additional supporting information, see the *Installation Procedure* section of the *Adabas Installation Manual*.

Build Language-specific Messages (Optional)

To build language-specific messages

1. Copy the template message module EXX970.SRCE(NA2MSG0) to another member - for example, EXX970.SRCE(NA2MSG9) - and then modify the message texts to suit your own language requirements.

   Note:
   NA2MSG0, NA2MSG1, and NA2MSG2 are reserved names.

2. Assemble and link your modified source module using the sample JCL EXX970.SRCE(SAGJ106), ensuring that you create a unique load module in the EXX970.LOAD library.

3. Modify the ERRTXT-MODULE parameter in the DEFAULTS=SECURITY section of the attribute file to reflect the name of your unique load module.

Start (Restart) Broker Kernel

The Broker must be restarted to pick up changes to the Broker attribute file and to initialize Broker kernel under z/OS to perform security checks.

Basic installation of EntireX Security for Broker kernel is now complete.

Setting up EntireX Security for Broker Stubs

This section describes the steps for installing EntireX Security for Broker stub under z/OS. The installation consists of the following steps:

- Assemble the SAFCFG Configuration Module
- Link the Security Components
- Rename SECUEXIT

Notes:

1. If you are running your application(s) at ACI version 7 or below, the following steps are required to install EntireX Security for the Broker stubs in all environments where applications execute either as clients or servers. See *List of Components per Platform* to see where EntireX Security for broker stubs is supported.
2. The mainframe stubs now employ high performance direct cross memory to send and receive data from buffers in the application's working storage. This is utilized for sending/receiving more than 32 KB of data. Therefore, when encryption is active, the application programmer must not rely on the contents of the `SEND` buffer after issuing the `SEND` command, since the contents of the `SEND` buffer will be encrypted when sending more than 32 KB of data. We recommend you code all applications so that you do not rely on the contents of the `SEND` buffer after calling Broker. This will be required in the future for all `SEND` commands regardless of whether the data exceeds 32 KB. Therefore, the application's `SEND` buffer must not be in read-only memory, where encryption is activated.

These steps are not required if you are running your application(s) at ACI version 8 or above.

Assemble the SAFCFG Configuration Module

The SAFCFG configuration module is required for applications running on z/OS using ACI version 7 or below.

To assemble the SAFCFG configuration module

- Run job WAL*vrs*.JOBS(SAFI010), which assembles and links SAFCFG (load module).

  Note:
  This module comes with preconfigured defaults. See source module WAL*vrs*.SRCE(SAFCFG). If encryption is required, set the macro assembly parameter as follows: `BKPRIV=1`.

Link the Security Components

For applications running on z/OS using ACI 7 or below, the Broker stub security component must be linked with the following stubs: `BROKER`, `CICSETB`, `NATETB23`, `COMETB`, `MPPETB`.

To link the Broker stub security component

- Relink all applications that contain ACI stub modules `BROKER`, `CICSETB`, `NATETB23`, `COMETB`, or `MPPETB` to include the following modules:
  - `NA2PETS` Broker security stub logic module
  - `SAFCFG` System parameter module

  See *Administering Broker Stubs*.

  Location of sample `INCLUDE` statements: EXX970.JOBS(EXXJ109).

  Note:
  These components are needed for backward compatibility if your applications issue any commands using ACI version 7 or below. Applications using ACI version 8 or above do not require these additional components in the stubs.
  For ACI version 7 or below, these components must be added to the stub environment utilized by the application. Failure to link these components along with the stub when using ACI version 1 though 7 can result in message "SEFM225 MESSAGE FROM BACK LEVEL STUB" being issued by Broker kernel.

Rename SECUEXIT

`SECUEXIT` must be made available for applications running on z/OS using ACI version 7 or below.

To make `SECUEXIT` available

1. Rename `SECUEXI0` to `SECUEXIT` in library EXX970.LOAD so that it is available to applications running the IBM C stub.
2. Ensure that `SECUEXIT` is available in EXX970.LOAD for all applications.

Notes:

1. These steps are needed for backward compatibility if your applications issue any commands using ACI version 7 or below. Applications using ACI version 8 or above do not require these additional components in the stubs.
2. For ACI version 7 or below, these components must be added to the stub environment utilized by the application.

Installation of EntireX Security for Broker stubs is now complete. Now you can install the security components for the Broker stubs on the remaining operating systems where your application components are located.

See also *Setting up EntireX Security for Broker Stubs* under UNIX | Windows.