CA XCOM™ Data Transport® for z/OS

Administration Guide

Release 11.6

CA Technologies Product References

This guide references the following CA products:

- CA 7® Workload Automation Smart Console Option (CA 7)
- CA ACF2™ (CA ACF2)
- CA NetMaster® File Transfer Management (CA NetMaster FTM)
- CA Roscoe® (CA Roscoe)
- CA Top Secret® (CA Top Secret)
- CA XCOM™ Data Transport® (CA XCOM Data Transport)
- CA TCPaccess™ Communications Server (CA TCPaccess CS for z/OS)

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Changes to Parameters and Statements

This guide includes new Default Options Table parameters. It also includes parameter and statement changes.

New Parameters

The following new parameters are documented in this guide:

XCOMHIST

Parameter type: CONFIG Member

XCOMHIST_USER

Parameter type: CONFIG Member

XCOMHIST_PASSWORD

Parameter type: CONFIG Member

XCOMHIST_OWNER

Parameter type: CONFIG Member

XCOMHIST_TBL

Parameter types: CONFIG Member

CMPRS_PDS_ALLOW

Parameter type: CONFIG Member

CMPRS_SYSOUT_CL

Parameter types: CONFIG Member

DEALLOCMSG

Parameter types: CONFIG Member

CREATEDELETE

Parameter type: CONFIG Member

SYSID

Parameter type: CONFIG Member

SYSNAME

Parameter types: CONFIG Member

FACILITY

Parameter types: CONFIG Member

OPERCMDS

Parameter types: CONFIG Member

CREATEDELOVRD

Parameter types: CONFIG Member

AVGREC

Parameter types: CONFIG Member

EATTR

Parameter types: CONFIG Member

LDSNLRECL

Parameter types: CONFIG Member

PLEXQ

Parameter types: PARM and CONFIG Member

ZIIP

Parameter types: PARM and CONFIG Member

DEFAULT_CHARSET

Parameter types: CONFIG & Dest Member

DEFAULT_INPUTERROR

Parameter types: CONFIG & Dest Member

DEFAULT_CONVERROR

Parameter types: CONFIG & Dest Member

DEFAULT_DELIM

Parameter types: CONFIG & Dest Member

Changed Parameter

Changes to the SECURITY= parameter are documented in this guide. The SECURITY= parameter has an additional value, SAF.

Changed Statements

The changed use of list statements is documented in this guide.

Using the type TYPE=LIST statements in control library members to define a list destination, you can now also use TYPE=SUPERLIST statements. A superlist is a list of lists, which expands the number of member names that you can include in your destination member.

Contents

Chapter 1: Configuring and Customizing Your Product 19

Chapter 2: Configuration Parameters 91

Chapter 1: Configuring and Customizing Your Product

This chapter describes the tasks to configure and customize CA XCOM Data Transport for z/OS. If you are using PLEXQ for the first time, review the chapter Understanding the XCOMPLEX in this guide before performing the installation. Collectively, the tasks form Step 13 in the chapter Installing Your Product in the *CA XCOM Data Transport for z/OS Installation Guide*.

This section contains the following topics:

Define Your Product to VTAM (Optional)

CA XCOM Data Transport is defined as a VTAM application using an application definition table. This section explains what to include in an application definition table to define CA XCOM Data Transport to VTAM. This definition as a VTAM application is not required if the Default Options Table SNA parameter is set to NO.

Note: CICS requires SNA=YES. CICS is deprecated in CA XCOM r11.6. Use the ISPF interface. See the *CA XCOM Data Transport for z/OS Installation Guide* for information on configuring CA XCOM for TSO/ISPF.

What Is in an Application Definition Table

An application definition table consists of a VBUILD statement along with one or more APPL statements.

The VBUILD statement establishes CA XCOM Data Transport as an application program major node. The CA XCOM Data Transport major node contains a set of minor nodes that represent the specific CA XCOM Data Transport resources that VTAM can activate and deactivate as a group.

The minor nodes are defined with APPL statements. The minor node names are also known as APPLIDs.

VBUILD Statement

The format of the VBUILD statement is as follows:

```
name VBUILD TYPE=major node type
```

A valid value for the name is any string of up to eight alphanumeric characters. VBUILD is the major node name of CA XCOM Data Transport; it is the same as the member name in the VTAM definition library (VTAMLST). When an application such as CA XCOM Data Transport is being defined to VTAM, the value of major node type is APPL. TYPE=APPL indicates to VTAM simply an application major node definition. (If TYPE is not coded, APPL is also the default value of TYPE).

APPL Statement

The format of the APPL statement is as follows:

```
name APPL [parameter]
          [parameter]
          . . .
          [parameter]
```

A valid value for the *name* is any string of up to eight alphanumeric characters. APPL is the minor node name of CA XCOM Data Transport. The parameters define the communication characteristics of the application that is associated with the minor node.

Example of an Application Definition Table

You can use the data set CAI.CBXGSAMP(APPLXCOM) as a model for constructing an application definition table for CA XCOM Data Transport at your installation. For a listing of this dataset, see the appendix "Sample Files" in the *CA XCOM Data Transport for z/OS User Guide*. The following extract contains the VBUILD statement and some of the APPL statements from this data set:

```
APPLXCOM VBUILD TYPE=APPL
XCOMAPPL APPL   AUTH=(NOPO,ACQ,VPACE),MODETAB=YOURTABL,
                SONSCIP=YES,ACBNAME=XCOMAPPL,VPACING=5,
                DLOGMOD=XCOMMODE,PARSESS=YES
XCOM00   APPL   AUTH=(NOPO,ACQ,VPACE),MODETAB=YOURTABL,
                SONSCIP=YES,ACBNAME=XCOM00,VPACING=5,
                DLOGMOD=XCOMMODE
. . .
. . .
. . .
XCOM05   APPL   AUTH=(NOPO,ACQ,VPACE),MODETAB=YOURTABL,
                SONSCIP=YES,ACBNAME=XCOM05,VPACING=5,
                DLOGMOD=XCOMMODE
```

In the example, the application major node name is APPLXCOM. The major node APPLXCOM contains the minor nodes named XCOMAPPL and XCOM00 to XCOM05.

APPL Statements

The first APPL statement in the CA XCOM Data Transport application definition table identifies the APPLID that the CA XCOM Data Transport server uses. The name that is associated with the CA XCOM Data Transport server is XCOMAPPL. When CA XCOM Data Transport partners identify the CA XCOM Data Transport server, they use the name defined in the first APPL statement of the application definition table.

The remaining APPL statements define APPLIDs that batch jobs and ISPF dialog users use. Batch jobs and ISPF dialogs transfer data directly (without server intervention) to other systems, or they use the server to schedule transfers.

ACBs in an XCOMPLEX Environment

Important! The existing XCOMPLEX facility has been deprecated. The following information is provided for backward compatibility only. All new installations should use the new PLEXQ implementation. Existing XCOMPLEX users should migrate to the PLEXQ infrastructure for their XCOMPLEX functionality. Refer to section “Create a PLEXQ Environment (Optional)” in this guide.

If you are running in an XCOMPLEX environment, define a sufficient number of minor ACBs for each XCOMPLEX Worker Server and XCOMPLEX Admin Server to handle the workload. Use the following guidelines when setting up your environment to ensure an appropriate number of minor ACBs are defined:

- Determine how many XCOMPLEX Worker Servers are workers to the XCOMPLEX Admin Server and how many transfers are requested from the XCOMPLEX Admin Server. If you have two XCOMPLEX Worker Servers, XCOMA and XCOMB, with 10 minor ACBs defined for each (XCOMA00 to XCOMA09 and XCOMB00 to XCOMB09 respectively). Then the XCOMPLEX Admin Server XCOMC has a sufficient number of minor ACBs defined to be equal to or greater than all the XCOMPLEX Worker Servers minor ACBs combined. If you define XCOMC with minor ACBs XCOMC00 to XCOMC25, then it ensures your XCOMJOBs will not fail due to a lack of minor ACBs accommodates any future expansion.
- Determine the type of jobs. For example:
  - TYPE=SCHEDULE jobs use a minor node only for the length of time that is required to schedule the transfer.
  - TYPE=EXECUTE jobs use a minor node for the entire transfer.
  - TYPE=INQUIRE jobs use a minor node until the status for the transfer is COMPLETED, or the inquire time has expired.
  - TYPE=HISTORY jobs use a minor node for the length of time that is required to run the job.
- Determine the number of ISPF dialogs to use. These use minor nodes as well. The length of time the ISPF dialogs hold on to the minor ACBs follows the XCOMJOBs. Using ISPF to do TYPE=SCHEDULE dialogs requires less time than TYPE=EXECUTE or TYPE=INQUIRE dialogs.

Use the Sample Table to Define Your Product at Your Installation

If you want to modify the sample application definition table that is supplied on the distribution tape, note in particular the following points:

- The APPLID of the server

  XCOMAPPL is the default APPLID for the CA XCOM Data Transport server. If your installation is running only one version of CA XCOM Data Transport, you can retain XCOMAPPL as the server name. However, if you are running more than one version of CA XCOM Data Transport within the same VTAM environment or in the same network, provide each CA XCOM Data Transport server with a unique APPLID. Set up an application definition table for each server.

  If the server for your installation uses an APPLID other than XCOMAPPL, let other CA XCOM Data Transport installations know that APPLID.

- The value of the MODETAB parameter

  Provide the MODETAB parameter in each APPL statement with the name of the logon mode (logmode) table in use at your installation. VTAM searches the logmode table for an entry with the name specified in the DLOGMOD parameter. The logmode describes the session parameters used in an application (defined in an APPL statement) when it participates as a Secondary Logical Unit (SLU).

- The default LOGMODE name

  The DLOGMOD parameter in each APPL statement specifies the applications logmode entry name. The default logmode is XCOMMODE. This name matches the logmode table entry name used in the sample configurations in all other implementations of CA XCOM Data Transport. We recommend that you retain this default.

- The value of the ACBNAME parameter

  The ACBNAME parameter contains the name that an application program specifies in an ACB control block when it wishes to establish a connection with the VTAM application.

  Normally, the name coded in the APPL statement and the value of the ACBNAME are identical. If the ACBNAME parameter is not coded, the name of the APPL definition statement is then used as its value. The ACBNAME used in the APPL statement must match the ACBNAME parameter either in the CA XCOM Data Transport Default Options Table or in the EXEC statement of the server JCL.

  Use XCOM*nn* as the value of ACBNAME in the application definitions batch jobs. If another name is used, assign this name as the value of the JOBACB parameter in the CA XCOM Data Transport Default Options Table.

Important! Do not specify APPC=YES in the APPL definition statement. CA XCOM Data Transport implements a record-level LU 6.2 API based on VTAM macros.

Define the Logon Mode Table Entries (Optional)

In the application definition table (discussed in the preceding section), a logon mode (logmode) table is assigned to each application through the MODETAB parameter. The logmode table contains one or more LOGMODE entries. Each LOGMODE entry defines a particular application session characteristic such as the pacing count and the request unit (RU) size. The association between an application and its LOGMODE entry is established by using the DLOGMOD parameter in the application definition table.

You must have at least one uniquely named LOGMODE entry in the logmode table that your local system uses. If the Default Options Table SNA parameter is set to NO, the definition of LOGMODE entries specific to XCOM is not required.

Construct a Logon Mode Table

A sample logon mode table is provided in CAI.CBXGSAMP(XCOMTABL) on the distribution tape. For a listing of this dataset, see the appendix “Sample Files” in the *CA XCOM Data Transport for z/OS User Guide*. Use the sample logon mode table as a model for constructing a logon mode table for CA XCOM Data Transport at your installation.

A logon mode table consists of three VTAM macros: MODETAB, MODEENT, and MODEEND.

MODETAB

The first macro in a logon mode table is MODETAB. MODETAB indicates the beginning of the logmode table. You can specify the name of the logon mode table with MODETAB. In the sample logon mode table, the MODETAB macro is specified as follows:

```
XCOMAPPL  MODETAB
```

In this example, XCOMAPPL is the name of the sample logon mode table. If the name of the logon mode table is specified in the MODETAB macro, it must be the same as specified in the MODETAB parameter in the CA XCOM Data Transport servers APPL statement.

MODEENT

The MODEENT macro is used to create the LOGMODE entries in a logon mode table. A logon mode table can contain several LOGMODE entries. The sample logon mode table, for instance, provides LOGMODE entries for several RUSIZE values (1 K, 2 K, 4 K, 16 K, 32 K, 512 K, and so on).

The following fragment from the sample logon mode table illustrates how the LOGMODE entries are constructed.

```
XCOMMODE  MODEENT  LOGMODE=XCOMMODE,
                   FMPROF=X'13',
                   TSPROF=X'07',
                   PRIPROT=X'B0',
                   SECPROT=X'B0',
                   COMPROT=X'50B1',
                   RUSIZES=X'8686',
                   PSNDPAC=5,
                   SRVCPAC=5,
                   SSNDPAC=5,
                   PSERVIC=X'060200000000000000002C00'
```

The first item in a LOGMODE entry is the mode entry name (which in the displayed sample is XCOMMODE). The next keyword is MODEENT. The next item is the parameter LOGMODE, which specifies the name of the LOGMODE entry. Following this parameter is a collection of other VTAM parameters that define the session characteristics of the LOGMODE entry XCOMMODE.

MODEEND

The MODEEND indicates the end of the logon mode table.

Default LOGMODE

A LOGMODE entry whose name is the same as the value of the DLOGMOD parameter in the application definition table is the default LOGMODE. The default LOGMODE for CA XCOM Data Transport is XCOMMODE (see the description of the MODEENT parameter). Keep this name as your installations default LOGMODE. If you decide to use a LOGMODE entry name other than XCOMMODE, inform your partner CA XCOM Data Transport installations of the mode name you are using.

Provide a LOGMODE Entry for Parallel Sessions

Your logmode table has to include a LOGMODE entry called SNASVCMG. SNASVCMG an IBM-supplied mode entry that allows an application to use parallel sessions.

Request Unit (RU) Size and Performance

Two values in the mode entry that can have a significant effect on performance are:

- The sending RU (request unit) size
- The receiving RU size

Once CA XCOM Data Transport is operational, evaluate the specified RU size. For information on overriding the RU size for a particular remote system, see the chapter Configuration Parameters.

Override LOGMODE Table Entries

The mainframe uses the default mode name to start a session. The default mode name can be specified in the CA XCOM Data Transport control library rather than in the NCP Line, Group, PU, LU, or switched network definitions. Specifying the default mode name in the CA XCOM Data Transport control library is easier than updating VTAM tables and it requires DLOGMOD=XCOM in the CA XCOM Data Transport Default Options Table.

Normally, the VTAM logon mode table is updated to change the RU size or pacing values. CA XCOM Data Transport permits the overriding of these values for a particular remote system in the CA XCOM Data Transport control library (CAI.CBXGPARM). For more information, see the chapter “CA XCOM Data Transport Configuration Parameters.”

Important! Do not use separate mode tables for LU 6.2. If PCs are being used, Combining APPC mode entries in the same mode table as the 3270 mode entry is recommended. VTAM does not allow any node to override the mode table name. CA XCOM Data Transport can override the mode entry, but not the mode table name.

Parameter Override Relations

Much of the configuration and customization of CA XCOM Data Transport is concerned with the selection of appropriate values for the parameters contained in the Default Options Table and the CA XCOM Data Transport control library.

Note: In the index of this guide, the Default Options Table parameters are called system parameters and the control library parameters are called destination parameters.

Control Library Parameters

Several global parameters that are listed in the Default Options Table or CONFIG member appear also in the individual destination members in the CA XCOM Data Transport control library. For example, DROPSESS, LOGMODE, and LOSERS. Such parameters can be set to one value in the CONFIG member and to a different value in the CA XCOM Data Transport control library. Thus, CA XCOM Data Transport could be configured to have DROPSESS=YES as a global parameter but DROPSESS=QEMPTY in the destination member.

PARM Parameters and SYSIN01 Parameters

In addition to being either a global default (system) parameter or a control library (destination) parameter, a parameter could also function as:

- PARM parameter (specified in the operand field of the EXEC statement in JCL)
- SYSIN01 parameter (specified in the SYSIN01 DD statement in JCL)

For example, LOGMODE is interpretable as a global default (system), control library (destination), or PARM parameter. Likewise, the default/control library parameter DROPSESS can also be used as a SYSIN01 parameter.

XCOMJOB Parameters and SYSIN01 Parameters

For a description of the XCOMJOB and SYSIN01 parameters, see the chapter “The Batch Interface” in the *CA XCOM Data Transport for z/OS User Guide*.

Resolve Multiple Interpretable Parameters

CA XCOM Data Transport employs a special parameter evaluation hierarchy to resolve parameters with multiple interpretations to their correct values. The order in which CA XCOM Data Transport processes the various CA XCOM Data Transport parameter categories is as follows (1 = highest priority, 4 = lowest priority):

1. SYSIN01 parameters
2. CA XCOM Data Transport control library parameters
3. PARM field parameters in the EXEC statement invoking XCOMJOB or XCOMXFER
4. Default Options Table parameters or CONFIG member parameters

Create the New TYPE=CONFIG Member

CA XCOM Data Transport has a set of system parameters that govern its operation across all of its various interfaces. The CA XCOM Data Transport system parameters take effect as soon as CA XCOM Data Transport is started and, unless overridden, they remain in effect as long as CA XCOM Data Transport is active. These system parameters are kept in the CA XCOM Data Transport TYPE=CONFIG member.

For installations converting from a previous release, any existing, customized Default Options Tables can be assembled and linked using the #DFLTAB macro. The #DFLTAB macro is located in the CBXGMAC library that is provided as part of the CA XCOM r11.6 installation. A TYPE=CONFIG member is created the first time a Default Options Table is used to invoke CA XCOM r11.6 address space (XCOMJOB, XCOMXFER, or XCOMXADM). The created TYPE=CONFIG member has the same global parameter options specified in the Default Options Table. The TYPE=CONFIG member is created in the FIRST PDS data set that is allocated to the XCOMCNTL DD statement. The TYPE=CONFIG member maintains the same name as the Default Option Table it was created from.

NOTE: If you do not want CA XCOM to make any updates in your current XCOMCNTL data sets, allocate a new data set and place it first in the XCOMCNTL DD concatenation. Once a TYPE=CONFIG member has been created, it is used for subsequent executions. Changes to configuration parameters and defaults are maintained there, and the Default Options Table is no longer used.

If the Default Options Table is not assembled with the current version of the #DFLTAB macro, the server, or batch job, invocation terminates with messages similar to the following messages:

```
XCOMM002II DEFAULT OPTIONS TABLE SHOULD BE REASSEMBLED.
GEN-LEVEL/VERSION (xxxx-xxx) DOES NOT MATCH XCOMXFER (yyyy-yyy).
```

Note: For descriptions of CA XCOM Data Transport system parameters, see the chapter Configuration Parameters.

Configure the Default Options

CA XCOM Data Transport has a set of system parameters that govern its operation across all of its various interfaces. The CA XCOM Data Transport system parameters take effect as soon as CA XCOM Data Transport is started and, unless overridden, they remain in effect as long as CA XCOM Data Transport is active. These system parameters have historically been kept in the CA XCOM Data Transport Default Options Table. These parameters are now stored in a TYPE=CONFIG control library PDS member.

Important! A TYPE=CONFIG member is automatically created in the first data set in the XCOMCNTL DD concatenation. The first time a Default Options Table is used, for which no TYPE=CONFIG member exists. Subsequent server initialization using the same DFLTAB or CONFIG name uses the TYPE=CONFIG member instead of the assembled Default Options Table.

When migrating from r11.5 or below, it is mandatory that the Default Options Table is reassembled before starting a CA XCOM Data Transport address space for the first time. Reassembling the table ensures that default values for any new Default Option Table parameters are included in the creation of the TYPE=CONFIG member. Failure to reassemble the Default Option Table results in the following error message:

```
XCOMM002IE DEFAULT OPTIONS TABLE SHOULD BE REASSEMBLED.

GEN-LEVEL/VERSION (xxxx-xxx) DOES NOT MATCH XCOMXFER (yyyy-yyy).
```

The CA XCOM Data Transport address space is then terminated.

Note: For descriptions of CA XCOM Data Transport system parameters, see the chapter Configuration Parameters.

Edit the Sample CONFIG Member

The data set CAI.CBXGPARM (XCOMCNFG) on the distribution media provides a sample CA XCOM Data Transport CONFIG member. For a printed sample of the dataset, see the Sample XCOMCNFG file in the appendix Sample Files in the *CA XCOM Data Transport for z/OS User Guide*.

You can edit the TYPE=CONFIG member to customize CA XCOM Data Transport to the requirements of your installation.

To edit the CONFIG member

1. Copy the contents of data set CAI.CBXGPARM (XCOMCNFG), that contains the sample CA XCOM Data Transport CONFIG member.
2. Open the newly created copy of the CA XCOM Data Transport CONFIG member with a text editor of your choice.
3. Assign each of the parameters your installation default value.
4. Save the updated CONFIG member.

Note: The CONFIG member parameters are described in the chapter, Configuration Parameters.

Before the changes made to the CONFIG member take effect, stop and restart CA XCOM Data Transport.

Leave the CONFIG Member Unedited?

CA XCOM Data Transport assigns default values for each CONFIG member parameter. CA XCOM Data Transport takes these values for any parameter not specified or overwritten at installation time. However, in most cases, some level of customization is required to run CA XCOM Data Transport successfully.

More Than One CONFIG Member

If you want to specify different environments for use with CA XCOM Data Transport, you can create a CONFIG member for each environment. To enable a particular environment-specific CONFIG member, specify its name as the value of the XCOMJOB, XCOMXADM, or XCOMXFER parameter CONFIG. CA XCOM Data Transport then uses this alternative CONFIG member.

Configure the CEEOPTS Language Environment Parameters (Optional)

Important: In order to run SSL transfers it is no longer a requirement to update the CEEOPTS member to reflect the CAPKI_HOME variable. Modifying the CA XCOM Data Transport defaults for the LE options (CEEOPTS) is only necessary if changes to LE storage handling or diagnostics are required. In most cases, it is not necessary.

CA XCOM Data Transport uses the IBM Language Environment (LE) to perform network communications over TCP/IP sockets.

The LE runtime parameters can be overridden by using the CEEOPTS DD statement in your CA XCOM Data Transport server or XCOMJOB JCL. The parameters can be placed in a sequential dataset or in a member of a PDS. In either case, the data set must be in fixed or fixed-block format and must reside on a QSAM supported DASD device. A sample of the CEEOPTS DD statement follows:

```
//CEEOPTS DD DSN=your.private.parmlib(CEEOPTS),DISP=SHR
```

The LE runtime parameters provide the ability to tune the characteristics of the Language Environments that CA XCOM Data Transport created and uses. Including storage allocation, error handling, LE tracing, reporting options, and others.

Notes:

- The member CEEOPTS in the library *yourhlq*.CBXGPARM contains the Runtime Options as distributed with CA XCOM Data Transport by default.
- For more information about the application-specific Runtime Options Module (CEEUOPT), as it relates to your version of the IBM operating system, see the Language Environment customization guide for your system.

Assemble and Link Edit User Exits (Optional)

Any user exit parameters in the Default Options set to YES or the name of a user exit module, assemble and link them before any of their exits can be used. The assembly JCL is provided in the library CAI.CBXGJCL(ASM#TBLS). For a description of the user exits in this library, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

Note: In the assembly step, specify CAI.CBXGMAC as the SYSLIB. Also, make sure the correct SYSIN, SYSPUNCH, and SYSLMOD libraries are specified in ASM#TBLS. To simplify future product updates and to preserve the original versions of sample modules, we recommend that load modules generated for product tables and user exits be placed in a private library, rather than in the SMP/E controlled CA XCOM Data Transport target libraries.

Define Destinations

Important! The XCOMCNTL data set is required; defining destinations is optional.

Destination definitions, that are members of the CA XCOM Data Transport control library CAI.CBXGPARM (XCOMCNTL), identify remote LUs, IP names, and their characteristics to the CA XCOM Data Transport server. An XCOMCNTL dataset from an earlier release is compatible and could be used.

Note: For CA XCOM Data Transport to perform transfers to remote LUs, it is not necessary that these LUs are known to the CA XCOM Data Transport server. However, it is necessary that they be defined to VTAM.

Why Define Destinations?

Destination definitions are needed for the following reasons:

Note: XCOMPLEX destination members are no longer required and are not recommended. Connectivity is created automatically between the XCOMPLEX Admin and XCOMPLEX Worker Servers. Support for these types of members is provided only for compatibility with prior releases.

- They allow the CA XCOM Data Transport default parameters to be overridden.

  Several CA XCOM Data Transport parameters can be used both in the CONFIG member (or Default Options Table) and in the CA XCOM Data Transport control library. The parameters defined in the CA XCOM Data Transport control library take precedence over their counterparts in the CONFIG member because of the parameter evaluation hierarchy that CA XCOM Data Transport uses.

  A parameter P is assigned the value X in the CONFIG member (P=X) but a different value Y in a control library member (destination definition). When CA XCOM Data Transport performs a transfer and the XCOMCNTL destination member co*ntainin*g P=Y is enabled, CA XCOM Data Transport considers the P definition in the cont*rol lib*rary member and the definition in the CONFIG member is ignored. When CA XCOM Data Transport performs a transfer and the destination *definit*ion containing P=Y is disabled, CA XCOM Data Transport reads the P d*efinitio*n from the CONFIG member and the definition in the control library member is ignored.

- They allow trusted transfers to z/OS to be performed.

  A trusted transfer allows a remote partner to send a transfer without specifying a password. Instead, the user id of the incoming transfers is compared against any TRUSTID table entries in an XCOMCNTL destination member. For more information, see Set Up Trusted Access Security in chapter 3.

- They allow indirect transfers to be performed.

  An indirect file transfer occurs when CA XCOM Data Transport is used as an intermediate node to pass data between two LUs that are not directly connected with each other. In such a transfer, CA XCOM Data Transport stores the data received from one system and forwards it to another system.

- They are required to provide access to the CA XCOM Data Transport Process SYSOUT (PSO) interface.

  The CA XCOM Data Transport PSO interface allows output from non-CA XCOM Data Transport jobs to be forwarded from the JES queue to remote destinations.

- They allow multiple LUs to be defined as transfer destinations.

  A multi-LU destination is either a group or a list. For a definition of the difference between these destination types, see Group Versus List Destinations in this chapter.

  Note: IP names require destination definitions if you want to override the parameters for that partner.

Construct Destination Tables

The main coding rules to be observed when writing destination definitions are highlighted and illustrated in the following:

- A destination definition consists of a sequence of definition statements having the format *parameter=parameter_value*; for example, GETSESS=YES, where GETSESS is a destination parameter (that is, a parameter that can be used in a destination definition) and YES is its value.

  Note: For descriptions of the destination parameters, see the following sections in the chapter “Configuration Parameters”:

  - Destination Parameters for Single LUs and Groups of LUs
  - List Destination Parameters
  - User Destination Parameters

- Each definition statement must start in the first column of the PDS member and contain no blanks. The first blank signals the end of the statement.
- A destination definition can include comments. The comments must be preceded by an asterisk (*). A comment can start in the first column of the table or it can follow a definition statement on the same line (provided there is at least one blank between the definition statement and the comment).

Fragment of a Destination Definition

The following is a fragment of a properly constructed destination definition. The first seven lines of the definition are comments (an asterisk in the first column). The next six lines each contain a definition statement (TYPE=DEST, and so on) starting in the first column, and each statement is followed by a series of blanks (the first one indicates the end of the statement) and a comment (identified as such by the preceding asterisk).

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
********************************************************************************
*                                                                              *
*CA XCOM Data Transport for z/OS                                                *
*                                                                              *
*FUNCTION:CA XCOM Data Transport r11.6 to UNIX system                           *
*                                                                              *
********************************************************************************
TYPE=DEST                  *This is a dest member
LU=LU2310                  *Logical unit name
WRITER=UNIX                *Name of JES writer for PSO support
GETSESS=NO                 *Session must be operator/remotely activated
ACCSEC=NO                  *Access security fields not used
PSOWAIT=YES                *JES spool scan delay is on
. . .
. . .
. . .
```

Destination Types

The destination defined in a control library member can be any of the following:

- A single LU or IP node
- A group of LUs
- A list of LUs and/or IP nodes, and/or groups of LUs
- A superlist, that is, a list of lists of LUs and/or IP nodes, and/or groups of LUs
- A user

For descriptions of the parameters required for the definition of each destination type, see the following sections in the chapter “Configuration Parameters”:

- Destination Parameters for Single LUs and Groups of LUs
- List Destination Parameters

Group Destinations and List Destinations

CA XCOM Data Transport treats a destination defined as a group of SNA LUs differently from a destination defined as a list of SNA LUs and TCP/IP nodes or as a superlist of lists:

- If CA XCOM Data Transport performs a transfer to a group, the recipient of the transfer is the first available SNA LU in the designated group.
- If CA XCOM Data Transport performs a transfer to a list or to a superlist, CA XCOM Data Transport transfers the data to each of the LUs and TCP/IP nodes referred to in the destination member.

How to Code Different Destination Types

Each destination definition must contain a type specification. This section shows the statements that must (or can) be coded in the control library member for each destination type, with information about any special constraints for that destination category.

Single LU or IP node

```
TYPE=DEST
LU=luname
or
IPNAME=ip_address
[IPPORT=port_number]
```

The TYPE=DEST statement must be the first non-comment statement in the control library member.

luname

The VTAM name of the LU being defined.

ip_address* and *port_number

Identify a node within a TCP/IP network. The specification of *port_number* is optional.

Note: In a TYPE=EXECUTE transfer, the name of the control library member must match the name of the LU, and the XCOMCNTL DD must be included in the TYPE=EXECUTE JCL. A TYPE=SCHEDULE transfer allows the LU name and the control library member name to be different.

For a complete list of the parameters that can be used in a single-LU destination definition, see Destination Parameters for Single LUs, Groups of LUs, and Single IPNAMEs in the chapter “Configuration Parameters.”

Group of LUs

```
TYPE=DEST
GROUP=membername
LU=luname1,...,luname16
```

The TYPE=DEST statement must be the first non-comment statement in the control library member.

The name of the control library member and the name of the group being defined must be the same.

The LU statement can specify up to 16 *luname*s.

Note: When multiple LUs are defined, PARSESS=YES cannot be specified, for example, the LUs belonging to a group are not parallel-session capable.

For a complete list of the parameters that may be used in a group destination definition, see Destination Parameters for Single LUs, Groups of LUs, and Single IPNAMEs in the chapter “Configuration Parameters.”

Group processing is supported for TCP/IP transfers, but only for a single IP address. In this way, an alias may be defined with CA XCOM for an IP address.

List of LUs and/or IP Nodes and/or Groups of LUs

```
TYPE=LIST
LU=luname[,...[,luname]]
and/or
[IPPORT=port_number]
IPNAME=ip_address[,...[,ip_address]]
and/or
GROUP=membername[,...[,membername]]
```

The TYPE=LIST statement must be the first non-comment statement in the control library member.

Multiple lunames, ip_addresses, and group_names can be specified in a single statement or multiple statements can be employed. When a single statement is used to list multiple destinations, the destination must be separated by commas. For example, to specify a list of three LU destinations (L1, L2, and L3) any of the following specifications can be used:

- LU=L1,L2,L3
- LU=L1,L2
  LU=L3
- LU=L1
  LU=L2
  LU=L3
- And so on

Notes:

- If IPPORT is specified, it applies to all following IPNAMEs up to the next IPPORT specification, if any.
- The maximum total number of all destinations in the list is at least 500 and will vary depending on the length of the parameters.
- Each destination on the list must have a member defined in the CA XCOM Data Transport control library.
- For a complete list of the parameters that can be used in a list destination definition, see List Destination Parameters in the chapter “Configuration Parameters.”

Superlist of Lists

```
TYPE=SUPERLIST
LIST=listname[,...[,listname]]
```

The TYPE=SUPERLIST statement must be the first non-comment statement in the control library member.

Multiple *listname*s can be specified in a single statement or multiple statements can be employed. When a single statement is used to list multiple destinations, the destination must be separated by commas. For example, to specify a list of three list destinations (L1, L2, and L3) any of the following specifications can be used:

- LIST=L1,L2,L3
- LIST=L1,L2
  LIST=L3
- LIST=L1
  LIST=L2
  LIST=L3
- And so on

Notes:

- CA XCOM Data Transport lists are limited to 32720 bytes of storage, as follows:
  - There is a 32-byte storage descriptor, leaving 32688 bytes for the list.
  - Each list entry occupies 4 to 10 bytes:
    - 1 byte for the TYPE
    - 1 flag byte
    - 1 byte for the LENGTH
    - 1 to 8 bytes for the name of the list.

  So, if each member name specified as a list or superlist is 8 characters long, a superlist can identify 2971 member names.
- Each destination list on the superlist must have a member defined in the CA XCOM Data Transport control library.
- For a complete list of the parameters that can be used in a list destination definition, see List Destination Parameters in the chapter “Configuration Parameters.”

Examples of Destination Definitions

This section contains a brief illustration of a destination definition for each destination type. For more examples, see the CA XCOM Data Transport control library, CAI.CBXGPARM.

Single LU Destination Member

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
*******************************************************************************
*                                                                             *
*CA XCOM Data Transport r11.6                                                 *
*                                                                             *
*FUNCTION:CA XCOM Data Transport r11.6 to UNIX system                         *
*                                                                             *
*******************************************************************************
TYPE=DEST                   *This is a destination definition
LU=LU2310
WRITER=UNIX                 *Name of JES writer for PSO support
GETSESS=NO                  *Session must be operator/remotely activated
ACCSEC=NO                   *Access security fields not used
PSOWAIT=YES                 *JES spool scan delay is on
PARSESS=NO                  *Parallel session support not required
SETUP=NO                    *Do not pass FORMS etc.
LOGMODE=XCOMMODE            *Logon mode table entry
SRPACE=5                    *Secondary receive pacing/NCP (PACING)
SSPACE=5                    *Secondary send pacing/no NCP/VTAM
PRPACE=5                    *Primary to VTAM/NCP receive pacing/VPACING
PSPACE=5                    *Primary to VTAM/NCP send pacing/VPACING
```

The first two lines in the above destination definition indicate that a single LU (LU2310) is being defined as a destination. Following the definition of the destination type is a series of statements that associate a set of communication characteristics with LU2310. These characteristics take effect as soon as the control library member containing the definition of LU2310 is enabled. Once enabled, they remain in effect until a command disabling them is issued (the commands for enabling and disabling control library members are described later in the section How to Enable and Disable a Destination Member).

Note: When the destination member for an LU is *not* enabled (or no destination member exists), CA XCOM Data Transport uses the parameters defined in the Default Options Table for any transfers to that LU.

Group Destination Member

The following destination member defines a group named GROUP1. The LUs that belong to this group are listed on the second non-comment line (LU=XCSAPP4,...) while the group itself is named on the third non-comment line.

Except for the assignment of multiple values (LUs) to the LU parameter and the presence of the statement GROUP=GROUP1, the group destination definition is the same as the preceding single LU destination definition.

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
*******************************************************************************
*                                                                             *
*CA XCOM Data Transport r11.6                                                 *
*                                                                             *
*FUNCTION:CA XCOM Data Transport r11.6 to UNIX system                         *
*                                                                             *
*******************************************************************************

TYPE=DEST                          *This is a destination definition
                                   *The next line lists the LUs in the group
LU=XCSAPP4,LU2310,LU23X05,TSO26001
GROUP=GROUP1                       *Name of JES writer for PSO support
GETSESS=NO                         *Session must be operator/remotely activated
ACCSEC=NO                          *Access security fields not used
PSOWAIT=YES                        *JES spool scan delay is on
PARSESS=NO                         *Parallel session support not required
SETUP=NO                           *Do not pass FORMS etc.
LOGMODE=XCOMMODE                   *Logon mode table entry
SRPACE=5                           *Secondary receive pacing/NCP (PACING)
SSPACE=5                           *Secondary send pacing/no NCP/VTAM
PRPACE=5                           *Primary to VTAM/NCP receive pacing/VPACING
PSPACE=5                           *Primary to VTAM/NCP send pacing/VPACING
```

List Destination Member

A list destination definition can contain only the following two statements:

```
TYPE=LIST
LU=XCOMQA,TS223,TS222
```

The statement TYPE=LIST indicates that the destination being defined is a list. The LUs that belong to the destination list are defined in the LU statement. Each LU in the list must have a member defined in the CA XCOM Data Transport control library, for example:

```
TYPE=DEST               *This is a destination member
LU=TS223                *VTAM node name
WRITER=TS223            *JES writer name
LOGMODE=XCOMMODE
GETSESS=YES


TYPE=DEST               *This is a destination member
LU=TS222                *VTAM node name
WRITER=TS222            *JES writer name
LOGMODE=XCOMMODE
GETSESS=YES


TYPE=DEST               *This is a destination member
LU=XCOMQA               *VTAM node name
WRITER=XCOMQA           *JES writer name
LOGMODE=XCOMMODE
GETSESS=YES
```

Superlist Destination Member

A list destination definition can contain only the following two statements:

```
TYPE=SUPERLIST
LIST=XCOMQA,TS223,TS222
```

The statement TYPE=SUPERLIST indicates that the destination being defined is a list. The lists that belong to the destination superlist are defined in the LIST statement. Each list in the superlist must have a member defined in the CA XCOM Data Transport control library, for example:

```
Member XCOMQA
TYPE=LIST
IPPORT=8044
IPNAME=QADEST1.MYCC.COM

Member TS223
TYPE=LIST
IPPORT=8046
IPNAME=QADEST2.MYCC.COM

Member TS222
TYPE=LIST
IPPORT=8044
IPNAME=QADEST2.MYCC.COM
```

How to Enable and Disable a Destination Member

The properties assigned to a particular destination member take effect only when that destination member is enabled. The destination definition remains enabled until a command to disable it is issued.

There are two basic ways to enable control library members:

- Using the CA XCOM Data Transport MODIFY command ENABLE

  The ENABLE command allows the user to enable only one control library member at a time. Control members can be disabled using the CA XCOM Data Transport DISABLE command. The ENABLE and DISABLE commands can be issued at any time.

- Using CA XCOM Data Transport START parameter

  The START parameter allows several destination members to be enabled automatically but only at CA XCOM Data Transport initialization.

Below, both methods of enabling/disabling control library members are described in greater detail. For complete details, see the chapter "Operation and Control" in the *CA XCOM Data Transport for z/OS User Guide*.

ENABLE and DISABLE Commands

To enable a control library member

Use the following ENABLE command:

```
F XCOM,ENABLE,member_name
```

member_name

The control library member to be enabled.

Note: If the *member_name* is a TYPE=SUPERLIST destination member, the ENABLE attempts to enable all the members contained in the superlist.

Example

The MODIFY command F XCOM,ENABLE,LU2310 activates the destination member LU2310 in the CA XCOM Data Transport control library.

To disable a control library member

Use the following DISABLE command:

```
F XCOM,DISABLE,member_name{,ALL|FORCE|FORCEALL}
```

member_name

The control library member to be deactivated.

ALL

Indicates that all lists in the specified superlist are to be disabled.

FORCE

Indicates that the specified superlist is to be disabled, but that the individual lists in the superlist are not to be disabled.

FORCEALL

Indicates that a superlist named in another superlist is to be disabled, as well as all individual lists named in the superlist.

Note: The optional parameters ALL, FORCE, and FORCEALL are supported only when the *member_name* is a TYPE=SUPERLIST destination member.

Example

The MODIFY command F XCOM,DISABLE,LU2310 deactivates the destination member LU2310 in the CA XCOM Data Transport control library.

CONFIG Parameter

The user defines a member using one of the methods described in the previous section entitled Create the new TYPE=CONFIG Member or use the default version that is shipped with CA XCOM Data Transport. If this member name differs from XCOMDFLT, the default, the CONFIG parameter specifies the name of the TYPE=CONFIG member.

Because the CONFIG parameter value is assigned early in the CA XCOM Data Transport initialization process, it is specified on the EXEC card in the JCL initiating the CA XCOM Data Transport server or batch utility.

member_name

A control library member, that contains the initialization parameters and their respective value assignments that are implemented for the current CA XCOM Data Transport execution.

Note: The parameter keywords must begin in column 1. Parameter values are continued across multiple lines by placing a “+” character immediately after the last non-blank character on a line. Subsequent continuation lines must begin in columns 2 – 16.

If no CONFIG member name is assigned to the CONFIG parameter,CA XCOM Data Transport reads the default member matching the value of the DFLTAB parameter. If the DFLTAB parameter is also missing, CA XCOM Data Transport uses XCOMDFLT as the CONFIG member name for the parameter values.

Customize Code Page Conversion Tables (Optional)

CA XCOM Data Transport has a code page conversion table feature that enables CA XCOM Data Transport to perform data translation based on the type of data (ASCII, BINARY, or EBCDIC) being received at a particular destination. When a session is established for a destination requiring translation, CA XCOM Data Transport searches an internal link list to find the requested code page conversion tables in the sending destination's definition. If the specified code page conversion tables are available, the destination acquires access to them and translation occurs for all direct data transfers. No data translation occurs for indirect file transfers. If the conversion tables are unavailable, the file transfers proceed without data translation.

Data translation occurs on the CA XCOM Data Transport server *receiving* the data file. Therefore, the code page conversion table and destination definition need to be defined and enabled on the receiving CA XCOM Data Transport server for data translation to occur. You must code and enable the code page conversion tables before the destination members.

For conversion to take place in SNA destination members, the member name used in XCOMCNTL must be the same as the LU=*name* in the destination member statement.

The Construction of the Code Page Conversion Table

The code page conversion tables are members of the CA XCOM Data Transport control library. They are constructed in the same way as the CA XCOM Data Transport destination tables, that is:

- Each non-comment statement in the table has the format *parameter=parameter_value* (for example, TYPE=CONVERT).
- Each statement must begin in the first column of the table.
- The first blank forms the end of a statement.

The following figure illustrates the format of the code page conversion table:

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
TYPE=CONVERT
NAME=xxxxxxxx
CNVVAL00=nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn
CNVVAL01=nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn
. . .
. . .
. . .
CNVVALF0=nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn,nn
```

The Parameters of the Code Page Conversion Table

A code page conversion table is constructed using three parameters, CNVVAL*nn* (*nn* being a hexadecimal number), NAME, and TYPE. These parameters are described in the next few sections. The sixteen rows headed by the keyword CNVVAL*nn* are the actual conversion table that is responsible for character conversion. Each of the sixteen rows of the conversion table consists of 16 hexadecimal numbers representing the characters of the code page to be converted.

Customize the Code Page Conversion Table

You can create a customized code page conversion table by changing the hexadecimal table characters in the NOTRANS table to your specifications. See the sample code page conversion table in the section discussing the CNVVAL*nn* parameter. After you have made the desired changes to hexadecimal character values, save the new table under a different member name.

Conversion Table Parameters

To construct a code page conversion table, you need to code three parameters, CNVVAL*nn*, NAME, and TYPE. These parameters are described in the following sections.

CNVVAL00F0

Specifies the number of the first character *position* on one of the 16 rows in the code page conversion table.

nn_1,nn_2,...nn_{16}

Specifies 16 hexadecimal character values that form a row in the code page conversion table.

Default: None

Notes:

- There are 256 character positions in the code page conversion table and the related code page (the NOTRANS table). In both, each successive character position has a numeric value that is one higher than the numeric value of the preceding character position. The character position numbers extend from 00 to F0 (255 in decimal).
- In the code page conversion table, the hexadecimal number *nn* representing a character may or may not be the same as the number of the position in which it appears. In the code page to be converted, the two values match in every instance. Character conversion is based on comparing character values that are mapped into the same position in the code page and the code page conversion table.

Example:

As an example, consider the following code page conversion table, which converts all uppercase EBCDIC characters into their lowercase counterparts:

```
TYPE=CONVERT
NAME=LOWER
*
*   SINGLE BYTE CHARACTER SET TRANSLATION TABLE
*
*  THIS IS A SAMPLE CONVERSION TABLE TO TRANSLATE UPPER CASE TO LOWER CASE CHARACTERS
*
CNVVAL00=00,01,02,03,04,05,06,07,08,09,0A,0B,0C,0D,0E,0F
CNVVAL10=10,11,12,13,14,15,16,17,18,19,1A,1B,1C,1D,1E,1F
CNVVAL20=20,21,22,23,24,25,26,27,28,29,2A,2B,2C,2D,2E,2F
CNVVAL30=30,31,32,33,34,35,36,37,38,39,3A,3B,3C,3D,3E,3F
CNVVAL40=40,41,42,43,44,45,46,47,48,49,4A,4B,4C,4D,4E,4F
CNVVAL50=50,51,52,53,54,55,56,57,58,59,5A,5B,5C,5D,5E,5F
CNVVAL60=60,61,62,63,64,65,66,67,68,69,6A,6B,6C,6D,6E,6F
CNVVAL70=70,71,72,73,74,75,76,77,78,79,7A,7B,7C,7D,7E,7F
CNVVAL80=80,81,82,83,84,85,86,87,88,89,8A,8B,8C,8D,8E,8F
CNVVAL90=90,91,92,93,94,95,96,97,98,99,9A,9B,9C,9D,9E,9F
CNVVALA0=A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,AA,AB,AC,AD,AE,AF
CNVVALB0=B0,B1,B2,B3,B4,B5,B6,B7,B8,B9,BA,BB,BC,BD,BE,BF
CNVVALC0=C0,81,82,83,84,85,86,87,88,89,CA,CB,CC,CD,CE,CF
CNVVALD0=D0,91,92,93,94,95,96,97,98,99,DA,DB,DC,DD,DE,DF
CNVVALE0=E0,E1,A2,A3,A4,A5,A6,A7,A8,A9,EA,EB,EC,ED,EE,EF
CNVVALF0=F0,F1,F2,F3,F4,F5,F6,F7,F8,F9,FA,FB,FC,FD,FE,FF
```

Up to character position C0 (in the thirteenth row, CNVVALC0), the characters' numeric values and their positions match. In the ninth row (CNVVAL80), for instance, character position 81 is occupied by a character whose hexadecimal value is 81 (this is the lowercase a in the EBCDIC code page). However, in the nine character positions following positions C0 and D0 and in the eight positions following E1, the expected character values C1-C9 (hex for the uppercase characters A-I), D1-D9 (J-R), and E2-E9 (Q-Z) do not occur. The character values mapped into positions C1-C9, D1-D9, and E2-E9 are the EBCDIC lowercase characters a-i, j-r, and q-z (81-89, 91-99, and A2-A9), respectively.

When a destination that requires character translation receives a character like A (whose hexadecimal value in the EBCDIC code page is C1), CA XCOM Data Transport checks position C1 in the destination's code page conversion table (which must be enabled) and replaces the received character (C1) with the character mapped into conversion table position C1 (which in the above sample table is 81 or the lowercase a). This replacement is an instance of character conversion.

If you called the above conversion table in the CA XCOM Data Transport Destination Table by setting CVEBCDIC=LOWER, only EBCDIC data transfers would be converted. No data conversion would occur for ASCII or binary data.

You must set the Destination Table Parameters CVASCII, CVBINARY and CVEBCDIC to LOWER to convert all data types from uppercase characters to lowercase characters. Depending on the translation needs of the destination server, all three parameters (CVASCII, CVBINARY, CVEBCDIC) may be coded with the same, different, or no code page conversion table names.

NAME

Specifies the name of the code page conversion table.

xxxxxxxx

Specifies a string of up to 8 alphanumeric characters to be used as the name of the code page conversion table.

Default: None

Note: The NAME parameter must be the second non-comment statement in the code page conversion table.

TYPE

Specifies the type of the table contained in a control library member.

CONVERT

Specifies that the table contained in the control library member is a code page conversion table.

Default: None

Note: TYPE=CONVERT must be the first non-comment statement in the code page conversion table.

Define the System Administrator Table (Optional)

CA XCOM Data Transport provides a facility that enables you to define various levels of CA XCOM Data Transport system administration and operating control. This facility is driven by the CA XCOM Data Transport System Administrator Table. A sample macro to create this facility is provided in the CA XCOM Data Transport Samples Library (CAI.CBXGSAMP(XCOMADMT)). For an example of the coding for the #ADMTAB macro, see the appendix “Sample Files” in the *CA XCOM Data Transport for* z/OS *User Guide*.

Modify the System Administrator Table

To define the CA XCOM Data Transport System Administrator to CA XCOM Data Transport, you need to modify the System Administrator Table.

To modify the System Administrator Table

1. Open the data set CAI.CBXGSAMP(XCOMADMT), which contains the CA XCOM Data Transport System Administrator Table.
2. Assign each parameter an appropriate value.

   The System Administrator Table parameters are described in the next section.
3. Assemble and link the System Administrator Table.

   The assembly JCL is provided in the library CAI.CBXGJCL(ASM#TBLS).

   Note: In the assembly step, specify CAI.CBXGMAC as the SYSLIB. Make sure that the correct SYSIN, SYSPUNCH, and SYSLMOD libraries are specified in ASM#TBLS.

System Administrator Table Parameters

This section describes the parameters that are used in the construction of the System Administrator Table.

ACCESS

Defines the CA XCOM Data Transport Administrator's authority to use the CA XCOM Data Transport TSO/ISPF functions and operator console commands.

ALL

Specifies that the Administrator can access all TSO/ISPF functions and issue all commands.

ALT

Specifies that the Administrator can alter the transfer's start date/time and execution priority.

DACT

Specifies that the Administrator can display active transfers.

DEL

Specifies that the Administrator can delete transfers from the scheduled transfer queue.

DHST

Specifies that the Administrator can display history records.

DSCH

Specifies that the Administrator can display scheduled transfers.

HOLD

Specifies that the Administrator can hold transfers.

NONE

Specifies that the Administrator cannot access any TSO/ISPF functions and cannot issue any commands.

REL

Specifies that the Administrator can release held transfers.

RESM

Specifies that the Administrator can resume suspended transfers.

SUSP

Specifies that the Administrator can suspend transfers.

TERM

Specifies that the Administrator can terminate active transfers.

Default: ALL

Notes:

- For more information about the CA XCOM Data Transport TSO/ISPF functions, see the chapter “The Menu Interface (TSO/ISPF Panels)” in the *CA XCOM Data Transport for z/OS User Guide*.
- For more information about the CA XCOM Data Transport operator control commands, see the chapter “Operation and Control” in the *CA XCOM Data Transport for z/OS User Guide*.

ADMIN

Defines the CA XCOM Data Transport Administrator's user ID.

xxxxxxxx

Specifies the CA XCOM Data Transport Administrator's user ID. The user ID can contain up to 8 alphanumeric characters.

Default: None

When a request is received by the XCOMXFER server, CA XCOM Data Transport checks the user's authority definition in the CA XCOM Data Transport Administrator Table and allows the user to perform the requested function if authority is granted under the ACCESS parameter.

GROUP

Defines a set of users to be controlled by the Administrator.

xxxxxxxx

Specifies the name of a group of users to be controlled by the Administrator. The name can contain up to 8 alphanumeric characters.

Default: No restriction of users

Notes:

- In the string of characters that is the value of GROUP, you can use an asterisk (*) as a wildcard to represent an arbitrary character sequence.
- If a request is issued calling for a user that is not in the group, the request is denied.

Define the Server in a Standalone Environment

The CA XCOM Data Transport server uses SNA LU 6.2 sessions to communicate with TSO users, batch jobs, CICS, TPF, and all other address spaces in which users might ask CA XCOM Data Transport to do work, unless you are using TCP/IP Scheduling and Inquiry. It maintains a queue of file transfers it will initiate on behalf of users who have scheduled them. It is also the target for file transfers initiated by other CA XCOM Data Transport nodes. If using TCP/IP for transfers with z/OS, see the IBM publications for information about the Open Edition Security with TCP/IP.

Complete the Server Storage Usage Worksheet

For the appropriate Server Storage Usage Worksheet, see the CA XCOM Data Transport web pages for the z/OS platform at http://ca.com/support. Completing the worksheet allows you to determine the appropriate storage usage for the CA XCOM Data Transport server and appropriate values for some of the Default Option Table parameters to be configured. For more information, see the chapter “Configuration Parameters.”

Create the Address Space

Defining the CA XCOM Data Transport server amounts to describing the address spaces that must be created before CA XCOM Data Transport can perform any of its functions. After basic installation and customization are completed, the next step is to create the primary CA XCOM Data Transport address space. The sample JCL provided in CAI.CBXGJCL(XCOM) can be edited as appropriate to your installation and used to initiate the CA XCOM Data Transport server. It should be added to SYS1.PROCLIB or some other PROCLIB.

DD Statements for the History File

Remember that the DD statements for the history file alternate indices points to the PATH and not to the alternate index. For a copy of this sample, see the sample JCL provided in CAI.CBXGJCL(XCOM).

Override Processing Options

The PARM parameter is optional and has been included to demonstrate one method of overriding the installation-wide processing options established through the CA XCOM Data Transport CONFIG Member. Certain processing options set in the CA XCOM Data Transport CONFIG Member can be overridden through the EXEC PARM operands of the CA XCOM Data Transport server JCL. (The TIMEOUT parameter is not used by the CA XCOM Data Transport server.) These overrides prevail as long as this server is running.

The following processing options can be overridden by the PARM parameter:

- ACBNAME, AGE, ALLOC
- CATALOG, CLASS, COMPNEG, CONFIG
- DFLTAB, DIR, DOMAIN, DUMPCL
- EDESC, EROUT, ERRINTV
- FERL
- IDESC, IROUT
- JESINTV
- LOG, LOGCLASS, LOGDEST, LOGMODE
- NETNAME
- PLEXQ, PRI, PSO, PSOUNIT, PSOVOL
- REMAGE, REPCR
- SEC, SERL, SMF, SMFNUM, START, SUPPLIST, SWAIT
- TCPSTACK, TERL
- UNIT, USERD
- VERL, VOL, VTAMGNAM
- XCOMPLEX

Non-Swappable

The CA XCOM Data Transport server does not require any special dispatching priority. It is recommended but not required that CA XCOM Data Transport be non-swappable. However, if the environment where CA XCOM Data Transport and VTAM are highly utilized is characterized by high paging or CPU usage, you should make CA XCOM Data Transport non-swappable and/or assign it to a special performance group with higher dispatching priority.

Operating Your Product

CA XCOM Data Transport operation can be initiated from the z/OS console by issuing the START command. CA XCOM Data Transport can also be submitted as a TSO or batch job. We recommend operating CA XCOM Data Transport as a server (that is, by using the START command at the z/OS console).

For a complete list of CA XCOM Data Transport operator commands, see the chapter "Operation and Control" in the *CA XCOM Data Transport for z/OS User Guide*.

Create a PLEXQ Environment (Optional)

CA XCOM Data Transport for z/OS can distribute and manage locally initiated transfers among servers in a CA XCOM Data Transport PLEXQ. A PLEXQ consists of one or more CA XCOM servers. Transfers intended for servers in the PLEXQ may be generically scheduled to the group name used by the PLEXQ server(s) as defined via the PLEXQ parameter.

Through the use of the IBM Parallel Sysplex Signaling Services, CA XCOM Data Transport manages the distribution of transfer workload amongst servers which are members of the same PLEXQ.

To create a PLEXQ environment, simply specify the same name in the PLEXQ parameter either in the CONFIG member or as part of the EXEC PARM for each CA XCOM server which is to participate in the same PLEXQ.

Multiple PLEXQ environments may be defined within a SYSPLEX, however, a server may participate in only ONE PLEXQ group at any given time.

NOTE: All participant members of a PLEXQ must be the same release.

Define the PLEXQ Server(s) (Optional)

To make a CA XCOM Data Transport server a member of a PLEXQ (see Define the CA XCOM Data Transport Server in a Standalone Environment in this chapter) configure it as follows:

To define a Server as a member of a PLEXQ

1. Specify the name of the PLEXQ group using the PLEXQ parameter or by specifying it in the EXEC PARM for the server JCL.
2. At least one PLEXQ server must be available in the PLEXQ or it will not be possible to schedule work to the PLEXQ.
3. Each PLEXQ server must have its own XCOMRRDS and history file, and be configured to run as a standalone server.

NOTE: All participant members of a PLEXQ must be the same release.

Schedule Transfers to the PLEXQ

Transfers can be scheduled either to the PLEXQ group or directly to a server using the SNA or TCP/IP protocol. Transfers are scheduled to a PLEXQ by using the STCPLEXQ parameter in the EXEC PARM of the XCOMJOB utility program. Connections established to a PLEXQ do not use SNA or TCP/IP protocols, but rather a proprietary messaging protocol which uses SYSPLEX Signaling Services as its transport layer.

Attempts to schedule or inquire on transfers to a PLEXQ when no servers are active in the PLEXQ will be rejected.

Same Release

All servers which are members of the same PLEXQ group must be at the same CA XCOM Data Transport release.

Default Options

Parameters for transfers scheduled to a PLEXQ group member server can be taken from the Default Options for the server which ultimately receives the transfer request or from another specified Default Options configuration. Parameters specified in the SYSIN01 and destination members will override these parameters.

Sample PARM Statements

The following is a sample PARM statement. If it is used when sending transfers to a PLEXQ group member server, then CA XCOM Data Transport will take defaults from XCPTDFB0, which is a Default Options CONFIG member (or Table) for the target PLEXQ group server.

```
PARM=('TYPE=SCHEDULE,DFLTAB=XCPTDFB0,STCPLEXQ=PRODXCOM')
```

Define the XCOMPLEX in a Coupling Facility Environment (Optional, Deprecated)

Important: The existing XCOMPLEX facility has been deprecated. The following information is provided for backward compatibility only. All new installations use the new PLEXQ implementation. Existing XCOMPLEX users migrate to the PLEXQ infrastructure for their XCOMPLEX functionality.

CA XCOM Data Transport for z/OS can distribute and manage locally initiated transfers among XCOMPLEX Worker Servers in an XCOMPLEX. An XCOMPLEX consists of an Admin Server and one or more XCOMPLEX Worker Servers. All transfers for servers in the XCOMPLEX are scheduled to the XCOMPLEX Admin Server. The XCOMPLEX Admin Server does not actually perform transfers.

By using the IBM Parallel Sysplex Coupling Facility, various CA XCOM Data Transport lists are maintained to aid in the distribution of the workload.

To define the XCOMPLEX.

1. Define the overall IBM Coupling Facility data set. CAI.CBXGJCL(XCPF) contains a sample job to define the IBM Coupling Facility data set. This step is not needed if you have an existing IBM Coupling Facility defined.
2. Define a policy data set for the XCOMPLEX. This policy data set defines the XCOMPLEX to the IBM Coupling Facility. CAI.CBXGJCL(POLCFRM) contains a sample job to define XCOMPLEX structures to the Coupling Facility. An XCOMPLEX name can be one to 16 alphanumeric characters. The name for the XCOMPLEX must be unique and must not be the same as the acbname for any server in the XCOMPLEX. Servers are in a common XCOMPLEX, as long as they share a common Coupling Facility. There can be a maximum of 61 XCOMPLEX Worker Servers for each XCOMPLEX. For more information about working with the Coupling Facility, see the IBM documentation.

   Note: Multiple XCOMPLEXes can be defined, even on the same system.
3. Define which servers make up the XCOMPLEX. The name of the XCOMPLEX is specified in the Default Options Table using the XCOMPLEX parameter. All servers in the XCOMPLEX are identified by specifying the same name for the XCOMPLEX parameter in their respective Default Options Tables. Each server can belong to only one XCOMPLEX.

NOTE: All participant members of the XCOMPLEX must be the same release.

Define the XCOMPLEX Admin Server in a Coupling Facility Environment (Optional)

The XCOMPLEX Admin Server manages the XCOMPLEX. The XCOMPLEX Admin Server requires different JCL from the XCOMPLEX Worker Servers. An XCOMPLEX Admin Server is a server that is brought up with PGM=XCOMXADM on the EXEC statement of the server JCL.

To define the XCOMPLEX Admin Server.

1. Define the XCOMPLEX Admin Server. CAI.CBXGJCL(XCOMADM) contains a sample job for the XCOMPLEX Admin Server.
2. Specify the name of the XCOMPLEX in the Default Options Table using the XCOMPLEX parameter. Specify the name of the XCOMPLEX by specifying in the EXEC PARM of the XCOMPLEX Admin Server JCL.
3. The Admin Server must have its own XCOMRRDS and history file.

Notes:

- The XCOMPLEX Admin Server does not schedule transfers itself. At least one XCOMPLEX Worker Server must be available for transfers that are scheduled. If there are no XCOMPLEX Worker Servers, the transfers are rejected.
- All participant members of the XCOMPLEX must be the same release.

Define the XCOMPLEX Worker Server in a Coupling Facility Environment (Optional)

An XCOMPLEX Worker Server is set up the same way as the standalone server with the following requirements:

To define the XCOMPLEX Worker Server.

1. Specify the name of the XCOMPLEX in the Default Options using the XCOMPLEX parameter. Specify the name of the XCOMPLEX by specifying it in the EXEC PARM for the XCOMPLEX Worker Server JCL.
2. For the XCOMPLEX Admin Server to schedule any transfers, at least one XCOMPLEX Worker Server must be available in the XCOMPLEX.
3. Each Worker Server must have its own XCOMRRDS and history file.

NOTE: All participant members of the XCOMPLEX must be the same release.

Schedule Transfers in the XCOMPLEX

Transfers can be scheduled either through the Admin Server or directly to a Worker. Transfers are scheduled to the Admin Server in exactly the same way as they are scheduled to stand alone servers, using the same JCL or the same panels. The Admin Server does not queue any of them, but immediately distributes them to the Worker Servers. At least one Worker Server must be active and connected to the XCOMPLEX since the Admin Server itself does not actually queue transfers. Any transfers sent to the Admin Server when no Worker Servers are active will be rejected. If any transfers are sent to the Admin Server that are not scheduled (for example, TYPE=EXECUTE), the Admin Server will reject them with an error message.

Same Release

In an XCOMPLEX environment, the Admin Server and Worker Servers must be at the same CA XCOM Data Transport release. This restriction is not programmatically enforced, but the mixing of releases within a single XCOMPLEX is not a supported environment.

Default Options

Parameters for transfers scheduled to the XCOMPLEX Admin Server can be taken from the Default Options for the Admin Server or from another specified Default Options configuration. Parameters specified in the SYSIN01 and destination members will override these parameters.

Sample PARM statements

The following is a sample PARM statement. If it is used when sending transfers through the XCOMPLEX Admin Server, then CA XCOM Data Transport will take defaults from XCPTDFB0, which is a Default Options Table for the XCOMPLEX Admin Server.

```
PARM=('TYPE=SCHEDULE,DFLTAB=XCPTDFB0')
```

There is another sample PARM statement below. If it is used when sending transfers to the XCOMPLEX Admin, then CA XCOM will take defaults from XCPXDFB0, which is a Default Options Table shared by the XCOMPLEX Worker Servers. The transfers will still be routed through the XCOMPLEX Admin Server since the ACBNAME and STCAPPL parameters point to the XCOMPLEX Admin Server.

```
PARM=('TYPE=SCHEDULE,DFLTAB=XCPXDFB0,ACBNAME=XCPT,STCAPPL=XCPT')
```

Configure Virtual IP Addresses—Remotely-Initiated Transfers Only (Optional)

CA XCOM Data Transport for z/OS can utilize Virtual IP Addressing with multiple CA XCOM Data Transport servers sharing a virtual IP address. CA XCOM Data Transport does not share the PORT, so each server must be on a different stack, or you must configure your TCP/IP stack to share a port. For the CA TCPaccess CS stack, specify ACCESS(SHR) for the PORTRULE definition or for the IBM stack, specify SHAREPORT in your TCP/IP profile. To use this support, your system must be configured for Virtual IP addresses. Consult your network support group for additional information. They will provide the virtual IP address and port.

TCP/IP distributes scheduled CA XCOM Data Transport transfers to the different CA XCOM Data Transport servers. The IBM Coupling Facility is required for this feature. This does not require the XCOMPLEX unless checkpointing is in use. The XCOMPLEX is required to be able to restart a transfer from a checkpoint when using Virtual IP addresses.

CA XCOM Data Transport accepts or rejects connection requests based on the configuration of TCP/IP and the Default Option Table parameters SERVADDR, SERVPORT, SSLPORT, SERVADDRV6, SERVPORTV6, SSLPORTV6, TCPIPV6, and TCPSTACK:

SERVADDR

This parameter can be used to define an incoming IP address. If specified, XCOM will accept connection requests only for this IP address. This may be subject to the TCP/IP configuration and what connections are permitted by the TCP/IP stack.

SERVPORT

Specifies the TCP/IP port that XCOM listens on.

SSLPORT

Specifies the SSL port that XCOM listens on.

SERVADDRV6

This parameter can be used to define an incoming TCP/IPv6 address. If specified, XCOM will accept connection requests only for this TCP/IPv6 address. This may be subject to the TCP/IP configuration and what connections are permitted by the TCP/IP stack.

SERVPORTV6

Specifies the TCP/IPv6 port that XCOM listens on.

SSLPORTV6

Specifies the TCP/IPv6 SSL port that XCOM listens on.

TCPIPV6

Specifies whether TCP/IPv6 is to be used by the CA XCOM Data Transport server.

TCPSTACK

Specifies whether XCOM TCP/IPv6 listener subtasks will be started in order to handle incoming TCP/IP connections.

For more information about these parameters, see their descriptions in Default Options Table Parameters in the chapter "Configuration Parameters."

Configure VTAM Generic Names—Remotely-Initiated Transfers (Optional)

CA XCOM Data Transport for z/OS can be used with VTAM Generic Name Support. It can be used on the same system or different systems, but all servers using the same VTAM Generic Name must be in the same SYSPLEX, since IBM VTAM Generic Name Support uses the Coupling Facility.

The IBM Coupling Facility is required. VTAM Generic Name support does not require the XCOMPLEX.

VTAM uses the IBM Work Load Manager to distribute the incoming transfers based on the level of existing sessions. VTAM takes existing parallel sessions into consideration when distributing the transfers:

- The VTAMGNAM parameter must be specified in the Default Options Table for each XCOMPLEX Worker Server. Do not specify the VTAMGNAM parameter for the XCOMPLEX Admin Server.
- Remote partners should send SNA transfers to the VTAM generic name supplied by your network support group.

Note: Any or all of these facilities can be used with CA XCOM Data Transport. Each facility can be used independently of the others.

Assemble and Link Edit the JES2-Dependent Module

This section describes how to assemble and link edit the CAI.CBXGSAMP member XCOMJ001.

JES2 Installations

The CAI.CBXGSAMP member XCOMJ001 contains JES2 macros. You need to assemble and link edit it for your installation. The JCL for this task is provided in the CBXGJCL member ASM#TBLS.

To assemble and link edit the CAI.CBXGSAMP member XCOMJ001

1. Uncomment the DD statement in the SYSLIB concatenation that refers to data set SYS1.AHASMAC and ensure that it points to the data set where the JES2 macros reside at your installation.
2. Uncomment the EXEC statement that refers to XCOMJ001.
3. Execute the JOB to assemble and link edit the XCOMJ001 module.
4. Make sure you assemble XCOMJ001 on the same z/OS release as the system on which it will be used.

The use of this module is strictly optional. If it is not installed, CA XCOM Data Transport will obtain information about SYSOUT data sets only from the JES2 Process SYSOUT interface. In general, more accurate and detailed information can be obtained directly from JES2, as illustrated by this module.

This module is required if you are using PSO and need more spool attributes for reports than are gathered by default.

Configure for LSR Support (Optional)

IBM's Local Shared Resources (LSR) facility allows the sharing of common control blocks such as I/O control blocks, buffers, and channel programs. Using this facility increases performance when queuing and processing transfer requests. To use the IBM LSR facility, the BLSR region has to be started. In the JCL used to start the XCOMXFER STC, make the following JCL change.

Note: LSR is recommended only for installations with very high volumes of scheduled transfers that are being run concurrently. EXECUTE transfers do not use XCOMRRDS, thus installations running a high volume of TYPE=EXECUTE transfers do not benefit from using LSR.

To configure for LSR support

Change this line:

```
//XCOMRRDS DD  DSN=XCOM.RRDS,DISP=SHR
```

to:

```
//DSVXRRDS DD  DISP=SHR,DSN=XCOM.RRDS
//XCOMRRDS DD  SUBSYS=(BLSR,'DDNAME=DSVXRRDS',
//            'BUFND=302',
//            'STRNO=151',
//            'RMODEB0=ALL',
//            'DEFERW=NO')
```

The recommended settings for the XCOMRRDS parameters are as follows:

BUFND

2 * (MAXTASK +1)

For example, if MAXTASK=150 then these parameters need to be set to 2 * (150 + 1) which is 302.

STRNO

MAXTASK + 1

For example, if MAXTASK=150 then this parameter should be set to 151.

RM0DEB0

ALL indicates that buffers above the 16 MB line are being used.

DEFERW

Indicates whether VSAM deferred write (DFR) is to be used.

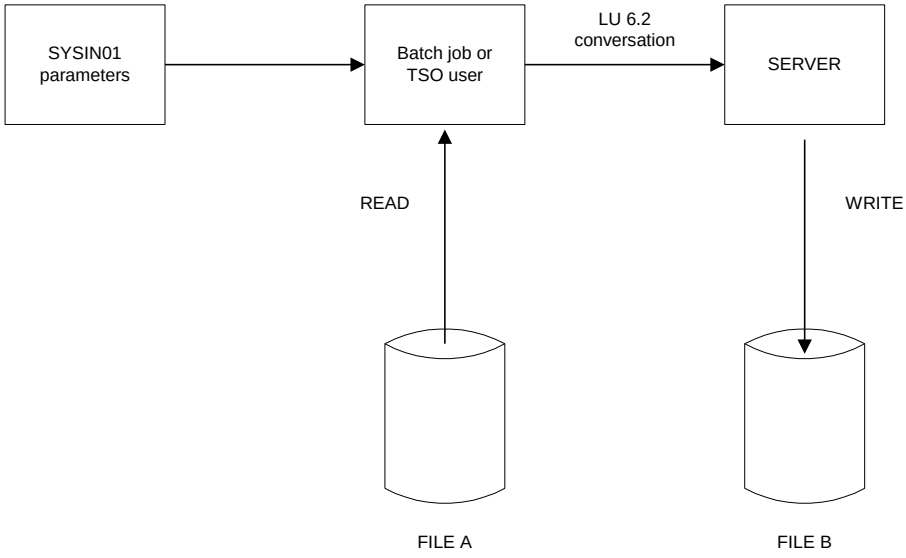
Note: Setting this parameter to NO causes data to be written to disk, ensuring that transfer information is not lost if an abnormal termination occurs. However, setting this parameter to NO decreases performance, because data is written to disk instead of to memory.

Setting DEFERW to YES improves performance, but if an abnormal termination occurs with this parameter set to YES, data that is written to memory is lost because it will not have been written out to the XCOMRRDS data set.

For more detailed information about the IBM LSR facility, see the online IBM documentation.

Verify the Installation

The following figure illustrates a file transfer under CA XCOM Data Transport. The CA XCOM Data Transport nodes involved in a file transfer ordinarily reside on separate platforms. But you can verify that you have installed CA XCOM Data Transport correctly by transferring a file between two CA XCOM Data Transports executing on the same platform.



TYPE=EXECUTE Transfer

Activate the Server (XCOMXFER)

The CA XCOM Data Transport region on the right in the above figure is the file transfer server, which is where the program XCOMXFER executes.

Activate the Application Major Node

The application major node that defines the ACB for the server must be active before XCOMXFER can execute.

Member APPLXCOM of CAI.CBXGSAMP contains VTAM definitions.

To activate the application major node

Copy the sample to one of the libraries in VTAM's VTAMLST concatenation. Issue the following VTAM command at the z/OS console:

```
V NET,ACT,ID=APPLXCOM
```

Note: The major node activation is required, *even when using only TCP/IP transfers*, for communication with the CICS user interface.

Display ACBs

After activating the application major node, you can display the ACBs by issuing the following command:

```
D NET,ID=APPLXCOM,E
```

In response, VTAM displays the ACBs in the application major node, as shown in the following table. Note that the status of the ACBs should be CONCT.

```
IST097I DISPLAY ACCEPTED
IST075I NAME = APPLXCOM, TYPE = APPL SEGMENT 027
IST486I STATUS= ACTIV, DESIRED STATE= ACTIV
IST360I APPLICATIONS:
IST080I XCOMAPPL CONCT   XCOMM00  CONCT   XCOMM01  CONCT
IST080I XCOMM02  CONCT   XCOMM03  CONCT   XCOMM04  CONCT
IST080I XCOMM05  CONCT
IST314I END
```

Specify the START Parameter

The START parameter can be specified through the EXEC PARM options of the CA XCOM Data Transport server JCL. In the following sample EXEC statement, the START parameter is set to the value STARTUP. STARTUP is a member of the XCOMCNTL data set that is initialized at system startup.

```
//SERVER  EXEC  PGM=XCOMXFER,                          X
//              TIME=1440,                             X
//              REGION=128M,                           X
//              PARM='ACBNAME=XCOMAPPL,START=STARTUP'
```

Note: If no destinations are to be enabled during initialization of the CA XCOM Data Transport server, the member STARTUP can contain a comment statement only. If the member STARTUP is not found in the XCOMCNTL data set, you will see the message XCOMM0265E for the member, and no TYPE=SCHEDULE transfers will be activated.

Start Your Product

When you have activated the XCOMXFER server and specified the START parameter, you can start CA XCOM Data Transport.

To start CA XCOM Data Transport

Enter one of the following commands on the z/OS console (or any TSO/ISPF or NETVIEW session) where the user is authorized to invoke z/OS system commands:

- START XCOM
- S XCOM

Several informational messages are displayed when CA XCOM Data Transport is started, as shown in the following example:

```
XCOMM0672I CA XCOM(TM) DATA TRANSPORT (R) RELEASE r11.6 - GENERATION LEVEL 0410
SPO0
XCOMM0004I    START=STARTB0
XCOMM0004I    DFTLAB=XCLSDFB0
XCOMM0004I    ACBNAME=XCOMLS
XCOMM0008I XCOMLS ACB OPENED SUCCESSFULLY
XCOMM0009I COMPILED UNDER VTAM V6 R1 M2 - EXECUTING UNDER VTAM V6 R1 M2 5695-117
XCOMM0763I DEFAULT TABLE XCLSDFB0 LOADED. GENERATED 11/28/04 AT 09.07
XCOMM0027I ESTAE ROUTINE HAS BEEN ESTABLISHED
XCOMM0037I PSO SUBTASK ATTACHED SUCCESSFULLY
XCOMM0056I CA XCOM(TM) RELEASE r11.6 (GEN LEVEL 1203 SPO0) IS UP AND ACCEPTING LOGONS
XCOMM0330I PRIMARY SUBSYSTEM IS JES2
XCOMM0803I STARTING XCOM TCP/IP LISTENER
XCOMM0788I XCOM TCP/IP LISTENER IS ACTIVE ON PORT 8044, IP ADDRESS=
```

As it initializes, CA XCOM Data Transport issues one XCOMM0559I message for each member that is enabled, as shown below:

```
XCOMM0559I  AS400   ENABLED SUCCESSFULLY
XCOMM0559I  MVSPAR  ENABLED SUCCESSFULLY
```

Start the XCOMPLEX Admin Server (Deprecated)

Important! The existing XCOMPLEX facility has been deprecated. The following information is provided for backward compatibility only. All new installations should use the new PLEXQ implementation. Existing XCOMPLEX users should migrate to the PLEXQ infrastructure for their XCOMPLEX functionality.

Typically, the XCOMPLEX Admin Server is brought up first, then the XCOMPLEX Worker Servers, although this is not required.

To start an XCOMPLEX Admin Server

Enter one of the following commands on the z/OS console (or any TSO/ISPF or NETVIEW session) where the user is authorized to invoke z/OS system commands:

- START XCOMAD
- S XCOMAD

Many of the messages that are displayed when you start the XCOMPLEX Admin Servers are the same as when you start a standalone server (see below), but the following are some of the additional messages you will see. Note the XCOMM0045I message, which indicates that an XCOMPLEX Worker Server has successfully connected to the Admin Server.

```
XCOMM0990I ACTUAL XCOMDFLT PARMS:
.
.
.
XCOMM0991I XCOMPLEX         = XCOMPLEX_BAS
XCOMM0008I XBAS31A  ACB OPENED SUCCESSFULLY
XCOMM0009I COMPILED UNDER VTAM V6 R1 M9 - EXECUTING UNDER VTAM V6 R1 M9
           5695-11701-190
XCOMM0763I DEFAULT TABLE BASAD115 LOADED. GENERATED 02/12/08 AT 23.43
XCOMM0027I ESTAE ROUTINE HAS BEEN ESTABLISHED
XCOMM0056I              CA XCOM(TM) r11.6 (GEN LEVEL 1203 SP00) IS UP AND ACCEPTING
           LOGONS
XCOMM0056I ON CPU 2097 SERIAL # 0CE000 IN 31-BIT MODE MVS SP7.0.9
XCOMM1000I STARTING CROSS-SYSTEM COUPLING FACILITY SERVICE TASK
XCOMM0066I *        TRACE    REQUEST COMPLETED SUCCESSFULLY
XCOMM0806I STARTING XCOM TCP/IP SSL LISTENER
XCOMM0821I STARTING XCOM TCP/IPV6 SSL LISTENER
XCOMM1068I CROSS-COUPLING FACILITY LISTENER TASK STARTING
XCOMM0803I STARTING XCOM TCP/IP LISTENER
XCOMM0820I STARTING XCOM TCP/IPV6 LISTENER
XCOMM0566E TYPE=PARAMETER NOT FIRST CARD IN XCOMCNTL MEMBER CICSJSA
XCOMM0559I XCOMRTL  ENABLED  SUCCESSFULLY
XCOMM0559I XCOMMVS2 ENABLED  SUCCESSFULLY
XCOMM0559I XCOMNG0  ENABLED  SUCCESSFULLY
XCOMM0807I XCOM TCP/IP SSL LISTENER ACTIVE ON PORT 46656, STACK ***ALL**
```

```
XCOMM0819I TCP/IP FUNCTION GETHOSTID RETURNS ADDRESS 141.202.65.31
XCOMM0788I XCOM TCP/IP LISTENER ACTIVE ON PORT 46655, STACK ***ALL**
XCOMM0819I TCP/IP FUNCTION GETHOSTID RETURNS ADDRESS 141.202.65.31
XCOMM0823I XCOM TCP/IPV6 SSL LISTENER ACTIVE ON PORT 46658, STACK ***ALL**
XCOMM0819I TCP/IP FUNCTION GETHOSTID RETURNS ADDRESS ::ffff:141.202.65.31
XCOMM0822I XCOM TCP/IPV6 LISTENER ACTIVE ON PORT 46657, STACK ***ALL**
XCOMM0819I TCP/IP FUNCTION GETHOSTID RETURNS ADDRESS ::ffff:141.202.65.31
XCOMM1019I XCF CONNECT STR=XCOMPLEX_BAS    , CONN=XBAS31A , RC=00000000,
           RSN=00000000, DISP=NEW
XCOMM1020I MAXIMUM SERVERS SUPPORTED=00000061
XCOMM1054I MAXIMUM LIST ENTRIES SUPPORTED=00001392
XCOMM1021I ALLOCATING MASTER ENTRY IN LIST #0000
XCOMM1028I ADDING SERVER ENTRY FOR NODE (XBAS31A ) WHICH IS USING LIST #0001
XCOMM0454I OPERLIM EXCEEDS XCF MAX ENTRIES. OPERLIM REDUCED TO VALUE IN XCOMM1054I
           MESSAGE
XCOMM1005I XCOMPLEX SERVICES ENABLED FOR NODE (XBAS31A )
XCOMM0451I XBAS31W  NOW CONNECTED TO XCOMPLEX ADMIN XBAS31A
```

Start the XCOMPLEX Worker Server (Deprecated)

Important! The existing XCOMPLEX facility has been deprecated. The following information is provided for backward compatibility only. All new installations should use the new PLEXQ implementation. Existing XCOMPLEX users should migrate to the PLEXQ infrastructure for their XCOMPLEX functionality.

Normally the XCOMPLEX Admin Server is brought up first, then the XCOMPLEX Worker Servers, although this is not a requirement.

To start an XCOMPLEX Worker Server

Enter one of the following commands on the z/OS console (or any TSO/ISPF or NETVIEW session) where the user is authorized to invoke z/OS system commands:

- START XCOMA
- S XCOMA

Several informational messages are displayed when CA XCOM Data Transport is started, as shown in the following excerpt:

```
XCOMM0991I XCOMPLEX         = XCOMPLEX_BAS
XCOMM0008I XBAS31W  ACB OPENED SUCCESSFULLY
XCOMM0009I COMPILED UNDER VTAM V6 R1 M9 - EXECUTING UNDER VTAM V6 R1 M9
           5695-11701-190
XCOMM0763I DEFAULT TABLE BASWK115 LOADED. GENERATED 03/17/08 AT 21.51
XCOMM0027I ESTAE ROUTINE HAS BEEN ESTABLISHED
XCOMM0037I PSO SUBTASK ATTACHED SUCCESSFULLY
XCOMM0330I PRIMARY SUBSYSTEM IS JES2
XCOMM1000I STARTING CROSS-SYSTEM COUPLING FACILITY SERVICE TASK
XCOMM0056I               CA XCOM(TM) r11.6 (GEN LEVEL 1203 SP00) IS UP AND
           ACCEPTING LOGONS
XCOMM0056I ON CPU 2097 SERIAL # 0CE000 IN 31-BIT MODE MVS SP7.0.9
XCOMM1068I CROSS-COUPLING FACILITY LISTENER TASK STARTING
XCOMM0803I STARTING XCOM TCP/IP LISTENER
XCOMM0788I XCOM TCP/IP LISTENER ACTIVE ON PORT 06653, STACK ***ALL**
XCOMM0819I TCP/IP FUNCTION GETHOSTID RETURNS ADDRESS 141.202.65.31
XCOMM0559I XBAS31A  ENABLED  SUCCESSFULLY
XCOMM0559I XCOMMVS2 ENABLED  SUCCESSFULLY
XCOMM0559I BASLIST  ENABLED  SUCCESSFULLY
XCOMM1019I XCF CONNECT STR=XCOMPLEX_BAS    , CONN=XBAS31W , RC=00000000,
           RSN=00000000, DISP=OLD
XCOMM1020I MAXIMUM SERVERS SUPPORTED=00000061
XCOMM1054I MAXIMUM LIST ENTRIES SUPPORTED=00001392
XCOMM1017I XCF READING MASTER ENTRY FROM LIST #0000
XCOMM1013I XCF REQ=READ             RC=00000000 REASON=00000000 LIST=0000
           ENTRY=XCOMPLEX_BAS
XCOMM1022I LOCATING SERVER ENTRY IN LIST #0000
XCOMM1026I UPDATING MASTER ENTRY IN LIST #0000
```

```
XCOMM1028I ADDING SERVER ENTRY FOR NODE (XBAS31W ) WHICH IS USING LIST #0002
XCOMM0454I OPERLIM EXCEEDS XCF MAX ENTRIES. OPERLIM REDUCED TO VALUE IN
           XCOMM1054I MESSAGE
XCOMM1005I XCOMPLEX SERVICES ENABLED FOR NODE (XBAS31W )
```

Perform a Direct File Transfer (TYPE=EXECUTE)

The region on the left in the figure in the section Verify the Installation (TYPE=EXECUTE Transfer) represents a batch job or TSO address space that is executing a file transfer between itself and a CA XCOM Data Transport server. The program XCOMJOB is executing in the region. Notice that XCOMJOB is executing a direct (non-queued) transfer, also known as a TYPE=EXECUTE transfer (because of the specification PARM='TYPE=EXECUTE' in the JCL for XCOMJOB). In this type of transfer, the batch job or TSO address space *synchronously* transfers a file to or from a CA XCOM Data Transport server. The batch job or TSO address space enters a wait state until the transfer is complete.

The members XCOMJOBE and XCOMIVP1 in CAI.CBXGJCL contain sample jobs that initiate TYPE=EXECUTE transfers.

To perform a direct file transfer (TYPE=EXECUTE)

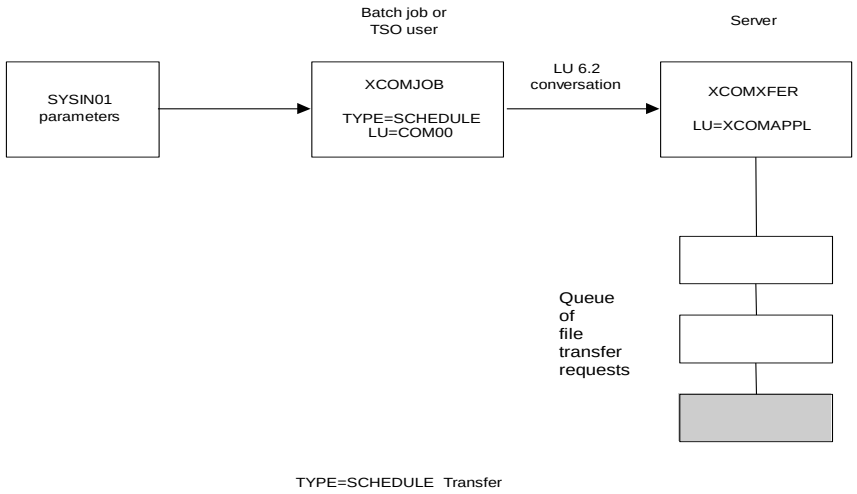
Add a JOB card and modify the XCOMIVP1 job (as indicated) before submitting it.

When the job completes, the XCOMLOG data set contains lines similar to the following if the transfer was successful:

```
XCOMM0008I XCOM00 ACB OPENED SUCCESSFULLY
XCOMM0009I COMPILED UNDER VTAM V6 R1 M2 - EXECUTING UNDER VTAM V6 R1 M2 5695-117
TYPE=SEND
LU=XCOMAPPL
FILETYPE=FILE
FILEOPT=CREATE
LFILE=FILE.A
FILE=FILE.B
XCOMM0155I XCOMAPPL - LOGON EXIT ENTERED
XCOMM0402I REQUEST NUMBER 002000 ASSIGNED TO TRANSFER REQUEST
XCOMM0137I 00000020 RECORDS SENT SUCCESSFULLY - FILE=FILE.A
XCOMM0151I XCOMAPPL SESSION ENDED
```

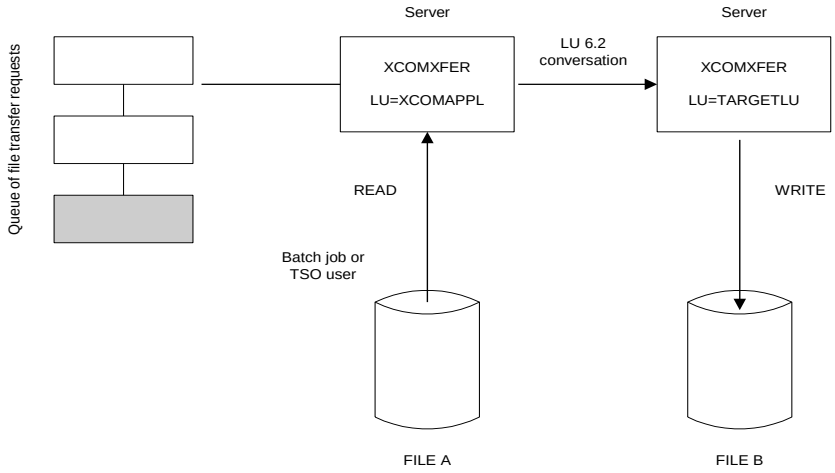
Perform a Scheduled Transfer (TYPE=SCHEDULE)

If you do not want a batch job or TSO address space to wait for a transfer to complete, you can execute XCOMJOB specifying PARM='TYPE=SCHEDULE.' In this case, the XCOMJOB region does not transfer the file itself. Rather, as illustrated in the following figure, it adds a request to a CA XCOM Data Transport server's queue of file transfer requests.



TYPE=SCHEDULE Transfer

As illustrated in the next figure (A Scheduled File Transfer), the CA XCOM Data Transport server selects requests from the queue and initiates an LU 6.2 conversation with another CA XCOM Data Transport server. The XCOMJOB that originally scheduled the request treats this as an asynchronous file transfer and terminates once the request has been scheduled.



A Schedule File Transfer

The previous figure illustrates a TYPE=SCHEDULE file transfer involving two CA XCOM Data Transport servers. Members XCOMJOBS and XCOMIVP2 in CAI.CBXGJCL contain jobs that initiate TYPE=SCHEDULE transfers. In the XCOMIVP2 member, SYSIN01 includes the STARTDATE=04237 parameter, which specifies the date when the CA XCOM Data Transport server will initiate the file transfer.

When Julian date 04237 arrives, the CA XCOM Data Transport server with LU XCOMAPPL selects this transfer request from its queue and initiates a file transfer to the CA XCOM Data Transport node that it identified as TARGETLU. If the STARTDATE parameter is omitted, the CA XCOM Data Transport server initiates the transfer as soon as it can.

After the job in XCOMIVP2 completes, you can display the file transfer request that has been added to the CA XCOM Data Transport server's queue by issuing this command at the z/OS console:

```
F XCOM,SHOW
```

XCOM

The address space name of the CA XCOM Data Transport server.

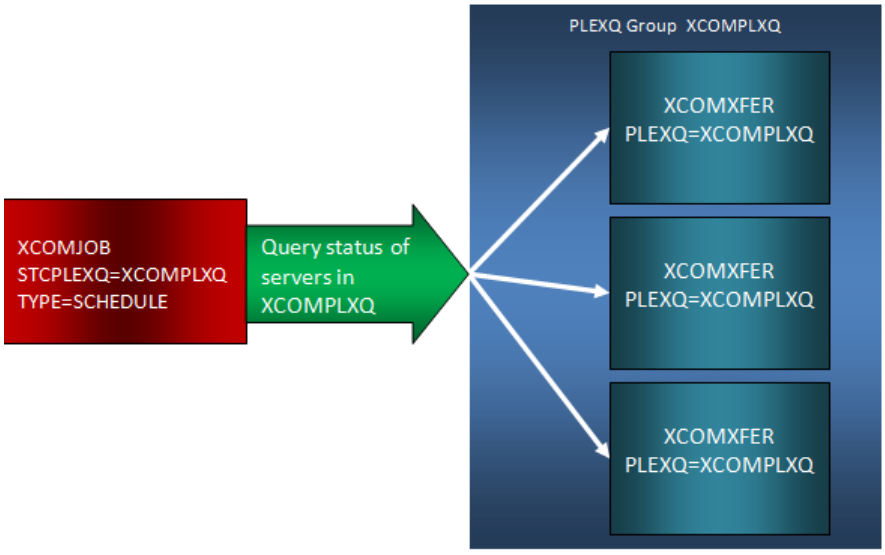
The CA XCOM Data Transport server responds to this command by issuing an XCOMM0389I message for each transfer request on its queue. In this case, CA XCOM Data Transport responds with the following lines:

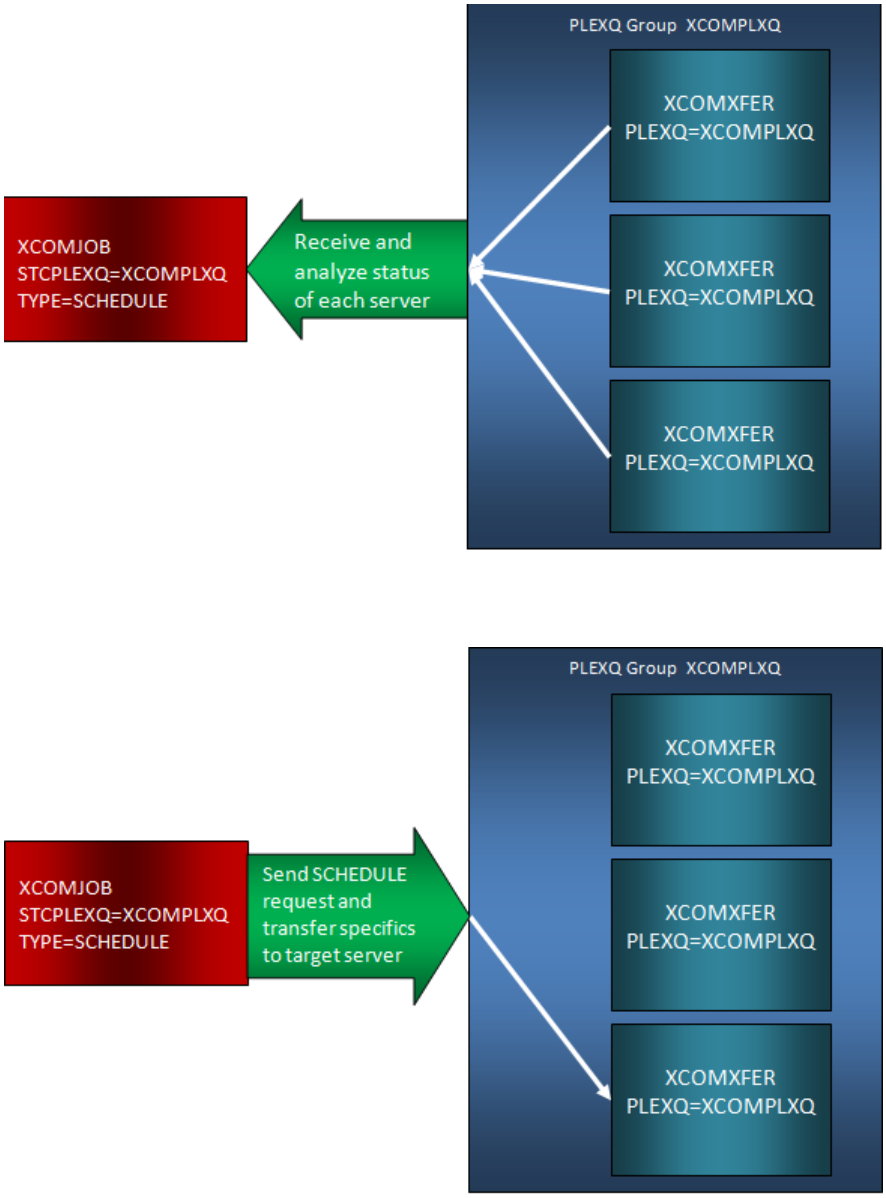
```
XCOMM0013I SHOW
XCOMM0389I REQ#=001162, STATUS=INACTIVE, NAME=TARGETLU, DATE=04237, TIME=0000,
PRI=10, AGE=010
```

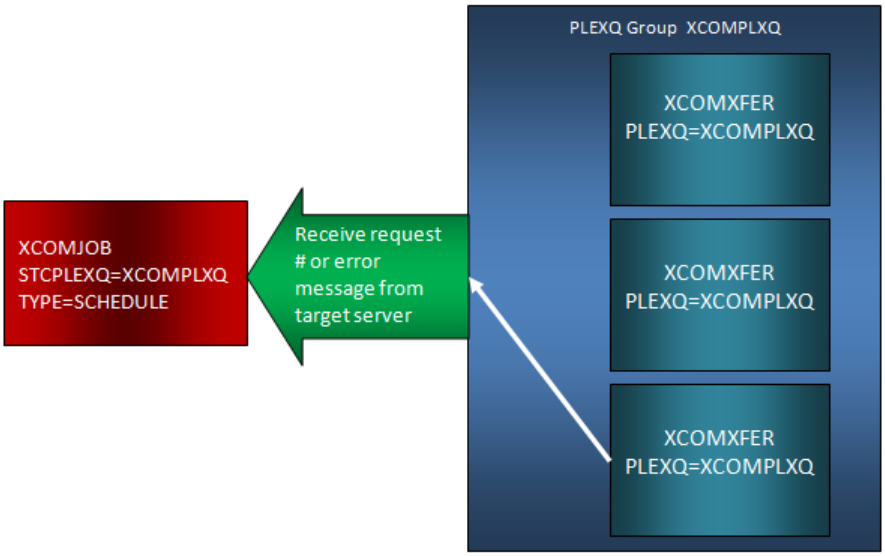
Perform a Scheduled Transfer in the PLEXQ (TYPE=SCHEDULE)

When scheduling transfers within the PLEXQ, you must use TYPE=SCHEDULE. In this case, the XCOMJOB batch utility connects to the PLEXQ group and initiates a query to all member servers within the PLEXQ group named in the STCPLEXQ EXEC PARM. The servers in the PLEXQ group each respond with its current status. The XCOMJOB utility programmatically selects the best candidate server to receive the schedule request, and proceeds to direct the scheduling activity to the appropriate server. Inquire activity (if any) is subsequently directed to the appropriate server by the XCOMJOB utility. All exchanges of data and information are routed through the IBM SYSPLEX Coupling Facility Signaling Services. Member XCOMJOQS in CAI.CBXGJCL contains a job that initiates a TYPE=SCHEDULE transfer using the PLEXQ environment.

The following diagrams represent the sequence of events involved in scheduling a transfer to a PLEXQ group:



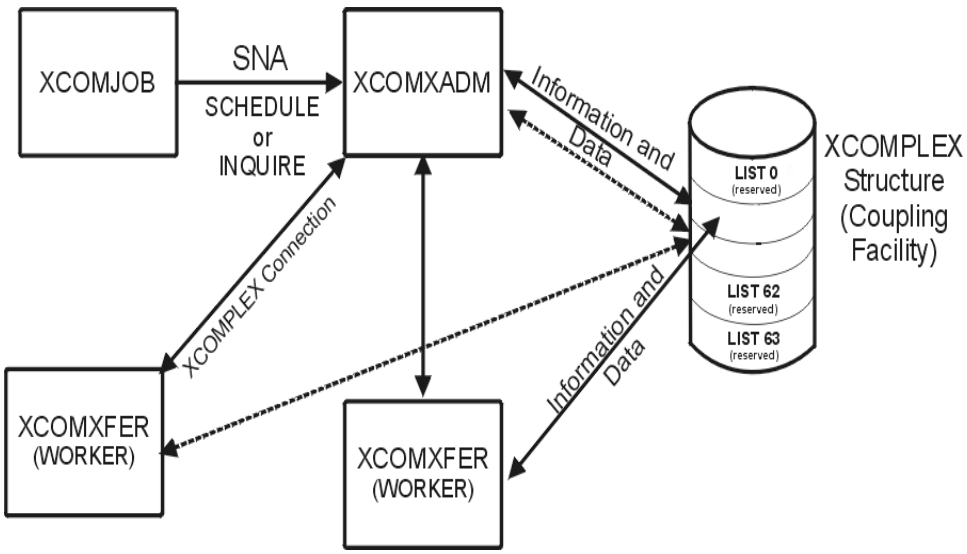

PLEXQ Group XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
XCOMJOB
STCPLEXQ=XCOMPLXQ
TYPE=SCHEDULE
Receive and
analyze status
of each server
PLEXQ Group XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
XCOMJOB
STCPLEXQ=XCOMPLXQ
TYPE=SCHEDULE
Send SCHEDULE
request and
transfer specifics
to target server


PLEXQ Group XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
XCOMXFER
PLEXQ=XCOMPLXQ
Receive request
or error
message from
target server
XCOMJOB
STCPLEXQ=XCOMPLXQ
TYPE=SCHEDULE

Perform a Scheduled Transfer in the XCOMPLEX (TYPE=SCHEDULE) (Deprecated)

Important! The existing XCOMPLEX facility has been deprecated. The following information is provided for backward compatibility only. All new installations should use the new PLEXQ implementation. Existing XCOMPLEX users should migrate to the PLEXQ infrastructure for their XCOMPLEX functionality.

When scheduling transfers within the XCOMPLEX, you must use TYPE=SCHEDULE and the transfers must all be scheduled to the XCOMPLEX Admin Server. The XCOMPLEX Admin Server then distributes the transfers to the XCOMPLEX Worker Servers. The XCOMPLEX Admin Server keeps track of the status of all the XCOMPLEX Worker Servers so it can route the transfer to the appropriate XCOMPLEX Worker Server. The XCOMPLEX Admin Server then takes care of routing any RESTARTS or INQUIRES to the proper XCOMPLEX Worker Server. All exchanges of data and information are routed through the IBM Coupling Facility. Member XCOMJOXS in CAI.CBXGJCL contains a job that initiates a TYPE=SCHEDULE transfer using the XCOMPLEX Admin Server.



Invoke Your Product Through the ISPF Interface

You can also perform TYPE=EXECUTE and TYPE=SCHEDULE file transfers from the CA XCOM Data Transport ISPF dialog.

To invoke the dialog

Enter *one* of the following commands from ISPF option 6:

- XCOM62
- CA$XCOM

If you have customized the ISPF Primary Option Menu or another ISPF panel, you can invoke the CA XCOM Data Transport dialog directly from that panel.

For more information about the CA XCOM Data Transport ISPF interface, see the chapter “The Menu Interface (TSO/ISPF Panels)” in the *CA XCOM Data Transport for z/OS User Guide*.

Use the CA XCOM Data Transport Health Checks to Tune the CA XCOM Data Transport Regions

The IBM Health Checker for z/OS allows you to identify potential problems in your z/OS environment by checking system or product parameters and system status against recommended settings. The IBM Health Checker for z/OS is structured as a framework that includes a health check started task and various separate check routines provided by IBM or other vendors. CA XCOM Data Transport provides health checks for virtual storage usage (above and below the line) as well as task-level monitoring. When exception conditions are found, the CA XCOM Data Transport health checks provide detailed recommendations on how to correct the problem. The CA XCOM Data Transport health checks also make best practice recommendations for using CA XCOM Data Transport.

These health checks are automatically activated when the CA XCOM Data Transport region is started if the following components are configured and running on your system:

- IBM Health Checker for z/OS—This free component is distributed with every supported level of z/OS. If you have not yet set up this component, see the IBM Health Checker for z/OS User Guide for details on how to set it up.
- CA Health Checker Common Service—a free CA Common Services component. The load library for this component must be authorized and installed in the link list on the target system where the CA XCOM Data Transport health checks will be run. A new address space is not required for the CA Health Checker Common Service. For the CA Common Services release levels that support this functionality, see http://support.ca.com.

CA XCOM Data Transport provides the following health checks:

- XCOM_ABOVE_16M@stcname
- XCOM_BELOW_16M@stcname
- XCOM_MAXLOC_LEVEL@stcname
- XCOM_MAXREM_LEVEL@stcname
- XCOM_MAXTASK_LEVEL@stcname

For more information about using the CA XCOM Data Transport health checks, see the appendix "CA XCOM Data Transport Health Checks." This information includes check parameters that can be configured to change default thresholds or settings used in the checks.

How to Use Your Product with Other Products

CA XCOM Data Transport includes interfaces to a number of popular software packages. An introduction and explanation of each is provided in this section.

Abend-AID

When using Abend-AID and an SVC dump is requested by CA XCOM Data Transport support, turn off Abend-AID dumps by adding the following statement in the CA XCOM Data Transport Server JCL.

```
//ABNLIGNR  DD  DUMMY
```

FDR/ABR

In the startup JCL for the CA XCOM Data Transport server, add the following statement. This causes FDR/ABR to respond to CA XCOM Data Transport and to let it proceed with other transfers while waiting for FDR managed data sets to be recalled from archive.

```
//ABRSYNCH  DD  DUMMY
```

CICS Notification Facility

Important! The existing CICS panels have been deprecated. The following information is provided for backward compatibility only. All new installations use the new ISPF implementation. Existing CICS users migrate to the ISPF panels.

CA XCOM Data Transport provides a facility that can be used to direct notification messages to a predefined CICS transaction through an LU 6.2 conversation.

To use the CICS notification facility.

1. Define the CICS transaction in the CICS Program Control Table (PCT).
2. Define CA XCOM Data Transport as a remote system in the CICS Terminal Control Table (TCT).

CICS TCT Entry

The CICS TCT must have an entry for the CA XCOM Data Transport APPLID. A sample for this entry is provided in CAI.CBXGJCL(XCOMCSD). This entry must do all of the following:

- Indicate that the LU has a protocol of APPC
- Have a NETNAME set equal to the APPLID
- Have an access method set to VTAM
- Set single session to “1,0”
- Set idle connect to NO
- Set in service to YES
- Set the mode name to XCOMMODE or equivalent

Control Library Member

Also, a CA XCOM Data Transport control library (CAI.CBXGPARM) member must be in place on the CA XCOM Data Transport system invoking the CICS transaction.

About the CICS Notification Transaction

CA XCOM Data Transport starts this user-written CICS transaction at the conclusion of a file transfer. This is important to those who want to signal their online CICS region that a file transfer has completed so that transactions that need that data can be invoked.

The ID of the transaction to be started is defined by the NTFYTPN parameter in the CA XCOM Data Transport Default Options Table. If you want to use this facility, you need to define the transaction to your CICS region in advance, either by adding an entry to your Program Control Table or by using the CEDA transaction. The following paragraphs include information on how an end user can request CICS notification.

When initiating the file transfer through JCL, you should use the NOTIFY and NOTIFYNAME parameters and/or-if CICS notification is to occur on the remote system-the RNOTIFY and RNOTIFYNAME parameters to identify the CICS system on which the transaction should be invoked (see XCOMJOB DD Statement in the chapter "The Batch Interface" in the *CA XCOM Data Transport for z/OS User Guide*). For example, if your CICS APPLID is CICSP, set NOTIFY to CICS and NOTIFYNAME to CICSP.

Code the CICS Notification Transaction

The transaction that processes CA XCOM Data Transport notifications must conform to certain opening and closing logic rules. For a sample CICS transaction, see CAI.CBXGSAMP(XCOMSAMP).

To code the CICS notification transaction.

1. Perform a CICS RECEIVE to obtain the message that the CA XCOM Data Transport server sent. This message is a copy of the CA XCOM Data Transport history record describing the transfer that the HSTDSECT macro mapped. A 16-byte header precedes the record and can be ignored. Make sure that your work area is large enough to accommodate the history record and the header. Check sample XCOMSAMP for recommendations. A command example:

   ```
   EXEC CICS RECEIVE
   ```

2. Check the confirm indicator in the CICS execute interface block. If it is set to X'FF', then issue a CICS CONFIRM. A commands example:

   ```
   IF DFHCONF = HIGH-VALUES THEN

   EXEC CICS SEND CONFIRM
   ```

3. Deallocate the conversation with CA XCOM Data Transport:

   ```
   EXEC CICS FREE
   ```

4. Do whatever application processing you wish.

5. End the program:

   ```
   EXEC CICS RETURN
   ```

DCB ABEND Exit Software

CA XCOM Data Transport does not support, nor is it compatible with any product that modifies the DCB abend exit as this can cause unpredictable results. In many cases, this software appears to function successfully with XCOM until there is a problem. At that point, abends and errors can appear since XCOM is not able to regain control from the DCB abend exit software to handle the problem.

Security Interfaces

For information about the security interfaces IBM RACF, CA ACF2, and CA Top Secret, see the chapter “Overview of Security” in the *CA XCOM Data Transport for z/OS User Guide.*

Scheduling Packages

Scheduling packages help insure that the proper activities are performed at the proper time and in the proper sequence for a given set of tasks. For example, the job that prints the payroll checks cannot run until the job that calculates the amounts is completed. The calculation job cannot run until all the departments have reported their employees' hours for the week.

These scheduling systems use triggers to indicate the successful completion of one task, thereby starting the next. The more traditional triggers, with examples, are as follows:

Another Batch Job

Job X is complete; now run Job Y and Job Z.

Time/Date

Run job BUDGET at noon on the 30th of every month.

Command

The data entry department finishes their order entries for the day, and tells the system through a command that triggers the inventory, shipping, and billing jobs.

File Creation

Department A creates a file that holds their employees' hours for the week.

The user does not need an interface between CA XCOM Data Transport and the scheduling system to schedule file transfers. CA XCOM Data Transport can run as a batch job just like any other job in the system, and thus can be scheduled like any other job in the system. If an outgoing transfer is part of the required job stream, then a CA XCOM Data Transport batch job specifying SEND can be included in the schedule. Other jobs can be triggered following the successful (or even unsuccessful) completion of the CA XCOM Data Transport job. Likewise, inbound transfers can be scheduled by running a CA XCOM Data Transport batch job specifying RECEIVE.

If CA XCOM Data Transport is going to be used as a trigger for further processing, it is important to understand the difference between queued and non-queued transfers. A batch job requesting a file transfer to be queued has completed successfully when that transfer has been scheduled. Since the job has reached its completion once the transfer request is in the queue (TYPE=SCHEDULE), the job itself is not aware of whether the file transfer has actually taken place; it cannot relay information about the actual file transfer.

TYPE=INQUIRE jobs may be used to provide a status for TYPE=SCHEDULE jobs. Non-queued transfers, TYPE=EXECUTE, provide information about the actual job transfer and a return code based on the success or failure of the file transfer. For an explanation of the TYPE=SCHEDULE, TYPE=INQUIRE, and TYPE=EXECUTE, see the *CA XCOM Data Transport for z/OS* User *Guide*.

CA 7 Interface

CA XCOM Data Transport contains an interface to CA's popular mainframe scheduling package, CA 7 Workload Automation Smart Console Option. The interface is used only with incoming file transfers, and is enabled by specifying CA7EXIT=YES in the CA XCOM Data Transport Default Options Table. When the interface is enabled and the transfer is of type TYPE=SCHEDULE, CA XCOM Data Transport, using the CA 7 facility U7SVC, reports to CA 7 that a file has been updated successfully. The user has the option of using this file creation as a CA 7 trigger for other jobs.

Note: Transfers of type TYPE=EXECUTE cannot use the U7SVC facility even when CA7EXIT=YES is specified in the Default Options Table.

Using the previous payroll example, consider the following scenario:

A series of payroll jobs must run to print salary checks and calculate tax payments. These jobs cannot run until all the departments have reported their employees' time for the week.

This information is kept on various midrange and smaller systems located in each department. It is agreed that the departments will transfer the records as soon as they are accumulated and will give them a file name unique to each department. If the CA XCOM Data Transport CA 7 interface is enabled, CA XCOM Data Transport informs CA 7 each time a file is created, allowing CA 7 to use all of those files as triggers for the payroll batch stream. In this manner, the payroll stream does not start until all necessary files are in place.

Important! Using this interface, you might get a CA7DATA DD statement error and an IEC1301 error message unless you *specify DCB=(BLKSIZE=80,RECFM=F,L*RECL=80) on the CA7DATA DD card.

Other Scheduling Packages

You can implement interfaces to other scheduling packages by using one of the following:

- User Exit 1 (see the appendix “User Exits” in the *CA XCOM Data Transport for z/OS User Guide*)
- SMF records

CA NetMaster FTM

The Default Options Table includes two parameters to allow CA NetMaster support. They are NETMAST and RECVRID (see the chapter “Configuration Parameters”). For more information about using these parameters with CA XCOM Data Transport, see the appropriate NetMaster guides or contact their support staff.

Server Failover Recovery

The recovery process allows for the rerouting of locally initiated transfers from an inactive CA XCOM Data Transport for z/OS server to an active one. This rerouting of transfers is known as failover processing, because it provides a recovery process if a server terminates. However, the active and pending transfers from any CA XCOM Data Transport RRDS data set can only be rerouted, as long as the RRDS data set is not in use by another process (that is, a CA XCOM Data Transport server or any batch utility).

The following transfers are not eligible for failover processing:

- Remotely initiated send and receive file transfers; these transfers remain in the inactive CA XCOM Data Transport for z/OS server's RRDS data set.
- Remotely initiated send job and send report transfers; these transfers are deleted from the inactive CA XCOM Data Transport for z/OS server's RRDS data set.

How to Perform a Server Recovery

To perform failover processing

Execute the CA XCOM Data Transport batch utility XCOMJOB with the execution parameter TYPE=RECOVER specified.

The following is sample JCL to perform failover processing.

Sample JCL (TYPE=RECOVER)

The member CAI.CBXGJCL (XCOMJOBR) is a sample JCL for performing a server recovery.

Notes:

- The CA XCOM Data Transport queue (XCOM RRDS data set) cannot be opened by any CA XCOM Data Transport server or other application while the failover processing is active.
- The failover process (TYPE=RECOVER) does not use any SYSIN01 control statements. Any SYSIN01 control statements are ignored for a TYPE=RECOVER XCOMJOB execution.
- The CA XCOM Data Transport queue (XCOM RRDS data set) that is used for input by failover processing is referenced by the XCOMRRDS DD JCL statement within the batch job that invokes XCOMJOB.

Parameters

The following XCOMJOB execution parameters are valid for failover processing.

TYPE=RECOVER

Required.

Directs XCOMJOB to initiate failover processing.

Default: None

STCAPPL=acbname

Optional.

Specifies the server to which the transfers are to be rerouted and that an SNA connection is to be used for communication.

STCAPPL is mutually exclusive with STCIP and STCPORT.

Default: ACBNAME value from the CA XCOM CONFIG Member.

STCIP=ip.namelip.addr

Optional.

Specifies the IP address or name of the server to which the transfers are to be rerouted and that a TCP/IP connection is to be used for communication with the target CA XCOM Data Transport server.

STCIP is mutually exclusive with STCAPPL.

Default: None

STCPORT=port#

Optional.

Specifies the TCP/IP port on which the target CA XCOM Data Transport server is listening.

STCPORT is mutually exclusive with STCAPPL.

Default: None

DFLTAB=dfltname

Optional.

Specifies the name of the Default Options Table to be converted for use as a TYPE=CONFIG member and be used as the source for configurable options for this invocation of XCOMJOB. If a CONFIG Member already exists with this same, it will be used instead, and no conversion will take place. If both the DFLTAB and CONFIG parameters are specified, the CONFIG parameter takes precedence.

Default: XCOMDFLT

CONFIG=cnfgname

Optional.

Specifies the name of the CONFIG Member to be used as the source for configurable options for this invocation of XCOMJOB. If both the DFLTAB and CONFIG parameters are specified, the CONFIG parameter takes precedence.

Default: XCOMCNFG

SECURE_SCHEDULE=YES|NO

Optional.

Specifies whether the Secure Sockets Layer (SSL) is to be used to communicate with the target CA XCOM Data Transport server.

Default: NO

TRACE=0-9|YES|NO

Optional.

Specifies whether CA XCOM Data Transport tracing is to be performed on the network communication between the XCOMJOB utility and the target CA XCOM Data Transport server.

level

A value from 0 thru 9 which enables and specifies the level of the CA XCOM Data Transport trace function instead of the VTAM trace. Level 0 provides the minimal trace data while level 9 provides maximum trace data. Levels in between build incrementally on level 0.

Default: NO

CONFIG Member

Parameters for transfers scheduled to the XCOMPLEX Admin Server can be taken from the CONFIG Member for the Admin Server or from another specified CONFIG Member. Parameters specified in the SYSIN01 and destination members will override these parameters.

Sample Rules

Two sample rules have been provided for use with CA OPS/MVS which, when customized for the local installation, will automatically invoke failover processing if a monitored CA XCOM Data Transport server terminates abnormally. Comments within each sample member offer guidance for correctly customizing them to meet local requirements.

These sample members are:

XCOPSMSG

This CA OPS/MVS rule detects CA XCOM Data Transport messages XCOMM0198I and XCOMM0199I and sets a CA OPS/MVS variable to indicate whether the CA XCOM Data Transport server address space terminated normally or abnormally.

XCOPSEOJ

This CA OPS/MVS rule detects the end-of-job condition for the specified CA XCOM Data Transport server. If the CA XCOM Data Transport server terminated abnormally, the predefined failover JCL is submitted to the local system for processing.

Chapter 2: Configuration Parameters

This chapter describes the various categories of parameters in terms of which CA XCOM Data Transport for z/OS is configured. Within each category, the parameters are described in alphabetical order.

This section contains the following topics:

CONFIG Member Parameters

This section describes the parameters for the CA XCOM Data Transport CONFIG Member.

ACBNAME

Specifies the VTAM ACBNAME that the server attempts to open at initialization time.

XCOMAPPL

Specifies XCOMAPPL as the ACBNAME.

xxxxxxxx

Specifies an ACBNAME other than XCOMAPPL. This name can be up to eight alphanumeric characters.

Default: XCOMAPPL

Note: The value of the ACBNAME parameter should be the same as the VTAM APPL statement name.

ACFUSER

Designates the name of the CA ACF2 control block from which CA XCOM Data Transport will obtain a user ID.

ACEE

Specifies that CA XCOM Data Transport is to obtain the user ID from the ACEE.

ACFUID

Specifies that CA XCOM Data Transport is to use the CA ACF2 UID string from the ACUCB for authorization checking.

ASXB

Specifies that CA XCOM Data Transport is to obtain the user ID from the ASXB.

Default: ASXB

Note: This parameter concerns only CA ACF2 users. Consult the CA ACF2 administrator for more information about this parameter.

AGE

Specifies the length in days of the queue purging interval applied to locally initiated transfer requests.

1 to 999

Specifies the length in days of the queue purging interval.

Default: 10 (days)

Notes:

- This feature prevents scheduled requests that are incomplete from remaining indefinitely on the pending queue. After the purging interval, these requests will be deleted and a history record will be generated reflecting this change. This relieves the CA XCOM Data Transport administrator from periodically checking the CA XCOM Data Transport queue to delete old requests, thus allowing unsupervised operation of CA XCOM Data Transport.
- The REMAGE parameter serves the same purpose for remotely initiated transfer requests.

ALERT_CONV

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

The value of the ALERT_CONV parameter is the severity level of the event named by the first term.

Specifies that CA XCOM Data Transport generates alerts for conversation related events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_CONV is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_CONV parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_CONV parameter according to its specification in the CONFIG Member.

ALERT_FILE

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

Specifies that CA XCOM Data Transport generates alerts for file related events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_FILE is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_FILE parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_FILE parameter according to its specification in the CONFIG Member.

ALERT_GEN

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

Specifies that CA XCOM Data Transport generates alerts for general events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_GEN is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_GEN parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_GEN parameter according to its specification in the CONFIG Member.

ALERT_SEC

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

Specifies that CA XCOM Data Transport generates alerts for security related events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_SEC is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_SEC parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_SEC parameter according to its specification in the CONFIG Member.

ALERT_SESS

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

Specifies that CA XCOM Data Transport generates alerts for session related events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_SESS is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_SESS parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_SESS parameter according to its specification in the CONFIG Member.

ALLOC

Specifies the default type of space allocation that is to be used for file transfers to a new data set.

BLKS

Specifies that space is allocated in blocks.

CYLS

Specifies that space is allocated in cylinders.

TRKS

Specifies that space is allocated in tracks.

RECORDS

Specifies that space is allocated in records.

Default: CYLS

Note: Related parameters are CATALOG, PRI, SEC, UNIT, VOL, DIR, PSOUNIT, and PSOVOL.

APPLSEC

Specifies the value to be used as the APPL value when requesting validation of the user ID and password by the security interface.

ACBNAME

Causes CA XCOM Data Transport to use the value entered with the ACBNAME keyword in this table or in the PARM field of the EXEC JCL statement that starts CA XCOM Data Transport.

APPLID

Causes CA XCOM Data Transport to use the value entered with the APPLID keyword in this table or in the PARM field of the EXEC JCL statement that starts CA XCOM Data Transport.

DEFAULT

Causes CA XCOM Data Transport to use a value of XCOM62 (this provides compatibility with earlier releases of the CA XCOM Data Transport product).

NETNAME

Causes CA XCOM Data Transport to use the value entered with the NETNAME keyword in this table or in the PARM field of the EXEC JCL statement that starts CA XCOM Data Transport.

NONE

Causes CA XCOM Data Transport not to request APPLID security.

STCAPPL

Causes CA XCOM Data Transport to use the value entered in the PARM field of the EXEC JCL statement that starts CA XCOM Data Transport.

'*xxxxxxxx*'

Causes CA XCOM Data Transport to use the value entered within the single quotes as the APPL value. The value is a string of up to eight alphanumeric characters.

Default: DEFAULT

Notes:

- CA XCOM Data Transport's sample IBM RACF and CA Top Secret interfaces (XCOMRACF and XCOMTOPS, respectively) use the APPLSEC parameter as the value passed in the APPL parameter when making RACROUTE VERIFY requests. Sites may use the value of this parameter for other purposes by recoding the sample security interface.
- CA XCOM Data Transport passes this value to Exit 5 (XCOMEX05) in the field SECAPPL of the Exit 5 parameter list (mapped by the SECDSECT macro).

AVGREC

Specifies the multiplier for Primary and Secondary allocation units when allocating based on number of records.

U

Indicates that the PRI and SEC parameters specify the number of records to allocate for.

K

Indicates that PRI and SEC parameters specify the number of records in thousands (so it would be the number specified multiplied by 1024).

For example, specifying 3 would be stating 3K or 3072 records.

M

Indicates that PRI and SEC parameters specify the number of records in millions (so it would be the number specified multiplied by 1048576).

For example, specifying 2 would be stating 2M or 2097152 records.

Default: U

Note: This parameter applies only when the alloc value in the SPACE parameter specifies a value of REC indicating that a file is being allocated based on a specific number of records.

BANNER

Specifies whether CA XCOM Data Transport is to create a banner page when transferring a report.

YES

Specifies that CA XCOM Data Transport is to create a banner page.

NO

Specifies that CA XCOM Data Transport is not to create a banner page.

Default: YES

Note: If EXIT10=YES has been specified in the Default Options Table, Exit 10 (module XCOMEX10) is invoked to create the banner page on behalf of CA XCOM Data Transport; otherwise, CA XCOM Data Transport creates the default banner page.

CA7EXIT

Specifies whether CA XCOM Data Transport will attach a CA 7 interface subtask at the end of a successful file transfer.

YES

Specifies that CA XCOM Data Transport is to attach a CA 7 interface subtask.

NO

Specifies that CA XCOM Data Transport is not to attach a CA 7 interface subtask.

Default: NO

Notes:

- CA7EXIT is useful for installations that use the CA 7 scheduling system and need to use a successful inbound file transfer as a trigger for subsequent job scheduling.
- This parameter is effective only with transfers of type TYPE=SCHEDULE (that is, queued transfers).
- CA XCOM Data Transport uses the CA 7 U7SVC program for data set triggering. For more information about CA 7 setup, see the *CA 7 Interface Guide* and the *CA 7 Data Base Maintenance Guide*.

CATALOG

Indicates whether new data sets created by CA XCOM Data Transport should be cataloged.

YES

Specifies that the data sets are to be cataloged.

NO

Specifies that the data sets are not to be cataloged.

Default: YES

Notes:

- CATALOG is similar to the DISP parameter of the JCL DD statement.
- Related parameters are ALLOC, SEC, DIR, PRI, UNIT, VOL, PSOUNIT, and PSOVOL.

CKPT

Specifies the interval for checkpoints taken during a file transfer.

0 to 9999 records

Specifies the number of records to transfer for a checkpoint interval.

Notes:

- The checkpoint/restart facility resumes interrupted transfers from the point at which the most recent checkpoint was taken.
- Each time a checkpoint is taken, the output buffers on the receiving system are written to the disk. A checkpoint interval that is too short will slow down file transfers; a checkpoint interval that is too long increases the risk of data loss. We recommend that you set the Checkpoint Count to at least 1000. On Token Ring, Ethernet, and other high-speed networks, the Checkpoint Count should be set to the highest allowable value, if needed. Set this parameter to 0 to disable checkpointing.
- If the receiving system is z/OS or VSE, the Checkpoint Count should be set to a multiple of the blocking factor. For example, if the DCB attributes are RECFM=FB LRECL=80 BLKSIZE=8000, the Checkpoint Count should be a multiple of 100.

Default: 1000

CLASS

Specifies the default SYSOUT class assigned to an incoming report when the remote system does not provide one. This can be any valid JES SYSOUT class.

A

Specifies the SYSOUT class A.

x

Specifies a SYSOUT class other than A. This can be designated with any alphanumeric character.

Default: A

CMPRS_PDS_ALLOW

Specifies whether PDS compression is allowed.

YES

Allows users to request PDS compression on a transfer-by-transfer basis, or by using a DEST member.

Note: This setting also implicitly provides automatic compression in response to an out-of-space condition.

NO

Disables the compression function within the server(s) to which it applies.

X37

Automatically invokes PDS compression if a z/OS system abend B37, D37, or E37 occurs during a transfer into a PDS data set.

Default: NO

Note: Any retryable transfer that is a result of an automatic compression from an out-of-space conditions (such as #XCOMM0221E DATASET OUT OF SPACE - PDS DATASET WILL BE COMPRESSED) is terminated if the retry also results in an out-of-space condition

CMPRS_SYSOUT_CL

Identifies the name of a PDS compression SYSOUT class.

Default: Use the XCOM class.

COMPNEG

Specifies whether compression negotiation is performed.

YES

Specifies that the data compression method is negotiated.

NO

Specifies that the data compression method is not negotiated.

Default: YES

Notes:

- If COMPNEG=YES and the compression method suggested by the initiator or partner LU is known to both participants, that method is used to compress the transfer data.
- If the suggested compression method is not known to a participant involved in the transfer, Run Length Encoding of blanks and zeros is applied to the data to be transferred.
- If an unknown compression type is requested, the file transfer is rejected.

CONTIG

Use contiguous space allocation.

YES

Specifies that contiguous space allocation is used.

NO

Specifies that contiguous space allocation is not used.

Default: NO

CREATEDELETE

Specifies whether the CREATEDELETE transfer (SYSIN01) parameter should be permitted.

Important! Review the CREATEDELETE transfer parameter before permitting its use.

YES

Specifies that CREATEDELETE should always be attempted if possible; so the CREATEDELETE transfer parameter is always set to YES.

NO

Specifies that the use of CREATEDELETE is not permitted; so the CREATEDELETE transfer parameter is always set to NO.

ALLOW

Specifies that the use of CREATEDELETE is permitted.

Default: NO

CREATEDELOVRD

Specifies whether CA XCOM Data Transport should allow a CREATEDELETE to occur if the target data set is protected with an expiration date (EXPDT) and the data set has not yet expired.

YES

Allow the CREATEDELETE to proceed; the target data set is reallocated as described by the CREATEDELETE option.

NO

Deny the CREATEDELETE; the transfer fails, because the data set expiration date has not been reached.

Default: NO

CRUSSDIR

Specifies whether CA XCOM Data Transport is allowed to create a USS directory if FILEOPT=CREATE and the directory does not exist.

Default: Y

Note: If CRUSSDIR=N and the directory does not exist, then CA XCOM Data Transport displays an error message.

DEALLOCMSG

Specifies the MSG147/MSG367 after deallocation.

YES

Specifies that the following messages are to be issued after the file received is deallocated.

XCOMM0147I nnnnnnnnnn RECORDS RECEIVED SUCCESSFULLY - FILE=DSNAME.

XCOMM0367I nnnnnnnnnn BLOCKS nnnnnnnnnn RECORDS RECEIVED SUCCESSFULLY - FILE=DSNAME.

NO

Specifies that the messages are issued immediately after the file transfer is completed before the file is deallocated by the CA XCOM server.

Default: NO

DEFAULT_CHARSET

This parameter specifies the default character set CA XCOM Data Transport uses for Unicode transfers (CODE=UTF8 or CODE=UTF16).

CCSID#nnnnn/tttttttt

nnnnnn - specifies the CCSID number that corresponds to the character set. Valid values are 1 - 65535.

tttttttt (optional) – specifies the technique search order IBM Unicode Services uses when performing conversion. From 1 to 8 characters are specified. Valid values to use are:

R - Roundtrip conversion

E - Enforced Subset conversion

C - Customized conversion

L - Language Environment Behavior conversion

M - Modified for special use conversion

B - Bidi transformation (Bi-directional) conversion

0-9 - User defined conversions

Default: CCSID#37 (US EBCDIC)

Notes:

- If the technique search order is not specified, Unicode Services defaults to ‘RECLM’.
- DEFAULT_CHARSET is used when z/OS is the local partner and LOCAL_CHARSET is not specified, or when z/OS is the remote partner and REMOTE_CHARSET is not specified.

DEFAULT_CONVERROR

This parameter identifies the action when the input file contains characters that cannot be converted as they are not included within the output character sets character repertoire.

REPLACE

Replace each unconvertible character with the default substitution characters defined for the Unicode character set.

REPLACE#nnnnnnn

Replace each unconvertible character with the Unicode character that the decimal value nnnnnnn identifies. If the specified replacement character cannot be represented in the output character set, then the transfer is failed. This option is not supported for z/OS systems, where the replacement character is defined in the conversion table that is defined to IBM Unicode Services. This option is treated as REPLACE. The replacement character has a valid range of 1 – 1114111.

SKIP

The erroneous data is disregarded, but a warning message is issued at the end of the transfer. The message identifies that this condition occurred and provides a total count of the number of skipped characters. This option is not supported for z/OS systems and is treated as REPLACE.

FAIL

The transfer terminates with an error condition.

Default: FAIL

Note: DEFAULT_CONVERROR is used when z/OS is the remote partner and MBCS_CONVERROR is not specified.

DEFAULT_DELIM

This parameter specifies an optional encoding for which the specified DEFAULT_CHARSET is based. If specified, encoding must be EBCDIC and the first option in the list.

Additionally it specifies a list of delimiters to use for USS-based output files when FILEDATA=TEXT.

Used only for UNICODE transfers (CODE=UTF8 or CODE=UTF16).

Valid options:

EBCDIC – The specified character-set is EBCDIC encoded.

NA – Not applicable, the system default delimiter is used.

NL – New line

CR – Carriage return

LF – Line feed

CRLF – Carriage return/Line feed

LFCR – Line feed/Carriage return

CRNL – Carriage return/New line

Default: EBCDIC:NA

Notes:

- If EBCDIC is specified, it must be the first option in the list.
- DEFAULT_DELIM is used when z/OS is the local partner and LOCAL_DELIM is not specified. DEFAULT_DELIM is also used when z/OS is the remote partner and REMOTE_DELIM is not specified.

DEFAULT_INPUTERROR

This parameter identifies the appropriate action when the input file contains data that is not consistent with the specified input character set.

REPLACE

Replace each piece of erroneous data with the default substitution characters defined for the Unicode character set.

REPLACE#nnnnnnn

Replace each piece of erroneous data with the Unicode character that the decimal value nnnnnnn identifies. This option is not supported for z/OS systems, where the replacement character is defined in the conversion table that is defined to IBM Unicode Services. This option is treated as REPLACE. The replacement character has a valid range of 1 – 1114111.

SKIP

The erroneous data is disregarded, but a warning message is issued at the end of the transfer. The message identifies that this condition occurred and provides a total count of the number of ignored bytes. This option is not supported for z/OS systems and is treated as REPLACE.

FAIL

The transfer terminates with an error condition.

Default: FAIL

Note: DEFAULT_INPUTERROR is used when z/OS is the local partner and MBCS_INPUTERROR is not specified.

DIR

Specifies the number of PDS directory blocks to be allocated for file transfers to new data sets.

0 to 16,777,215

Specifies the number of directory blocks.

Default: 40 (blocks)

Note: Related parameters are ALLOC, SEC, PRI, CATALOG, UNIT, VOL, PSOUNIT, and PSOVOL.

DLOGMOD

Specifies the source of the logon mode name to be used when CA XCOM Data Transport initiates a session to a remote LU.

VTAM

Indicates that CA XCOM Data Transport will initiate sessions with single-session-capable LUs using SIMLOGON or REQSESS where the NIB LOGMODE=X'00', requesting that the VTAM SSCP owning the LU obtain the session parameters from the MODTAB and DLOGMOD VTAMLST associated with that LU and return them to CA XCOM Data Transport via the logon exit CINIT or SCIP BIND RU.

XCOM

CA XCOM Data Transport will initiate sessions with single- or parallel-session-capable LUs using SIMLOGON or REQSESS with the NIB LOGMODE set equal to the LOGMODE parameter from the destination member, if any, or from the Default Options Table.

Default: VTAM

Notes:

- DLOGMOD=VTAM is ignored by parallel-session LUs, and non-SNA service manager sessions use “XCOM” as the value of DLOGMOD. SNA service manager CNOS sessions are initiated with a logmode of SNASVCMG, as required by LU 6.2 protocol.
- The logmode name usually returned to the VTAM logon exit in the CINIT RU or in the SCIP exit in the BIND RU is used to construct the mode name user vector in the bind as required by LU 6.2 session establishment. If the mode name is not available in the CINIT or BIND RU control vectors X'0D' or X'2D' the LU 6.2 mode name is set from the LOGMODE parameter of the enabled destination member, if any, associated with the LU or from the Default Options Table.
- Note that while the LU 6.2 protocol does not require that the VTAM logmode name match the LU 6.2 mode name, CA XCOM Data Transport currently assumes that they do match and sets the LU 6.2 mode name in bind user vector X'02 to match the VTAM logmode.

DOMAIN

Identifies the Windows domain server used to validate the remote user ID and password.

xxxxxxxxxxxxxxx

Identifies the domain server used to validate the remote user ID and password. The name can contain up to 15 characters.

Default: None

Note: Used with transfers to Windows only.

DROPSESS

This parameter indicates whether CA XCOM Data Transport drops an LU-LU session at the conclusion of a scheduled file transfer.

YES

Indicates that CA XCOM Data Transport drops the session.

NO

Indicates that CA XCOM Data Transport does not drop the session.

QEMPTY

Indicates that CA XCOM Data Transport is to process all the transfers to a particular LU in the request queue before dropping the session.

ALL

Indicates that CA XCOM Data Transport drops all sessions, including SNASVCMG, at the conclusion of a scheduled file transfer. If all transfers for the particular LU in the request queue have finished.

<timeout value>

Indicates that if there is no activity on the session for the specified time interval, the session is dropped. This includes the SNASVCMG session. Valid values for the timeout interval are 1-60 (Seconds).

Default: NO

Notes:

- CA XCOM Data Transport for VAX and some CA XCOM Data Transport for UNIX products do not support z/OS-initiated session establishment. Therefore, DROPSESS has no effect when the target of the transfer request is one of these platforms.
- DROPSESS=ALL is only used for infrequently-used SNA partners to avoid potential problems with SNASVCMG sessions being dropped and possibly established simultaneously with heavy volume of transfers. DROPSESS=ALL is similar in function to DROPSESS=QEMPTY but drops the SNASVCMG session as well.

DUMPCL

Specifies the SYSOUT class CA XCOM Data Transport uses when creating diagnostic dumps.

A

Specifies the SYSOUT class A.

x

Specifies a SYSOUT class other than A. This class can be designated with any alphabetic character.

Default: A

DYNALMNT

Specifies whether CA XCOM Data Transport is to allow the mounting of a tape or DASD volume when processing the dynamic allocation of a data set.

YES

Specifies that CA XCOM Data Transport will request the mounting of a tape or DASD volume if it is not mounted when the file is being allocated.

NO

Specifies that CA XCOM Data Transport will *not* request the mounting of a tape or DASD volume if it is not mounted when the file is being allocated. The transfer will be terminated.

Default: NO

Note: The DYNALMNT default parameter can be specified in the XCOMDFLT table or overridden on the XCOMJOB/XCOMXFER EXEC parameter. The value of DYNALMNT can also be modified using the operator DFLT command.

Use DYNALMNT to Stack Files

If DYNALMNT=YES, you can stack files on a volume or tape.

To stack files on a volume or tape

Use the following sample JCL in SYSIN01:

```
TYPE=SEND
IPNAME=123.456.78.90
IPPORT=8044
CKPT=0
TAPE=YES
FILEOPT=CREATE
FILETYPE=FILE
SECURE_SOCKET=NO
LFILE=your.dataset
FILE=your.tape.dataset
LABEL=(1,SL)
UNIT=VTAPE
HOLD=NO
XTCNET=TAPE
XTCJOB=VTAPE
XTCGOODREL=VTAPE1
*
NEWXFER
LFILE=your.dataset
FILE=your.tape.dataset1
LABEL=(2,SL)
VOL=296934
HOLD=YES
XTCNET=TAPE
XTCJOB=VTAPE1
```

EATTR

This parameter identifies if the dataset can have extended attributes if the dataset be allocated on an Extended Address Volume (EAV).

OPT

Specifies that a dataset can optionally have extended attributes.

NO

Specifies that a dataset cannot have extended attributes.

Default: NO

Note: This parameter is applicable only for data set creation.

EDESC

Specifies the sixteen bits (coded in hexadecimal) of a z/OS message descriptor code.

Value (hexadecimal)	Descriptor Codes	Definition
8000	1	System failure
4000	2	Immediate action required
2000	3	Eventual action required
1000	4	System status
0800	5	Immediate command response
0400	6	Job status
0200	7	Application program processor
0100	8	Out-of-line message
0080	9	Status display
0040	10	Dynamic status display
0020	11	Critical eventual action required
0010	12	Important information message
	13	Reserved for future use

Default: 0200

Note: The message descriptor codes are selected by the bit configuration, where bit 0 corresponds to descriptor code 1. These codes are assigned to error messages ending with E. The descriptor code is used by CA XCOM Data Transport when issuing WTO macro instructions. z/OS inserts an indicator (a blank space followed by a + sign) at the start of the message.

ENCRYPT

Specifies whether a SYSIN01 file created using the ISPF interface is encrypted.

YES

Causes a SYSIN01 file created using the ISPF interface to be encrypted in conformance with the Data Encryption Standard (DES).

NO

Causes a SYSIN01 file created using the ISPF interface not to be encrypted and to conform to the standard EBCDIC character set.

Default: NO

Notes:

- Encryption methods other than the DES are used outside the U.S.
- Because of federal regulations restricting the export of products with cryptographic capabilities, this parameter may not be available or effective in some non-U.S. CA XCOM Data Transport products.
- If ENCRYPT=YES is specified all batch jobs must use encrypted SYSIN01 files created through the ISPF interface.

EROUT

Specifies the sixteen bits (coded in hexadecimal) of z/OS console routing codes for error messages ending with E.

Value (hexadecimal)	Descriptor Codes	Definition
8000	1	Master console action
4000	2	Master console information
2000	3	Tape pool
1000	4	Direct access pool
0800	5	Tape library
0400	6	Disk library
0200	7	Unit record pool
0100	8	Teleprocessing control
0080	9	System security
0040	10	System error/maintenance

Value (hexadecimal)	Descriptor Codes	Definition
0020	11	Programmer information
0010	12	Emulators
0008	13	Installation use
0004	14	Installation use
0002	15	Installation use
0001	16	Installation use
0000	none	For CA XCOM Data Transport use only

Default: 4020

Notes:

- The console routing codes are selected by bit configuration, where bit 0 corresponds to routing code 1, bit 1 corresponds to routing code 2, and so on. At z/OS system generation, each operator's console is assigned routing codes corresponding to the functions that the installation wants that console to perform.
- Specifying EROUT=0000 in the Default Options Table or as part of the PARM field of the EXEC JCL statement causes the suppression of all XCOMM*nnnn*E messages from the SYSLOG, and most error messages generated by XCOMJOB. All messages will continue to be a part of the CA XCOM Data Transport log.

ERRINTV

Specifies the interval in minutes at which CA XCOM Data Transport is to reset error flags on the pending request queue and retry session establishment.

1 to 999

Specifies in minutes the frequency with which error flags are set.

Default: 7 (minutes)

Notes:

- The error flags are a result of session disruptions or initial BIND failures.
- The CA XCOM Data Transport operator console command RESET can be used to manually reset an error flag. (For a description of the RESET command, see the *CA XCOM Data Transport for z/OS User Guide*.)
- A transfer may retry before the number of minutes set in the error interval, depending on when in the error interval cycle the transfer failed. The error flags are reset at the end of every error interval period, which, by default, is every seven minutes. Therefore, if a transfer fails when the error interval is five minutes into the cycle, the transfer will be retried in two minutes when all of the error flags are reset.

EXECUTE

Specifies whether CA XCOM Data Transport allows direct (TYPE=EXECUTE) transfers.

YES

Specifies that CA XCOM Data Transport allows direct transfers

NO

Specifies that CA XCOM Data Transport does not allow direct transfers

Default: YES

Note: For a description of the TYPE=EXECUTE transfer, see the *CA XCOM Data Transport for z/OS User Guide*.

EXIT01

Specifies whether CA XCOM Data Transport User Exit 1 should be enabled. Exit 1 is invoked at the end of a successfully completed file transfer.

YES

Specifies that the exit is enabled, and will use load module XCOMEX01.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 1.

Default: No.

Note: A sample version of this exit is provided in CAI.CBXGSAMP (XCOMEX01). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT02

Specifies whether CA XCOM Data Transport User Exit 2 should be enabled. Exit 2 allows the extraction of information from a JES2 address space that is not available via the subsystem interface.

YES

Specifies that the exit is enabled, and will use load module XCOMEX02.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 2.

Default: No.

EXIT03

Specifies whether CA XCOM Data Transport User Exit 3 should be enabled. EXIT03 allows the extraction of information from a JES3 address space that is not available via the subsystem interface.

YES

Specifies that the exit is enabled and will use load module XCOMEX03.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 3.

Default: No.

EXIT04

Specifies whether CA XCOM Data Transport User Exit 4 should be enabled. Exit 4 is used to supplement several layers of operator security facilities provided by CA XCOM Data Transport.

YES

Specifies that the exit is enabled and will use load module XCOMEX04.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 4.

Default: No.

Note: A sample version of this exit is provided in CAI.CBXGSAMP(XCOMEX04). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT05

Specifies whether CA XCOM Data Transport User Exit 5 should be enabled. Exit 5 is used to supplement file transfer security facilities provided by CA XCOM Data Transport.

YES

Specifies that the exit is enabled and will use load module XCOMEX05.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 5.

Default: No.

Note: A sample version of this exit is provided in CAI.CBXGSAMP(XCOMEX05). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT06

Specifies whether CA XCOM Data Transport User Exit 6 should be enabled. Exit 6 is invoked when a file transfer request changes state (from queued to active, for example).

YES

Specifies that the exit is enabled and will use load module XCOMEX06.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 6.

Default: No.

Note: A sample version of this exit is provided in CAI.CBXGSAMP(XCOMEX06). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT07

Specifies whether CA XCOM Data Transport User Exit 7 should be enabled. Exit 7 is invoked to validate a transfer request after all SYSIN01 parameters have been checked.

YES

Specifies that the exit is enabled and will use load module XCOMEX07.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 7.

Default: No.

Note: A sample version of this exit is provided in CAI.CBXGSAMP(XCOMEX07). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT08

Specifies whether CA XCOM Data Transport User Exit 8 should be enabled. Exit 8 is invoked during initialization and termination of the CA XCOM Data Transport server.

YES

Specifies that the exit is enabled and will use load module XCOMEX08.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 8.

Default: No.

Note: A sample version of this exit is provided in CAI.CBXGSAMP(XCOMEX08). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT09

Specifies whether CA XCOM Data Transport User Exit 9 should be enabled. Exit 9 is invoked when an F XCOM,EXIT,userdata command is issued.

YES

Specifies that the exit is enabled and will use load module XCOMEX09.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 9.

Default: No.

Note: A sample version of this exit is provided in CAI.CBXGSAMP(XCOMEX09). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT10

Specifies whether CA XCOM Data Transport User Exit 10 should be enabled. Exit 10 is invoked to create a banner page when a Send Report request is processed.

YES

Specifies that the exit is enabled and will use load module XCOMEX10.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 10.

Default: No.

Note: A sample version of this exit is provided in CAI.CBXGSAMP(XCOMEX10). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT11

Specifies the name of exit module. (reserved)

EXIT12

Specifies whether CA XCOM Data Transport User Exit 12 should be enabled. Exit 12 executes a user-written security routine that determines for each user the LUs with which the user is authorized to perform transfers.

YES

Specifies that the exit is enabled and will use load module XCOMEX12.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 12.

Default: No.

Notes:

- The value of the EXIT12 parameter is checked only if also LUSECURE=YES is specified in the Default Options Table.
- A sample version of Exit 12 is provided in CAI.CBXGSAMP(XCOMEX12). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

EXIT13

Specifies whether CA XCOM Data Transport User Exit 13 should be enabled. Exit 13 executes a user-written security routine that verifies a given user's authority to issue a particular console, ISPF or CICS command.

YES

Specifies that the exit is enabled and will use load module XCOMEX13.

NO

Specifies that the exit is not enabled.

Load Module name

Specifies that the named module is to be loaded and called as User Exit 13.

Default: No.

Note: A sample version of Exit 13 is provided in CAI.CBXGSAMP(XCOMEX13). For more information about this user exit, see the appendix User Exits in the *CA XCOM Data Transport for z/OS User Guide*.

FACILITY

This parameter specifies the security resource class to use for validating the target partner of a transfer request with the security package.

FACILITY

Specifies that security resource class to use is FACILITY.

xxxxxxxx

Specifies a name other than FACILITY CA XCOM Data Transport uses as a resource class name. This name can consist of up eight alphanumeric or special characters.

Default: FACILITY

Note: CA XCOM Data Transport can accept special characters, but not all the special characters are valid for the security product. See the appropriate security guide for restrictions.

FERL

Specifies the number of times CA XCOM Data Transport is to retry a transfer after certain file errors or file allocation errors have occurred.

0

Specifies that CA XCOM Data Transport should not attempt to retry a transfer after the first file allocation or other file error.

1 to 254

Specifies the number of times CA XCOM Data Transport is to retry a transfer after encountering file and file allocation errors.

255

Specifies that CA XCOM Data Transport should retry the transfer indefinitely.

Default: 255

Note: CA XCOM Data Transport retries a transfer at the interval specified by the ERRINTV parameter. Because the FERL parameter specifies the number of retries, the transfer is attempted FERL+1 times (that is, the first attempt is not included in the count).

GETSESS

Specifies how CA XCOM Data Transport establishes a session with a remote LU.

YES

Indicates that the CA XCOM Data Transport server attempts session establishment with a remote LU as soon as the request for a transfer has arrived.

NO

Indicates that the CA XCOM Data Transport server is to wait for the operator to manually activate the LU through CA XCOM Data Transport's console command ACTIVATE or for the remote LU's attempt to establish a local LU session.

Default: NO

Note: CA XCOM Data Transport for some CA XCOM Data Transport for UNIX products do not support z/OS-initiated session establishment. Therefore, GETSESS has no effect when the target of the transfer request is one of these platforms.

HISTORY

Specifies the type of history record to be written.

NONE

Specifies that a history record is not written.

ODBC

Specifies that a ODBC history record is written.

VSAM

Specifies that a VSAM history record is written.

Default: VSAM

HISTORY_OUT_DD

Specifies the DD name that will be used to write user history records.

xxxxxxxx

Specifies a one- to eight-character DD name that is used to save history records retrieved during a TYPE=HISTORY or TYPE=INQUIRE process.

Default: XCOMHOUT

HISTORY_WRITE

Specifies whether TYPE=INQUIRE is to write history records to the DD specified by HISTORY_OUT_DD.

YES

TYPE=INQUIRE writes history records to the DD specified by HISTORY_OUT_DD for all transfers where the inquiry is successful.

NO

TYPE=INQUIRE does not attempt to write history records.

Default: NO

IDESC

Specifies the sixteen bits (coded in hexadecimal) of a z/OS message descriptor code.

Value (hexadecimal)	Descriptor Codes	Definition
8000	1	System failure
4000	2	Immediate action required
2000	3	Eventual action required
1000	4	System status
0800	5	Immediate command response
0400	6	Job status
0200	7	Application program processor
0100	8	Out-of-line message
0080	9	Status display
0040	10	Dynamic status display
0020	11	Critical eventual action required
0010	12	Important information message
	13	Reserved for future use

Default: 0200

Notes:

- The codes are selected by the bit configuration where bit 0 corresponds to descriptor code 1, bit 1 to descriptor code 2, and so on. These codes are assigned to messages ending with I.
- For more information, see the EDESC parameter.

INQWAIT

Specifies how long CA XCOM Data Transport should wait - in hours (*hh*), minutes (*mm*) and seconds (*ss*) - before again determining and reporting the status of a requested data set transfer when, due to a TYPE= INQUIRE setting, CA XCOM Data Transport has determined and reported that the transfer has not been completed.

hhmmss

Specifies in hours (*hh*), minutes (*mm*) and seconds (*ss*) the length of the time that CA XCOM Data Transport should wait before reporting the status of a transfer request.

Default: 000100 (1 minute)

Note: This parameter's value is expressed as a number of up to six digits (for example, 010000 for 1 hour). For an explanation of TYPE=INQUIRE, see the *CA XCOM Data Transport for z/OS User Guide*.

IPPORT

Specifies the default TCP/IP target port.

1 to 65535

Specifies the default TCP/IP target port used when IPPORT is omitted.

Default: 8044

Note: This parameter is used for all supported types of IP transfers: IPv4, IPv6, IPv4 SSL, and IPV6 SSL. The IPPORT specified here must match the correct listening port of the target system. For example, SSL transfers should specify an IPPORT that matches the remote system’s SSL listening port.

IROUT

Specifies the sixteen bits (coded in hexadecimal) of a z/OS console routing code.

Value (hexadecimal)	Descriptor Codes	Definition
8000	1	Master console action
4000	2	Master console information
2000	3	Tape pool
1000	4	Direct access pool
0800	5	Tape library
0400	6	Disk library
0200	7	Unit record pool
0100	8	Teleprocessing control
0080	9	System security
0040	10	System error/maintenance
0020	11	Programmer information
0010	12	Emulators
0008	13	Installation use
0004	14	Installation use
0002	15	Installation use
0001	16	Installation use
0000	none	For CA XCOM Data Transport use only

Default: 4020

Notes:

- The console routing codes are selected by the bit configuration, where bit 0 corresponds to routing code 1, bit 1 to routing code 2, and so on. These codes are assigned to messages ending with I.
- Specifying IROUT=0000 in the Default Options Table or as part of the PARM field of the EXEC JCL statement causes the suppression of all XCOMM*nnnn*I messages, with the exception of XCOMM0056I and XCOMM0001I from the SYSLOG, and most informational messages generated by XCOMJOB. All messages will continue to be a part of the CA XCOM Data Transport log.

JESINTV

Sets the interval in seconds at which the CA XCOM Data Transport server is to scan the JES spool for output directed to destinations or writers specified in CAI.CBXGPARM destination members.

15 to 9999

Specifies in seconds the frequency with which CA XCOM Data Transport is to scan the JES spool for output.

Default: 60 (seconds)

Note: When output is found, it is copied by the Process SYSOUT interface to a temporary DASD data set, which is then sent as a report to the LU specified in the destination member. Copying to a temporary data set may be delayed (until a suitable session becomes available to transfer the output) by specifying PSOWAIT=YES.

JESOPER

Reserved

JOBACB

Specifies the default ACB prefix.

XCOM

Specifies the ACB prefix XCOM.

xxxxxx

Specifies an ACB prefix other than XCOM. This prefix consists of one to six alphanumeric characters with an alphabetic or national character in the first position.

Default: XCOM

Notes:

- The JOBACB parameter (that is, the ACB prefix) is used by XCOMJOB to obtain an ACB name by concatenating the prefix with a two-digit number selected from 00,01,...,99. This name is performed in a manner similar to that in which TSO and NetView obtain ACB names for their users.
- For example, when XCOM is specified at initialization time, XCOMJOB attempts to open the XCOM00 ACB name. If it succeeds, then that ACB will be used. Otherwise, the next one will be tried, XCOM01, and so on, up to 99 concurrent sessions.
- To override this parameter at execution time, use PARM='ACBNAME=*xxxxxxxx*' rather than PARM='JOBACB=*xxxxxx*'.

LCLNTFYL

Specifies the local notification level for transfers initiated from the CA XCOM for z/OS server.

A (All)

Notify on transfer completion.

W (Warn)

Notify only if the transfer received a warning or error.

E (Error)

Notify only if the transfer received an error.

Default: A

You can specify this parameter in the XCOMDFLT table, in the destination member, or in the SYSIN01. Its presence is checked for first in the SYSIN01, then in the destination member, and lastly in the XCOMDFLT default table.

LDATACLS

Specifies the name of the data class to use when allocating a new SMS-managed data set if the name is not specified by the partner.

Note: This parameter applies only to mainframe SMS data sets.

xxxxxxxx

Specifies the name of the data class to use when allocating a new SMS-managed data set if the name is not specified by the partner. The name consists of up to eight alphanumeric characters.

Default: None

LDSNLRECL

This parameter specifies the default LRECL for data sets using record allocation.

Range: 1 to 65535

Default: 132

LDSNTYPE

This parameter indicates the data set type definition, that is specified as LIBRARY or a partner PDS if not specified. If omitted, the site system default is used.

LIBRARY

Defines a PDSE

PDS

Defines a partitioned data set.

BASIC

Defines a legacy sequential dataset.

LARGE

Defines a large format sequential dataset.

EXTREQ

Defines an extended format dataset.

EXTPREF

Specifies an extended format is preferred. If the extended format is not possible, a basic format is used.

Default: None

LIBNEG

Specifies whether multiple members of a source PDS can be received in a sequential data set on the target.

YES

Specifies that multiple members of a PDS are mapped into a sequential data set on the target.

NO

Specifies that a multi-member PDS cannot be received in a sequential data set on the target.

Default: YES

Note: The initiating CA XCOM Data Transport system examines the LIBNEG parameter when it determines that the source data set is a library and the target data set is sequential. If LIBNEG=YES, the data from the members of the library is written to a target sequential data set. The target data set does not contain any indication that the original source data set was structured as a library. If LIBNEG=NO, the transfer terminates with an error.

LMGMTCLS

Specifies the name of the management class to use when allocating a new SMS-managed data set if the name is not specified by the partner.

Note: This parameter applies only to mainframe SMS data sets.

xxxxxxxx

Specifies the name of the management class to use when allocating a new SMS-managed data set if the name is not specified by the partner. The name consists of up to eight alphanumeric characters.

Default: None

LOG

Specifies whether the CA XCOM Data Transport transfer log should be written.

YES

Specifies that the transfer log is written.

NO

Specifies that the transfer log is not written.

Default: YES

Note: It is recommended that LOG=YES be used, as the log is an important source of information on file transfers and contains some information that does not appear in the JES job log.

LOGCL

Indicates the JES SYSOUT class for the CA XCOM Data Transport transfer log (see the LOG parameter).

X

Specifies the SYSOUT class X.

x

Specifies a SYSOUT class other than X. This value can be any alphabetic character.

Default: X

LOGDEST

Specifies the JES destination for the CA XCOM Data Transport transfer log (see the LOG parameter).

xxxxxxxx

A string of up to eight alphanumeric characters that specifies a JES destination.

Default: None

LOGMODE

Specifies the VTAM logmode name that CA XCOM Data Transport uses if DLOGMOD=XCOM was specified and no enabled destination member exists for the LU being activated.

XCOMMODE

Specifies the logmode name XCOMMODE.

xxxxxxxx

Specifies a logmode name other than XCOMMODE. The logmode name consists of up to eight alphanumeric characters.

Default: XCOMMODE

Notes:

- The default LOGMODE=XCOMMODE is also used as the LU 6.2 mode name passed to the session partner in the user portion of the bind if no enabled destination member exists for the LU being activated and the LOGON exit CINIT or SCIP exit bind RU does not contain a logmode name.
- For more information, see the DLOGMOD parameter.

LOSERS

Indicates the default number of contention loser sessions for LUs supporting parallel sessions.

0 to 127

Specifies the number of contention loser sessions.

Default: 4

Note: This parameter is used only when PARSESS=YES is coded.

LOWERCASE_PSWD

Indicates whether CA XCOM Data Transport is to translate the password to upper case.

YES

Specifies that CA XCOM Data Transport is *not* to translate the password to upper case. This setting is also dependent on the security system setting.

NO

Specifies that CA XCOM Data Transport is to translate all incoming passwords to upper case.

Default: NO

Notes:

- If there is no security system then passwords are translated to upper case regardless of the LOWERCASE_PSWD specification.
- If the security system does not support lower case passwords then passwords are translated to upper case regardless of the LOWERCASE_PSWD specification.

LSTORCLS

Specifies the name of the storage class to be used for a new SMS-managed data set if the name is not specified by the partner.

Note: This parameter applies to mainframe SMS data sets only.

xxxxxxxx

Specifies the name of the storage class to be used for a new SMS-managed data set if the name is not specified by the partner. The name consists of up to eight alphanumeric characters.

Default: None

LUSECURE

Specifies whether CA XCOM Data Transport is to verify a user's authority to perform transfers to the designated LU.

YES

Specifies that CA XCOM Data Transport check every user's authority to perform transfers to the designated LU.

NO

Specifies that CA XCOM Data Transport does not check the user's authority to perform transfers to the designated LU.

Default: NO

Notes:

- If LUSECURE=YES, CA XCOM Data Transport will check the value of the EXIT12 parameter. For CA XCOM Data Transport to do this, LUSECURE must be coded before EXIT12.
- The PARM keyword must not be used to override LUSECURE.

LU6ABND

Specifies how CA XCOM Data Transport handles an active conversation when a 0864 VTAM sense code is received.

RETRY

Specifies that CA XCOM Data Transport is to retry an active conversation upon receipt of a 0864 sense code.

TERMINATE

Specifies that CA XCOM Data Transport is to terminate an active conversation upon receipt of a 0864 sense code

Default: TERMINATE

Note: When specifying LU6ABND=RETRY, beware of the danger of getting in the 0864 retry loop because, in most cases, 0864 will signify a function ABEND on the system from which it was issued.

MAXDEL

Specifies the maximum number of concurrently active delete requests.

1 to 255

Specifies how many delete requests can be active at the same time.

Default: 32

MAXLOC

Specifies the maximum number of locally initiated transfers that can be active at one time.

0

Indicates that no maximum limit has been set.

1 to 9999

Specifies the highest number of locally initiated transfers that can be active at the same time.

Default: 75

MAXMOUNTWAIT

Specifies the maximum time that CA XCOM Data Transport is to wait for a device mount (for example, for a tape or disk) to be satisfied.

1 to 255

Specifies the number of minutes that CA XCOM Data Transport is to wait for a device mount (for example, for a tape or disk) to be satisfied before terminating the transfer.

Default: 10

MAXPACK

This parameter specifies the maximum packing length in bytes when PACK=LENGTH parameter is specified.

2048 to 31744 (bytes)

Specifies the maximum packing length (record packing buffer size) when the PACK=LENGTH parameter is specified.

Default: 2048 (bytes)

Notes:

- The use of PACK=LENGTH and the target buffer size is recommended to improve file transfer performance.
- In order to utilize zIIP processors for data compression, we recommend specifying a packing length of at least 4096.

MAXREM

Specifies the maximum number of remotely initiated transfers that can be active at one time.

0

Indicates that no maximum limit has been set.

1 to 9999

Specifies the highest number of remotely initiated transfers that can be active at the same time.

Default: 75

MAXRPTB

Specifies the maximum block size in bytes to be used when CA XCOM Data Transport writes a report to the JES SYSOUT queues.

512 to 32760

Specifies in bytes the maximum block size.

Default: 32760

Notes:

- If a value less than 512 is specified, the parameter defaults to 512. If a value greater than 32760 is specified, the parameter defaults to 32760.
- The value specified will affect indirect transfers as well as report transfers.

MAXTASK

This parameter specifies the maximum number of file transfers that the CA XCOM Data Transport server can perform concurrently.

0

Indicates that no maximum limit has been set.

1 to 9999

Specifies the highest number of transfers CA XCOM Data Transport can perform concurrently.

Default: 150

Notes:

- The accumulated values of MAXLOC and MAXREM do not have to equal the value of MAXTASK.
- Setting MAXTASK=0 or a high value can result in the CA XCOM Data Transport Server receiving S878 abend codes if insufficient storage is allocated to the CA XCOM Data Transport region.
- For information on how to tune these parameters, see the Server Storage Worksheet available on the CA website (http://ca.com/support). You can also contact CA Technical Support for the latest worksheet.
- DEFAULT_CONVERROR is used when z/OS is the remote partner and MBCS_CONVERROR is not specified.

MSGFMT

Specifies the placement of the CA XCOM Data Transport message ID when CA XCOM Data Transport messages are displayed on the system console.

STANDARD

Causes the CA XCOM Data Transport message ID to be moved to the beginning of the line preceding the prefix.

XCOM

Indicates that messages are displayed as in prior versions of CA XCOM Data Transport, that is, with a prefix consisting of the Remote LU name, Request number, and Transfer ID-all preceding the message ID.

Default: XCOM

Note: Specifying MSGFMT=STANDARD is useful in installations using the z/OS Message Processing Facility (MPF) to limit the traffic of messages to the system console; likewise, automated operations products can use MSGFMT=STANDARD to perform specific actions when a message is issued.

MSTRCATU

Specifies whether a z/OS master catalog update will occur if there is no user catalog alias for the high-level index of the data set to be created.

YES

Specifies that the z/OS master catalog will be updated.

NO

Specifies that the z/OS master catalog will not be updated.

Default: YES

NETMAST

Specifies an eight character module name that identifies the NetMaster module to invoke.

NO

Specifies that NetMaster is not used.

YES

Specifies that the default NetMaster module to invoke, XCOMNMFX, is used.

xxxxxxxx

Specifies a name other than XCOMNNFX is used for the NetMaster module to invoke.

Default: No

NETNAME

Specifies the name passed by CA XCOM Data Transport to remote systems in CINIT or BIND vectors to override the PLU name in the user portion of the BIND.

XCOMAPPL

Specifies that CA XCOM Data Transport use the name XCOMAPPL to override the PLU name in the user portion of the BIND.

xxxxxxxx

Specifies a name other than XCOMAPPL used by CA XCOM Data Transport to override the PLU name in the user portion of the BIND. This name can consist of up eight alphanumeric characters.

Default: XCOMAPPL

Note: The specification of NETNAME is required for some systems, such as the AS/400. The value of this parameter must match that of the ACBNAME parameter in the Default Options Table. For detailed information, see specific system components.

NTFYTPN

Specifies the name of the CICS transaction to be started via an LU 6.2 ALLOCATE verb when NOTIFY=CICS is specified on a CA XCOM Data Transport request.

XCM1

Specifies that the CICS transaction to be started use the name XCM1.

xxxx

Specifies that the CICS transaction to be started uses a name other than XCM1. This name can contain up to four alphanumeric characters.

Default: XCM1

Note: The CICS TCT table must be set up for CA XCOM Data Transport to be used. CICS is deprecated for XCOM r11.6. Use the ISPF interface instead.

OPERCMDS

Specifies the security resource class used for validating operator command access with the security package.

OPERCMDS

Specifies that security resource class used is OPERCMDS.

xxxxxxxx

Specifies a name other than OPERCMDS used by CA XCOM Data Transport to use as a resource class name. This name can consist of up eight alphanumeric or special characters.

Default: OPERCMDS

Note: CA XCOM Data Transport can accept special characters, but not all the special characters are valid for the security product. See the appropriate security guide for restrictions.

OPERLIM

Specifies the maximum number of transfer requests to be displayed on the TSO/ISPF screen or at the remote system by the operator, or transfers to be returned by the TYPE=HISTORY request.

1 to 9999

Specifies the maximum number of transfer requests to be displayed.

Default: 5000

Notes:

- The number of requests displayed directly affects the processing speed in the TSO/ISPF session. To obtain more data, use selection criteria (for example, date, request, and so on) to limit the amount of data retrieved.
- If 0454I messages are displayed during XCOMPLEX Admin start up, review your ISPF inquiry needs to determine if the OPERLIM value or XCF table size should be adjusted.
- For TYPE=HISTORY requests, the SYSIN01 parameter OLIMIT can be used to override this value.

OPERSEC

Specifies the CA XCOM Data Transport operator/user authorization level.

The effect of the OPERSEC parameter depends on the operator control functions ACTION and DISPLAY.

If the operator's control function is ACTION, the operator may control the status of requests, using such commands as CANCEL, RESUME, SUSPEND, etc.

If the operator's control function is DISPLAY, the operator may obtain a display of active, pending, and history transfer requests.

The values of the OPERSEC parameter correlate with the operator control functions as follows:

- For operator control function ACTION

  NONE

  Indicates that CA XCOM Data Transport will run no security check, giving the user unrestricted access to the ACTION control function.

  OPER

  Indicates that one of the following applies:

  - User authority can be assigned via the CA XCOM Data Transport Administrator Table.
  - Users can write their own authorization via Exit 4.
  - Authority can be directed to TSO.

  SAF

  Indicates that a standard SAF security check is made to determine if the user has the appropriate authority.

  USER

  Indicates that one of the following applies:

  - User authority can be assigned via the CA XCOM Data Transport Administrator Table.
  - Users can write their own authorization via Exit 4.

- For operator control function DISPLAY

  NONE

  Indicates that CA XCOM Data Transport will run no security check, giving the user unrestricted access to the DISPLAY control function.

  OPER

Indicates that CA XCOM Data Transport will run no security check, giving the user unrestricted access to the DISPLAY control function.

SAF

Indicates that a standard SAF security check is made to determine if the user has the appropriate authority.

USER

Indicates that one of the following applies:

- User authority can be assigned via the CA XCOM Data Transport Administrator Table.
- Users can write their own authorization via Exit 4.

Default: NONE

PLEXQ

Specifies the SYSPLEX Signaling Services group name used by the CA XCOM Data Transport server (XCOMXFER).

Use this parameter to specify an eight character SYSPLEX Signaling Services group name to which the current execution is to connect. Each character can be an upper case letter, a numeric digit, or one of the characters '$', '#', or '@'. The first character cannot be between 'A' and 'I' inclusive. The first three characters cannot be 'SYS'. The group name cannot be UNDESIG.

PRI

This parameter specifies the default amount of primary space that is allocated to new data sets in the units that the ALLOC parameter specifies.

1 to 16,277,215

Specifies the amount of primary space that is allocated to new data sets. The ALLOC parameter determines the unit of allocation.

Default: 10 (blocks, cylinders, tracks, or records depending on the value of the ALLOC parameter)

Note: Related parameters are ALLOC, AVGREC, SEC, DIR, CATALOG, UNIT, VOL, PSOUNIT, and PSOVOL.

PSO

Specifies whether CA XCOM Data Transport will start a PSO subtask to transfer JES spool files.

YES

CA XCOM Data Transport will automatically start a PSO subtask to transfer JES spool files. The PSO subtask will query the JES spool at regular intervals to see if there are any spool files that need to be transferred.

NO

Prevents the starting of the PSO subtask. JES spool will *not* be queried for spool files. However, any overhead associated with the checking for spool files will also be eliminated.

Default: YES

Important! The PSO default parameter can be specified on the XCOMXFER EXEC parameter only. It *cannot* be specified in the default table or CONFIG member.

Note: With PSO=YES, use the JESINTV parameter to specify the interval in seconds at which the JES spool is scanned.

PSOCKPT

Specifies the interval for checkpoints taken during a PSO transfer.

0 to 9999 records

Specifies the number of PSO records to transfer before a checkpoint is taken.

Default: 1000

Notes:

- The checkpoint/restart facility resumes interrupted PSO transfers from the point at which the most recent checkpoint was taken.
- The smaller the checkpoint interval, the greater the effect on the throughput, due to frequent checkpointing. Setting this parameter less than 10 would severely degrade performance. Usually values from 100 to 1000 are sufficient. Set this parameter to 0 if you do not want to do checkpointing.

PSODISP

This parameter does *both* of the following:

1. Specifies the disposition of a PSO data set if CA XCOM Data Transport is unable to successfully complete a PSO transfer.
2. Sets the DISP flag in a PSO transfer. DISP determines whether to keep or delete the report file after printing on the remote system. This value is ignored when the remote system is an IBM mainframe.

DELETE

1. Deletes the PSO data set after an unsuccessful PSO transfer.
2. Deletes the report file after it is printed on the remote system.

KEEP

1. Keeps the PSO data set after an unsuccessful PSO transfer.
2. Keeps the report file after it is printed on the remote system (by removing the temporary files).

Defaults: KEEP

Notes:

- If PSODISP=DELETE, manual intervention is required to requeue the transfer to CA XCOM Data Transport.
- If PSODISP=KEEP, manual intervention may be required to reclaim space on the remote system.

PSOPREF

Specifies the high-level qualifier used by the PSO interface when allocating temporary data sets. This value is also used by PDSE program library transfers when creating temporary data sets.

XCOMPSO

Specifies the high-level qualifier XCOMPSO.

Up to 20 alphanumeric characters

Specifies a high-level qualifier other than XCOMPSO. The high-level qualifier can contain up to 20 alphanumeric characters. You can specify multiple high-level qualifiers, up to 20 characters. The prefix must follow MVS naming conventions.

Default: XCOMPSO

PSOSECUR

Specifies whether CA XCOM Data Transport calls User Exit 5 for PSO and indirect file transfers.

YES

Specifies that CA XCOM Data Transport calls User Exit 5.

NO

Specifies that CA XCOM Data Transport does not call User Exit 5.

Default: NO

PSOUNIT

Specifies the generic unit name used by the PSO interface when allocating temporary data sets.

SYSALLDA

Specifies the unit name SYSALLDA.

xxxxxxxx

Specifies a unit name other than SYSALLDA. This name can contain up to eight alphanumeric characters.

Default: SYSALLDA

PSOVOL

Specifies the DASD volume used by the PSO interface when allocating temporary data sets. This value is also used by PDSE program library transfers when creating temporary data sets.

xxxxxx

Specifies the name of the DASD volume. The name can contain up to 6 alphanumeric characters.

Default: None

PSWDCHK

Indicates whether the password specified by the user for a transfer request is passed to z/OS dynamic allocation as the data set password.

YES

Specifies that the password is passed to z/OS.

NO

Specifies that the password is not passed to z/OS.

Default: YES

QSTART

This parameter specifies the handling of the XCOMRRDS transfer queue during initialization of the CA XCOM Data Transport server.

WARM

This option performs a WARM start of the server in “normal” mode, as for previous releases. The XCOMRRDS transfer queue is read and requests are queued for pending work in the XCOMRRDS data set.

COLD

This option performs a COLD start of the server. It removes all pending work from the RRDS. (This is roughly the equivalent of deleting and defining the XCOMRRDS data set, with the exception that the next transfer request number is not reset to 1000.)

HOLD

This option performs a HOLD start of the server. It reads the XCOMRRDS transfer queue data set and builds requests for all pending work. The difference between HOLD and WARM is that all LOCALLY initiated transfer requests are placed in a HOLD status.

Default: WARM

RCALPROC

Specifies the name of an external procedure that will be used for the recall of archived data sets.

xxxxxxxx

Specifies the one- to eight-character name of the recall procedure.

Default: None

Notes:

- If no RCALPROC is specified, then CA XCOM Data Transport will allow the local archive product to recall the data set during dynamic allocation (SVC99). In some cases dynamic allocation may acquire an enqueue on the system resource SYSZTIOT and hold the enqueue until the data set has been recalled.
- With a RCALPROC specified, CA XCOM Data Transport will request that dynamic allocation not recall the data set but instead return a code that indicates that the data set has been archived.

  When CA XCOM Data Transport receives the code from dynamic allocation that the data set has been archived, the following occurs:

  - The transfer is suspended.
  - The recall processing will be done outside of the CA XCOM Data Transport server's region using a START command. RCALPROC is used to define the name of the procedure (commonly called a proc) that is used to recall the data set.
  - The retry of the transfer is controlled by the ERRINTV parameter.

    Example

    If RCALPROC=XCOMRCAL is defined, then the XCOMRCAL procedure should be coded as follows to recall the data set:

    ```
    //XCOMRCAL PROC XDSN=
    //*=============================================================*
    //BR14    EXEC  PGM=IEFBR14
    //*=============================================================*
    //RCALDS   DD  DISP=SHR,DSN=&XDSN
    //         PEND
    ```

    The parameter XDSN will be set to the name of the archived data set.
  - We recommend that users take note of the time it takes to recall their migrated or archived data sets and use the higher end of the recall time as the ERRINTV value in the CA XCOM Data Transport default table if the RCALPROC is coded. If the recall time is much higher than the normal ERRINTV currently in use, then we suggest that you set up a separate CA XCOM Data Transport server to do transfers of data sets that are regularly migrated or archived to tape so you can set the ERRINTV value to that of the recall time without affecting the performance of your regular transfers.

- RCALPROC is not supported on the XCOMPLEX Admin Server.

RECVRID

Specifies the receiver ID for CA NetMaster File Transfer Management.

xxxxxx

Specifies the receiver ID for CA NetMaster File Transfer Management.

Default: $RFFTEVR

Notes:

- RECVRID is not supported on the XCOMPLEX Admin Server. For more information, contact CA NetMaster File Transfer Management support.
- Related parameter NETMAST.

RELEASE

Specifies whether the remote partner is to release unused DASD space when creating a new file. You can set RELEASE in destination members, in the XCOMDFLT table, and as a SYSIN01 parameter.

YES

The remote partner is to release unused DASD space.

The unused DASD space that is specified for the transfer is released when the file is closed at the end of the transfer.

NO

The remote partner is not to release unused DASD space.

REMAGE

Specifies the queue purging interval in days for remotely initiated transfer requests.

0 to 999

Specifies the length of the queue purging interval in days.

Default: 5

Note: The use of REMAGE prevents remotely initiated transfer requests that are incomplete from remaining indefinitely on the pending queue. After aging, these requests are deleted from the pending queue and a history record is generated reflecting this change. This relieves the CA XCOM Data Transport administrator from periodically checking the CA XCOM Data Transport queue to delete old requests, thus allowing unsupervised operation of CA XCOM Data Transport.

REPCR

Specifies whether CA XCOM Data Transport is to attempt to do a CREATE when receiving a transfer with FILEOPT=ADD|REPLACE and the data set does not exist.

YES

Specifies that CA XCOM Data Transport will attempt to do a CREATE when receiving a transfer with FILEOPT=ADD|REPLACE and the data set does not exist.

NO

Specifies that CA XCOM Data Transport will fail the transfer when receiving a transfer with FILEOPT=ADD|REPLACE and the data set does not exist.

Default: NO

Notes:

- The REPCR default parameter can be specified in the default table or overridden by the XCOMJOB/XCOMXFER EXEC parameter.
- The value of REPCR can also be modified using the operator DFLT command.

RMTNTFYL

Specifies the remote notification level for transfers initiated from the CA XCOM for z/OS server.

A (All)

Notify on transfer completion.

W (Warn)

Notify only if the transfer received a warning or error.

E (Error)

Notify only if the transfer received an error.

Default: A

You can specify this parameter in the CONFIG member, in the destination member, or in the SYSIN01. Its presence is checked for first in the SYSIN01, then in the destination member, and lastly in the CONFIG member.

ROSPROC

Specifies the name of the CA Roscoe started task or job that receives notification of the completion of a file transfer.

ROSCOE

Specifies that the name of the CA Roscoe started task is ROSCOE.

xxxxxxxx

Specifies that the name of the CA Roscoe started task is other than ROSCOE. This name can contain up to eight alphanumeric characters.

Default: ROSCOE

Notes:

- The ROSPROC parameter is required for users of CA Roscoe.
- The name assigned to ROSPROC is used to issue the following CA Roscoe command:

  `MODIFY` *`xxxxxxxx`*`,SEND`

ROUND

Specifies the round space allocation.

YES

Specifies that round space allocation is used.

NO

Specifies that round space allocation is not used.

Default: NO

SEC

This parameter specifies the default amount of secondary space that is allocated to new data sets in the units that the ALLOC parameter specifies.

1 to 16,277,215

Specifies the amount of secondary space that is allocated to new data sets. The ALLOC parameter determines the unit of allocation.

Default: 5 (blocks, cylinders, tracks, or records, depending on the value of the ALLOC parameter)

Note: Related parameters are ALLOC, AVGRECUNIT, PRI, DIR, CATALOG, UNIT, VOL, PSOUNIT, and PSOVOL.

SECURE_SOCKET

Specifies whether to use an OpenSSL socket or non-OpenSSL socket for transfers.

YES

Performs a secure transfer. The transfer uses an OpenSSL socket and must connect to an SSL listener on the remote partner.

NO

Performs a non-secure transfer. The transfer uses a non-OpenSSL socket and must connect to a non-SSL listener on the remote partner.

Default: NO

SECURITY

Identifies the data security interface used.

ACF2

Indicates that the security interface is CA ACF2.

NONE

Indicates that no security interface is in use.

RACF

Indicates that the security interface is IBM RACF.

SAF

Indicates that the transfer subtasks are to run under the local user's ACEE.

TOPS

Indicates that the security interface is CA Top Secret.

Default: NONE

Notes:

- The PARM keyword cannot be used to override the SECURITY parameter.
- USS support enforces SAF security for USS files, regardless of the SECURITY= parameter in the defaults table. SECURITY=NONE/ACF2/RACF/TOPS is honored for all other files.
- SECURITY=SAF cannot be used with the XCOMPLEX Admin Server (XCOMXADM), because XCOMXADM cannot perform transfers.
- If User Exit 5 is selected, it will be used in addition to the general security interface defined by the SECURITY parameter. For more information about User Exit 5, see the appendix “User Exits” in the *CA XCOM Data Transport for z/OS User Guide*.

SERL

Specifies the number of times CA XCOM Data Transport tries to establish a session with the partner LU after the first attempt at session establishment has failed. Used for SNA transfers only.

0

Specifies that CA XCOM Data Transport does not attempt session establishment after the first session establishment error.

1 to 254

Specifies the number of retries.

255

Specifies that CA XCOM Data Transport retries session establishment indefinitely.

Default: 255

Notes:

- CA XCOM Data Transport retries a transfer at the interval specified by the ERRINTV parameter. Because SERL specifies the number of retries, the transfer is attempted SERL+1 times (that is, the initial attempt to establish a session is not included in the count).
- For group transfers, a session establishment error is considered to have occurred only after attempts at session establishment with *all* LUs in the group have failed.
- For example, if a group contains three LUs and SERL=2 is specified, CA XCOM Data Transport must try session establishment with each LU in the group before the session establishment error count is incremented. If all three attempts fail, the error count is set to 1 and CA XCOM Data Transport retries session establishment with each LU (this is the first retry). If the three session establishment attempts fail again, the error count is set to 2 and all three LUs are retried (this is the second retry). If all three fail this time, too, CA XCOM Data Transport stops making further session establishment attempts because the session establishment retry limit (SERL) has been reached.

SERVADDR

Specifies the only IP address from which the CA XCOM Data Transport for z/OS server will accept incoming requests.

IP address

Specifies a local IP address that remote partners should use to send to this server.

Notes:

- The default table parameters SERVADDR and TCPSTACK affect stack affinity.
- If the SERVADDR parameter is specified, CA XCOM Data Transport will use only that stack for incoming transfers.
- For example, in sites with multiple stacks, only the IP address (stack) specified in SERVADDR will accept incoming requests. This means that each remote partner sending to this CA XCOM Data Transport server must specify the IP address (stack) specified in SERVADDR as the partner.
- This parameter defaults to none.

SERVADDRV6

Specifies the only TCP/IPv6 address from which the CA XCOM Data Transport for z/OS server will accept incoming requests.

IP Address

Specifies a local TCP/IPv6 address that remote partners should use to send to this server.

Default: None

Notes:

- The default table parameters SERVADDRV6, TCPSTACK, and TCPIP affect stack affinity.
- If the SERVADDRV6 parameter is specified, CA XCOM Data Transport will use only that stack for incoming transfers.
- For example, in sites with multiple stacks, only the TCP/IPv6 address (stack) specified in SERVADDRV6 will accept incoming requests. This means that each remote partner sending to this CA XCOM Data Transport server must specify the TCP/IPv6 address (stack) specified in SERVADDRV6 as the partner.
- This parameter defaults to none.

SERVPORT

Specifies the default TCP/IP listener port for this CA XCOM Data Transport server.

1 to 65535

Specifies the default TCP/IP listener port.

Default: 8044

Note: Change this value only if this CA XCOM Data Transport server is listening on a port other than 8044. Remote partners sending to this CA XCOM Data Transport server must specify an IPPORT that matches this value.

SERVPORTV6

Specifies the default TCP/IPv6 listener port for this CA XCOM Data Transport server.

1 to 65535

Specifies the default TCP/IPv6 listener port.

Default: 8046

Notes:

- Change this value only if this CA XCOM Data Transport server is listening on a port other than 8046. Remote partners sending to this CA XCOM Data Transport server must specify an IPPORT that matches this value.
- Related parameters TCPIPV6, TCPIP

SMF

Specifies whether an SMF record is created at the completion of a transfer.

YES

Specifies that an SMF record is created.

NO

Specifies that no SMF record is created.

Default: NO

Notes:

- The use of the SMF parameter requires APF authorization.
- The PARM keyword may not be used to override the SMF parameter.

SMFNUM

Designates the ID number for SMF records created by CA XCOM Data Transport.

0 to 999

Specifies an SMF record ID number.

Default: 192

Notes:

- When selecting a particular SMF ID for CA XCOM Data Transport recording, make sure that the member SYS1.PARMLIB(SMFPRM*xx*) specifies that the selected record type ID is written to the SYS1.MANx files.
- For information about SMF, see the IBM *SPL: System Management Facility*.
- The PARM keyword must not be used to override this parameter.

SNA

Specifies whether SNA is to be used by CA XCOM Data Transport.

YES

Specifies the use of SNA for data communications.

NO

Specifies to not use SNA for data communications.

Default: YES

Note: CICS requires SNA=YES. CICS is deprecated in XCOM r11.6. Use ISPF interface instead.

SSL

Specifies whether OpenSSL is to be used by the CA XCOM Data Transport server.

ONLY

Specifies that all TCP/IP transfers performed by this server will use OpenSSL.

ALLOW

Specifies that both OpenSSL and non-OpenSSL TCP/IP transfers can be performed by this server.

NONE

Specifies that OpenSSL TCP/IP transfers may not be performed by this server.

Default: NONE

SSLPORT

Specifies the default OpenSSL TCP/IP secure port.

1 to 65535

Specifies the port that the OpenSSL TCP/IP listener task will monitor for incoming transfers. This is also used as the default port for outbound OpenSSL TCP/IP transfers.

Default: 8045

SSLPORTV6

Specifies the default OpenSSL TCP/IPv6 secure port.

1 to 65535

Specifies the port that the OpenSSL TCP/IPv6 listener task will monitor for incoming TCP/IPv6 transfers.

Default: 8047

Note: Related parameters TCPIPV6, TCPIP, SSL

START

Specifies the name of the CA XCOM Data Transport control library (CAI.CBXGPARM) member that contains the names of all the CAI.CBXGPARM members that are to be automatically enabled at server startup.

XCOMSTRT

Specifies XCOMSTRT as the name of the CAI.CBXGPARM startup member.

xxxxxxxx

Specifies a name other than XCOMSTRT as the name of the CAI.CBXGPARM startup member. This name can be up eight alphanumeric characters long.

Default: XCOMSTRT

STCPROTOCOL

Specifies the started task communication protocol.

SNA

Specifies that an XCOM started task will perform SNA communications.

TCP

Specifies that an XCOM started task will perform TCP/IP communications (non-SSL).

SSL

Specifies that an XCOM started task will perform Secure Socket Layer (SSL) communications.

Default: SNA

SUP_ALLOC_INFO

Specifies whether dynamic allocation informational messages (IGD*nnn*I) are to be displayed in the job log. These messages continue to be displayed in the JESYSMSG data set regardless of this setting.

YES

Causes CA XCOM Data Transport to tell z/OS Dynamic Allocation to suppress informational messages only. That is, warning messages and error messages are not suppressed.

NO

Specifies that informational messages are to be displayed in the job log.

S

Default: NO

SUPPLIST

Specifies whether CA XCOM Data Transport is to suppress XCOMM0397I and XCOMM0398I messages when processing transfers of PDS or PDSE (source) members.

YES

Specifies that CA XCOM Data Transport will suppress the XCOMM0397I and XCOMM0398I messages.

NO

Specifies that CA XCOM Data Transport will allow the following messages to be issued to the console when processing transfers of PDS or PDSE (source) members:

- XCOMM0397I *membname* BEING TRANSMITTED
- XCOMM0398I *membname* BEING RECEIVED

Default: NO

Notes:

- The SUPPLIST default parameter can be specified in the default table or overridden by the XCOMJOB/XCOMXFER EXEC parameter.
- The value of SUPPLIST can also be modified using the operator DFLT command.

SURCHK

Indicates whether additional security checking is to be performed when a request to send a job is processed that specifies a user ID other than that of the user making the request.

YES

Specifies that additional security checking is to be performed.

NO

Specifies that no additional security checking is to be performed.

Default: NO

Notes:

- If SURCHK=YES is specified, CA XCOM Data Transport makes an additional security check, using the security resource class specified in the SURCLS parameter (see the section SURCLS that follows) with an entity name of the surrogate user ID.
- If the user making the transfer request has READ or higher access to this resource, the request is executed; otherwise, the request fails.

SURCLS

Specifies a security resource class name, which CA XCOM Data Transport uses to verify users' authority to employ surrogate user IDs when making requests to send jobs.

xxxxxxxx

Specifies the class name of CA XCOM Data Transport's security resource. The name can contain up to eight alphanumeric characters.

Default: None

Notes:

- This parameter can be specified only if SURCHK=YES has been specified.
- If specified, CA XCOM Data Transport validates any surrogate user ID request by making a security check in the specified resource class, using the requested surrogate user ID as the entity name. If the user making the request has READ or higher access to this resource, the request is executed; otherwise, the request fails.

SWAIT

Specifies the number of seconds that CA XCOM Data Transport waits for a session to be established after the request for session establishment has been queued.

1 to 32767

Specifies the time limit in seconds within which a queued session establishment request must result in the establishment of a session.

Default: 30 (seconds)

Note: If CA XCOM Data Transport's first attempt at session establishment fails, a request for session establishment is placed in a request queue. A session must be established within the time specified by the SWAIT parameter. If no session is established within the specified time, a session establishment error is considered to have occurred, which results in incrementing the value of the SERL parameter (see the description of SERL).

SYSID

Specifies the system ID (one to four characters). SYSID and SYSNAME together provide a unique system identifier.

A special value, *SMF, indicates that the z/OS SMFID is to be used for the system ID value.

You can specify this parameter on the JCL PARM card for the CA XCOM Data Transport started task to override the value in the Default Options Table.

Default: *SMF

SYSNAME

Specifies the system name (one to eight characters). SYSID and SYSNAME together provide a unique system identifier.

A special value, *JOBNAME, indicates that the job name of the CA XCOM Data Transport started task or XCOMJOB batch job is to be used as the system name value.

You can specify this parameter on the JCL PARM card for the CA XCOM Data Transport started task to override the value in the Default Options Table.

Default: *JOBNAME

TCPIP

Specifies if TCP/IPv4 support is enabled for transfers.

NO

TCP/IPv4 support is not enabled.

YES

TCP/IPv4 support is enabled (the TCP/IP option must be installed).

Default: YES

Note: The TCPIP parameter (as well as all related TCPIP parameters such as TCPSTACK) is ignored in the XCOMPLEX Admin Server region since this region only routes schedule transfer requests to the appropriate XCOMPLEX Worker Servers and does not do the actual transfers. TCPIP is not supported or needed by the XCOMPLEX Admin Server.

TCPIPV6

Specifies whether TCP/IPv6 is to be used by the CA XCOM Data Transport server.

NONE

There will be no TCP/IPv6 listener tasks started to monitor inbound TCP/IPv6 connections.

ALLOW

TCP/IPv6 listener tasks will be started to monitor inbound TCP/IPv6 connections.

ONLY

There will be no TCP/IPv4 listener tasks started to monitor inbound TCP/IPv4 connections. The TCP/IPv6 listener tasks can handle TCP/IPv4 requests.

Default: NONE

Notes:

- The following table shows the relationship between the TCPIP=, TCPIPV6=, and SSL= parameters.
- Related parameters: TCPIP, SERVPORTV6, SSLPORTV6, SERVADDRV6.

TCPIPv6 Parameter Table

TCPIP= Value	SSL= Value	TCPIPV6= Value	IPv4 Listeners	IPv6 Listeners
NO	NONE	NONE	No	No

TCPIP= Value	SSL= Value	TCPIPV6= Value	IPv4 Listeners	IPv6 Listeners
NO	Not NONE	Not NONE	N/A*	N/A*
YES	NONE	NONE	Non-secure	No
YES	ALLOW	NONE	Non-secure and SSL	No
YES	ONLY	NONE	SSL only	No
YES	NONE	ALLOW	Non-secure only	Non-secure only**
YES	ALLOW	ALLOW	Non-secure and SSL	Non-secure and SSL**
YES	ONLY	ALLOW	SSL only	SSL only**
YES	NONE	ONLY	No	Non-secure only**
YES	ALLOW	ONLY	No	Non-secure and SSL**
YES	ONLY	ONLY	No	SSL only**

* The N/A values are not allowed by the assembly of the #DFLTAB.

** XCOM TCP/IPv6 listeners can handle TCP/IPv4 connections as well as TCP/IPv6 connections.

TCPLUSEC

This parameter governs how the TCP/IP address will be identified in the call to the security system.

BINARY

The IP address is converted to displayable hex and is passed to security as an eight-digit hexadecimal number.

For example, IP address 255.255.255.255 becomes “FFFFFFFF”. Likewise, 127.128.16.3 becomes “7F801003”.

TCPIP

The literal “TCPIP” is passed to security.

TCP/IP

The literal “*TCP/IP*” is passed to security.

Default: TCP/IP

Note: CA XCOM Data Transport passes this value to Exit 5 (XCOMEX05) in the field SECLUN of the Exit 5 parameter list for file security (mapped by the SECDSECT macro).

TCPRTIME

Specifies in seconds the maximum time that CA XCOM Data Transport waits for a TCP/IP response during a TCP/IP receive function.

0 to 99999

Specifies, in seconds, the maximum time that CA XCOM Data Transport waits for a TCP/IP response during a TCP/IP receive function.

Default: 0 (seconds)

Notes:

- The session may also be timed-out by the TCP/IP stack after expiration of the keep-alive time. The keep-alive time is defined by the TCP/IP stack. For more information on setting the keep-alive time, see the TCP/IP stack documentation.
- A value of 0 indicates that CA XCOM Data Transport will not time out during a TCP/IP function. However, the session may still be timed out by the keep-alive time limit.

TCPSESS

Specifies the maximum number of TCP/IP file transfers that the server can perform concurrently to any one IP address.

0

Indicates that no maximum limit has been set.

1 to 128

Specifies the maximum number of concurrent transfers the server can perform to any one IP address.

Default: 15

Note: This parameter should be specified to throttle TCP/IP file transfers. This parameter governs locally-initiated transfers only.

TCPSOCKD

The TCP/IP Socket option TCP_NODELAY. This refers to the Nagle algorithm for send coalescing. By default, small sends may be delayed. This should have no impact for normal CA XCOM Data Transport record sizes. This parameter is used for TCP/IP transfers only.

YES

Small sends may be delayed. (Does not disable the Nagle algorithm.)

NO

All sends are immediate. (Disables the Nagle algorithm.)

Default: YES

Note: Socket options affect the way CA XCOM Data Transport uses the TCP/IP stack implementation.

TCPSRCVB

TCP/IP Socket option SO_RCVBUF. This parameter specifies the buffer size for receives. Use 0 for the default size provided by the socket implementation. The value for TCPSRCVB can be smaller than the value for TCPTBUF. Used for TCP/IP transfers only.

0 to 256000

Specifies the size of the TCP/IP receive buffer.

Default: 64

Notes:

- A value in the range 1 to 250 is interpreted as K bytes while 251 and above is interpreted as bytes.
- This value is rounded to the next highest 4K.
- The maximum value allowed for this parameter may differ from one z/OS version to another. If errors occur during server startup, then the value of this parameter needs to be lowered.

TCPSSNDB

TCP/IP Socket option SO_SNDBUF. This parameter specifies the buffer size for sends. Use 0 for the default size provided by the socket implementation. The value for TCPSSNDB can be smaller than the value for TCPTBUF. Used for TCP/IP transfers only.

0 to 256000

Specifies the buffer size for sends for TCP/IP transfers.

Default: 64

Notes:

- A value in the range 1 to 250 is interpreted as K bytes while 251 and above is interpreted as bytes.
- This value is rounded to the next highest 4K.
- Socket options affect the way CA XCOM Data Transport uses the TCP/IP stack implementation.
- The maximum value allowed for this parameter may differ from one z/OS version to another. If errors occur during server startup, then the value of this parameter will need to be lowered.

TCPSTACK

The job name or started task name of the TCP/IP stack that CA XCOM Data Transport will listen on. If multiple stacks are running at your site, use this parameter to specify the required stack.

xxxxxxxx

Specifies up to eight alphanumeric characters indicating the job name or started task name of the TCP/IP stack that CA XCOM Data Transport will listen on.

Default: None

Notes:

- An empty value causes the CA XCOM Data Transport TCP/IP interface to listen on every TCP/IP stack running on the system. To use a specific stack, specify the job name of the TCPIP stack.
- This parameter affects only the stack that CA XCOM Data Transport is listening on.
- For locally initiated transfers, the system selects the stack to use.

TCPTBUF

This parameter specifies the internal buffer size for sends and receives. The default size allows multiple CA XCOM Data Transport records to be received in a single socket call. With this default, if your CA XCOM Data Transport record size is less than 32K, CA XCOM Data Transport will attempt to receive multiple records in a single socket call. Used for TCP/IP transfers only.

0 to 65536

Specifies the internal buffer size for sends and receives for TCP/IP transfers.

Default: 32768

TCPTCHKF

Indicates the frequency with which CA XCOM Data Transport checks to see if incoming error information is available when sending data. For example, if the value is 5, a check is made every fifth time that data is sent, to determine if data is available for receiving. Larger values give better performance. Smaller values minimize the sending of data after the partner reports an error. Used for TCP/IP transfers only.

0 to 9999

The interval that data sends are checked for incoming error information.

Default: 10

TCPTTIME

Specifies the maximum time to wait, in seconds, for a partner to terminate TCP/IP communications. If a transfer terminates normally, both sides of the conversation coordinate the termination, and there should be no need to wait. This timeout will occur only during an error in the termination of the connection. Used for TCP/IP transfers only.

0 to 999

Specifies the maximum number of seconds to wait for partner to terminate TCP/IP communications.

Default: 20 seconds

TERL

Specifies the number of file allocation errors, VTAM errors, and retries during TCP/IP session establishment that CA XCOM Data Transport will allow before terminating the transfer.

0

Specifies that no errors are retried (regardless of the settings of FERL, SERL, or VERL).

1 to 32766

Specifies the total number of times that CA XCOM Data Transport attempts to correct errors.

32767

Specifies that errors are retried indefinitely.

Default: 32767

Notes:

- If the number of errors for a particular category of retryable errors (FERL, SERL, VERL) or any combination of such categories is equal to the value of TERL, CA XCOM Data Transport stops further retry attempts and fails the transfer. For example, suppose that TERL=10, FERL=6, SERL=6, and VERL=4. If CA XCOM Data Transport has retried file allocation 5 times and session establishment 5 times, the total error retry limit (TERL) has been reached. Therefore, CA XCOM Data Transport makes no further error retry attempts, although the individual error retry limits have not been reached.
- TERL=32767 (that is, indefinite number of error retries) does not override a value of FERL, SERL, or VERL that indicates a finite number of retries (such as any value greater than 0 but smaller than 255). For example, if TERL=32767 and SERL=5, CA XCOM Data Transport stops retrying session establishment errors after the fifth error has occurred, instead of continuing indefinitely.

TIMEOUT

Specifies in seconds the maximum duration of time that CA XCOM Data Transport waits for a VTAM or TCP/IP response before aborting a session with a partner.

0 to 99999

Specifies in seconds the length of time that CA XCOM Data Transport is to wait for a VTAM or TCP/IP response.

Default: 600 (seconds)

Notes:

- For direct (TYPE=EXECUTE) transfer requests, session establishment will be tried until the limit is reached. For CA XCOM Data Transport server requests, if the TIMEOUT limit is reached during an active transfer, the transfer will be aborted and placed into VTAM error status.
- For a TCP/IP transfer, the TIMEOUT parameter will determine how long to wait for a connection to a remote IP node.
- If the TIMEOUT limit is reached during SNA session establishment, the pending session will be aborted. Also, any transfers relating to the failing LU will be placed into error status for the duration specified by the error interval (see the ERRINTV parameter).
- Proper use of TIMEOUT ensures that CA XCOM Data Transport does not hang indefinitely, waiting for a VTAM or TCP/IP response. This is particularly important in the TSO/ISPF environment, where XCOMJOB is invoked for operator functions or transfer requests.

UMASK

This parameter is used to set the permissions assigned to a file when the file is being created on a UNIX system for the first time. The value is expressed as an octal number (base 8). The octal number has the same meaning as in the standard UNIX umask command. This value is removed from the XCOM default of 666 (rw-rw-rw) for files and 777(rwxrwxrwx) for directories.

Range: 000 to 777

Default: 022

Notes:

- For directories—CA XCOM Data Transport sets permissions for a created directory to 7*xx*, no matter what owner UMASK value that was specified. Group and other permissions, represented by *xx*, represent the permissions with the specified UMASK removed.
- For files – While the file is being transferred, CA XCOM Data Transport sets permissions for a created file to 6*xx*, where *xx* represents the permissions with the specified UMASK removed. After the transfer has been completed, CA XCOM Data Transport sets the owner permission with the specified UMASK removed.

UNIT

Specifies the default unit name when allocating new data sets.

SYSALLDA

Specifies the direct access device name SYSALLDA.

xxxxxxxx

Specifies a device name other than SYSALLDA. The name can be up to eight alphanumeric characters.

Default: SYSALLDA

Note: Related parameters are ALLOC, PRI, SEC, DIR, CATALOG, and VOL.

USERD

Specifies system-wide user data to be included in the logging information for file transfers initiated by the system.

XCOMMVS

Specifies the user data XCOMMVS.

xxxxxxxxxx

Specifies user data other than XCOMMVS. The length of the user data may be up to 10 bytes.

Default: XCOMMVS

Note: The user data can be used for the correlation of CA XCOM Data Transport activities in multi-CA XCOM Data Transport environments.

USEROVR

Specifies whether the remote or local user ID parameters USERID and LUSER can be used.

YES

Specifies that the parameters USERID and LUSER can be used.

NO

Indicates that the USERID and LUSER parameters are ignored and the user ID used for security authorization is the same as for the batch job or TSO session.

Default: YES

Note: For information about the LUSER and USERID parameters, see the *CA XCOM Data Transport for z/OS User Guide*.

USERPRO

Specifies whether user IDs are propagated.

YES

Specifies that CA XCOM Data Transport is to forego verification of the user ID on the remote system, because it was already verified on the local system.

NO

Specifies that CA XCOM Data Transport is to verify the user ID on the remote system even if it was already verified on the local system.

Default: NO

Note: USERPRO is relevant when the server you are using resides on a different z/OS system. In this case, if USERPRO=NO, LUSER and LPASS will be required in the SYSIN01 stream. If USERPRO=YES, LUSER and LPASS are optional. If USERPRO=YES and no LUSER is specified, the user ID for the TYPE=SCHEDULE job (or the TSO user ID if in ISPF) will be propagated to the server that is to initiate the transfer.

VERL

Used by VTAM and TCP/IP to determine the number of times CA XCOM Data Transport attempts to retry a file transfer that has received a network error.

0

Specifies that the transfer is not retried.

1 to 254

Specifies the number of retry attempts.

255

Specifies that the transfer is retried indefinitely.

Default: 255

Note: CA XCOM Data Transport retries a transfer at the interval specified by the ERRINTV parameter. Because VERL specifies the number of retries, the transfer is attempted VERL+1 times (that is, the initial attempt to establish a session is not included in the count).

VOL

Specifies the default volume serial number to be used when allocating new data sets.

xxxxxxxx

Specifies the volume serial number, which can contain up to eight alphanumeric characters.

Default: None

Note: Related parameters are ALLOC, PRI, SEC, DIR, CATALOG, and UNIT.

VTAMGNAM

Specifies the VTAM Generic Resource Name that the server attempts to open at initialization time.

Note: This parameter is valid for only the CA XCOM Data Transport server job.

xxxxxxxx

Specifies the one- to eight-character alphanumeric Generic Resource Name.

Default: None

Note: Do not specify this for the XCOMPLEX Admin Server as the XCOMPLEX Admin Server cannot accept transfers.

WINNERS

Specifies the maximum number of contention winners for parallel session partners.

0 to 127

Specifies the maximum number of contention winners.

Default: 4

Note: The WINNERS parameter is used when an LU is defined as PARSESS=YES.

XCOM_CONFIG_SSL

Specifies the HFS SSL configuration file path and file name.

1 to 256 characters

Specifies the HFS path and file name of the SSL configuration file used by CA XCOM Data Transport for secure transfers.

Note: A sample SSL configuration file, configssl.cnf, is provided with the installation.

Default: None

XCOMHIST

Specifies the name of the ODBC Data Source location as defined in SYSIBM.LOCATIONS; is analogous to the CA XCOM Data Transport Default Options Table parameter.

Range: 1 to 128 characters

XCOMHIST_OWNER

Specifies the ODBC owner for the history table.

Range: 1 to 128 characters

Default: XCOMUSER

XCOMHIST_PASSWORD

The encrypted password for the XCOMHIST_USER. If the XCOMHIST_USER does not require a password, this parameter should be omitted.

Range: Exactly 70 characters

Note: The XCOMENCR utility program should be used to create this encrypted password. See sample program CAI.CBXGJCL(XCOMENCR).

XCOMHIST_TBL

Specifies the ODBC history table name.

Default: XCOM_HISTORY_TBL

XCOMHIST_USER

This parameter specifies the authorization ID to use when connecting to the ODBC database for history.

Range: 1 to 128 characters

Default: XCOMUSER

XCOMPLEX

Specifies the name of the XCOMPLEX facility that the CA XCOM Data Transport Administrator administers.

Specify the same XCOMPLEX name for each CA XCOM Data Transport server that is a member of the XCOMPLEX facility.

Note: This parameter is valid for the CA XCOM Data Transport server job and the CA XCOM Data Transport administrator job.

xxxxxxxxxxxxxxxx

Specifies the 1- to 16-character alphanumeric name of the XCOMPLEX to join.

Default: NONE (XCOMPLEX will be disabled)

ZIIP

Specifies whether CA XCOM Data Transport should utilize zIIP processors to reduce CPU utilization costs.

Specify Yes or No to indicate if a zIIP can be utilized to offload some processing from the main CPUs.

Default: YES

Note: ZIIP can only be specified in a TYPE=CONFIG member. If the XCOM Default Option Table is used, the default value will be used.

Destination Parameters for Single LUs, Groups of LUs, and Single IPNAMEs

This section describes the parameters for single LUs, groups of LUs, and single IPNAMEs.

ACCSEC

Indicates the status of the access security subfield in the ATTACH FMH-5.

YES

Indicates that the partner system's LU 6.2 implementation supports or requires an access security subfield in the ATTACH FMH-5.

NO

ACCSEC=NO indicates that the partner system's LU 6.2 implementation does not support or require an access security subfield.

Default: NO

Notes:

- This parameter applies only to partner systems such as AS/400, whose LU 6.2 implementation supports or requires an access security subfield in the ATTACH FMH-5.
- If ACCSEC=YES is coded, the PASSWORD and USERID parameters must be specified in the TSO/ISPF panel or the SYSIN01 statement stream when a transfer is requested. The password will be decrypted before it is put into the FMH-5.

ALERT_CONV

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

The value of the ALERT_CONV parameter is the severity level of the event named by the first term.

Specifies that CA XCOM Data Transport generates alerts for conversation related events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_CONV is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_CONV parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_CONV parameter according to its specification in the CONFIG Member.

ALERT_FILE

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

Specifies that CA XCOM Data Transport generates alerts for file related events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_FILE is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_FILE parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_FILE parameter according to its specification in the CONFIG Member.

ALERT_GEN

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

Specifies that CA XCOM Data Transport generates alerts for general events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_GEN is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_GEN parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_GEN parameter according to its specification in the CONFIG Member.

ALERT_SEC

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

Specifies that CA XCOM Data Transport generates alerts for security related events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_SEC is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_SEC parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_SEC parameter according to its specification in the CONFIG Member.

ALERT_SESS

Causes CA XCOM Data Transport to generate NetView generic alerts for conditions that meet the specified event category and severity level.

Specifies that CA XCOM Data Transport generates alerts for session related events.

The possible values are the following:

Information

Specifies that CA XCOM Data Transport generates alerts when any information is available pertaining to the designated event category.

Warning

Specifies that CA XCOM Data Transport generates alerts for the designated event category when warnings are issued during the event.

Error

Specifies that CA XCOM Data Transport generates alerts for the designated event category when errors occur during the event.

Action

Specifies that CA XCOM Data Transport generates alerts for the designated event category when the event requires corrective action on the part of the user or operator.

NONE

Specifies that CA XCOM Data Transport generates no alerts for the specified event category.

Default: NONE (for severity level)

Notes:

- The severity levels are cumulative, that is, a severity level of Information includes events of the severity levels Information, Warning, Error and Action; a security level of Error includes events of the security levels Error and Action, and so on.
- If ALERT_SESS is specified in the CONFIG Member and the CA XCOM Data Transport control library, the values given in the former are merged with those in the latter. However, circumstances may occur where CA XCOM Data Transport cannot use the ALERT_SESS parameter as specified in the control library; in such cases, CA XCOM Data Transport uses the ALERT_SESS parameter according to its specification in the CONFIG Member.

CODETABL

Specifies the one- to three-character prefix to the file names, atoe.tab and etoa.tab, that contain the external ASCII-to-EBCDIC and EBCDIC-to-ASCII custom character conversion tables on the CA XCOM Data Transport for Windows and the CA XCOM Data Transport for UNIX platforms. These custom character conversion tables determine which external translation tables are to be used when a transfer is sent to these platforms. This parameter is valid only if INTERNAL_CONVERSION_TABLES=NO is set on the receiving platform and the platforms are at r11 or above.

Default: None

COMPNEG

Specifies whether compression negotiation is performed.

YES

Specifies that the data compression method is negotiated.

NO

Specifies that the data compression method is not negotiated.

Default: YES

Notes:

- If COMPNEG=YES and the compression method suggested by the initiator or partner LU is known to both participants, that method is used to compress the transfer data.
- If the suggested compression method is not known to a participant involved in the transfer, Run Length Encoding of blanks and zeros is applied to the data to be transferred.
- If an unknown compression type is requested, the file transfer is rejected.

COMPRESS

This parameter specifies whether data compression is used for a transfer.

YES

CA XCOM Data Transport provides Run Length Encoding (RLE) only for blanks and binary zeroes.

NO

No data compression takes place.

RLE

CA XCOM Data Transport provides complete RLE for all repeating characters.

COMPACT

CA XCOM Data Transport provides complete RLE (as in RLE) plus a byte compaction scheme suitable for uppercase EBCDIC text.

COMPACTL

The COMPACTL compression parameter is the same as COMPACT. However, this value is most suitable for lowercase EBCDIC text.

LZSMALL

CA XCOM Data Transport compresses the data according to the small memory model of Lempel-Ziv 77 compression.

LZMEDIUM

CA XCOM Data Transport compresses the data according to the medium memory model of Lempel-Ziv 77 compression.

LZLARGE

CA XCOM Data Transport compresses the data according to the large memory model of Lempel-Ziv 77 compression.

HUFFMAN

Greater compression than RLE but not as much as the Lempel-Ziv 77 modes.

LZRW3

General-purpose algorithm that runs fast and gives reasonable compression.

ZLIB(*n*)

Greater compression than LZRW3 but less than LZSMALL, LZMEDIUM, and LZLARGE. The **n** value can be 1 through 9.

Default: YES

Note: Specifying a value other than NO enables use of zIIP to perform the compression function. zIIP processing is used for compression only if the buffer size to compress is >= 4096 bytes. Specifying PACK=LENGTH and use a high MAXPACK value (>= 4096) for TCP/IP transfers or RUSIZE (>=4096) for SNA transfers to use zIIP.

COMPRESS_PDS

COMPRESS_PDS is the parameter that causes the actual PDS compression to happen. If your CA XCOM Data Transport administrator has enabled the programmatic PDS compression feature in a CA XCOM Data Transport region, you can use the COMPRESS_PDS option to control if and when output PDS data sets get compressed as part of the transfer.

NONE

Suppresses the compression of an output PDS dataset as part of a CA XCOM Data Transport transfer.

BEFORE

Compresses an output PDS dataset before the transfer of user data begins.

AFTER

Compresses an output PDS dataset after the transfer of user data has completed.

BOTH

Compresses an output PDS dataset both before and after the transfer of user data.

Default: NONE

Notes:

- COMPRESS_PDS applies only to PDS data sets that is, or have been, opened for output as the target of a CA XCOM Data Transport transfer.
- If the COMPRESS_PDS option is present in the DEST member for a particular transfer partner, and that DEST member is made available to the XCOMJOB utility that is used to schedule a transfer to that partner, it becomes the default value for all transfers initiated with that partner CA XCOM Data Transport in that invocation of XCOMJOB.
- If your CA XCOM Data Transport administrator configures CMPRS_PDS_ALLOW=YES or CMPRS_PDS_ALLOW=X37, the COMPRESS_PDS=NONE option cannot suppress the compression of an output PDS data set if a z/OS system abend B37, D37, or E37 occurs.

Output

Output from the compression utility is handled in accordance with the setting of the CMPRS_SYSOUT_CL server-level parameter. The spool data sets (if they are allocated) will have the following prefixes:

- XB for compressions performed before a transfer, as in the case with COMPRESS_PDS=BEFORE or on a restarted transfer request
- XA for transfers performed after a transfer

The decimal transfer request number is appended to the prefix to provide a unique spool entry for each compression operation. This naming convention allows for the correlation of compression utility output with a specific file transfer if there is a need for problem determination research after the transfer.

Example:

For a request number 034271, the following spool entry names would apply:

- The output from the utility used to compress the PDS data set before the transfer would be named XB034271.
- For a compression performed after the data transfer, the SYSOUT dataset would be named XA034271.

CONVTYPE

Specifies the type of LU 6.2 conversation to be used by a z/OS-initiated conversation.

MAPPED

Specifies that the conversation should be mapped.

BASIC

Specifies that the conversation should be basic.

Default: MAPPED

Notes:

- This parameter should only be used when communicating with IBM TPF systems.
- CA XCOM Data Transport accepts basic or mapped conversations regardless of this parameter's specification.

CPUTYPE

Specifies the CPU type of the remote LU.

AS400

Specifies the CPU type AS400.

DG

Specifies the CPU type DG.

MVS

Specifies the CPU type MVS.

PC

Specifies the CPU type PC.

STRATUS

Specifies the CPU type STRATUS.

SUN

Specifies the CPU type SUN.

VM

Specifies the CPU type VM.

Default: None

Notes:

- This parameter is used for informational purposes only.
- The range of CPU types that can serve as remote LUs in CA XCOM Data Transport sessions is not limited to the set of values given in the preceding table.

CREATEDELETE

It specifies whether the CREATEDELETE transfer (SYSIN01) parameter should be permitted.

ALLOW

The use of CREATEDELETE is permitted

NO

The use of CREATEDELETE is not permitted; so the CREATEDELETE transfer parameter is always set to NO.

YES

CREATEDELETE should always be attempted if possible; so the CREATEDELETE transfer parameter is always set to YES.

Default: NO

For the Default Options Table (XCOMDFLT)

As specified in the Default Options Table (XCOMDFLT)

For the destination member (XCOMCNTL), or for an XCOMJOB PARM parameter or XCOMXFER PARM parameter

Notes:

Important! Review the CREATEDELETE transfer parameter before permitting its use.

The default of NO can be overridden for individual partners in the destination member.

CVASCII

Specifies the name of the code page conversion table that is to be used in translating ASCII data at the destination.

xxxxxxxx

Specifies the name of a customized code page conversion table. The name can consist of up to eight alphanumeric characters.

Default: None

CVBINARY

Specifies the name of the code page conversion table that is to be used in translating binary data at the destination.

xxxxxxxx

Specifies the name of a customized code page conversion table. The name can consist of up to eight alphanumeric characters.

Default: None

CVEBCDIC

Specifies the name of the code page conversion table that is to be used in translating EBCDIC data at the destination.

xxxxxxxx

Specifies the name of a customized code page conversion table. The name can consist of up to eight alphanumeric characters.

Default: None

DATACLAS

Specifies the name of the data class to use when allocating a new SMS-managed data set.

xxxxxxxx

Specifies the one- to eight-character data class name to use when allocating a new SMS-managed data set.

Note: This parameter applies only to mainframe SMS data sets.

Default: None

DEFAULT_CHARSET

This parameter specifies the default character set CA XCOM Data Transport uses for Unicode transfers (CODE=UTF8 or CODE=UTF16).

CCSID#nnnnn/tttttttt

nnnnnn - specifies the CCSID number that corresponds to the character set. Valid values are 1 - 65535.

tttttttt (optional) – specifies the technique search order IBM Unicode Services uses when performing conversion. From 1 to 8 characters are specified. Valid values to use are:

R - Roundtrip conversion

E - Enforced Subset conversion

C - Customized conversion

L - Language Environment Behavior conversion

M - Modified for special use conversion

B - Bidi transformation (Bi-directional) conversion

0-9 - User defined conversions

Default: CCSID#37 (US EBCDIC)

Notes:

- If the technique search order is not specified, Unicode Services defaults to ‘RECLM’.
- DEFAULT_CHARSET is used when z/OS is the local partner and LOCAL_CHARSET is not specified, or when z/OS is the remote partner and REMOTE_CHARSET is not specified.

DEFAULT_CONVERROR

This parameter identifies the action when the input file contains characters that cannot be converted. The characters are not included within the output character sets character repertoire.

REPLACE

Replace each unconvertible character with the default substitution characters defined for the Unicode character set.

REPLACE#nnnnnnn

Replace each unconvertible character with the Unicode character that the decimal value nnnnnnn identifies. If the specified replacement character cannot be represented in the output character set, then the transfer is failed. This option is not supported for z/OS systems, where the replacement character is defined in the conversion table that is defined to IBM Unicode Services. This option is treated as REPLACE. The replacement character has a valid range of 1 – 1114111.

SKIP

The erroneous data is disregarded, but a warning message is issued at the end of the transfer. The message identifies that this condition occurred and provides a total count of the number of skipped characters. This option is not supported for z/OS systems and is treated as REPLACE.

FAIL

The transfer terminates with an error condition.

Default: The DEFAULT_CONVERROR parameter in the destination member or CA XCOM Data Transport Default Options Table specifies the default.

DEFAULT_DELIM

This parameter specifies an optional encoding for which the specified DEFAULT_CHARSET is based. If specified, encoding must be EBCDIC and the first option in the list.

Additionally it specifies a list of delimiters to use for USS-based output files when FILEDATA=TEXT.

Used only for UNICODE transfers (CODE=UTF8 or CODE=UTF16).

Valid options:

EBCDIC – The specified character-set is EBCDIC encoded.

NA – Not applicable, the system default delimiter is used.

NL – New line

CR – Carriage return

LF – Line feed

CRLF – Carriage return/Line feed

LFCR – Line feed/Carriage return

CRNL – Carriage return/New line

Default: EBCDIC:NA

Notes:

- If EBCDIC is specified, it must be the first option in the list.
- DEFAULT_DELIM is used when z/OS is the local partner and LOCAL_DELIM is not specified. DEFAULT_DELIM is also used when z/OS is the remote partner and REMOTE_DELIM is not specified.

DEFAULT_INPUTERROR

This parameter identifies the appropriate action when the input file contains data that is not consistent with the specified input character set.

REPLACE

Replace each piece of erroneous data with the default substitution characters defined for the Unicode character set.

REPLACE#nnnnnnn

Replace each piece of erroneous data with the Unicode character that the decimal value nnnnnnn identifies. This option is not supported for z/OS systems, where the replacement character is defined in the conversion table that is defined to IBM Unicode Services. This option is treated as REPLACE. The replacement character has a valid range of 1 – 1114111.

SKIP

The erroneous data is disregarded, but a warning message is issued at the end of the transfer. The message identifies that this condition occurred and provides a total count of the number of ignored bytes. This option is not supported for z/OS systems and is treated as REPLACE.

FAIL

The transfer terminates with an error condition.

Default: The DEFAULT_INPUTERROR parameter in the destination member or CA XCOM Data Transport Default Options Table specifies the default.

DEST

Specifies the JES destination name, which CA XCOM Data Transport's Process SYSOUT interface uses as a qualifying argument in the JES SSI call to retrieve all output for a given destination or external writer.

xxxxxxxx

Specifies a JES destination name of up to eight alphanumeric characters.

Default: None

Notes:

- For many JES releases, the JES destination name must be defined in the JES parameters. Otherwise, an error message is issued by CA XCOM Data Transport because of a failed dynamic allocation.
- This parameter and the WRITER parameter are mutually exclusive. They cannot both be specified in the same CA XCOM Data Transport control library member.

DOMAIN

Identifies the Windows domain server used to validate the remote user ID and password.

xxxxxxxxxxxxxxx

Identifies the domain server used to validate the remote user ID and password. The name can contain up to 15 characters.

Default: None

Note: Used with transfers to Windows only.

DROPSESS

This parameter indicates whether CA XCOM Data Transport drops an LU-LU session at the conclusion of a scheduled file transfer.

YES

Indicates that CA XCOM Data Transport drops the session.

NO

Indicates that CA XCOM Data Transport does not drop the session.

QEMPTY

Indicates that CA XCOM Data Transport is to process all the transfers to a particular LU in the request queue before dropping the session.

ALL

Indicates that CA XCOM Data Transport drops all sessions, including SNASVCMG, at the conclusion of a scheduled file transfer. If all transfers for the particular LU in the request queue have finished.

<timeout value>

Indicates that if there is no activity on the session for the specified time interval, the session is dropped. This includes the SNASVCMG session. Valid values for the timeout interval are 1-60 (Seconds).

Default: The value that DROPSESS has in the Default Options Table.

Notes:

- CA XCOM Data Transport for VAX and some CA XCOM Data Transport for UNIX products do not support z/OS-initiated session establishment. Therefore, DROPSESS has no effect when the target of the transfer request is one of these platforms.
- DROPSESS=ALL is only used for infrequently used SNA partners to avoid potential problems with SNASVCMG sessions being dropped and possibly established simultaneously with heavy volume of transfers. DROPSESS=ALL is similar in function to DROPSESS=QEMPTY but drops the SNASVCMG session as well.

DSNTYPE

Specifies the data set definition.

Note: This parameter applies only to mainframe SMS data sets.

LIBRARY

Defines a PDSE

PDS

Defines a partitioned data set

BASIC

Defines a legacy sequential dataset.

LARGE

Defines a large format sequential dataset.

EXTREQ

Defines an extended format dataset.

EXTPREF

Specifies an extended format is preferred. If the extended format is not possible, a basic format will be used.

Note: These values are IBM standards for SMS processing.

Range: One to eight characters

Default: None

FERL

Specifies the number of times CA XCOM Data Transport is to retry a transfer after certain file errors or file allocation errors have occurred.

0

Specifies that CA XCOM Data Transport should not attempt to retry a transfer after the first file allocation or other file error.

1 to 254

Specifies the number of times CA XCOM Data Transport is to retry a transfer after encountering file and file allocation errors.

255

Specifies that CA XCOM Data Transport should retry the transfer indefinitely.

Default: 255

Note: CA XCOM Data Transport retries a transfer at the interval specified by the ERRINTV parameter. Because the FERL parameter specifies the number of retries, the transfer is attempted FERL+1 times (that is, the first attempt is not included in the count).

GETSESS

Specifies how CA XCOM Data Transport establishes a session with a remote LU.

YES

Indicates that the CA XCOM Data Transport server attempts session establishment with a remote LU as soon as the request for a transfer has arrived.

NO

Indicates that the CA XCOM Data Transport server is to wait for the operator to manually activate the LU through CA XCOM Data Transport's console command ACTIVATE or for the remote LU's attempt to establish a local LU session.

Default: The value that GETSESS has in the Default Options Table

Note: CA XCOM Data Transport for VAX and some CA XCOM Data Transport for UNIX products do not support z/OS-initiated session establishment. Therefore, GETSESS has no effect when the target of the transfer request is one of these platforms.

GROUP

This parameter can specify either of the following:

- One or more groups of LUs
- An alias for a single IP address

xxxxxxxx

Specifies *one* of the following:

- The name of a group of LUs. The name can contain up to eight alphanumeric characters.
- An IP address or IP name

Default: None

Notes:

- The GROUP parameter allows the multi-threading of file transfers to nodes that support multiple logical units but not parallel sessions.
- When a transfer to a group is requested, CA XCOM Data Transport attempts to initiate a session with each LU in the group until it succeeds in establishing a session with one of them. CA XCOM Data Transport will then use that LU as the partner node to send the file to the remote system.
- You can specify GROUP instead of an LU parameter in the SYSIN01 or TSO/ISPF panel when scheduling a transfer to a group of one or more LUs.
- If you specify GROUP, the GROUP name must match the name of a CA XCOM Data Transport control library member.

Example 1

The following is an example of how to use the GROUP parameter to set up an alias for an IP name or IP address.

- In the DEST member:

```
IPNAME=ipaddress (or ipname)
GROUP=HOST1
```

- In the JCL:

```
GROUP=HOST1
```

Example 2

The following is an example of how to use the GROUP parameter with a group of LUs.

- In the DEST member:

```
LU=LU1,LU2,LU3
GROUP=NJLUS
```

- In the JCL:

```
GROUP=NJLUS
```

IPNAME

Identifies the IP name or address of the remote system for a TCP/IP transfer.

1 to 64 alphanumeric characters

Specifies the name or address of the remote TCP/IP system involved in a transfer. This name can contain up to 64 alphanumeric characters and it must be one that has been defined to the domain name server. The address can be in IPv4 or IPv6 notation.

IPPORT

Specifies the default TCP/IP target port.

1 to 65535

Specifies the default TCP/IP target port used when IPPORT is omitted.

Default: 8044

Note: This parameter is used for all supported types of IP transfers: IPv4, IPv6, IPv4 SSL, and IPV6 SSL. The IPPORT specified here must match the correct listening port of the target system. For example, SSL transfers should specify an IPPORT that matches the remote system's SSL listening port.

LCLNTFYL

Specifies the local notification level for transfers initiated from the CA XCOM for z/OS server.

A (All)

Notify on transfer completion.

W (Warn)

Notify only if the transfer received a warning or error.

E (Error)

Notify only if the transfer received an error.

Default: A

You can specify this parameter in the XCOMDFLT table, in the destination member, or in the SYSIN01. Its presence is checked for first in the SYSIN01, then in the destination member, and lastly in the XCOMDFLT default table.

LIBNEG

Specifies whether multiple members of a source PDS can be received in a sequential data set on the target.

YES

Specifies that multiple members of a PDS are mapped into a sequential data set on the target.

NO

Specifies that a multi-member PDS cannot be received in a sequential data set on the target.

Default: YES

Note: The initiating CA XCOM Data Transport system examines the LIBNEG parameter when it determines that the source data set is a library and the target data set is sequential. If LIBNEG=YES, the data from the members of the library is written to a target sequential data set. The target data set does not contain any indication that the original source data set was structured as a library. If LIBNEG=NO, the transfer terminates with an error.

LOGMODE

Specifies the name of the VTAM mode entry that CA XCOM Data Transport will use to initiate a session to the remote system or group.

xxxxxxxx

Specifies a mode entry name. The name can contain up to eight alphanumeric characters.

Default: The default VTAM logon mode definition or the CA XCOM Data Transport Default Options Table value.

Notes:

- LOGMODE is used when DLOGMOD=XCOM is coded in the Default Options Table.
- For more information about setting up VTAM logmodes for CA XCOM Data Transport, see Define the Logon Mode Table Entries in the chapter "Configuring and Customizing Your Product."

LOSERS

Indicates the default number of contention loser sessions for LUs supporting parallel sessions.

0 to 127

Specifies the number of contention loser sessions.

Default: The CA XCOM Data Transport Default Options Table value or CONFIG member.

Note: This parameter is used only when PARSESS=YES is coded.

LU

Specifies up to 16 LU names.

***xxxxxxxx*$_1$, . . ., *xxxxxxxx*$_{16}$**

Specifies the name of an LU. The name can contain up to eight alphanumeric characters.

Default: None

Notes:

- The CA XCOM Data Transport control library data set consists of fixed-block records of length 80 bytes, and CA XCOM Data Transport does not permit their continuation past this limit. If a single 80-byte-long LU statement cannot accommodate all of the LU names that need to be specified, multiple LU statements may be used. For example, to specify 16 LUs, you could employ a separate LU statement for each LU; or, you could use two LU statements, each specifying the same or a different number of LUs, and so on.
- For a single LU destination member, the LU name has to match a member name.

Example

The following fragment from a destination member specifies six LUs:

```
. . .
LU=LU2310,LU2319,LU2200
LU=LUKXT,LUSAS1
LU=LU100
. . .
```

You can also specify the six LUs with a single LU statement:

```
. . .
LU=L2310,LU2319,LU2200,LUKXT,LUSAS1,LU100
. . .
```

MAXPACK

This parameter specifies the maximum packing length in bytes when PACK=LENGTH parameter is specified.

2048 to 31744 (bytes)

Specifies the maximum packing length (record packing buffer size) when the PACK=LENGTH parameter is specified.

Default: 2048 (bytes)

Notes:

- The use of PACK=LENGTH and the target buffer size is recommended to improve file transfer performance.
- In order to utilize zIIP processors for data compression, we recommend specifying a packing length of at least 4096.

MGMTCLAS

Specifies the name of the management class to use when allocating a new SMS-managed data set.

xxxxxxxx

Specifies the one- to eight -character management class name to use when allocating a new SMS-managed data set.

Note: This parameter applies only to mainframe SMS data sets.

Default: None

MODEL

Specifies the name of a CA XCOM Data Transport control library member that is used as a prototype of another CA XCOM Data Transport control library member.

xxxxxxxx

Specifies the name of the CA XCOM Data Transport control library member that serves as the model of another CA XCOM Data Transport control library member.

Default: None

Notes:

- A parameter defined in the prototype member may be set to a different value in the member built after the prototype. In that case, the parameter's definition in the non-prototype member will override the parameter's definition in the prototype member.
- The member specified in the MODEL parameter may not itself contain a MODEL parameter.

NEWDEST

Specifies the value that CA XCOM Data Transport should pass to the remote system as a destination for PSO transfers.

DEST

Specifies that CA XCOM Data Transport will pass the original JES destination to the remote system.

NONE

Specifies that CA XCOM Data Transport will pass no destination to the remote system.

WRITER

Specifies that CA XCOM Data Transport will pass the original JES writer name to the remote system.

xxxxxxxx

Specifies a character constant value that CA XCOM Data Transport will pass to the remote system. This value can be 1 to 21 characters in length.

Default: NONE

NEWWTR

Specifies the value CA XCOM Data Transport should pass to the remote system as a writer for PSO transfers.

DEST

Specifies that CA XCOM Data Transport will pass the original JES destination to the remote system.

NONE

Specifies that CA XCOM Data Transport will pass no destination to the remote system.

WRITER

Specifies that CA XCOM Data Transport will pass the original JES writer name to the remote system.

xxxxxxxx

Specifies a character constant value of length one to eight characters that CA XCOM Data Transport will pass to the remote system.

Default: NONE

PACK

This parameter indicates whether record-packing is used and it can substantially improve performance.

CRLF

Specifies that carriage returns and line feed characters is inserted at the end of each record. All systems use this type of record packing.

LENGTH

Specifies that the records are packed into fixed-size data transfer blocks. Each record begins with a 2-byte long prefix that indicates the length of the record and determines how many records can be packed into a block. The default block size is 2 K but it can be increased to 31 K by using the MAXPACK parameter in the CA XCOM Data Transport Default Options Table or in the CA XCOM Data Transport control library member.

When transferring a text file to a partner that supports record separators (for example, CRLF (0D0A) on Windows and LF (0A) on UNIX). The record separators are inserted. No separators are inserted in binary files that are transferred with PACK=LENGTH.

NO

Specifies that no record packing is used.

Default: NO

Notes:

- PACK=LENGTH with MAXPACK=31744 is selected when using CA XCOM Data Transport over TCP/IP. If PACK=LENGTH is coded, then you must also code RECSEP=NO.
- Platforms that support PACK=LENGTH are IBM AS/400, UNIX-based systems, and z/OS systems. See the appropriate documentation for the platform to see if this value is supported.
- For more information, see Pack Data Records in the chapter The Menu Interface (TSO/ISPF Panels) in the *CA XCOM Data Transport for z/OS User Guide*.
- For Unicode-based transfers (CODE=UTF8 or CODE=UTF16), PACK=LENGTH is enforced for the transfer. An informational message is issued when a transfer is modified to use PACK=LENGTH. This is for performance considerations in performing data conversion.

PARSESS

Specifies whether parallel sessions are allowed when CA XCOM Data Transport's VTAM application is sending a BIND to start communication with a remote LU.

YES

Specifies that parallel sessions are allowed.

NO

Specifies that parallel sessions are not allowed.

Default: NO

Note: Direct (TYPE=EXECUTE) transfers do not take advantage of the optimization required for parallel sessions.

PRPACE

Specifies the window size (in RUs) for pacing done between the primary logical unit and the boundary VTAM or NCP node. Using PRPACE substantially improves performance.

1 to 63

Specifies the pacing window size.

Default: The PRPACE value derived from the VTAM logmode table entry used by VTAM during initial session establishment and passed to CA XCOM Data Transport via the LOGON exit CINIT or SCIP exit bind RU.

Notes:

- A PRPACE value of 7 is recommended.
- PRPACE is equivalent to the VTAM VPACING operand.

PSOCKPT

Specifies the interval for checkpoints taken during a PSO transfer.

0 to 9999 records

Specifies the number of PSO records to transfer before a checkpoint is taken.

Default: The value for PSOCKPT in the Default Options Table.

Notes:

- The checkpoint/restart facility resumes interrupted PSO transfers from the point at which the most recent checkpoint was taken.
- The smaller the checkpoint interval, the greater the effect on the throughput, due to frequent checkpointing. Setting this parameter less than 10 would severely degrade performance. Usually values from 100 to 1000 are sufficient. Set this parameter to 0 if you do not want to do checkpointing.

PSODISP

This parameter does *both* of the following:

1. Specifies the disposition of a PSO data set if CA XCOM Data Transport is unable to successfully complete a PSO transfer.
2. Sets the DISP flag in a PSO transfer. DISP determines whether to keep or delete the report file after printing on the remote system. This value is ignored when the remote system is an IBM mainframe.

DELETE

1. Deletes the PSO data set after an unsuccessful PSO transfer.
2. Deletes the report file after it is printed on the remote system.

KEEP

1. Keeps the PSO data set after an unsuccessful PSO transfer.
2. Keeps the report file after it is printed on the remote system (by removing the temporary files).

Defaults: KEEP

Notes:

- If PSODISP=DELETE, manual intervention is required to requeue the transfer to CA XCOM Data Transport.
- If PSODISP=KEEP, manual intervention may be required to reclaim space on the remote system.

PSOPASS

Specifies the password included with Process SYSOUT reports that are sent to the remote system.

xxxxxxxx

Specifies a password of up to 31 alphanumeric characters.

Default: None

Notes:

- The password is included in the SNA Function Management Header-5 that actually begins the file transfer process.
- The use of the PSOPASS parameter is required only when ACCSEC=YES or when the remote system requires a user ID and password on a report transfer.

PSOPREF

Specifies the high-level qualifier used by the PSO interface when allocating temporary data sets. This value is also used by PDSE program library transfers when creating temporary data sets.

XCOMPSO

Specifies the high-level qualifier XCOMPSO.

Up to 20 alphanumeric characters

Specifies a high-level qualifier other than XCOMPSO. The high-level qualifier can contain up to 20 alphanumeric characters. You can specify multiple high-level qualifiers, up to 20 characters. The prefix must follow MVS naming conventions.

Default: XCOMPSO

PSOUSER

Specifies the user ID included with Process SYSOUT reports that are sent to the remote system.

xxxxxxxx

Specifies a user ID of up to eight alphanumeric characters.

Default: None

Notes:

- The user ID is included in the SNA Function Management Header-5 that actually begins the file transfer process.
- The use of the PSOUSER parameter is required only when ACCSEC=YES or when the remote system requires a user ID on a report transfer.

PSOWAIT

Specifies whether CA XCOM Data Transport is allowed to skip several iteration cycles for the current destination before scanning the JES queues for work

YES

Indicates that CA XCOM Data Transport is to check the JES queues for Process SYSOUT (PSO) data only if a session exists for one of the LUs in the group.

NO

Indicates that CA XCOM Data Transport is to check the JES queues on every cycle.

Default: NO

Notes:

- Coding PSOWAIT=YES can save significant CPU cycles when many entries are coded in the CA XCOM Data Transport control library (CAI.CBXGPARM). Coding PSOWAIT=YES reduces system overhead if the control library contains more 100 PSO-type nodes.
- As a general guideline, code PSOWAIT=NO if the remote LU is on a leased line.

PSPACE

Specifies the window size (in RUs) for pacing done between the secondary logical unit and the boundary VTAM or NCP node.

1 to 63

Specifies the pacing window size.

Default: The PSPACE value derived from the VTAM logmode table entry used by VTAM during initial session establishment and passed to CA XCOM Data Transport via the LOGON exit CINIT or SCIP exit bind RU.

Notes:

- A PSPACE value of 7 is recommended.
- The use of PSPACE can substantially improve performance.
- PSPACE is equivalent to the VTAM VPACING operand.

RECSEP

Specifies whether record separators are added to the data-delimiting records when they are written.

YES

Specifies that record separators are added to the data-limiting records.

NO

Specifies that record separators are not added to the data-limiting records.

Default: YES

Notes:

- The RECSEP parameter is ignored if CODE=BINARY is specified.
- As a rule, RECSEP is not applicable if both the sending and receiving systems are EBCDIC.

RELEASE

Specifies whether the remote partner is to release unused DASD space when creating a new file. You can set RELEASE in destination members, in the XCOMDFLT table, and as a SYSIN01 parameter.

YES

The remote partner is to release unused DASD space.

The unused DASD space that is specified for the transfer is released when the file is closed at the end of the transfer.

NO

The remote partner is not to release unused DASD space.

RMTNTFYL

Specifies the remote notification level for transfers initiated from the CA XCOM for z/OS server.

A (All)

Notify on transfer completion.

W (Warn)

Notify only if the transfer received a warning or error.

E (Error)

Notify only if the transfer received an error.

Default: A

You can specify this parameter in the CONFIG member, in the destination member, or in the SYSIN01. Its presence is checked for first in the SYSIN01, then in the destination member, and lastly in the CONFIG member.

RRUSIZE

Specifies the maximum RU size to be received from other LUs.

128 to 65536

Specifies the maximum RU size.

Default: The RRUSIZE value derived from the VTAM logmode table entry used by VTAM during initial session establishment and passed to CA XCOM Data Transport via the LOGON exit CINIT or SCIP exit bind RU.

Notes:

- This parameter is relevant only to sessions where the system is the primary logical unit.
- Using larger values can substantially improve performance on high-speed links, in installations with channel adapters, and 3088s. The secondary logical unit (the BIND receiver) determines what RU sizes are to be used on a session and can override any RU sizes sent in the BIND by the primary logical unit. If you want to use larger RU sizes, read the appropriate sections of the IBM VTAM and NCP manuals relating to the BFRS, MAXDATA, and TRANSFR parameters. These parameters require modifications to accommodate larger sizes.

SECURE_SOCKET

Specifies whether to use an OpenSSL socket or non-OpenSSL socket for transfers.

YES

Performs a secure transfer. The transfer uses an OpenSSL socket and must connect to an SSL listener on the remote partner.

NO

Performs a non-secure transfer. The transfer uses a non-OpenSSL socket and must connect to a non-SSL listener on the remote partner.

Default: NO

SERL

Specifies the number of times CA XCOM Data Transport tries to establish a session with the partner LU after the first attempt at session establishment has failed. Used for SNA transfers only.

0

Specifies that CA XCOM Data Transport does not attempt session establishment after the first session establishment error.

1 to 254

Specifies the number of retries.

255

Specifies that CA XCOM Data Transport retries session establishment indefinitely.

Default: 255

Notes:

- CA XCOM Data Transport retries a transfer at the interval specified by the ERRINTV parameter. Because SERL specifies the number of retries, the transfer is attempted SERL+1 times (that is, the initial attempt to establish a session is not included in the count).
- For group transfers, a session establishment error is considered to have occurred only after attempts at session establishment with *all* LUs in the group have failed.
- For example, if a group contains three LUs and SERL=2 is specified, CA XCOM Data Transport must try session establishment with each LU in the group before the session establishment error count is incremented. If all three attempts fail, the error count is set to 1 and CA XCOM Data Transport retries session establishment with each LU (this is the first retry). If the three session establishment attempts fail again, the error count is set to 2 and all three LUs are retried (this is the second retry). If all three fail this time, too, CA XCOM Data Transport stops making further session establishment attempts because the session establishment retry limit (SERL) has been reached.

SETUP

Specifies whether CA XCOM Data Transport should pass the SYSOUT print and class values from the JES queue to the remote system.

YES

Indicates that either the JCL parameter values specified by the user or the JES-assigned default values are passed.

For information about coding SETUP=YES with a VAX system, see the Notes section that follows.

NO

Indicates that print class and form values of binary zeroes (nulls) are passed to the remote system.

For information about coding SETUP=NO with a VAX system, see the Notes section that follows.

Default: YES

Notes:

- The SETUP parameter is referenced only for transfers that use the Process SYSOUT interface.
- Specify SETUP=YES only if you are sure that all the form names and print class values passed to the remote system are valid on the remote system.
- Specify SETUP=NO if the Process SYSOUT interface is being used to communicate with a VAX system. VAX systems require every form name used to be predefined; invalid form names will cause the print job to ABEND. In a case where the VAX site does require special forms, specify SETUP=YES, ensuring that all the z/OS form names and print classes have been predefined to the partner VAX.

SRPACE

Specifies the secondary receive pacing count in RUs that CA XCOM Data Transport puts into the BIND request.

1 to 63

Specifies the secondary receive pacing count.

Default: 5 (RUs)

Notes:

- The secondary receive pacing count sets the pacing between the boundary NCP (or VTAM) and the peripheral logical unit. It is equivalent to the PACING operand on the NCP definition macros.
- This parameter can be overridden by the secondary logical unit.
- If you specify SRPACE=0, the CA XCOM Data Transport VTAM application program will not override this value in the BIND.

SRUSIZE

Specifies the message size limit in the BIND request for request units sent on the session.

128 to 65536

Specifies the message size.

Default: The value derived from the VTAM logmode table entry used by VTAM during initial session establishment and passed to CA XCOM Data Transport via the LOGON exit CINIT or SCIP exit bind RU.

Notes:

This parameter is relevant only to sessions where the system is the primary logical unit.

- In general, remote lines use small RU sizes and channel-attached devices use large RU sizes. High-speed remote links (for instance, T1 lines) usually benefit from large RU sizes.
- SRUSIZE can be overridden by the secondary logical unit.

SSPACE

Specifies the pacing window size in RUs for messages sent by the secondary LU.

1 to 63

Specifies the pacing window size.

Default: 5 (RUs)

Notes:

- There is no VTAM or NCP operand corresponding to CA XCOM Data Transport's SSPACE parameter.
- Use care in overriding the BIND parameters suggested by VTAM.

STORCLAS

Specifies the name of the storage class for a new SMS-managed data set.

xxxxxxxx

Specifies the one- to eight-character storage class name to use when allocating a new SMS-managed data set.

Note: This parameter applies only to mainframe SMS data sets.

Default: None

SWAIT

Specifies the number of seconds that CA XCOM Data Transport waits for a session to be established after the request for session establishment has been queued.

1 to 32767

Specifies the time limit in seconds within which a queued session establishment request must result in the establishment of a session.

Default: 30 (seconds)

Note: If CA XCOM Data Transport's first attempt at session establishment fails, a request for session establishment is placed in a request queue. A session must be established within the time specified by the SWAIT parameter. If no session is established within the specified time, a session establishment error is considered to have occurred, which results in incrementing the value of the SERL parameter (see the description of SERL).

TCPRTIME

Specifies in seconds the maximum time that CA XCOM Data Transport waits for a TCP/IP response during a TCP/IP receive function

1 to 99999

Specifies in seconds the time that CA XCOM Data Transport is to wait for a response during a TCP/IP receive function.

Default: The value specified for TCPRTIME in the Default Options Table.

Notes:

- The session may also be timed-out by the TCP/IP stack after the Keep Alive Time period expires. The Keep Alive Time is defined by the TCP/IP stack. For more information on setting Keep Alive Time, see your TCP/IP documentation.
- The TCPRTIME and TIMEOUT parameters are mutually exclusive.

TCPSESS

Specifies the maximum number of TCP/IP file transfers that the server can perform concurrently to any one IP address.

0

Indicates that no maximum limit has been set.

1 to 128

Specifies the maximum number of concurrent transfers the server can perform to any one IP address.

Default: 15

Notes:

This parameter should be specified to throttle TCP/IP file transfers.

- This parameter is for TYPE=SCHEDULE transfers only.
- The value specified for this parameter should be less than the value specified in the Default Options Table. If the specified value is greater than the value specified in the Default Options Table or is not specified, the value defaults to the TCPSESS value specified in the Default Options Table. The MODIFY command LIST will list this value as maximum=*nnn*. If there is an asterisk appended onto this LIST item, the value for TCPSESS was taken from the Default Options Table.

TERL

Specifies the number of file allocation errors, VTAM errors, and retries during TCP/IP session establishment that CA XCOM Data Transport will allow before terminating the transfer.

0

Specifies that no errors are retried (regardless of the settings of FERL, SERL, or VERL).

1 to 32766

Specifies the total number of times that CA XCOM Data Transport attempts to correct errors.

32767

Specifies that errors are retried indefinitely.

Default: 32767

Notes:

- If the number of errors for a particular category of retryable errors (FERL, SERL, VERL) or any combination of such categories is equal to the value of TERL, CA XCOM Data Transport stops further retry attempts and fails the transfer. For example, suppose that TERL=10, FERL=6, SERL=6, and VERL=4. If CA XCOM Data Transport has retried file allocation 5 times and session establishment 5 times, the total error retry limit (TERL) has been reached. Therefore, CA XCOM Data Transport makes no further error retry attempts, although the individual error retry limits have not been reached.
- TERL=32767 (that is, indefinite number of error retries) does not override a value of FERL, SERL, or VERL that indicates a finite number of retries (such as any value greater than 0 but smaller than 255). For example, if TERL=32767 and SERL=5, CA XCOM Data Transport stops retrying session establishment errors after the fifth error has occurred, instead of continuing indefinitely.

TIMEOUT

Specifies in seconds the maximum duration of time that CA XCOM Data Transport waits for a TCP response before aborting a session with a TCP/IP partner for a TCP/IP dest member. Specifying a TIMEOUT=*value* in the DEST member for a TCP/IP partner activates a SELECT prior to the RECEIVE, which allows CA XCOM Data Transport to timeout.

1 to 99999

Specifies in seconds the length of time that CA XCOM Data Transport is to wait for a TCP response.

Default: The TIMEOUT value specified in the Default Options Table.

Notes:

- If the TIMEOUT parameter is specified for a SNA DEST member, it is accepted but ignored. SNA transfers will always use the TIMEOUT value specified in the Default Options Table.
- For TCP dest members, if the TIMEOUT limit is reached and no response was received from TCP, the transfer will be aborted. It will be subjected to the VERL retry rules if specified. Proper use of this parameter ensures that CA XCOM Data Transport does not hang indefinitely waiting for a TCP response.
- The TIMEOUT and TCPRTIME parameters are mutually exclusive.

TRUSTID

This parameter must be coded to allow incoming TRUSTED transfers from this remote destination member. Each user ID on the remote destination that needs to do TRUSTED transfers must have a TRUSTID entry coded.

```
TRUSTID=USERID{,GROUPID}
```

USERID

The user ID (1 to 12 characters) for which a trusted transfer is to be performed.

GROUPID

The group ID (1 to 8 characters) for which a trusted transfer is to be performed.

Note: If the GROUPID is omitted, it is assumed to have the same value as the USERID.

Default: None

Notes:

- For a sample destination member with TRUSTID, see Implement Trusted Access Security for Transfers to z/OS in the chapter "Security Considerations."
- The data set specified by XCOMCNTL (default CAI.CBXGPARM) must be secured (that is, password protected) to prevent unauthorized users from getting trusted access.
- The LIST command does not show the members in the TRUSTID list; it only shows that TRUSTID is present in the destination member.
- The TRUSTID and GROUPID parameters are not case sensitive.
- The TRUSTID and GROUPID parameters cannot be used with TYPE=LIST destination members.

TYPE

Specifies the type of the destination being defined in the CA XCOM Data Transport control library member.

DEST

Specifies that the destination defined in the given CA XCOM Data Transport control library member is a single LU or a group of LUs.

Default: None

Notes:

- This parameter must always be coded as TYPE=DEST for destinations that consist of a single LU or a group of LUs.
- The TYPE parameter must be the first non-comment card to appear in the CA XCOM Data Transport control library member.

VERL

Used by TCP/IP to determine the number of times CA XCOM Data Transport will attempt to retry a file transfer that has received a VTAM error.

0

Specifies that the transfer is not retried.

1 to 254

Specifies the number of retry attempts.

255

Specifies that the transfer is retried indefinitely.

Default: 255

Note: CA XCOM Data Transport retries a transfer at the interval specified by the ERRINTV parameter. Because this parameter specifies the number of retries, the transfer will be attempted VERL+1 times (that is, the initial attempt to establish a session is not included in the count).

WINNERS

Specifies the maximum number of contention winners for parallel session partners.

0 to 127

Specifies the maximum number of contention winners.

Default: The CA XCOM Data Transport Default Options Table value or CONFIG member.

Note: The WINNERS parameter is used when an LU is defined as PARSESS=YES.

WRITER

Specifies a JES writer name to be used as a search argument to retrieve output from the JES spool.

xxxxxxxx

Specifies a JES writer name of up to eight alphanumeric characters.

Default: None

Notes:

- This parameter and the DEST parameter are mutually exclusive. They cannot both be specified in the same CA XCOM Data Transport control library member. CA XCOM Data Transport handles the WRITER parameter in exactly the same manner as the DEST parameter.
- CA XCOM Data Transport does not verify if the writer is a valid JES writer, it is not necessary to predefine the writer locally.

XCOM_CONFIG_SSL

Specifies the HFS SSL configuration file path and file name.

1 to 256 characters

Specifies the HFS path and file name of the SSL configuration file used by CA XCOM Data Transport for secure transfers.

Note: A sample SSL configuration file, configssl.cnf, is provided with the installation.

Default: None

List Destination Parameters

This section describes the parameters for list destinations.

GROUP

This parameter can specify either of the following:

- One or more groups of LUs
- An alias for a single IP address

xxxxxxxx

Specifies *one* of the following:

- The name of a group of LUs. The name can contain up to eight alphanumeric characters.
- An IP address or IP name

Default: None

Notes:

- When used with a group of LUs, this parameter allows the multi-threading of file transfers to nodes that support multiple logical units but not parallel sessions.
- When a transfer to a group is requested, CA XCOM Data Transport attempts to initiate a session with each LU in the group until it succeeds in establishing a session with one of them. CA XCOM Data Transport then uses that LU as the partner node to send the file to the remote system.
- You can specify multiple groups (member names) by using a single GROUP statement or multiple GROUP statements. However, you *must* use multiple GROUP statements, if the groups to be specified do not all fit in the same 80-byte-long fixed-block record (the record format of the CA XCOM Data Transport control library data sets). The multiple groups specified on a GROUP statement must be separated with commas.
- The number of different destinations (GROUPs, IPNAMEs, LUs) specified in a list can be more than 500.
- When used with an IP name or address, this parameter specifies an alias.
- The following is an example of how to use the GROUP parameter to set up an alias for an IP name or IP address.
  - In the DEST member:

    ```
    IPNAME=ipaddress (or ipname)
    GROUP=HOST1
    ```

  - In the JCL:

    ```
    GROUP=HOST1
    ```

IPNAME

Identifies the IP name or address of the remote system for a TCP/IP transfer.

1 to 64 alphanumeric characters

Specifies the name or address of the remote TCP/IP system involved in a transfer. This name can contain up to 64 alphanumeric characters and it must be one that has been defined to the domain name server. The address can be in IPv4 or IPv6 notation.

Notes:

- You can specify multiple IP addresses by using a single IPNAME statement or multiple IPNAME statements. However, multiple IPNAME statements *must* be used, if the IP addresses to be specified do not all fit in the same 80-byte-long fixed-block record (the record format of the CA XCOM Data Transport control library data sets). Each of the multiple IP addresses specified on an IPNAME statement must be separated with a comma.
- The number of different destinations (GROUPs, IPNAMEs, LUs) specified in a list can be more than 500.
- The DOMAIN parameter is not supported in a LIST destination member. The DOMAIN is picked up from the individual destination members, if any, or from the Default table.

IPPORT

Specifies the default TCP/IP target port.

1 to 65535

Specifies the default TCP/IP target port used when IPPORT is omitted.

Default: 8044

Notes:

- This parameter must match the SERVPORT specification of the target server and normally should not be changed.
- The coding of IPPORT is optional. It cannot be used independently of the IPNAME parameter in a list destination definition. If IPPORT is coded, it applies to the following IPNAME(s) up to the next specification (if any) of IPPORT.

LU

Specifies *one or more* LU names.

xxxxxxxx

Specifies the name of an LU. The name can contain up to eight alphanumeric characters.

Default: None

Notes:

- You can specify multiple LU names by using a single LU statement or multiple LU statements. However, multiple LU statements *must* be used, if the LU names to be specified do not all fit in the same 80-byte-long fixed-block record (the record format of the CA XCOM Data Transport control library data sets). Each of the multiple LU names specified on an LU statement must be separated with a comma.
- The number of different destinations (GROUPs, IPNAMEs, LUs) specified in a list can be more than 500.

TYPE

Specifies the type of the destination being defined in the CA XCOM Data Transport control library member.

LIST

Specifies that the destination defined in the given CA XCOM Data Transport control library member is a list of LUs.

Notes:

- TYPE=LIST must be coded in all list destination definitions.
- The TYPE parameter must be the first non-comment card to appear in the CA XCOM Data Transport control library member.

Default: None

Superlist Destination Parameters

This section describes the parameters for superlist destinations.

TYPE

Specifies the type of the destination being defined in the CA XCOM Data Transport control library member.

SUPERLIST

Specifies that the destination defined in the given CA XCOM Data Transport control library member is a superlist.

Notes:

- TYPE=SUPERLIST must be coded in all superlist destination definitions.
- The TYPE parameter must be the first non-comment card to appear in the CA XCOM Data Transport control library member.

Default: None

LIST

This parameter specifies a list member defined in the CA XCOM Data Transport control library.

LIST=*listname*[,...[,*listname*]]

Notes:

- Each *listname* must have a member defined in the CA XCOM Data Transport control library.
- If each member name specified as a list or superlist is 8 characters long, a superlist can identify 2971 member names.
- Multiple listnames can be specified in a single statement or multiple statements can be employed. When a single statement is used to list multiple destinations, the destination must be separated by commas. For example, to specify a list of three list destinations (L1, L2, and L3) any of the following specifications can be used:
  - LIST=L1,L2,L3
  - LIST=L1,L2
    LIST=L3
  - LIST=L1
    LIST=L2
    LIST=L3

Default: None

Chapter 3: Security Considerations

This chapter discusses CA XCOM Data Transport security through various levels and interfaces.

This section contains the following topics:

Security Planning

Before implementing CA XCOM Data Transport for z/OS, a complete review of security issues should be made by those responsible for data security. This review should include those installing CA XCOM Data Transport, security administrators, auditors, and the systems staff involved in supporting the security software. Because of the expanded data access capabilities introduced by CA XCOM Data Transport, the security planning should address the need to define additional access privileges.

Consider the security environment at both ends when performing a CA XCOM Data Transport file transfer. Security validation for the local system is performed on the local system. Security validation for the remote system is performed on the remote system.

CA XCOM Data Transport software is subject to the same resident security checks as other z/OS applications. Therefore, the CA XCOM Data Transport started task needs a security profile that allows the full range of file transfers to be implemented; otherwise, even when the user has authorization to access a data set, a 913 abend code may be generated.

Security Checking

The security checking of CA XCOM Data Transport processing done by your resident security software falls into the following three categories.

Locally initiated file transfers

CA XCOM Data Transport checks if predefined user access privileges permit such processing on the local system. For example, for a send file transfer there must be access privileges to read the file being sent.

Remotely initiated file transfers

CA XCOM Data Transport checks whether the predefined security profile for the user ID/password specified by the remote user allows the requested processing on the local system. For example, for a send file transfer involving the update of a file, the resident security software will check whether that user ID has update access privileges to that file.

File transfers dependent on user logon

The CA XCOM Data Transport server can be configured to check for a user logon (for some or all LUs) before any file transfers can occur on those LUs. The remote partner (Windows family) must support this feature.

Overview of Security

CA XCOM Data Transport provides security for four different types of resources. The resources that can be secured are the files, data in the files, commands, and LUs. This section offers a brief characterization of the essential features of CA XCOM Data Transport security as it pertains to the various resources.

File Security

CA XCOM Data Transport makes calls to IBM RACF, CA Top Secret, and CA ACF2 to verify whether a given user ID is authorized to read or update a given data set.

Command Security

CA XCOM Data Transport makes standard SAF calls to determine whether a given user ID or console is authorized to issue CA XCOM Data Transport commands. The commands whose access status is verified include z/OS console commands and commands that can be issued through ISPF and CICS menu interfaces.

Partner Security

CA XCOM Data Transport makes standard SAF calls to determine whether a given user ID is authorized to perform transfers with a given partner. CA XCOM Data Transport checks the direction (send or receive) in which the transfers are performed is authorized. Also whether the partner making a transfer request can be the initiator of transfers.

Trusted Access Security

The Trusted Access feature allows a transfer to be sent to a remote partner without actually specifying the user ID and password in the transfer. Trusted Access transfers can be sent to and from z/OS, Windows, and UNIX partners.

History Database Security

When CA XCOM Data Transport is using an ODBC database to store history records, users must be granted privileges to the history table. These privileges allow the user to update and/or read from the table. Users that connect to the history database are checked for access rights to records in the table. The history database can be shared among multiple systems and platforms.

Invoking Security

Each security function can be turned on separately by way of the assembled Default Options Table in the CA XCOM Data Transport load library. Also, each security function has an associated user exit, which increases the flexibility of CA XCOM Data Transport's security and allows security to be adapted to site-specific conditions.

File Access Security

There are three levels of security checking for file transfers under CA XCOM Data Transport:

- User ID/password validation
- Data set access privileges for the specified user ID
- Optional file security user exit (XCOMEX05)

Validate the Indicated User ID/Password

The first level is initiated when CA XCOM Data Transport issues a call from its address space to the security software for validation of the indicated user ID/password.

Notes:

- CA XCOM Data Transport always protects the password thru the use of encryption. For a description of CA XCOM Data Transport password protection, see the section *Password Protection by Encryption*
- CA XCOM Data Transport can change passwords on the remote system. For a description of the password parameter, see the chapters The Menu Interface (TSO/ISPF Panels) and The Batch Interface in the *CA XCOM Data Transport for z/OS User Guide*.

Validate Data Set Access Privileges

If the user ID/password is valid, CA XCOM Data Transport goes to the second level of security checking. The CA XCOM Data Transport security interface passes to the resident security software the user ID and the name of the data set to be accessed. The security software checks whether access should be granted and passes the results back to the CA XCOM Data Transport security interface.

File Security User Exit (XCOMEX05)

If the first two stages of security checking are successful, the optional file security user exit (XCOMEX05) is then invoked. This allows for additional site-specific security validation.

Additional Security Considerations

Note the following:

- When CA XCOM Data Transport is run as a batch job (XCOMJOB) of type TYPE=EXECUTE, the external security manager performs the functions of the first two stages of security checking, as it would for any other batch job. Hence, the CA XCOM Data Transport standard security interface is not invoked. The file security user exit, however, is still invoked.
- The file security user exit is invoked when receiving reports, whereas the standard CA XCOM Data Transport security exit is not.

How to Use the File Security User Exit

To use the user security exit, specify EXIT05=YES in the CA XCOM Data Transport Default Options Table. In this case, the user must provide the XCOMEX05 module in the XCOMLOAD library (CAI.CBXGLOAD).

Partner Security

Partner LU security is concerned with controlling whether a given user is authorized to perform transfers with a particular partner LU or IP name or address. CA XCOM Data Transport implements partner LU security through the parameters FACILITY, LUSECURE and EXIT12, which are coded in the Default Options Table or Configuration member.

SAF Security Call—Partner Security

If LUSECURE=YES is coded in the Default Options Table, CA XCOM Data Transport makes a standard SAF call to the security package (CA ACF2, IBM RACF, or CA Top Secret) to determine whether the user is trying to initiate a transfer has READ authority to the class resource named in the security call. The security class queried is determined by the value of the FACILITY configuration parameter. The default resource class name used by CA XCOM Data Transport is FACILITY.

The format of the security call is as follows:

`XCOM.`*`applsec`*`.{LU|IP}.`*`destname`*`.{SEND|RECEIVE}.{L|R}`

The components of the security call are explained in the following table:

XCOM

The literal XCOM must be specified as the first element of every security call.

applsec

Specifies the value of the APPLSEC parameter in the Default Options Table, unless it is NONE, in which case the expression XCOM will appear in this position. This slot in the security call identifies the CA XCOM Data Transport server.

LU

The literal LU indicates that the partner is an SNA partner.

IP

The literal IP indicates that the partner is a TCP/IP partner.

destname

Specifies the destination name that is to participate in the transfer. It may be an SNA LU name, TCP/IP name or TCP/IP address. For file security, the Default Options Table parameter TCPLUSEC determines the format used to pass the TCP/IP address to the security system. For partner security using TCP/IP, the actual TCP/IP name or address as provided in the destination member or SYSIN01 is used as the destname for the security resource.

SEND

The literal SEND indicates the direction of the transfer from the server's point of view.

RECEIVE

The literal RECEIVE indicates the direction of the transfer from the server's point of view.

L

The literal L indicates a locally initiated transfer.

R

The literal R indicates a remotely initiated transfer.

When to Use Partner LU Security

The possible uses of the security scheme discussed above include the following:

- Restricting the users' access to a group of PCs
- Controlling the direction in which file transfers may be performed (send only, receive only)
- Delimiting an LU's ability to initiate local or remote transfers
- Securing a PC or an LU that contains sensitive information

Examples of SAF Security Calls—Partner Security

Suppose a PC user with the user ID JOE on an LU named BOBSPC requests a CA XCOM Data Transport server (with APPLSEC=PRODXCOM) to send a file to BOBSPC. In this case, when the CA XCOM Data Transport server calls the security package, the security call takes the following form:

```
XCOM.PRODXCOM.LU.BOBSPC.SEND.R
```

This is asking the security package to check if the user ID JOE possesses the required READ authority for the CLASS=<FACILITY> resource in the above security call, that is, is the requestor of the transfer authorized to perform transfers to the LU named BOBSPC?

The next example involves a batch job of type TYPE=EXECUTE requesting that the CA XCOM Data Transport server receive a file from a PC named JILLSPC. Assume that APPLSEC=BATCHXC is coded in the Default Options Table. In this case, CA XCOM Data Transport will ask the security package if the user ID coded on the JOB card in the JCL is authorized to request transfers from the LU JILLSPC. The form of the security call is as follows:

```
XCOM.BATCHXC.LU.JILLSPC.RECEIVE.L
```

Note: The transfer request is specified as being locally initiated.

Partner LU Security (XCOMEX12)

If LUSECURE=YES and EXIT12=YES, security authorization is handled via Exit 12. The security check by Exit 12 may have one of three outcomes (return codes):

- The user is granted access to the desired resource (RC=0).
- The user is denied access to the desired resource (RC=8).
- The decision as to the user's access rights is referred to the security package (RC=4).

Note: Exit12 does not consult the security package for a decision to grant or deny access.

LUSECURE=YES must be coded for EXIT12=YES to take effect.

If LUSECURE=YES and EXIT12=NO, the user's security authorization is decided by the security package.

If LUSECURE=NO, CA XCOM Data Transport does not check the EXIT12 parameter.

More Information About Partner Security

For a description of Exit12, see the appendix “User Exits” in the *CA XCOM Data Transport for z/OS User Guide*.

A sample Exit12 is provided in CAI.CBXGSAMP(XCOMEX12).

For descriptions of the LUSECURE, EXIT12, and TCPLUSEC parameters, see CA XCOM Data Transport Default Options Table Parameters in the chapter “Configuration Parameters.”

For information about how to specify an IP Address for security, see the description of the Default Options Table parameter TCPLUSEC.

Note: For TCP/IP partners, both the TCP/IP name and the TCP/IP address should be defined in the security database.

Command Security

Command security is concerned with controlling whether a given user is authorized to issue a given CA XCOM Data Transport operator command. CA XCOM Data Transport operator commands can be issued directly from the system console or indirectly via the ISPF and CICS menu interfaces. CA XCOM Data Transport implements command security through the parameters OPERSEC and EXIT13, which are coded in the Default Options Table.

SAF Security Call—Command Security

If OPERSEC=SAF is coded in the Default Options Table, CA XCOM Data Transport makes a standard SAF call to a security package (CA ACF2, IBM RACF, or CA Top Secret) to determine whether the user trying to issue a CA XCOM Data Transport operator command has the authority to issue that command. The user is authorized to issue a particular command if his security profile satisfies the access level defined for the class resource named in the security call. The security resource class queried is the value of the OPERCMDS configuration parameter. The default resource class for CA XCOM Data Transport is OPERCMDS. The general format of the security call is as follows:

XCOM.*applsec*.*command*[.*parameter*[...]]

The components of the security call are explained in the following table:

XCOM

The literal XCOM must be specified as the first element of every security call.

applsec

Specifies the value of the APPLSEC parameter in the Default Options Table, unless it is NONE, in which case the expression XCOM will appear in this position. This component of the security call identifies the CA XCOM Data Transport server.

command

Specifies the name of a CA XCOM Data Transport operator command.

parameter

Specifies the parameters associated with the operator command.

If more than one parameter is used with a command, the parameters must be separated with a period from each other.

Note: There are operator commands with which no parameters are used.

Operator Commands and Their Security Calls

Each CA XCOM Data Transport operator command is secured through a different security call. The following table lists the CA XCOM Data Transport operator commands, indicates their access levels, and specifies their complete resource names. Following the specification of each resource name is a brief description of any variables except *applsec* contained in the name. For a description of the *applsec* variable, see SAF Security Call in this chapter. The variables contained in the resource names are shown in italics.

Command	Access	Resource Names
ACTIVATE	UPDATE	XCOM.*applsec*.ACTIVATE.*destname* **Variable:** *destname* specifies the name of the LU to be activated.
ALTER	UPDATE	XCOM.*applsec*.DATE.*destname*.*ownername* XCOM.*applsec*.EPRTY.*destname*.*ownername* XCOM.*applsec*.SPRTY.*destname*.*ownername* XCOM.*applsec*.TIME.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.
CANCEL	UPDATE	XCOM.*applsec*.CANCEL.*destname.type* **Variables:** *destname* specifies the name of the destination for which a session is to be cancelled. *type* specifies a CANCEL command option, that is, IMMED, PURGE, or SUSPEND.
CNOS	UPDATE	XCOM.*applsec*.CNOS.*membername* **Variables:** *membername* specifies the name of the control library member containing the definition of the LU for which a CNOS conversation is attempted.
DELETE	UPDATE	XCOM.*applsec*.DELETE.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.

Command	Access	Resource Names
DFLT	UPDATE	XCOM.*applsec*.DFLT.*parmname* **Variable:** *parmname* specifies a certain parameter in the Default Options Table. For a list of valid parameters, see the chapter "Operation and Control" in the *CA XCOM Data Transport for z/OS User Guide*.
DISABLE	UPDATE	XCOM.*applsec*.DISABLE.*membername* **Variable:** *membername* specifies the name of the control library member to be disabled.
DISPLAY	READ	XCOM.*applsec*.DISPLAY.*destname* XCOM.*applsec*.DISPLAY.* **Variable:** *destname* specifies the name of the destination about which session information is to be displayed. The asterisk (*) indicates that information all active sessions is to be displayed.
DUMP	UPDATE	XCOM.*applsec*.DUMP.*destname* **Variable:** *destname* specifies the name of the destination for which dump data is to be produced.
DUMPXCF	UPDATE	XCOM.*applsec*.DUMPXCF **Variable**: *applsec* specifies the value of the APPLSEC parameter in the Default Options Table.
ENABLE	UPDATE	XCOM.*applsec*.ENABLE.*membername* **Variables:** *membername* specifies the name of the control library member to be enabled.
EXIT	READ	XCOM.*applsec*.EXIT.*data* **Variable:** *data* specifies the eight-byte-long user data to be passed to the XCOMEX09 exit routine.
HOLD	UPDATE	XCOM.*applsec*.HOLD.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.
INFO	READ	XCOM.*applsec*.INFO

Command	Access	Resource Names
INQ	READ	XCOM.*applsec*.INQ
LIST	READ	XCOM.*applsec*.LIST.*membername* **Variable:** *membername* specifies the name of the control library member the contents of which are to be displayed.
LOGFREE	UPDATE	XCOM.*applsec*.LOGFREE
NOTRACE	UPDATE	XCOM.*applsec*.NOTRACE.*destname* **Variable:** *destname* specifies the name of the LU name or the IP name for which the trace function is to be disabled.
NOXTRACE	UPDATE	XCOM.*applsec*.NOXTRACE **Variable:** *applsec* specifies the value of the APPLSEC parameter in the Default Options Table.
NSASTAT	READ	XCOM.*applsec*.NSASTAT
RELEASE	UPDATE	XCOM.*applsec*.RELEASE.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.
SAVE	UPDATE	XCOM.*applsec*.*SAVE*
RESUME	UPDATE	XCOM.*applsec*.RESUME.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.
RSHOW	READ	XCOM.*applsec*.RSHOW.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.
SHOW	READ	XCOM.*applsec*.SHOW.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.

Command	Access	Resource Names
SNAP	UPDATE	XCOM.*applsec*.SNAP
STAT	READ	XCOM.*applsec*.STAT **Variable:** *applsec* specifies the value of the APPLSEC parameter in the Default Options Table.
STOP	CONTROL	XCOM.*applsec*.STOP.*type* **Variable:** *type* specifies a STOP command option, that is, IMMED.
SUSPEND	UPDATE	XCOM.*applsec*.SUSPEND.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.
TERM	UPDATE	XCOM.*applsec*.TERM.*destname*.*ownername* **Variables:** *destname* specifies the name of the destination involved in the transfer. *ownername* specifies the user ID under whose security authorization the transfer is performed.
TRACE	UPDATE	XCOM.*applsec*.TRACE.*destname* **Variable:** *destname* specifies the name of the LU or the IP name for which a trace is to be produced.
VERSION	READ	XCOM.*applsec*.VERSION
XRSHOW	READ	XCOM.*applsec*.XRSHOW **Variable:** *applsec* specifies the value of the APPLSEC parameter in the Default Options Table.
XSHOW	READ	XCOM.*applsec*.XSHOW **Variable:** *applsec* specifies the value of the APPLSEC parameter in the Default Options Table.
XTRACE	UPDATE	XCOM.*applsec*.XTRACE **Variable:** *applsec* specifies the value of the APPLSEC parameter in the Default Options Table.

Command	Access	Resource Names
ZIIP	READ	XCOM.*applsec*.ZIIP.*STATUS*
	UPDATE	XCOM.*applsec*.ZIIP.*ENABLE*
	UPDATE	XCOM.*applsec*.ZIIP.*DISABLE*
		Variable: *applsec* specifies the value of the APPLSEC parameter in the Default Options Table.

For a detailed description of the CA XCOM Data Transport operator commands, see the chapter "Operation and Control" in the *CA XCOM Data Transport for z/OS User Guide*.

Examples of SAF Security Calls—Command Security

Suppose that a console operator issues an ENABLE command for the control library member BOBSPC to a CA XCOM Data Transport server that has APPLSEC=PRODXCOM in its Default Options Table. In this case, when the CA XCOM Data Transport server calls the security package, the resource name takes the following form:

```
XCOM.PRODXCOM.ENABLE.BOBSPC
```

CA XCOM Data Transport executes the ENABLE command if the security package finds the access level UPDATE defined in the security profile of the console operator issuing the command.

Next, suppose a user wants to use the ISPF menus to suspend an active transfer to LU25 running under the authority of the user ID TOM. Assume the server is started with APPLSEC=PRODXCOM. In this case, when the CA XCOM Data Transport server calls the security package, the resource name takes the following form:

```
XCOM.PRODXCOM.SUSPEND.LU25.TOM
```

Again, before CA XCOM Data Transport executes the SUSPEND command, the security package must determine that the access level UPDATE has been defined for the user ID TOM.

Command Security for Consoles That Are Not Logged On

When the CA XCOM Data Transport server receives a CA XCOM Data Transport command from a console that is not logged on with a specific user ID, the server examines the console flags of the console. It determines on the basis of the console flags whether the console has the authority to issue a specific command.

In general, commands that request information (access level READ) are allowed from any console. On the other hand, the STOP (XCOM) command, which requires the access level CONTROL, can be issued only from the master console. The commands at the access level UPDATE require that the console have SYS, I/O, or CONS authority. All of these restrictions can be overridden through a user-written User Exit13.

Command Security User Exit (XCOMEX13)

In addition to coding OPERSEC=SAF, you may also code EXIT13=YES or EXIT13=load-module-name in the CONFIG Member to enable User Exit13. This exit allows you to write your own command security routines and thereby fine tune the control of command security.

IF OPERSEC=SAF and EXIT13 is enabled, command security is handled through Exit 13. The security check by Exit 13 can have one of three outcomes (return codes):

- The command issuer may use the desired command (RC=0). This decision is made by Exit13 without consulting the security package.
- The command issuer may not use the desired command (RC=8). This decision is made by Exit13 without consulting the security package.
- The decision as to the user's right to issue a particular command is referred to the security package (RC=4).

OPERSEC=SAF must be coded for EXIT13 to be invoked.

If OPERSEC=SAF and EXIT13=NO, the user's authority to use operator commands is decided by the security package.

If SECURITY=NONE, CA XCOM Data Transport does not check the EXIT13 parameter.

For more information about User Exit13, see the appendix “User Exits” in the *CA XCOM Data Transport for z/OS User Guide*.

A sample Exit13 is provided in CAI.CBXGSAMP(XCOMEX13).

History Database Security

The user ID defined in the Default Options Table with parameter XCOMHIST_USER must be granted use of the history table defined with parameters XCOMHIST_TBL and XCOMHIST_OWNER.

With VSAM history files, each CA XCOM Data Transport server worked with its own history file. However, using a relational database to store CA XCOM Data Transport history records allows multiple CA XCOM Data Transport servers (including CA XCOM Data Transport systems running on Windows and UNIX) to share the database. So you need to be able to restrict access to rows in the database, so that a user on system A is not allowed to see history for system B unless the user is given explicit permission. To provide this level of security, CA XCOM Data Transport Command Security has been enhanced with an additional ALLHIST command resource.

CA XCOM Data Transport implements command security through the parameters OPERSEC and EXIT13, which are coded in the Default Options Table.

If OPERSEC=SAF is coded in the Default Options Table, CA XCOM Data Transport makes a standard SAF call to a security package (CA ACF2, IBM RACF, or CA Top Secret) to determine whether the user has access to the ALLHIST command resource. This resource, when permitted to a user, allows that user to view history records for any system that is maintaining history in that database. If the user is not permitted to this resource then the user is allowed to see history records for the system of the originating request only.

Command: ALLHIST

Access: READ

Resource Name: XCOM.applsec.ALLHIST

applsec

The identifier for the CA XCOM Data Transport server as defined in the Default Options Table, unless it is NONE, in which case the expression XCOM appears in this position. This component of the security call identifies the CA XCOM Data Transport server.

Note: If OPERSEC=NONE is coded in the Default Options Table, CA XCOM Data Transport runs with no security check, giving the user unrestricted access to view history records for any system that is maintaining history in that database.

This level of security is in addition to the current security provided by CA XCOM Data Transport.

Invoke a Security Interface

The CA XCOM Data Transport security interface is selected through the SECURITY parameter in the CA XCOM Data Transport Default Options Table. The following z/OS security interfaces are provided for CA XCOM Data Transport:

- CA ACF2
- CA Top Secret
- IBM RACF
- SAF

Important! A thorough knowledge of each security software package is necessary to use the following information.

CA ACF2 Interface

Select the CA XCOM Data Transport CA ACF2 interface by specifying SECURITY=ACF2 in the CA XCOM Data Transport Default Options Table, or the SECURITY parameter keyword in the started task's CA XCOM Data Transport EXEC statement PARM field of the started task (see the chapter “Configuring and Customizing Your Product”).

The CAI.CBXGJCL library contains a sample JCL procedure that can be used to link CA XCOM Data Transport's CA ACF2 interface module to the version of CA ACF2 that is running. Edit CAI.CBXGJCL(LINKACF2) to fit your particular installation. Next, submit the edited procedure as a job. Be sure to check for any unresolved references in the linkage editor output.

General CA ACF2 Requirements

The following requirements concern the use of CA ACF2:

- A currently supported release of CA ACF2 must be installed.
- You must have linked the CA XCOM Data Transport CA ACF2 interface module to your CA ACF2 version (see the chapter “Configuring and Customizing Your Product”).
- If CA XCOM Data Transport is running as a started task, the installation must have turned on the STC validation option. If CA XCOM Data Transport is running as a job, then STC validation need not be turned on, but CA XCOM Data Transport must be running from an authorized library and link edited as such.
- The CA XCOM Data Transport logon ID must be given the MUSASS privilege and the JOBFROM privilege.

CA ACF2 Interface Description

CA XCOM Data Transport provides an interface with the CA ACF2 security software. Security validation through CA XCOM Data Transport's CA ACF2 security interface occurs whenever the CA XCOM Data Transport started task is used for file transfer. The initial security checking validates the logon ID and the password combination for the system.

When this validation is completed, CA XCOM Data Transport sends these parameters to the CA XCOM Data Transport security interface:

- Logon ID of user requesting access
- Password of user requesting access
- New Password (optional) if password has been changed
- LU or TCP/IP name or address of the node that initiated the file transfer request
- Name of the data set the user wants to access
- Type of access requested:
  - READ
  - ALLOCATE
  - WRITE

CA ACF2 System Verification Process

When a user submits a transfer request, the CA XCOM Data Transport security interface triggers the CA ACF2 security system verification process. CA ACF2 checks the user's resource access privileges to determine whether the user has READ/WRITE authority to the resource in question. For example, when a user requests an outbound data set transfer, the CA ACF2 security system verifies that the user is authorized to READ from the data set on the user's local system that would be sent to the remote target system. Also, when a user requests an inbound transfer of a remotely-stored data set whose name does not match the name of any data set on the user's local (target) system, CA ACF2 verifies that the user is authorized to ALLOCATE the remotely-stored data set.

CA ACF2 Status Code

After it has done a security check, CA ACF2 returns a status code that shows how the security system has responded to submission of the transfer request. That is, the status code indicates whether the user has permission to access the resource that would be involved in the requested manner (a READ, ALLOCATE or WRITE). A 913 abend indicates that a user has authorization to access a data set, but the CA XCOM Data Transport address space does not. In this case, CA XCOM Data Transport sends an error message to the user.

Note: The MUSASS privilege allows CA XCOM Data Transport to execute a pseudo-logon of the user and perform a validity check.

How Job Submission Works with CA ACF2 Enabled

At times the CA XCOM Data Transport started task submits jobs into the z/OS system. For example, a remote personal system running CA XCOM Data Transport for Windows might submit a job for execution on this z/OS system. For proper security checking in these situations, CA XCOM Data Transport must be assigned MUSASS and JOBFROM privileges. Then it can build and insert the JOBFROM control card into each job stream submitted on behalf of the user. This allows those jobs to inherit the specified user ID/password and the source information of the user who requested the submission. Consequently, job submission control and accountability can be enforced by CA ACF2.

CA XCOM Data Transport builds the JOBFROM card in the following format:

```
//*JOBFROM userID/luname
```

luname

Specifies the LU name of the node from which the user initiated the send job file transfer.

userID

Specifies the CA XCOM Data Transport user ID specified by the user that initiated the request.

Possible Error Conditions—CA ACF2

The following are some of the messages passed back through the CA XCOM Data Transport security interface to explain why an access request has been refused:

- CA ACF2 not available
- Password not matched
- Logon ID not found in the CA ACF2 Logon ID data base
- Password has expired
- Data set access denied

CA XCOM Data Transport passes the CA ACF2 message back when a security violation is encountered.

Installation with an Expired Password Exit

If the installation implements an Expired Password exit, the following problems might arise:

- The Logon ID and Password combination passed to the interface is flagged as expired.
- CA XCOM Data Transport processing continues as if the Logon ID and Password combination were valid, but the Expired Password exit assigns a new Password to the Logon ID without notifying the user.

CA Top Secret Interface

Select the CA XCOM Data Transport's CA Top Secret interface by specifying SECURITY=TOPS in the CA XCOM Data Transport Default Options Table, or specify TOPS by overriding the SECURITY parameter on the CA XCOM Data Transport EXEC statement of the started task JCL.

It is not necessary to assemble or link edit the XCOMTOPS security interface module unless it has been customized. In most cases, it is not necessary to customize the interface module.

CAI.CBXGSAMP(XCOMTOPS) contains the source code for the CA Top Secret interface. CAI.CBXGJCL(ASMTOPSU) contains sample JCL that can be used to assemble and link edit it. After customizing ASMTOPSU, edit ASMTOPSU to fit your particular installation. Next, submit the edited procedure as a job. Be sure to check for any unresolved references in the linkage editor output.

Access Resources for CA Top Secret

The CA XCOM Data Transport CA Top Secret interface uses standard SAF macros to validate access to resources. In addition to data set access checking, the CA XCOM Data Transport CA Top Secret interface can support security checking on the terminal (that is, logical unit), source of origin, time of access, and volume level. The CA XCOM Data Transport security interface determines whether the user ID allows access to the requested resources. For example, a job submitted by a remote CA XCOM Data Transport system runs under the access authorizations of the user ID associated with this job by the remote user. If the remote user does not explicitly specify a user ID, the job runs under the privileges of the local system's default user ID.

If the CA XCOM Data Transport security interface determines that the user is an authorized user, CA XCOM Data Transport opens the file. When CA XCOM Data Transport opens a file, CA Top Secret uses the access authority granted to the CA XCOM Data Transport address space rather than to the user. Therefore, CA XCOM Data Transport must be given access authority for all data sets (except for sensitive data sets that will never be needed by CA XCOM Data Transport users). 913 abends indicate that a user has authorization to access a data set, but the CA XCOM Data Transport address space does not. In this case, CA XCOM Data Transport sends an error message to the user.

Define a Facility

To implement the CA XCOM Data Transport CA Top Secret Interface, a facility must be defined for CA XCOM Data Transport in the System Facilities Matrix. To add a facility for CA XCOM Data Transport, modify one of the predefined USER facilities in the System Facilities Matrix. If all the USER facilities are already used, modify one of the other predefined facilities. For more information, see the CA Top Secret documentation.

Options for Defining a Facility

Define the facility for CA XCOM Data Transport using the following options:

AUTHINIT

CA XCOM Data Transport will run as an authorized program.

ID=*n*

Identifier used on CA Top Secret reports.

IJU

With CA XCOM Data Transport for z/OS, IJU should be coded, to allow CA XCOM Data Transport to handle the insertion of the user ID and password.

INITPGM=XCO

Generic program name (up to 3 digits).

MULTIUSER

Defines CA XCOM Data Transport as a multi-user address space.

NAME= XCOM

Defines the facility name. You can define a different name besides the one provided here.

NOABEND

Prevents a user violation from causing the address space to abend.

NOASUBM

No alternate job submission method.

NOLUMSG

No LU informational messages will be sent to the CA XCOM Data Transport interface.

NOSTMSG

No started task informational messages will be sent to the CA XCOM Data Transport interface.

SIGN(M)

Allows multiple simultaneous signons (user discretion). Each CA XCOM Data Transport user must be explicitly authorized to use the CA XCOM Data Transport Facility. Use this format:

```
TSSADDTO(USER99)FAC(XCOM)
```

If the following message occurs during a file transfer, it may mean that CA XCOM Data Transport has not been defined as a facility to CA Top Secret.

```
TSS Initialization Error
```

Multi-level Passwords

If a user possesses the MULTIPW attribute, add a password for the CA XCOM Data Transport facility. Use this format:

```
TSS ADD(USER01)PASSWORD(BUZWRD)FAC(XCOM)
```

When making a file transfer involving a z/OS system running CA Top Secret, specify the CA XCOM Data Transport password as the Remote Password parameter associated with the Remote User ID of a user who has the MULTIPW attribute. An incorrect password causes an error.

Define Your ACID

The Access Control ID (ACID) created for the CA XCOM Data Transport address space should be given the following attributes. For more information, see the CA Top Secret documentation.

If CA XCOM Data Transport is running as a started task, you may have to add the CA XCOM Data Transport proc to the allowable started task list. This depends on your installation's approach to securing Started Task Control (STC). It is always a good idea to protect the CA XCOM Data Transport started task, because it will usually be given powerful access privileges.

To define your ACID

Use this format:

```
TSS ADDTO(STC)PROC(XCOM'sSTC)ACID(XCOM'sACID)
```

FAC(STC)

Allows CA XCOM Data Transport to run as a started task.

MASTFAC(XCOM)

Must match the facility name defined above.

NOSUBCHK

Allows CA XCOM Data Transport to submit jobs on behalf of another user.

PASS(NOPW,0)

The operator will not be prompted for a password at CA XCOM Data Transport startup time.

Restrict Logical Unit Access

CA Top Secret considers a logical unit (LU) to be a TERMID resource type. Logical units are identified by their LU names.

To assign ownership of a particular logical unit

Use this format:

```
TSS ADD(ACTDEPT)TERMID(luname)
```

The CA Top Secret source of origin security feature can restrict a particular user or profile by permitting access to the system from designated LUs only.

To permit access to the system from designated LUs only

Use this format:

```
TSSADD(USER01)SOURCE(luname)
```

If access is denied to a logical unit, the following CA Top Secret message is generated:

```
TSS 974E TERMINAL(luname)ACCESS DENIED
```

Define a Resource Class

CA XCOM Data Transport r11.6 provides the ability for the installation to create resource classes for Partner and Command security. Resource classes provide the ability to separate the security rules for CA XCOM Data Transport from other products to improve performance on refreshing the resource class.

A resource class is added to the Resource Definition Table (RDT) and must then have an owner defined before access can be granted to it.

Define a Partner Resource Class

TSS ADDTO(RDT) RESCLASS(xcomfac) RESCODE(nnn) MAXLEN(255) ACLST(NONE, ALL, READ)

RESCLASS(xcomfac)

Defines the name of the resource class for partner security.

RESCODE(nnn)

Specifies the CA Top Secret resource code. Valid values are 001-03F for a General Resource or 101-13F for Prefixed Resource. A prefixed resource class is a resource that allows masking characters and has an ownership resource name length up to 26 characters. General resource classes only allow masking characters when defined with the MASK attribute, and have an ownership resource name length of eight characters.

MAXLEN(255)

Specifies the maximum permission length for the resource.

ACLST(NONE, ALL, READ)

Specifies the resource access levels. For partner security, NONE and READ are the only levels required. CA XCOM Data Transport only looks for read access to the resource when checking access to a partner.

Define a Command Resource Class

TSS ADDTO(RDT) RESCLASS(xcomfac) RESCODE(nnn) MAXLEN(255) ACLST(NONE, ALL, READ, UPDATE)

RESCLASS(xcomfac)

Defines the name of the resource class for command security.

RESCODE(nnn)

Specifies the CA Top Secret resource code. Valid values are 001-03F for a General Resource or 101-13F for Prefixed Resource. A prefixed resource class is a resource that allows masking characters and has an ownership resource name length up to 26 characters. General resource classes only allow masking characters when defined with the MASK attribute, and have an ownership resource name length of eight characters.

MAXLEN(255)

Specifies the maximum permission length for the resource.

ACLST(NONE, ALL, READ, UPDATE)

Specifies the resource access levels. For partner security, NONE, READ and UPDATE are the only levels required. CA XCOM Data Transport only looks for read or update access to the resource when checking access to a command.

Grant Ownership of a Resource Class

TSS ADDTO(MASTERQ) xcomfac(resource)

xcomfac(resource)

Specifies the class and resource being defined as owned by the specified ACID. The resource class is specified for **xcomfac** and the resource name is defined as needed for a partner or command. The format for partner resources is described in SAF Security Call—Partner Security. The format for command resources is described in SAF Security Call—Command Security.

Grant Permission to a Resource Class

TSS PERMIT(acid) xcomfac(resource) ACCESS(access level)

xcomfac(resource)

Specifies the resource in the specified class to be permitted to the specified ACID. The resource class is specified for **xcomfac** and the resource name is defined as needed for a partner or command. The format for partner resources is described in SAF Security Call—Partner Security. The format for command resources is described in SAF Security Call—Command Security.

ACCESS(access level)

Specifies the level of access to be granted to the ACID. This is based on the access levels defined to the resource class. For Partner security, you would grant either READ or NONE access based on whether the ACID is allowed to access the partner resource. For Command security, you would grant READ, UPDATE, or NONE access based on whether the ACID is allowed access to the command.

How the Security Interface Works

When a file transfer request is received, the system and interface perform the following actions:

- CA XCOM Data Transport first ensures that the user ID and password supplied by the remote user is valid. This is done with the RACINIT macro. The password sent across the line is encrypted. In addition to the user ID, the VTAM APPLID and the VTAM LU name fields are passed to the RACINIT macro via the APPL and TERMID operands. A new password can be specified by the user to change the current password.
- CA XCOM Data Transport also passes the VOLSER of the volume where the requested data set resides.
- The CA XCOM Data Transport CA Top Secret interface requests that an ACEE be created and saved by the RACINIT macro for use by the authorization routine.

  Note: User ID/password validation is done for data files and job type transfer requests, but not done for report type transfer requests.
- The CA XCOM Data Transport CA Top Secret Interface then passes the ACEE to the RACHECK macro that determines whether the user ID has access privileges.

  Three types of allocation checking are done for data set access requests. The allocation type used depends on the level of access requested:

  - READ authority
  - WRITE authority
  - CREATE authority
- The ACEE is deleted by using the RACINIT ENVIR=DELETE command after access checking. This purges the user ACEE from the system.

How Job Submission Works with CA Top Secret Enabled

CA XCOM Data Transport supports the initiation of jobs from a remote LU through the internal reader (INTRDR).

When a job is submitted, CA XCOM Data Transport performs the following actions:

- Validates the user ID and password that are on the job statement. (SURCHK=YES and SURCLS must be specified in the CA XCOM Data Transport Default Options Table.)

  This check is performed on the receiving system. This validation does not take place if the user ID on the job statement is the same as that used by the CA XCOM Data Transport started task on the receiving system, or if it is the same user ID under which the receiving CA XCOM Data Transport server is executing.

  If the combination is invalid or contains blanks, CA Top Secret determines what happens next. For example, the procedure might have failed outright, or it might be allowed under the access privileges of a default user ID.

- Checks for USER and PASSWORD parameters on the JOB statement. If there are none, CA XCOM Data Transport inserts those specified by the remote user on to the JOB statement. Access checking is based on this user ID, not CA XCOM Data Transport's authorization.

Note: The inserted password is visible only to the CA XCOM Data Transport security interface and does not appear on any system or job output.

Possible Error Conditions—CA Top Secret

When error or abnormal conditions occur, the CA XCOM Data Transport CA Top Secret interface extracts the CA Top Secret message and describes the failure. For more information about CA Top Secret messages, see the CA Top Secret *Messages and Codes Guide*. CA Top Secret messages are of the format TSS9999, where 9999 is the message number.

All error messages have the following in common:

- They are displayed on the CA XCOM Data Transport console.
- They are logged to the CA XCOM Data Transport log file.
- They are sent to the remote LU and logged/displayed on that machine.

For more information about abnormal conditions, ask the security administrator to review the CA Top Secret log.

IBM RACF Security Interface

Select the CA XCOM Data Transport IBM RACF Interface by specifying SECURITY=RACF in the CA XCOM Data Transport Default Options Table, or specify RACF by overriding the SECURITY parameter on the CA XCOM Data Transport EXEC statement on the started task JCL.

It is not necessary to assemble or link edit the XCOMRACF security interface module unless it has been customized. In most cases, it is not necessary to customize the interface module.

CAI.CBXGSAMP(XCOMRACF) contains the source code for the CA Top Secret Interface. CAI.CBXGJCL(ASMRACFU) contains sample JCL that can be used to assemble and link edit it. After customizing XCOMRACF, edit ASMRACFU to fit your particular installation. Next, submit the edited procedure as a job. Be sure to check for any unresolved references in the linkage editor output.

Access Restrictions

The CA XCOM Data Transport IBM RACF security interface uses standard IBM RACF macros to validate access to resources. In addition to data set access checking, the CA XCOM Data Transport security interface checks the source of origin, time of access, and volume level. The security interface determines whether the user ID is allowed to access the requested resources. For example, if a job is submitted by a remote CA XCOM Data Transport system, it runs under the authorizations of the remote user.

Access Authorization

When CA XCOM Data Transport opens a file, IBM RACF uses the access authority granted to the CA XCOM Data Transport address space rather than to the user. Therefore, CA XCOM Data Transport should be given access authority for all data sets (except for sensitive data sets that will never be needed by CA XCOM Data Transport users). 913 abends indicate that a user has authorization to access a data set, but the CA XCOM Data Transport address space does not. In this case, CA XCOM Data Transport sends an error message to the user.

Started Task Definition

When CA XCOM Data Transport runs as a started task, it must be given a started task definition by the security administrator. To implement this new definition, the IBM RACF started task table must be assembled and relinked (see the *IBM Security Server RACF* System *Programmer's Guide*). Make sure the UID associated with the started task has sufficient access authority. Also, CA XCOM Data Transport should be granted multiple user authorization.

APPLID Protection

If the CA XCOM Data Transport APPLID is to be protected, it must be defined to IBM RACF (see the *IBM Security Server RACF Security Administrator's Guide*). Furthermore, CA XCOM Data Transport users must be given explicit authority to use this APPLID.

How the RACF Security Interface Works

When a file transfer request is received, the system performs the following actions:

- CA XCOM Data Transport first ensures that the user ID and password are valid. This is done through the RACINIT macro. The password sent across the line is encrypted. In addition to the user ID, the VTAM APPLID and the VTAM LU name fields are passed to the RACINIT macro via the APPL and TERMID operands. A new password can be specified by the user. CA XCOM Data Transport also passes the VOLSER of the volume on which requested data sets reside.
- The CA XCOM Data Transport IBM RACF Interface requests that an ACEE be created by the RACINIT macro and saved for use by the authorization routine.

  Note: User ID/password validation is done for data files and job type transfer requests, but not for report type transfer requests.
- The CA XCOM Data Transport IBM RACF Interface then passes the ACEE to the RACHECK macro that determines whether the user ID has access privileges. For data set access requests, three types of allocation checking are done, depending on the level of access requested:
  - READ authority
  - WRITE authority
  - ALLOCATION authority
- After access checking is done, the ACEE is purged from the system with the following command:

  ```
  RACINITENVIR=DELETE
  ```

Note: Access events are logged by SMF under both the CA XCOM Data Transport and the user's UID.

How Job Submission Works with IBM RACF Enabled

CA XCOM Data Transport supports initiation of jobs from a remote LU through the internal reader. When a job is submitted, CA XCOM Data Transport performs the following actions:

- Validates the user ID and password provided by the remote user.
- Checks for the USER and PASSWORD parameters on the JOB statement. If there are none, CA XCOM Data Transport inserts those specified by the remote user onto the JOB statement. The inserted password is visible only to the security interface and does not appear on any system or job output. Access checking is based on this user ID, not CA XCOM Data Transport's authorization.

Note: If the CA XCOM Data Transport user ID and the user ID on the JOB statement differ, the job runs under the latter's privileges.

Possible Error Conditions—IBM RACF

When an error or abnormal condition occurs, the CA XCOM Data Transport's IBM RACF security interface returns an error message describing the failure. For example:

```
XCOMM5502E INIT ERR 08 00:INVALID PASSWORD
```

Recommended actions are also provided. For a listing of error messages, see the *CA XCOM Data Transport for z/OS Message Reference Guide*. All CA XCOM Data Transport's IBM RACF security interface messages have XCOMM55*nn*E message ID formats.

All error messages have the following in common:

- They are displayed on the CA XCOM Data Transport console.
- They are logged to the CA XCOM Data Transport log file.
- They are sent to the remote LU and logged/displayed on that machine.

For additional information on the cause of a CA XCOM Data Transport IBM RACF error message, ask the security administrator to review the IBM RACF log.

SAF Interface

Select the CA XCOM Data Transport SAF Interface by specifying SECURITY=SAF in the CA XCOM Data Transport Default Options Table, or specify SAF by overriding the SECURITY parameter on the CA XCOM Data Transport EXEC statement on the CA XCOM Data Transport server (XCOMXFER) JCL.

When using SECURITY=ACF2|TOPS|RACF, the CA XCOM Data Transport server runs under its own ACEE and the security checks are done using the specified external security package.

With the CA XCOM Data Transport SAF Interface , the transfer runs under one of the following ACEEs:

- For an incoming (remotely initiated) transfer, the ACEE of the user ID provided in the transfer header
- For an outgoing (locally initiated) transfer, the ACEE of the user ID that initiated the transfer

The ACEE is built using the IBM BPX1SEC callable service which invokes whichever external security package is in control of the CA XCOM Data Transport Server address space. This eliminates the need to give the CA XCOM Data Transport server READ/WRITE/UPDATE access to all data sets that are to be transferred using this server, as long as the user whose ACEE is used for a particular transfer has the proper accesses to the resources.

Because SECURITY=SAF is used only for the actual transfer process, it should not be used with the CA XCOM Data Transport XCOMPLEX Admin Server (XCOMXADM).

BPX1SEC Security Requirements

The use of the IBM BPX1SEC callable service has the following requirements:

- The ACEE of the CA XCOM Data Transport Server must be defined to OMVS with superuser status (UID=0)
- If the BPX.DAEMON resource in the FACILITY class is defined, then the ACEE of the CA XCOM Data Transport Server must have READ authority to BPX.DAEMON.

The following example show the CA Top Secret commands necessary to grant proper access.

Note: If you have SECURITY=ACF2 or SECURITY=RACF, see your CA ACF2 or IBM RACF documentation as required.

Access to BPX.DAEMON

To grant access to the BPX.SERVER

Issue the following CA Top Secret command:

TSS PER(xxxxxxxx) IBMFAC(BPX.DAEMON) ACC(READ)

xxxxxx

The ACEE of the CA XCOM Data Transport Server.

Security Considerations for USS Files

USS support enforces SAF security for all transfers that involve a USS file, regardless of the SECURITY= parameter in the defaults table or EXEC statement.

SECURITY=NONE/ACF2/RACF/TOPS is honored for all other types of files.

Password Protection by Encryption

CA XCOM Data Transport always protects the password thru the use of encryption. CA XCOM Data Transport encrypts the password in the configuration file as well as during transmission. The cipher used for encrypting the password can be selected by parameter.

How Configuration File Password Encryption Works

Eliminate Passwords from Parameter Files

Previous releases of CA XCOM Data Transport for z/OS required users to supply a password in their file transfer parameters when communicating to the IBM Midrange (AS/400, for example) as well as OS/2.

CA XCOM Data Transport makes the specification of a password optional in most cases, even when updating a secured resource on the remote system.

The Already Verified Indicator

CA XCOM Data Transport for z/OS supports a field called the “already verified indicator” in the function management header (FMH-5 or Attach Header). This bit is set if all of the following conditions are met:

- The user omits a password and does not override their password.
- The remote system is defined with ACCSEC=YES.
- The BIND specifies that ACCSEC is supported. The BIND must also specify that the FMH5 user ID “already verified” indicator will be allowed. This usually requires a configuration parameter on the IBM midrange system.

Set Up Trusted Access Security

The Trusted Access feature is used for z/OS-initiated transfers to CA XCOM Data Transport for Windows and UNIX platforms. Trusted Access also allows Windows- or UNIX-initiated transfers to CA XCOM Data Transport for z/OS.

Trusted Access allows transfers to be sent without specifying a user ID and password, as long as the remote partner has the z/OS system defined as trusted, with the same user ID as the sending system. The user ID specified for the z/OS system must match the user ID of the person logged on to the initiating partner. In this case, even if no user ID or password is provided, the transfer will be allowed to proceed.

Example

If your logon ID on a Windows, UNIX, or z/OS platform is USER01 and your logon ID on the target z/OS platform is also USER01, then the trusted transfer can run without specifying the user ID and password. CA XCOM Data Transport will automatically try to match the logon ID to the TRUSTID entries in the destination member for the remote partner.

Trusted Transfers to z/OS

For trusted transfers sent to z/OS, a destination member for the remote system must be enabled with a TRUSTID entry that matches the user ID sent by the remote system. The TRUSTID parameter consists of a user ID and an optional group ID. The user ID specified on a TRUSTID parameter must match the user ID of the person logged on to the initiating system.

For more information about the TRUSTID parameter, see Destination Parameters for Single LUs, Groups of LUs, and Single IPNAMEs in the chapter "Configuration Parameters."

Implement Trusted Access Security for Transfers to z/OS

To implement Trusted Access security, define a destination member with one or more TRUSTID entries.

The sample destination member below is used to determine if Trusted Access is permitted for requests from remote system 130.200.123.45. It contains four TRUSTID entries:

- The first entry has only the USERID coded, because it is a valid signon ID on the local z/OS system.
- The second entry is not a valid z/OS user and has a GROUPID coded to point to a valid z/OS user GEORGENG that will be used for a trusted transfer initiated by remote user TRUSTGNG.
- The third entry is not a valid z/OS user and has a GROUPID coded to point to a valid z/OS user USER003 that will be used for a trusted transfer initiated by remote user TRUSTGNG2.
- The fourth entry is a valid z/OS user and has a GROUPID coded to point an alternative valid z/OS user USER004 that will be used for a trusted transfer initiated by remote user TRUSTGNG3.

```
**********************************************************************
*  FUNCTION: DEST FOR GEORGE'S WINDOWS (TCP/IP)                      *
**********************************************************************
TYPE=DEST
WRITER=
IPNAME=130.200.123.45
IPPORT=8044
TRUSTID=GEORGENG            THIS IS A VALID z/OS signon
TRUSTID=TRUSTGNG,GEORGENG   THIS PC USERID will use GEORGENG on z/OS
TRUSTID=TRUSTGNG2,USER003   THIS PC USERID will use USER003  on z/OS
TRUSTID=TRUSTGN32,USER004   THIS PC USERID will use USER004  on z/OS
```

Trusted Transfers from z/OS

When initiating a transfer from CA XCOM Data Transport for z/OS, the TRUSTED parameter must be set to YES in SYSIN01 to indicate to the receiving system that this transfer is a trusted transfer. If the user initiating the trusted transfer is not defined as a Trusted user on the receiving platform, then USERID= must be set to a user ID or group ID that is defined on the receiving platform. USERID is accepted only if USEROVR is set to YES in the XCOMDFLT.

Trusted Access must be configured on the remote partner. The TRUSTED parameter is supported only by other systems running Advantage CA-XCOM Data Transport Version 3.1, Unicenter CA-XCOM Data Transport r11, or CA XCOM Data Transport r11.5 on platforms that currently support Trusted transfers. To verify this support, see the remote system's documentation.

Data Encryption Using Secure Socket Layer (SSL)

CA XCOM Data Transport uses OpenSSL to utilize the Secure Socket Layer (SSL) to perform secure TCP/IP transfers between platforms running CA XCOM Data Transport r11 and above that support secure (SSL) TCP/IP. A secure (SSL) TCP/IP transfer allows for the encryption of the transmitted data and adds a digital signature to the encryption of the transmitted data.

Important Considerations when Using OpenSSL

CA XCOM Data Transport employs OpenSSL to use the Secure Socket Layer (SSL) while performing secure TCP/IP transfers between platforms running at our r11 and above. OpenSSL has become a common approach to SSL implementation and is used by a number of vendors in addition to CA.

In some CA platform implementations, such as z/OS, UNIX, Linux, and Windows, we use a CA-modified version of OpenSSL that may be employed by other CA products. On these platforms, CA provides and supports the OpenSSL implementation as if it were our own software. However, the installation may appear as a separate step so that you can choose to use another CA-provided OpenSSL implementation that was installed with another CA product.

On other platforms, including HP’s Guardian and OpenVMS and IBM’s System i (formerly AS/400), we utilize the OpenSSL provided and supported by these vendors.

Between these two methodologies, we do not expect CA XCOM Data Transport customers to support their own implementation of OpenSSL or to rely upon the OpenSSL community to do the same. The first line of support is intended to be the vendor, and our licensed customers can contact CA Support with questions or issues with OpenSSL in any CA XCOM Data Transport implementation.

Chapter 4: Configuring the Network

Because CA XCOM Data Transport is capable of overriding existing mode definitions, additional LU definitions are often unnecessary. However, in some situations there may be a need for some Network Control Program (NCP) definitions.

This section contains the following topics:

Define Remote LUs (NCP Considerations)

This section describes NCP considerations to take into account when you configuring your network.

X.25 Switched Virtual Circuits

For X.25-switched virtual circuits, the PU definition and the associated MAXDATA parameter are specified in the VTAM switched major node definition rather than in the NCP generation. See X.25/NPSI SVC implementations in the appropriate NCP manuals.

VTAM Dialup Environment

Discussed below are some examples of VTAM macro definitions for a dialup (switched) major node environment.

Dial Out from the Host

The first example applies to dialing out from a z/OS host. Make sure you specify the correct MAXDATA value. The MAXDATA parameter of the PU macro designates the maximum information frame size used by the link layer of SNA. The appropriate specification will vary by system.

Note: Many versions of NCP require two ports for dialing out: one for the node, the other for an AT&T 801C-type autocaller.

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
OUTDIAL     VBUILD    TYPE=SWNET                                            X
                      MAXNO=1,              NUMBER OF PATH STATEMENTS       X
                      MAXGRP=1,             TELEPHONE NUMBERS AVAILABLE     X
*
PUCALL      PU        ADDR=01,                                              X
                      MAXDATA=265,                                          X
                      MAXOUT=7,                                             X
                      IDBLK=03D,                                            X
                      PUTYPE=2,                                             X
                      IDNUM=2674B,          USER SELECTED ID NUMBER         X
                      MODETAB=XCOMTABL,                                     X
                      PASLIM=7,                                             X
                      IRETRY=YES,                                           X
                      MAXPATH=1,
*
            PATH      DIALNO=12127664400,   TELEPHONE NUMBER TO CALL        X
                      PID=1,                ARBITRARY PATH NUMBER           X
                      GID=1,                ARBITRARY GROUP NUMBER          X
                      GRPNM=GROUPS,         NCP GROUP NAME FOR SWITCHED LINES X
                      REDIAL=1,
                      USE=YES
*
LUCALL      LU        LOCADDR=1,                                            X
                      DLOGMOD=XCOMMODE
```

Dial in to the Host

The second example shows a VTAM LU 6.2 switched major node dialing into a z/OS host. The example uses the sample physical unit and logical unit (PU/LU) macros defined in the previous table.

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
SWSAMPLE VBUILD  TYPE=SWNET                                              X
*
APPCPU   PU    ADDR=C1,                                                  X
               PUTYPE=2,                                                 X
               MAXDATA=521,                                              X
               MAXOUT=7,                                                 X
               IDBLK=050,                                                X
               IDNUM=EF02A,                                              X
               MODETAB=XCOMTABL,                                         X
               PASLIM=7,                                                 X
               IRETRY=YES,                                               X
*
APPCLU1  LU    LOCADDR=1,                                                X
               DLOGMOD=XCOMMODE
*
APPCLU2  LU    LOCADDR=2,                                                X
               DLOGMOD=XCOMMODE
```

Create Cross-domain Resources

Cross-domain resources (CDRCS) need to be addressed for users with multiple VTAM domains. Cross-domain resources must be created if dynamic CDRSC definition is not being used in conjunction with SSCP support.

Note: When VTAM validates switched-node definitions, it tries to validate inaccurately specified fields coded by users of midrange and smaller systems when these fields are sent in an XID. Normally, this affects AS/400 users. Be sure to properly configure the NETID and PUNAME fields.

SNA Considerations

This section describes NCP considerations to take into account when you configuring your network.

Specify Pacing and Performance

VTAM and NCP systems programmers may wish to review the following rules governing pacing specifications.

Pacing Parameters

There are three pacing parameters that can be specified in a LU-to-LU session to adjust session-level pacing. Specify these parameters in a LOGMODE table entry and the appropriate VTAM/NCP definitions for devices and applications. The actual values used for pacing during a LU-to-LU session depend on the following:

- Where the pacing values have been coded
- How many stages of pacing are used in a session
- The configuration of the session partners
- The BIND negotiations

Pacing Stages

You may define the following pacing stages:

- PSNDPAC - Primary Send Pacing
- SRCVPAC - Secondary Receive Pacing
- SSNDPAC - Secondary Send Pacing

Primary Send Pacing (PSNDPAC)

Use Primary Send Pacing to perform pacing between the PLU and one of the following:

- The boundary function NCP for the SLU (session with link attached peripheral node)
- Another APPL (APPL-to-APPL session)
- The VTAM boundary function (cross-domain session with locally attached node)

Pacing is used for OUTBOUND (PLU-to-SLU) traffic only. The value in the logmode PSNDPAC field overrides all other Primary Send pacing parameters.

Coding 0 for the PSNDPAC field in the logmode indicates that the Primary Send Pacing value will be governed by the VPACING value coded on the LU definition. The SLU's VPACING value in the APPL definition will be used if this is an APPL-to-APPL session. If you *do not* code VPACING and the PSNDPAC value is set to 0 in the logmode, the default VPACING value is 2 for a LU and 0 for an APPL.

Secondary Receive Pacing (SRCVPAC)

Use Secondary Receive Pacing to perform session pacing between the following:

- The NCP boundary function and the SLU (session with link-attached peripheral nodes)
- The PLU and the SLU (session with a locally attached LU in a single-domain environment)
- The VTAM boundary function and the SLU (cross-domain session with a locally attached node)

Pacing is used for OUTBOUND (PLU to SLU) traffic only. Specify a non-zero value in the SRCVPAC field of the logmode entry to override all other Secondary Receive Pacing values.

Coding this value as 0 in the logmode SRCVPAC field indicates the usage of the value coded in the PACING parameter for the LU. If this is a session with a locally attached device in a single-domain network, the value coded in the VPACING parameter on the LU definition is used. If these parameters are not coded in the LU definition and the logmode value is set to 0, the default value of 1 is used.

If no pacing is used for this stage, 0 *must* be coded in both the logmode SRCVPAC and the LU PACING definitions. If the session is with a locally attached node in a single-domain environment, the VPACING parameter must be set to 0 in the LU definition, or the session is not established. For more information, see the following boundary function example.

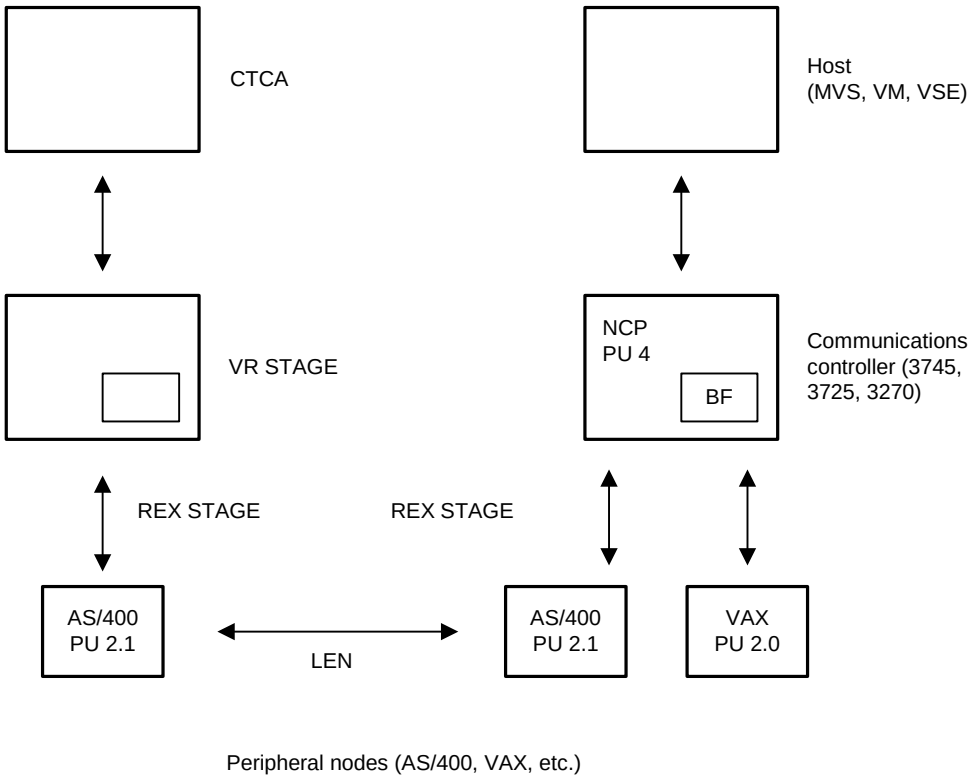
Secondary Send Pacing (SSNDPAC)

Use Secondary Send Pacing to perform session pacing between the following:

- The SLU and the PLU when one-stage pacing is used
- The session is with a link-attached peripheral node or the session is with a locally attached node in a single-domain environment or the session is an APPL-to-APPL session.
- The PLU and the boundary function of the VTAM to which the node is attached in a cross-domain session with a locally attached device

  This session will use two-stage pacing and refers to INBOUND (SLU-to-PLU) traffic only.

For one-stage pacing, the SSNDPAC value is used as a pacing switch. Coding a 0 for this field indicates that no inbound pacing will occur. For two-stage pacing, the SSNDPAC value in the logmode is used as the pacing specification for the first stage pacing (SLU to attaching VTAM). Specifying 0 as the SSNDPAC value in the logmode indicates that no inbound pacing will occur in the first stage. The VPACING value coded in the APPL statement is used to determine the pacing value for the *second* stage only. If this value is 0 or the parameter is omitted, then inbound pacing *does not* occur in the second stage. For more information, see the following figure.



Legend:

PU = Physical Unit
BF = Boundary Function
CTCA = Channel to Channel Adapter
VT Stage = Virtual Route Stage
REX Stage = Route Extension Stage
LEN = Low Entry Network

VPACING Specifications

An application can override the VPACING specification of its session partner (SLU) if AUTH=NVPACE is coded in the APPL definition. This is equivalent to coding VPACING=0 in the LU definition of the SLU. The default is AUTH=VPACE.

If you do not have control over which LU is the PLU or SLU in an APPL-to-APPL session, then code the values as follows:

- PSNDPAC in the logmode
- SSNDPAC in the logmode
- VPACING in the APPL definition on both PLU and SLU definitions

This yields the desired pacing values and maintain uniformity regardless of which APPL is the PLU.

Adaptive Pacing

Adaptive Pacing provides a means of controlling the rate of RU exchange between a VTAM host and an NCP. It also provides the same functional control between an NCP and specific devices connected to a front-end processor that supports Adaptive Session Pacing.

Remote Locations

For remote locations, the initial minimum pacing values are set during the establishment of a session. The issuer of the BIND request is responsible for setting these values. The values may change under system or application control depending on system buffer resources and traffic patterns within the network. The network allocates session buffers automatically to make efficient use of available resources. Under certain conditions, the pacing value is reduced to decrease the speed of the file transfer. In extreme cases, a node may even stop receiving data for a period of time.

Class of Service Tables

Class of Service (COS) tables may be used to prioritize transmissions and to ensure throughput for critical sessions. This is especially useful in the case of traffic traversing NCP-to-NCP links because higher-priority traffic is dispatched ahead of lower-priority traffic (for example, interactive traffic is dispatched ahead of batch traffic). Traffic flowing outbound from the NCP to a device can also be regulated by the NCP parameter, LSPRI. For more information about LSPRI, see the *NCP Reference Manual*.

BIND Functions

At this time, the extended BIND functions required for adaptive pacing are supported by only these IBM mainframe and midrange operating systems: z/OS, VM, VSE, and OS/400. In environments where the current VTAM and NCP products are installed and non-supported devices are in place, both adaptive and fixed pacing are in effect. Adaptive Pacing is used on the VR stage (owning VTAM to boundary NCP) and Fixed Pacing is used on the Route Extension (REX) stage (boundary NCP and the device).

Test Your Product in the Network

In data communications, it is essential to test whatever has been implemented before exposing it to a production environment. To assist in ensuring that CA XCOM Data Transport's installation in the network was properly completed, an Installation Verification Procedure (IVP) has been provided.

If an error is detected by the IVP, go back and review each step of the installation procedure to determine the cause of the problem.

Test the Server and the Batch Interface

You need to test the CA XCOM Data Transport Server (XCOMXFER) and the CA XCOM Data Transport batch interface (XCOMJOB) to verify that they have been correctly implemented.

To test the server and the batch interface

1. Update all appropriate load libraries, sample libraries, and so on.
2. Start the CA XCOM Data Transport server.
3. Customize and submit the CA XCOM Data Transport batch job provided in CAI.CBXGJCL.

Successful completion of the sample job verifies that the server and the batch interface have been correctly implemented. If the Security Interface was implemented, this helps to ensure that it was installed correctly.

Test the ISPF Dialogs

When the CA XCOM Data Transport ISPF library has been installed, use the CA XCOM Data Transport menu interface to determine if the installation process was successful. For more information, see the chapter “The Menu Interface (TSO/ISPF Panels)” in the *CA XCOM Data Transport for z/OS User Guide*.

Test the XCOMPLEX Worker Server and XCOMPLEX Admin Server Batch Interface

Important! The existing XCOMPLEX facility has been deprecated. The following information is provided for backward compatibility only. All new installations should use the new PLEXQ implementation. Existing XCOMPLEX users should migrate to the PLEXQ infrastructure for their XCOMPLEX functionality. Refer to section Create a PLEXQ Environment (Optional) in this guide.

You need to test the CA XCOM Data Transport XCOMPLEX Worker server (XCOMXFER), XCOMPLEX Admin Server (XCOMXADM), and batch interface to verify that they have been correctly implemented.

To test the XCOMPLEX Worker server (XCOMXFER), XCOMPLEX Admin Server (XCOMXADM), and batch interface

1. Verify the XCOMPLEX Admin and XCOMPLEX Worker Servers have the same value specified for XCOMPLEX= in the CA XCOM default table.
2. Start the XCOMPLEX Admin Server. Verify that the XCOMPLEX has been enabled for the XCOMPLEX Admin Server by checking for message XCOMM1005I.
3. Start the XCOMPLEX Worker Servers. Verify that the XCOMPLEX has been enabled for the XCOMPLEX Worker Servers by checking for message XCOMM1005I.
4. Verify that the connection has been made between the XCOMPLEX Admin and XCOMPLEX Worker Servers by checking for message XCOMM0451I in both the XCOMPLEX Worker Server and XCOMPLEX Admin Server logs for each XCOMPLEX Worker Server.
5. Use the MVS MODIFY command, STAT, from the XCOMPLEX Admin Server to show each XCOMPLEX Worker Server connected. For more information, see the chapter “Operation and Control” in the *CA XCOM Data Transport for z/OS User Guide*.
6. Follow steps to verify server and batch interface as described in “Test the Server and the Batch Interface” in this chapter. When running the IVP job, specify the ACBNAME and STCAPPL of XCOMPLEX Worker Server.
7. Use MVS MODIFY commands XRSHOW and XSHOW to show transfers. For more information, see the chapter “Operation and Control” in the *CA XCOM Data Transport for z/OS User Guide*.
8. Follow steps to verify the Server and batch interface as described in Test the Server and the Batch Interface in this chapter. When running the IVP job, specify the ACBNAME and STCAPPL of the XCOMPLEX Admin Server. Transfers should now be distributed through the XCOMPLEX Admin to the XCOMPLEX Worker Servers.
9. Use MVS MODIFY commands XRSHOW and XSHOW to show transfers. For more information, see the chapter “Operation and Control” in the *CA XCOM Data Transport for z/OS User Guide*.

Chapter 5: Understanding the PLEXQ

The CA XCOM Data Transport PLEXQ links multiple CA XCOM Data Transport servers together. CA XCOM uses the IBM Parallel Sysplex Signaling Services as the transport for communications between members of a PLEXQ. You can continuously accommodate increasing workloads by adding CA XCOM Data Transport servers to a PLEXQ. Multiple XCOMPLEX server groups may be set up. For information about configuring and bringing up PLEXQ Servers, see the chapter Configuring and Customizing Your Product.

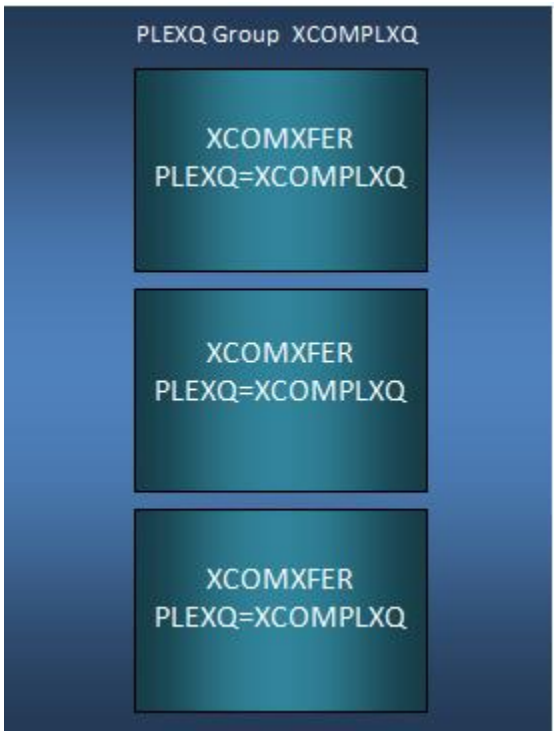
This section contains the following topics:

Structure of the PLEXQ

A CA XCOM Data Transport PLEXQ consists of a number of XCOMXFER servers of the same release. Each PLEXQ Group is given a unique name. All of the servers in the PLEXQ must specify the same name for the PLEXQ parameter in the CONFIG member or the EXEC PARM override. Each server may belong to only one PLEXQ at a time. Each server that is a member of a PLEXQ has its own request queue (XCOM RRDS). These servers may share a CA XCOM Data Transport control data set (XCOMCNTL). Member servers of a PLEXQ are autonomous peers which simply respond to requests received via incoming messages via SYSPLEX Signaling Services.

A PLEXQ can consist of one server, up to the maximum number of servers that can be practically supported by the user's SYSPLEX environment. There is no maximum imposed by the PLEXQ architecture itself.

The following is a representation of servers in a PLEXQ called XCOMPLXQ:



Communication Within a PLEXQ

During startup, servers which have a PLEXQ parameter defined connect to (or JOIN) a SYSPLEX Signaling Services GROUP. Servers which are to connect to a PLEXQ can be started in any order, as PLEXQ servers are peers in functionality.

Exchanges of data and information are routed through the IBM SYSPLEX Signaling Services. Member servers themselves process work or command requests that are made to a PLEXQ directly. Servers in a PLEXQ are polled and the XCOMJOB utility selects the server to receive the work that is being initiated based on the responses received. When scheduling a transfer, consideration is made as to the current workload within each server.

Note: CA XCOM Data Transport PLEXQ services provide functionality similar to the deprecated XCOMPLEX. Transfer workload is balanced based on the current activities in each PLEXQ member server. In a PLEXQ environment, there is no ADMIN server. Requests are made to the PLEXQ group itself, and individual servers interact directly with the XCOMJOB batch utility. This interaction provides superior performance, and reduced resource utilization when performing the same tasks as compared to an XCOMPLEX environment.

Scheduling Transfers

Transfers may be scheduled either through a connection to the PLEXQ group, or directly to a PLEXQ member server. Transfers are scheduled to the PLEXQ group by using the STCPLEXQ EXEC PARM in the XCOMJOB batch utility JCL. The STCPLEXQ parameter is mutually exclusive with the STCAPPL and STCIP/STCPORT parameters, and causes all scheduling or inquiry communications to be performed via the SYSPLEX Signaling Services rather than using the SNA or TCP/IP protocols.

An attempted transfer schedule request to a PLEXQ in which no servers are active will be rejected. Transfers may still be sent to a PLEXQ member server directly without going through the PLEXQ. Transfers scheduled directly to a PLEXQ member server are also considered in the XCOMJOB workload distribution algorithm. The total workload of each PLEXQ member server is evaluated before it is selected to receive a new transfer request by the XCOMJOB batch utility.

Workload Distribution

The XCOMJOB batch utility uses a proprietary ranking system to select the appropriate PLEXQ member server to receive a transfer request. When a local schedule request is initiated to a PLEXQ group, each available server which is a member of the target PLEXQ group is assessed and assigned a ranking. CA XCOM Data Transport considers the total number of transfers, with special consideration given to locally initiated transfers. The Worker who has the lo*west r*anking receives the transfer. In case of a tie, the first server in the PLEXQ group to respond to the poll for status will receive the transfer.

MAXTASK and MAXLOC Parameters

CA XCOM Data Transport uses the MAXTASK and MAXLOC default table parameters as well as the number of concurrent transfers running on each Worker Server to determine ranking. Note that one server may receive all transfers based on this ranking system. For example, if transfers complete quickly, the same Worker Server may always be available. Alternatively, if a server has many long running transfers or a low MAXLOC value, this Worker Server receives a high ranking. If a Worker Server reaches the MAXLOC, the maximum number of locally initiated transfers, this Worker Server also receives a high ranking. Transfers scheduled for a future time are not considered when determining rank.

Note: The MAXTASK and MAXLOC parameters provide the following default options:

- MAXTASK—The CONFIG parameter that specifies the maximum number of file transfers that the CA XCOM Data Transport server can perform concurrently.
- MAXLOC—The CONFIG parameter that specifies the maximum number of locally initiated transfers that can be active at one time.

STAT Modify Command

The CA XCOM Data Transport STAT modify command can be used to provide information as to the status of the servers which are members of a PLEXQ group. These statistics are a snapshot, and are not used as the basis of monitoring the workload distribution. The PLEXQ member servers are continuously processing requests and the ranking for servers changes as each new request comes in or is completed. Multiple requests can come in or, complete in a matter of seconds. See Using the CA XCOM Data Transport *MODIFY Commands in the cha*pter Operation and Control in the CA XCOM Data Transport for z/OS User Guide.

Checkpoint/Restart

PLEXQ member servers are responsible for processing restarts of all locally initiated transfers that have been scheduled to them, either directly or via the PLEXQ group.

Inquire

When a transfer is scheduled to a PLEXQ group, the message buffer returned identifying the transfer request number also contains specific information which will cause a subsequent INQUIRE request to be directed to the appropriate server to which the transfer was scheduled.

TYPE=OPER (Operator) Requests from ISPF to the PLEXQ

To allow users to perform TYPE=OPER transfers of PDSE program libraries, add XCOMPLEX to the AUTHPGM and the AUTHTSF tables of IKJTSO00 module in SYS1.PARMLIB. The CA XCOM Data Transport libraries used in your CLIST for the CA XCOM Data Transport ISPF interface must all be APF authorized also.

Specify a protocol of PLQ on the ISPF panels to submit requests through the PLEXQ. The server name is the name of the PLEXQ to route the request to.

You can also refresh the TSO library using the TSO UPDATE PARMLIB(00) member. For more information, see the *IBM TSO/E Customization manual*.

VIPA

CA XCOM Data Transport for z/OS can utilize virtual IP Addressing with multiple CA XCOM Data Transport servers sharing a virtual IP address. This feature is for remotely initiated transfers. Remote partners send to a single Virtual IP address and transfers are distributed across all servers defined with this Virtual IP address. This feature may be used with or without the PLEXQ. For more information, see Configure Virtual IP Addresses-Remotely-initiated Transfers Only in the chapter Configuring and Customizing Your Product.

VTAM GNAME

For incoming SNA data transfers to CA XCOM Data Transport for z/OS, VTAM Generic Resources directs the request to a particular Worker CA XCOM Data Transport region. The VTAM GNAME must be specified in the CA XCOM Data Transport Default Option Table for each server. This feature can be used with or without the PLEXQ. For more information, see Configure VTAM Generic Names—Remotely initiated Transfers in the chapter Configuring and Customizing Your Product.

Chapter 6: Generating SSL Certificates

This chapter describes how to generate certificates that can be used with CA XCOM Data Transport.

For more information about using OpenSSL, see *Network Security with OpenSSL* by John Vega, Matt Messier, and Pravir Chandra (O'Reilly & Associates).

This section contains the following topics:

SSL Mode

CA XCOM Data Transport uses SSL in client/server mode. In client/server mode, certificates and private keys are required for both the local (initiating) and remote (receiving) CA XCOM Data Transport partners. SSL considers the local CA XCOM Data Transport partner to be the client and the remote CA XCOM Data Transport partner to be the server.

When establishing the SSL connection, the server sends the server certificate to the client for verification. After the client verifies the server certificate, the client sends the client certificate to the server for verification. Both the client and the server must verify the Certification Authority (CA) certificate from the other.

To set up SSL for CA XCOM Data Transport

Important! When you install CA XCOM Data Transport, the SSL certificates (in Steps 1 to 4) are automatically generated. They are documented here in case you need to regenerate them at any time.

1. Set the expiration for the CA certificate (if required).
2. Create the CA certificate (if required).
3. Create the server certificate (if required).
4. Create the client certificate (if required).
5. Configure the CA XCOM Data Transport SSL server.
6. Configure the CA XCOM Data Transport client.

These tasks are described in the following sections.

Set Expiration

When generating a CA certificate, the default_days parameter in cassl.conf that controls the expiration of server and client certificates is not used for CA certificates. The certificate is generated with a default expiration of 30 days.

To change the default expiration

1. Add 'days *nnn*' to the makeca script line. The following line is an example of how the makeca script is shipped:

   ```
   Openssl req -x509 -newkey rsa -out ./certs/cassl.pem -outform PEM
   ```

2. To change the expiration to one year, change the line before running the makeca script:

   ```
   Openssl req -x509 -newkey rsa -out ./certs/cassl.pem -outform PEM -days 365
   ```

Create the CA Certificate

To create the CA certificate

1. Create a configuration file that is used as input to the openssl utility. A sample file, named cassl.conf, was installed in the ssl subdirectory of the CA XCOM Data Transport installation directory for UNIX and Windows platforms. For z/OS, cassl.conf was downloaded as part of a .TAR formatted file, and then copied to a user-specified path on the site's HFS file system. This .TAR file needs to have the SSL files extracted before it can be edited. Change to the ssl subdirectory and edit the [root_ca_distinguished_name] section, changing the values as appropriate for your system.

   Note: For UNIX, you must have 'root' authority to perform this task.

2. Issue the following command to run the makeca script:

   ```
   ./makeca
   ```

   This shell script uses the cassl.conf file to generate a certificate and key file. The certificate, cassl.pem, is saved in the 'certs' subdirectory. The key file, generated as casslkey.pem, is saved in the 'private' subdirectory.

   Note: When running the makeca script the first time, the pseudo-random number generator (PRNG) file does not exist and issues a warning to this effect. The makeca utility generates the PRNG file the first time it is run and does not issue this warning on subsequent executions. This is only a warning; you can continue with the next step.

3. To list the certificate just created, issue the following command to use the listca script:

   ```
   ./listca
   ```

This shell script displays the CA certificate and the information stored in the package.

Create the Server Certificate

To create the server certificate

1. Create a configuration file to use as input to the openssl utility. A sample file, serverssl.conf, was installed in the ssl subdirectory. Edit the [req_distinguished_name] section, changing the values to your specifications.
2. Using the script makeserver, issue the following command:

   ```
   ./makeserver
   ```

   The makeserver shell script uses the serverssl.conf file and the cassl.pem file to generate a server certificate and a key file. The server certificate, servercert.pem, is saved in the 'certs' subdirectory. The key file, generated as serverkey.pem, is saved in the 'private' subdirectory.
3. To list the certificate just created, issue the following command to use the listserver script:

   ```
   ./listserver
   ```

   This shell script displays the server certificate and information stored in the package.

Create the Client Certificate

To create the client certificate

1. Create a configuration file to use as input to the openssl utility. A sample file, clientssl.conf, was installed in the ssl subdirectory. Edit the [req_distinguished_name] section, changing the values to meet your system requirements.
2. Issue the following command to use the makeclient script:

   ```
   ./makeclient
   ```

   The makeclient shell script uses the clientssl.conf file and the cassl.pem file to generate a client certificate and a key file. The certificate, clientcert.pem, is saved in the 'certs' subdirectory. The key file, generated as clientkey.pem, is saved in the 'private' subdirectory.
3. To list the certificate just created, issue the following command to use the listclient script:

   ```
   ./listclient
   ```

   The listclient shell script displays the client certificate and information stored in the package.

Configure the SSL Server

When you configure the SSL server, you enable CA XCOM Data Transport to use the CA and server certificates for establishing server (remote) SSL connections.

To configure the SSL server

1. Review and modify the CA XCOM Data Transport SSL configuration file, configssl.cnf, so that the settings meet your site standards. Server connections use the RECEIVE_SIDE values.
2. Set the XCOM_CONFIG_SSL parameter in your CONFIG member/global file to point to your customized configssl.cnf file.

   Note: For z/OS, the path and file name must be an HFS file.
3. Configure CA XCOM Data Transport to receive remote SSL connections:
   - For z/OS, ensure that TCP/IP support is enabled in CA XCOM Data Transport and specify the TCP/IP port(s) that will accept SSL connection.

     For TCPIPv4, the following values must be set in the CONFIG member:

     - TCPIP=YES
     - SSL={ONLY|ALLOW}
     - SSLPORT=99999, where 99999 is a site defined port

     For TCPIPv6, the following values must be set in the CONFIG member:

     - TCPIP=YES
     - SSL={ONLY|ALLOW}
     - TCPIPV6={ONLY|ALLOW}
     - SSLPORTV6=99999, where 99999 is a site defined port

     Note: For the CA XCOM CONFIG member value combinations to determine which TCP/IP listeners will be started, see the table below.
   - For UNIX, during installation, manually add the txpis and/or txpis6 services and the TCP/IP port(s) that will accept SSL connection requests to the inetd configuration files.
   - For Windows, specify the TCP/IP port that that will accept SSL connection requests using the SSL Port Number on the TCP/IP tab in the Global Parameters GUI.
   - For Windows, specify the TCP/IP port that that will accept SSL connection requests, using the SSL Port Number and/or the Server IPv6 Port Number on the TCP/IP tab in the Global Parameters GUI. If using IPv6 support, ensure that the Choose Listener drop-down box under the Server IPv6 Port Number indicates the correct listener(s) to start.

4. Verify that the port that receives incoming SSL connections is a unique port that is not in use by any other application. The port used for incoming TCP/IP connections cannot also be used for incoming SSL connections. If CA XCOM Data Transport will be receiving both incoming TCP/IP connections and incoming SSL connections, then two ports are required.
5. For z/OS, restart the CA XCOM Data Transport server (started task).
6. For UNIX and Windows, restart the CA XCOM Data Transport service.

Default Options Table Parameter Values for TCP/IP Listeners

TCPIP= Value	SSL= Value	TCPIPV6= Value	IPv4 Listeners	IPv6 Listeners
YES	ALLOW	NONE	Non-secure and SSL	No
YES	ONLY	NONE	SSL only	No
YES	NONE	ALLOW	Non-secure only	Non-secure only*
YES	ALLOW	ALLOW	Non-secure and SSL	Non-secure and SSL*
YES	ONLY	ALLOW	SSL only	SSL only*
YES	NONE	ONLY	No	Non-secure only*
YES	ALLOW	ONLY	No	Non-secure and SSL*
YES	ONLY	ONLY	No	SSL only*

* XCOM TCP/IPv6 listeners can handle TCP/IPv4 connections as well as TCP/IPv6 connections.

Configure the Client

When you configure the client, you enable the CA XCOM Data Transport client to use the CA certificate and the server certificate when establishing client (local) SSL connections.

To configure the client

1. Review and modify the settings of the CA XCOM Data Transport SSL configuration file, configssl.cnf, as appropriate for your system. Client connections use the INITIATE_SIDE values.
2. Point the XCOM_CONFIG_SSL parameter in your CONFIG member/global file to your customized configssl.cnf file.

   Note: For z/OS, the path and file name must be an HFS file.

   - For z/OS, the XCOM_CONFIG_SSL parameter can also be specified as a destination member parameter.
   - For UNIX and Windows, the XCOM_CONFIG_SSL parameter can also be specified in your configuration (cnf) file.
3. Set the SECURE_SOCKET parameter to YES to indicate an SSL connection.
   - For z/OS, specify the SECURE_SOCKET parameter in the SYSIN01, the destination member, or the CONFIG member.
   - For UNIX and Windows, specify the SECURE_SOCKET parameter in the configuration (cnf) file.
4. Specify the port through which the remote CA XCOM Data Transport partner accepts SSL connections. Use one of the following parameters:
   - PORT for UNIX and Windows
   - IPPORT for z/OS
5. Initiate the transfer request.

Notes:

- CA XCOM Data Transport uses the OpenSSL toolset. The configssl.cnf file is used by CA XCOM Data Transport to configure OpenSSL.
- Set the XCOM_CONFIG_SSL parameter in your CONFIG member/global file to point to your customized configssl.cnf file.

Sample configssl.cnf File

```
#########################################################################
#                                                                       #
# This file is used by CA XCOM Data Transport to                        #
#  configure OpenSSL                                                    #
#                                                                       #
# Mandatory means that the parameter must contain a value:              #
#                                                                       #
#   # Mandatory                                                         #
#   [VERIFY_DEPTH]                                                      #
#   INITIATE_SIDE = 4                                                   #
#   RECEIVE_SIDE  = 4                                                   #
#                                                                       #
# is correct,                                                           #
#                                                                       #
#   # Mandatory                                                         #
#   [VERIFY_DEPTH]                                                      #
#   INITIATE_SIDE =                                                     #
#   RECEIVE_SIDE  = 4                                                   #
#                                                                       #
# is incorrect, INITIATE_SIDE must not be empty.                        #
#                                                                       #
# For optional sections, INITIATE_SIDE etc. may be empty.               #
#                                                                       #
# For CA XCOM, INITIATE_SIDE is used by the local machine.              #
#              RECEIVE_SIDE is used by the remote partner.              #
#                                                                       #
# Note: The directory and file names used in this sample refer to       #
#       the directories and files created using the makeca,             #
#       makeserver and makeclient sample scripts.  If the sample        #
#       scripts have been unloaded to a location other than             #
#       /usr/spool/xcom/ssl then these names will need to be            #
#       updated.                                                        #
#                                                                       #
#########################################################################

# Mandatory, note that CA XCOM uses the v3 protocol
[SSL_OPTION]
INITIATE_SIDE = SSL_OP_ALL|SSL_OP_NO_SSLv2
RECEIVE_SIDE  = SSL_OP_ALL|SSL_OP_NO_SSLv2|SSL_OP_SINGLE_DH_USE

# Mandatory
[VERIFY_DEPTH]
INITIATE_SIDE = 4
RECEIVE_SIDE  = 4

# Mandatory, note that CA XCOM uses the v3 protocol
[SSL_METHOD]
INITIATE_SIDE = v3
```

```
RECEIVE_SIDE  = v3

# Optional
[CIPHER]
INITIATE_SIDE = ALL:!AES:!ADH:!LOW:!EXP:MD5:@STRENGTH
RECEIVE_SIDE  = ALL:!AES:!ADH:!LOW:!EXP:!MD5:@STRENGTH

# Optional, specifies the method used for encryption by the 3DES cipher
# on the z/OS platform.
# The OpenSSL implementation used by XCOM provides 3DES encryption
# using a software encryption routine.  Due to the complex nature of
# 3DES, the software encryption routine may consume a large amount
# of CPU.  Using a compression routine (XCOM COMPRESS= parameter) may
# help limit the CPU by decreasing the amount of data passed through
# the software encryption routine.
# To further reduce the CPU usage required by 3DES, this section allows
# the software encryption routine to be replaced by call to ICSF
# for cryptographic coprocessor support.
# Options:
# CLEAR - stores the symmetric keys in clear text in memory during the
#         transfer and use the ICSF CSNBSYE/CSNBSYD encryption
#         functions.
# NO (default) - use the OpenSSL software encryption routine
# Notes:
# * Applies to 3DES on the z/OS platform only.
# * CLEAR requires that ICSF be installed and active on the system
#   with a cryptographic coprocessor.
[ICSF]
INITIATE_SIDE = NO
RECEIVE_SIDE  = NO

# Optional, specifies that the certificates are located in your site's
# security product.  The KEYRING and LABLCERT are passed to security to
# identify the keyring that contains the certificates.
# If specified, these values will override the values specified in the
# sections: CA, CA_DIRECTORY, CERTIFICATE and PRIVATEKEY.
# KEYRING and LABLCERT are only used by the z/OS platform.
[KEYRING]
INITIATE_SIDE =
RECEIVE_SIDE  =

# Optional, specifies that the certificates are located in your site's
# security product.  The KEYRING and LABLCERT are passed to security to
# identify the keyring that contains the certificates.
# If specified, these values will override the values specified in the
# sections: CA, CA_DIRECTORY, CERTIFICATE and PRIVATEKEY.
# KEYRING and LABLCERT are only used by the z/OS platform.
[LABLCERT]
```

```
INITIATE_SIDE =
RECEIVE_SIDE  =

# Mandatory
[CA]
INITIATE_SIDE = /usr/spool/xcom/ssl/certs/cassl.pem
RECEIVE_SIDE  = /usr/spool/xcom/ssl/certs/cassl.pem

# Mandatory
[CA_DIRECTORY]
INITIATE_SIDE = /usr/spool/xcom/ssl/certs
RECEIVE_SIDE  = /usr/spool/xcom/ssl/certs

# Mandatory
[CERTIFICATE]
INITIATE_SIDE = /usr/spool/xcom/ssl/certs/clientcert.pem
RECEIVE_SIDE  = /usr/spool/xcom/ssl/certs/servercert.pem

# Mandatory
[PRIVATEKEY]
INITIATE_SIDE = /usr/spool/xcom/ssl/private/clientkey.pem
RECEIVE_SIDE  = /usr/spool/xcom/ssl/private/serverkey.pem

# Mandatory, YES/NO (if NO, DH will be used)
[RSAKEY]
RECEIVE_SIDE = NO

# Optional (for RSA NO, see above). If RSA NO and DH files empty,
#  then internal program tables will be used.
[DH]
DH_512_RECEIVE_SIDE  =
DH_1024_RECEIVE_SIDE =
DH_2048_RECEIVE_SIDE =
DH_4096_RECEIVE_SIDE =

# Mandatory if a random daemon is not running on the system.
#  Length is set to the file length or to the number of bytes
#  to be read from a urandom device. Length can be -1 (read until EOF)
#  for a disk file but not for a urandom device, OpenSSL would read
#  until EOF which will never be reached for a urandom device.
#  If a length of -1 is specified for a file name containing "dev" then an error will occur.
# Optional if a random daemon is running on the system.
[RANDOM]
INITIATE_SIDE_FILE = /usr/spool/xcom/ssl/random.pem
INITIATE_SIDE_LENGTH = 1024
RECEIVE_SIDE_FILE = /usr/spool/xcom/ssl/random.pem
RECEIVE_SIDE_LENGTH = 1024

# Mandatory if local certificates were created with passwords.  If local
```

```
# certificates were not created with passwords, the value will be ignored.
# PASSWORD has to match the password used when generating the certificates.
[PASSWORD]
INITIATE_SIDE = password
RECEIVE_SIDE  = password

# Mandatory, YES/NO
[VERIFY_CERTIFICATE]
INITIATE_SIDE = YES
RECEIVE_SIDE  = YES

# Mandatory, YES/NO
[VERIFY_MACHINE]
INITIATE_SIDE = NO
RECEIVE_SIDE  = NO

# Optional, matches against the Subject Alternative Name DNS: field in the certificate
# HOST_NAME can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx host name matches the certificate sent by the receive side
# RECEIVE_SIDEx host name matches the certificate sent by the initiate side
[HOST_NAME]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the Serial Number: field in the certificate
# SERIAL_NUMBER can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx serial number matches the certificate sent by the receive side
# RECEIVE_SIDEx serial number matches the certificate sent by the initiate side
[SERIAL_NUMBER]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the name= field in the certificate
# NAME can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx name matches the certificate sent by the receive side
# RECEIVE_SIDEx name matches the certificate sent by the initiate side
[NAME]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the title= field in the certificate
# TITLE can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx title matches the certificate sent by the receive side
# RECEIVE_SIDEx title name matches the certificate sent by the initiate side
[TITLE]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the description= field in the certificate
```

```
# DESCRIPTION can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx description matches the certificate sent by the receive side
# RECEIVE_SIDEx description matches the certificate sent by the initiate side
[DESCRIPTION]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the emailAddress= field in the certificate
# EMAIL can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx email matches the certificate sent by the receive side
# RECEIVE_SIDEx email matches the certificate sent by the initiate side
[EMAIL]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the OU= field in the certificate
# ORGANIZATIONAL_UNIT_NAME can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx organizational unit name matches the certificate sent by the receive side
# RECEIVE_SIDEx organizational unit name matches the certificate sent by the initiate side
[ORGANIZATIONAL_UNIT_NAME]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the O= field in the certificate
# ORGANIZATION_NAME can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx organization name matches the certificate sent by the receive side
# RECEIVE_SIDEx organization name matches the certificate sent by the initiate side
[ORGANIZATION_NAME]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the L= field in the certificate
# LOCALITY_NAME can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx locality name matches the certificate sent by the receive side
# RECEIVE_SIDEx locality name matches the certificate sent by the initiate side
[LOCALITY_NAME]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the ST= field in the certificate
# STATE_OR_PROVINCE_NAME can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
# INITIATE_SIDEx state or province name matches the certificate sent by the receive side
# RECEIVE_SIDEx state or province name matches the certificate sent by the initiate side
[STATE_OR_PROVINCE_NAME]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =

# Optional, matches against the C= field in the certificate
# COUNTRY_NAME can contain one or more INITIATE_SIDEx and RECEIVE_SIDEx (x=1, 2, 3, etc.)
```

```
# INITIATE_SIDEx country name matches the certificate sent by the receive side
# RECEIVE_SIDEx country name matches the certificate sent by the initiate side
[COUNTRY_NAME]
INITIATE_SIDE1 =
RECEIVE_SIDE1  =
```

Using Certificates with Your Product

Certificates can be stored in one of two places for use by CA XCOM Data Transport for z/OS. They can be placed either in HFS data sets or in the z/OS system's security package. In either case, the certificates are loaded and processed dynamically at the time the secure session is being negotiated with the partner system. This provides flexibility, because certificates can be updated as needed while the CA XCOM Data Transport server remains active. If the certificates are stored in HFS data sets, the CA XCOM Data Transport server or batch job must have sufficient access authority to read these data sets. In this case, there are four relevant parameter sections in the configssl.cnf file which control certificate usage. These sections are:

- [CA]
- [CA_DIRECTORY]
- [CERTIFICATE]
- [PRIVATEKEY]

These sections provide the directory and file names that contain the actual certificate and encryption key data.

If the certificates are stored in one or more KEYRINGs that are maintained by the z/OS system's security package, the server or batch job must run with authority to use the appropriate KEYRING to which the certificates have been loaded. In this case, the required KEYRING is referenced in the [KEYRING] section in the configssl.cnf member. If a certificate other than the default is to be used, specify the certificate label in the configssl.cnf section [LABLCERT].

If the INITIATE_SIDE or RECEIVE_SIDE parameters are provided in the [KEYRING] section of a configssl.cnf data set, the four sections pertaining to accessing HFS files are ignored for the type of transfer to which the parameter applies. In other words, the INITIATE_SIDE parameter applies the KEYRING data to locally initiated transfers and the RECEIVE_SIDE parameter applies the KEYRING data to remotely initiated transfers only.

For more information about defining digital certificates to your z/OS security system, see the documentation for your particular security software:

- For CA Top Secret, see the *CA Top Secret for z/OS Cookbook*.
- For CA ACF2, see the *CA ACF2 for z/OS Cookbook*.
- For RACF, see IBM's *z/OS Security Server RACF Security Administrator's Guide*.

No matter where the certificates are stored, the server or batch job must run with the appropriate system and security definitions needed to create a UNIX System Services (USS) environment to run under.

Chapter 7: Utilizing zIIP

This chapter describes how to utilize zIIP processors with CA XCOM Data Transport to help reduce CPU utilization costs.

This section contains the following topics:

What is zIIP?

zIIP is a special processor that is restricted to executing specific types of SRB mode work. It is intended to free up general computing capacity and lower the overall cost of computing for CPU intensive workloads.

What features can run on zIIP?

CA XCOM Data Transport will attempt to offload data compression and de-compression functions to a zIIP when available. Given that data compression and de-compression accounts for a significant portion of CPU utilization for a data transfer, moving this processing to a zIIP will reduce utilization and costs.

Enabling zIIP

Enabling CA XCOM Data Transport to enable zIIP support requires that the installation have CA Common Services for z/OS installed at the r11 release or higher. An APAR is required to be installed for both the r11 and r12 releases of CA Common Services for z/OS which introduces the zIIP Enablement Service. For the r11 release, APAR RO27636 must be installed. For the r12 release, RO27110 must be installed. The common services library must be available to the XCOM Server started task or XCOMJOB TYPE=EXECUTE job.

By default, CA XCOM Data Transport will enable zIIP support providing that the zIIP Enablement Service is available to load. There is a PARM and configuration parameter which can be specified to disable the zIIP support. The parameter is ZIIP=YES|NO, with YES being the default.

Using zIIP

To insure that data compression and de-compression functions are eligible to be moved to the zIIP without impacting performance, CA XCOM Data Transport requires that the size of the data block being compressed or de-compressed is a minimum of 4096 (4 K) bytes. This is to insure that the overhead of switching the workload to the zIIP does not affect performance of the transfer.

We recommend that the configuration parameter MAXPACK is set to a minimum of 4096. The higher the value for these parameters, the more efficient the use of zIIP is.

For any transfer, PACK=LENGTH is specified to insure that the 4-K minimum size for compression on the zIIP is realized. Without this parameter, the record size of data in a file would need to be 4 K to move the compression or de-compression workload to zIIP.

Managing zIIP

CA XCOM Data Transport provides some facilities for managing and monitoring zIIP usage. The only goal of utilizing zIIP is to reduce CPU utilization and therefore the costs associated with that utilization.

History and SMF Records

A feature of zIIP Enablement Services provides an API to gather usage statistics for both normal and zIIP CPUs. For each transfer statistics about zIIP usage are written into the history record and SMF record. This data consists of six fields which can then be reported on through a TYPE=HISTORY job. Also supplied CA Easytrieve reports, or by way of the ISPF file transfer detail panel.

- Total CPU time used
- Total TCB mode CPU time
- Total SRB mode CPU time (The mode zIIP requires)
- Total zIIP CPU time
- Total zIIP eligible time that is spent on standard CPU
- Total time that is qualified to run on zIIP

For transfers that do not utilize compression, statistics are still gathered. However the numbers for zIIP times are minimal as the process of gathering the zIIP statistics utilizes the zIIP processor. CPU statistics are applicable for these transfers.

ZIIP Command

A new modify command for the CA XCOM Data Transport Server is being provided to both report on status and allow zIIP support to be enabled/disabled.

The ZIIP,STATUS command will provide information about the CPUs (standard and zIIP) as to how many are defined and online. It will also display the status of zIIP support for the various features of CA XCOM Data Transport that can exploit use of zIIP. Currently this is only data compression and de-compression. Finally, statistics will be displayed which show the total amount of processing time that was eligible to run on zIIP and that actually executed on zIIP.

The STATUS command is also issued on startup and termination of the CA XCOM Data Transport Server and XCOMJOB TYPE=EXECUTE start and termination.

The ZIIP,ENABLE and ZIIP,DISABLE commands allow an administrator to either enable or disable the zIIP support for future data transfers. Any transfers currently in progress will continue to operate in the state that was active at the start of the transfer.

Error Handling

In the event of an abend that occurs while the transfer is running on a zIIP, CA XCOM Data Transport will handle the abend, disable future use of zIIP, and restart the transfer from the last checkpoint. On the restart, zIIP will not be used due to the disabling of the support.

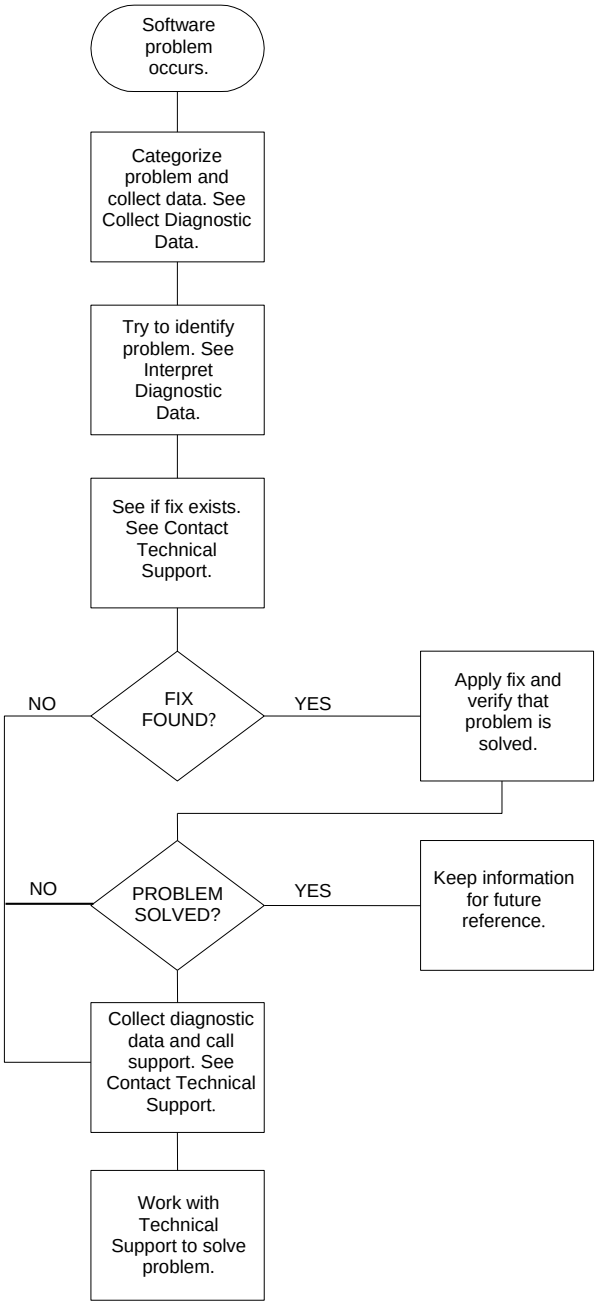
Chapter 8: Troubleshooting

This chapter contains information about identifying and resolving problems and contacting CA Technical Support.

This section contains the following topics:

Diagnostic Procedures

The flowchart below summarizes the procedures to follow if you have a problem with a CA software product. These procedures are detailed on the following pages.



Collect Diagnostic Data

In the list below, use the heading to categorize the problem your site has encountered. Then, follow the corresponding instructions to generate useful diagnostic data.

Installation

Review the installation procedures in the *Installation Guide*.

CA XCOM Data Transport abends

Contact CA Technical Support to see if the abend is a known problem or if an unformatted SVC dump is required. Save any dumps, CA XCOM Data Transport logs, and associated SYSLOGs you may already have.

Note: When using Abend-AID and an SVC dump is requested by CA XCOM Data Transport support, turn off Abend-AID dumps by adding the following statement in the CA XCOM Data Transport Server JCL:

```
//ABNLIGNR  DD  DUMMY
```

Collect Diagnostic Data about the XCOMPLEX

To gather information about the XCOMPLEX, issue the STAT MODIFY command (see Using the CA XCOM Data Transport MODIFY Commands in the chapter “Operation and Control” in the *CA XCOM Data Transport for z/OS User Guide*). The output of this command verifies that the XCOMPLEX is in communication with the Coupling Facility and validates the installation and configuration of the XCOMPLEX Admin Servers and XCOMPLEX Worker Servers. This information is displayed in the XCOMLOG.

If there is no response from the STAT command, verify that the XCOMPLEX parameter is set correctly for each XCOMPLEX Worker Server in the XCOMPLEX, by checking the value for that parameter in the XCOMLOG when the server comes up.

Collect Diagnostic Data about the PLEXQ

To gather information about the PLEXQ, issue the STAT MODIFY command (see Using the CA XCOM Data Transport MODIFY Commands in the chapter Operation and Control in the CA XCOM Data Transport for z/OS User Guide). The output of this command verifies that the PLEXQ is in communication with the SYSPLEX signaling services and validates the installation and configuration of the PLEXQ group servers. This information is displayed in the XCOMLOG.

Verify that the PLEXQ parameter is set correctly for each Worker Server in the PLEXQ group if there is no response from the STAT command. Check the value for the parameter in the XCOMLOG when the server comes up to verify that the PLEXQ parameter is set correctly. Also verify that the server is connected to the PLEXQ by checking the XCOMLOG for message XCOMM1101I CONNECTED TO PLEXQ GROUP xxxxxxxx.

Collect Diagnostic Data for ISPF Panel Problems

Use this trace only when requested by CA Technical Support.

To collect a trace of the CA XCOM Data Transport ISPF panels, enter

```
DODEBUG
```

at the command prompt on the Primary Option Menu panel. The following message

```
DODEBUG ACKNOWLEDGED
```

displays at the top right-hand corner of the panel, and a diagnostic trace is written to the screen. To stop the trace, exit the Primary Options Menu panel.

Interpret Diagnostic Data

When you have collected the specified diagnostic data, write down your answers to the following questions:

1. What was the sequence of events prior to the error condition?
2. What circumstances existed when the problem occurred and what action did you take?
3. Has this situation occurred before? What was different then?
4. Did the problem occur after a particular PTF was applied or after a new release of the software was installed?
5. Have you recently installed a new release of the operating system?
6. Has the hardware configuration (tape drives, disk drives, and so on) changed?

From your response to these questions and the diagnostic data, try to identify the cause and resolve the problem.

Troubleshoot Sending Reports with PSO

The following list shows problems and possible solutions for sending reports with PSO:

Reports just do not go

Possible solutions:

- Verify that PSO support is enabled for the CA XCOM Data Transport server. PSO can be disabled by specifying PSO=NO on the XCOMXFER PARM.
- Check the XCOMLOG and the JES Log for any messages.
- Disable and reenable the dest member
- Display the report on the JES Spool. Make sure that the dest member matches the attributes of the report on the JES Spool. If the REPORT is specified as DEST= in the destination member, it must show up as DEST on the SPOOL. If WRITER in the dest member, WRITER must show up. The writer or dest names must match.
- List the destination member:

  ```
  /F xcomprocname,LIST,destmembername
  ```

  The output is written into the XCOM log.
- Reset the destination member:

  ```
  /F xcomprocname,RESET,*
  ```
- Review security. If XCOM does not have the proper permissions, it cannot pick up any reports. It may not even be able to detect that there are any reports out there.
- Make sure that reports are not queued in a held class.
- Check the XCOMLOG and the JES Log for any messages.
- Depending on the problem, check the PSOVOL for space or security problems. PSODISP=KEEP can cause temporary data sets to build up.

The reports are picked up, but they do not reach the partner

Possible solutions:

- Check the XCOMLOG and the JES Log for any messages.
- Depending on the problem, check the PSOVOL for space or security problems. PSODISP=KEEP can cause temporary data sets to build up.

Appendix A: CA XCOM Data Transport Health Checks

This appendix describes health checks for CA XCOM Data Transport. The product owner for all CA XCOM Data Transport health checks is CA_XCOM.

This section contains the following topics:

Parameter Overrides for CA XCOM Data Transport Health Checks

The IBM Health Checker for z/OS allows the override of selected default parameters by specifying desired defaults on the POLICY statement in the HZSPRM*xx* member of parmlib. This is useful in changing such values as INTERVAL to a value more appropriate for your installation. Individual checks can also be written to support parameter overrides using the PARM( ) parameter on the POLICY statement. For a complete list of parameters that can be overridden see the chapter "Managing Checks" in the *IBM Health Checker for z/OS User Guide*.

These parameters can also be overridden by using the MODIFY command to pass the desired parameters to the IBM Health Checker for z/OS started task.

```
F hzsproc,UPDATE,CHECK(check_owner,check_name),PARM='chkparm'
```

Example

```
F HZS,UPDATE,CHECK(CA_XCOM, XCOM_ABOVE_16M@stcname),PARM='THRESHOLD(90)'
```

If a CA XCOM Data Transport check supports the override of a parameter, it is documented in the Parameters Accepted section in each check that follows. The default interval is provided in the Description section for each CA XCOM Data Transport check.

XCOM_ABOVE_16M@stcname

Description

This check monitors the amount of above-the-line storage that has been allocated within the CA XCOM Data Transport address space. The purpose of this check is to be able to provide an alert when above-the-line storage is being depleted beyond the configured threshold. The default interval for this check is every 30 minutes.

Best Practice

Provide sufficient above-the-line storage to the CA XCOM Data Transport region commensurate with the amount of concurrent file transfer activity that the region is expected to process. For guidance in calculating the appropriate virtual storage allocation, see the Storage Usage Worksheet, which is updated and made available for each new release of CA XCOM Data Transport.

Parameters Accepted

This check accepts one parameter, THRESHOLD(nnn). The value specified in THRESHOLD(nnn) is used to determine the percentage of above-the-line storage that can be allocated before a resource shortage exception is recognized. This parameter must be an integer in the range of 1 to 100. The default is THRESHOLD(80).

Reference

The current versions of the Storage Usage Worksheets for supported releases of CA XCOM Data Transport are available on the support.ca.com web site.

Exception Message

XCMH503W Available storage above-the-line is less than nnn% of the total amount.

Storage utilization has exceeded nnn% of the amount available to the CA XCOM Data Transport region. Storage shortages within a CA XCOM Data Transport region can cause unpredictable results including, but not limited to, miscellaneous abends, file I/O errors, and network communication errors. The storage shortages could cause failed data transmissions or abnormal termination of the region itself.

For more information, see the chapter Health Check Messages in the *CA XCOM Data Transport for z/OS Message Reference Guide*.

XCOM--XCOM_BELOW_16M@stcname

Description

This check monitors the amount of below-the-line storage that has been allocated within the CA XCOM Data Transport address space. The purpose of this check is to be able to provide an alert when below-the-line storage is being depleted beyond the configured threshold. The default interval for this check is every 30 minutes.

Best Practice

Provide sufficient below-the-line storage to the CA XCOM Data Transport region commensurate with the amount of concurrent file transfer activity that the region is expected to process. For guidance in calculating the appropriate virtual storage allocation, refer to the Storage Usage Worksheet, which is updated and made available for each new release of CA XCOM Data Transport.

Parameters Accepted

This check accepts one parameter, THRESHOLD(nnn). The value specified in THRESHOLD(nnn) is used to determine the percentage of below-the-line storage that can be allocated before a resource shortage exception is recognized. This parameter must be an integer in the range of 1 to 100. The default is THRESHOLD(80).

Reference

The current versions of the Storage Usage Worksheets for supported releases of CA XCOM Data Transport are available on the support.ca.com web site.

Exception Message

XCMH503W Available storage below-the-line is less than nnn% of the total amount.

Storage utilization has exceeded nnn% of the amount available to the CA XCOM Data Transport region. Storage shortages within a CA XCOM Data Transport region can cause unpredictable results including, but not limited to, miscellaneous abends, file I/O errors, and network communication errors. The storage shortages could cause failed data transmissions or abnormal termination of the region itself.

For more information, see the chapter Health Check Messages in the *CA XCOM Data Transport for z/OS Message Reference Guide*.

XCOM_MAXTASK_LEVEL@stcname

Description

This check monitors the total number of tasks that are active within the CA XCOM Data Transport region. The purpose of this check is to provide an alert when the number of active tasks reaches a threshold percentage of the maximum allowed by the MAXTASK parameter. The default interval for this check is every 30 minutes.

Best Practice

The MAXTASK value should be set to a level that will not allow more concurrent transfers to run than there are system resources to support. It is necessary to ensure that sufficient system resources (particularly virtual storage) are available to enable the number of transfers specified by MAXTASK to run concurrently.

Parameters Accepted

This check accepts one parameter, THRESHOLD(*nnn*). The value specified in THRESHOLD(*nnn*) is used to determine how close to the MAXTASK limit the total number of active tasks can be before an exception is recognized. This parameter must be an integer in the range of 1 to 100. The default is THRESHOLD(80).

Reference

For information about tuning the MAXTASK parameter, see the chapter "Configuration Parameters."

Exception Message

XCMH513W The total number of active tasks within the CA XCOM Data Transport region has exceeded *nnn*% of the configured limit as specified on the MAXTASK parameter.

The MAXTASK configuration parameter limits the number of active tasks that are allowed in the CA XCOM Data Transport region at any given time. An exception condition is recognized when the active task level reaches a certain threshold percentage of that limit. The default threshold percentage is 80%. This threshold may be modified by specifying a different percentage via the "THRESHOLD(*nnn*)" parameter for this particular health check.

When a CA XCOM Data Transport region reaches the MAXTASK limit, additional file transfer requests are delayed and retried at a later time. This may result in unacceptable processing delays for new file transfer requests. The overhead required to manage this exception condition may degrade the overall performance of the CA XCOM Data Transport region and adversely affect its ability to complete existing file transfer requests.

For more information, see the chapter "Health Check Messages" in the *CA XCOM Data Transport for z/OS Message Reference Guide*.

XCOM_MAXLOC_LEVEL@stcname

Description

This check monitors the number of locally-initiated tasks that are active within the CA XCOM Data Transport region. The purpose of this check is to provide an alert when the number of locally-initiated active tasks reaches a threshold percentage of the maximum allowed by the MAXLOC parameter. The default interval for this check is every 30 minutes.

Best Practice

The MAXLOC value should be set to a level that does not allow more concurrent transfers to run than there are system resources to support and that does not exceed the MAXTASK parameter.

Parameters Accepted

This check accepts one parameter, THRESHOLD(*nnn*). The value specified in THRESHOLD(*nnn*) is used to determine how close to the MAXLOC limit the total number of active tasks can be before an exception is recognized. This parameter must be an integer in the range of 1 to 100. The default is THRESHOLD(80).

Reference

For information about tuning the MAXLOC parameter, see the chapter "Configuration Parameters."

Exception Message

XCMH523W The total number of locally initiated active tasks within the CA XCOM Data Transport region has exceeded *nnn*% of the configured limit as specified on the MAXLOC parameter.

The MAXLOC configuration parameter limits the number of locally initiated active tasks that are allowed in the CA XCOM Data Transport region at any given time. An exception condition is recognized when the locally initiated active task level reaches a certain threshold percentage of that limit. The default threshold percentage is 80%. This threshold may be modified by specifying a different percentage via the THRESHOLD(*nnn*)" parameter for this particular health check.

When a CA XCOM Data Transport region reaches the MAXLOC limit, additional locally initiated file transfer requests are delayed and retried at a later time. This may result in unacceptable processing delays for new file transfer requests. The overhead required to manage this exception condition may degrade the overall performance of the CA XCOM Data Transport region and adversely affect its ability to complete existing file transfer requests.

For more information, see the chapter "Health Check Messages" in the *CA XCOM Data Transport for z/OS Message Reference Guide*.

XCOM_MAXREM_LEVEL@stcname

Description

This check monitors the number of remotely-initiated tasks that are active within the CA XCOM Data Transport region. The purpose of this check is to provide an alert when the number of remotely-initiated active tasks reaches a threshold percentage of the maximum allowed by the MAXREM parameter. The default interval for this check is every 30 minutes.

Best Practice

The MAXREM value should be set to a level that does not allow more concurrent transfers to run than there are system resources to support and that does not exceed the MAXTASK parameter.

Parameters Accepted

This check accepts one parameter, THRESHOLD(*nnn*). The value specified in THRESHOLD(*nnn*) is used to determine how close to the MAXREM limit the total number of active tasks can be before an exception is recognized. This parameter must be an integer in the range of 1 to 100. The default is THRESHOLD(80).

Reference

For information about tuning the MAXREM parameter, see the chapter "Configuration Parameters."

Exception Message

XCMH533W The total number of remotely initiated active tasks within the CA XCOM Data Transport region has exceeded *nnn*% of the configured limit as specified on the MAXREM parameter.

The MAXREM configuration parameter limits the number of remotely-initiated active tasks that are allowed in the CA XCOM Data Transport region at any given time. An exception condition is recognized when the remotely-initiated active task level reaches a certain threshold percentage of that limit. The default threshold percentage is 80%. This threshold may be modified by specifying a different percentage via the THRESHOLD(*nnn*)" parameter for this particular health check.

When a CA XCOM Data Transport region reaches the MAXREM limit, additional remotely-initiated file transfer requests are delayed and retried at a later time. This may result in unacceptable processing delays for new file transfer requests. The overhead required to manage this exception condition may degrade the overall performance of the CA XCOM Data Transport region and adversely affect its ability to complete existing file transfer requests.

For more information, see the chapter "Health Check Messages" in the *CA XCOM Data Transport for z/OS Message Reference Guide*.

Index

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

W

X