BROADCOM®

MAINFRAME SOFTWARE

XCOM™ Data Transport® for UNIX/Linux 11.6.1

Table of Contents

Announcements and News

News about your product, conferences, and community events.

Mainframe Technical Exchanges: March, June and October 2025

Join us for the Mainframe Technical Exchanges in Prague, Czech Republic (March 25-27), virtually (June 24-26), and in Plano, Texas (October 15-17). Connect with Mainframe Experts who will share the latest technical education and product demos and respond to your questions and feedback. These educational events are a great way to network with peers and experts from across the globe.

Bookmark this page, Join Us for the Mainframe Technical Exchanges , for current registration and event information.

Third-party Software Agreements

Third-party software agreement information for your product is now located in the Broadcom Legal Notices site. Select your product from the dropdown to view all legal notices.

2024 Virtual Mainframe Technical Exchange

We held this live event October 8 - 10, 2024. However, you can still connect with Broadcom mainframe technical experts for educational sessions and product updates virtually. To access the session recordings and view them on demand, Register Here.

You can view the recorded sessions through **February 2025.**

Security Advisories Consolidated CSV or JSON File

Broadcom offers a .CSV and a .JSON file that you can download using HTTPS or Secure FTP. These files contain a consolidated list of security advisories affecting all supported Broadcom mainframe products. These files let you easily search the Common Vulnerabilities and Exposures (CVE) information. You can also access the Security Advisory articles that include more details and context about the security or integrity exposure. Broadcom updates this file daily. For download instructions, see this Broadcom Support article (login required).

Release Notes

Includes new feature descriptions and product compatibility details.

The release notes explain the key features and details for XCOM Data Transport for UNIX/Linux (XCOM Data Transport) 11.6.

XCOM Data Transport is a flexible data transfer solution that quickly and securely moves business-critical data across a wide variety of heterogeneous platforms. This product provides compression and record-packing capabilities to extend system resources, improve bandwidth, and enhance enterprise efficiency.

With its reliable delivery and automated transmission recovery features, XCOM Data Transport helps maintain data integrity throughout the processing environment. This product can meet enterprise security standards with robust data encryption features. This product also provides auditing capabilities to ensure that teams remain informed and ready to respond to potential issues.

New Features

The new features in this XCOM Data Transport release offer you increased flexibility and efficiency.

Release 11.6 provides the following enhancements. To ensure availability of all features and fixes, ensure that your product is current on maintenance.

Enhanced Splunk Reporting Capability

This enhancement is provided in PTF LU15723.

XCOM Data Transport can now send critical alerts to the Splunk platform. You can now view XCOM Scheduler service alerts, failed transfer alerts, and their corresponding error messages through a Splunk dashboard. Previously, you could only view general transfer activity. The addition of these critical alerts allows you to use Splunk as a single point of reference for monitoring your XCOM data transfers.

As part of this enhancement, a new sample dashboard is available. The **XCOM Service Critical Alerts** dashboard displays critical alerts and error messages from all XCOM UNIX/Linux and Windows servers that have been integrated with Splunk. This dashboard is provided in an updated `xcom.spl` file.

The connection between XCOM and Splunk has also changed. Previously, the XENDCMD exit was used with a Java client to invoke the Splunk connection. Now, a new `xcomanalytics.cnf` file provides the parameters to connect with Splunk, and no Java client is needed. The new configuration file lets you specify exactly what transfer information to send to Splunk, which can result in cost savings for your organization. This connection method also enables XCOM to send data about transfers that fail to start. The previous connection method only reported on transfers that were able to start.

The new `xcomanalytics.cnf` file contains the following new parameters:

- ANALYTICS_API
- ANALYTICS_FIELDS
- ANALYTICS_HOST
- ANALYTICS_HTTP_AUTH_HEADER
- ANALYTICS_HTTP_CUST_HEADER
- ANALYTICS_PORT
- ANALYTICS_SCHEME

The `XCOM.GLB` global defaults file also has the following new parameters:

- ANALYTICS_CNFFILE
- ENABLE_ANALYTICS

For descriptions of these parameters, see the Reference section of this document.

For information about integrating XCOM with Splunk, see Integrate with Splunk Dashboards.

For information about the Splunk dashboards, see View the Sample XCOM Dashboards in Splunk.

Unicode and Multi-Byte Character Set Support for Data Transfer

Before the advent of Unicode, a significant number of character sets were devised to permit the representation of symbols used in the Chinese, Japanese, Korean, and Taiwanese (CJK) languages. Today, Unicode is favored and there is an ongoing transition from these legacy character sets to Unicode encodings, most notably UTF-8 and UTF-16.

Many CJK legacy multibyte character sets are ASCII based, as is the case for the most commonly used Unicode encodings (as an example, UTF-8, UTF-16).

In the IBM mainframe (predominantly EBCDIC) world however composite character sets are commonly employed, involving a Shift-in/Shift-out encoding method. This encoding mechanism enables a single-byte ASCII or EBCDIC character-set to be used for the representation of Latin characters, in tandem with a multibyte character set for the representation of non-Latin characters. Shift-in and shift-out control characters are then inserted in the data stream to signal a switch between the two embedded character sets. The CCSID 937 character set combines an EBCDIC single byte character-set with a Traditional Chinese multibyte character set. While the CCSID 938 character set combines an ASCII single byte character-set with the same Traditional Chinese multibyte character set.

CA XCOM Data Transport currently performs data transfers utilizing one of three data formats – ASCII, EBCDIC, or Binary.

This enhancement allows for transmission of text files that are encoded using multi-byte character sets, including in-flight conversion of data between different character sets. Two additional data formats can be specified for the CODE_FLAG parameter to allow for transmission of these files. In addition, new parameters have been added to the CA XCOM Data Transport global parameters and configuration parameters. These parameters allow you to specify the local and remote character sets to be used for file data conversion and actions for dealing with unconvertible characters.

CA XCOM Data Transport is utilizing the ICU (International Components for Unicode) toolkit to perform data conversion functions. For information on the ICU toolkit, please refer to the ICU website http://site.icu-project.org/.

The CODE_FLAG parameter allows for two new data formats – UTF8 and UTF16. When one of these formats is specified for a transfer, data is converted to that format for transmission to the remote partner.

The LOCAL_CHARSET and REMOTE_CHARSET parameters specify the character-set of the local and remote files for the transfer. These parameters are used in conjunction with CODE_FLAG=UTF8 or CODE_FLAG=UTF16 to perform the conversion of data. If not specified for the transfer, they default to the value specified for the DEFAULT_CHARSET global parameter.

In order to handle conversion issues between character sets, additional parameters MBCS_INPUTERROR and MBCS_CONVERROR specify what action is taken in the event of a character being encountered that cannot be converted. The sending partner uses MBCS_INPUTERROR and specifies to either replace the character with a replacement character or fail the transfer. The receiving partner uses MBCS_CONVERROR and specifies to either replace the character with a replacement character or fail the transfer. If not specified the value of DEFAULT_INPUTERROR and DEFAULT_CONVERROR global parameters will be used.

Parameters LOCAL_DELIM and REMOTE_DELIM specify the encoding scheme that the corresponding character-set uses and a list of delimiters which exists within the data as record separators.

New Global Parameters

Global parameters are added for Unicode transfers.

DEFAULT_CHARSET

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

DEFAULT_CONVERROR

This parameter specifies the appropriate action when the input file contains nonconvertible characters. This is because the characters are not included within the output character sets character repertoire.

DEFAULT_DELIM

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

Also, DEFAULT_DELIM specifies a colon-separated list of record delimiters that are used to mark and detect the end of a record.

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

DEFAULT_INPUTERROR

This parameter specifies the appropriate action when the input file contains data that is inconsistent with the input character set.

XCOM_ICUPATH

This parameter specifies the path to ICU shared libraries icudata and icuuc.

New Configuration Parameters

Configuration parameters are added for Unicode transfers.

LOCAL_CHARSET

This parameter specifies the local character set that XCOM Data Transport uses for Unicode transfers (CODE_FLAG=UTF8 or CODE_FLAG=UTF16).

LOCAL_DELIM

This parameter specifies an optional encoding for which the specified LOCAL_CHARSET is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list.

LOCAL_DELIM also specifies a colon separated list of record delimiters that are used to mark and detect the end of a record.

LOCAL_DELIM is used only for UNICODE transfers (CODE_FLAG=UTF8 or CODE_FLAG=UTF16).

MBCS_CONVERROR

This parameter identifies the action when the input file contains nonconvertible characters because they are not included within the output character sets character repertoire.

MBCS_INPUTERROR

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

REMOTE_CHARSET

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

REMOTE_DELIM

This parameter specifies an optional encoding for which the specified REMOTE_CHARSET is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list.

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

Edit Transfer Record Screen

Fields that are modified on the Edit Transfer Record screen for Unicode transfers:

Options Encoding
In addition to the existing options, UTF8 (31k pack) or UTF16 (31k pack) are added.

Fields that are added to the Edit Transfer Record screen for Unicode transfers:

Local System Parameters Character-set
Specifies the local character set that the XCOM Data Transport uses for Unicode transfers (Encoding=UTF8 or Encoding=UTF16).

Local System Parameters Record Delimiter
Specifies an optional encoding for which the specified Local Character-set is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list. Additionally, specifies a colon separated list of record delimiters that are used to mark and detect the end of a record.

Remote System Identification and Parameters Character-set
Specifies the remote character set that XCOM Data Transport uses for Unicode transfers (Encoding=UTF8 or Encoding=UTF16).

Remote System Identification and Parameters Record Delimiter
Specifies an optional encoding for which the specified Remote Character-set is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list. Additionally, specifies a colon separated list of record delimiters that are used to mark and detect the end of a record.

Misc Options Character-set Input Error
Identifies the appropriate action when the input file contains data that is not consistent with the specified input character set.

Misc Options Character-set Convert Error
Identifies the appropriate action when the input file contains characters that cannot be converted as they are not included within the output character sets character repertoire.

Detail History Record Screen

Fields that are modified on the Detail History Record screen for Unicode transfers:

Encoding
In addition to the existing options, UTF8 and UTF16 have been added.

Fields that are added to the Detail History Record screen for Unicode transfers:

Character Set Input Error & Replace Count

For transfers using Unicode encoding scheme, specifies the appropriate action when the input file contains data that is not consistent with the specified input character set. The replace count is the number of characters for which the action was taken. For transfers on z/OS systems, the count is the number of data buffers for which the action was taken.

Character Set Convert Error & Replace Count

For transfers using Unicode encoding scheme, specifies the action when the input file contains characters that cannot be converted. The characters are not included within the output character sets character repertoire. The replace count is the number of characters for which the action was taken. For transfers on z/OS systems, the count is the number of data buffers for which the action was taken.

Character Set

Specifies the character set of the data.

Record Delimiters

Specifies the encoding scheme for the character set and a set of possible delimiters to use for file processing.

Global Parameters Screen

Fields that are added to the Global Parameters screen for Unicode transfers:

Action to Take On Input Character Error

Specifies the default action when the input file contains data that is not consistent with the specified input character set.

Action to Take On Convert Character Error

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

Default Character set

Specifies the default character set XCOM Data Transport uses for Unicode transfers (Encoding=UTF8 or Encoding=UTF16).

Default Delimiter

Specifies default encoding for which the specified Character-set is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list. Additionally, specifies a colon separated list of record delimiters that are used to mark and detect the end of a record.

ICU Path

This parameter specifies the path to ICU shared libraries icudata and icuuc.

Transmission Password Encryption Cipher Selection

The cipher that is used to encrypt the password during transmission is controlled using the TRNENCRL_CIPHER/ STCTRNENCRL_CPIHER and TRNENCRR_CIPHER parameters. Each of these parameters provides a list of ciphers. TRNENCRL_CIPHER/STCTRNENCRL_CPIHER provides the list of requested ciphers for locally initiated connections. TRNENCRR_CIPHER provides a ranked list of permitted ciphers for remotely initiated connections.

To use Transmission Password Encryption Cipher Selection, confirm that:

- Both the local system and remote system support Transmission Password Encryption Cipher Selection
- The transmission protocol that is used is either TCP/IP or Secure TCP/IP (TLS/SSL)

A COMPAT option is provided for you to select the XCOM Data Transport proprietary password encryption algorithm. This algorithm in return provides backward compatibility if Transmission Password Encryption Cipher Selection is not usable.

New Global Parameters

TRNENCRL_CIPHER

This parameter specifies the default list of ciphers that are to encrypt the password fields for locally initiated transfers when the TRNENCRL_CIPHER parameter is not specified.

TRNENCRL_CIPHER

This parameter specifies the default list of ciphers that are to encrypt the password fields for locally initiated transfers when the TRNENCRL_CIPHER parameter is not specified.

TRNENCRR_DHBITS

This parameter specifies the size (in bits) of the prime number that is used during DH (Diffie-Hellman) exchange for remotely initiated transfers or meta-transfers. The exchanged value generates the key that is used to encrypt the password fields transmitted in the XCOM Data Transport header.

New Configuration Parameter

The configuration parameters added for Transmission Password Encryption Cipher Selection.

STCTRNENCRL_CIPHER

This parameter specifies the requested list of ciphers which are used to encrypt the password fields for locally initiated -c5 meta-transfer requests.

TRNENCRL_CIPHER

This parameter specifies the requested list of ciphers that are to encrypt the password fields for locally initiated transfers.

Edit Transfer Record Screen

The field added to the Edit Transfer Record screen for Transmission Password Encryption Cipher Selection.

Misc Options Local Cipher List Specifies the requested list of ciphers that are used to encrypt the password fields for locally initiated transfers.

Global Parameters Screen

Fields added to the Global Parameters screen for Transmission Password Encryption Cipher Selection.

Default Local Cipher List Specifies the Default list of ciphers that are used to encrypt the password fields for locally initiated transfers. Remote Permitted Cipher List Specifies the permitted list of ciphers that are used to encrypt the password fields for remotely initiated transfers. Remote DH Prime Number Size Specifies the size (in bits) of the prime number that is used during DH (Diffie-Hellman) exchange for remotely initiated transfers or meta-transfers. The exchanged value generates the key that is used to encrypt the password fields transmitted in the XCOM Data Transport header.

Enhanced Features

This section lists the feature enhancements for XCOM Data Transport Release 11.6.

Release 11.6

XCOM Data Transport release 11.6 proved the following feature enhancements:

History Search Enhancements

The Get History Records screen has been enhanced:

- An auto-refresh option has been added which allows the display to remain up to date with CA XCOM Data Transport activity.
- The Transfer Request Display can now be sorted ascending or descending based on request number column.
- The columns shown on the Transfer Request Display can be selected and saved. The Last Message is now available as an optional column.
- The Transfer Request Display can now be displayed in a separate window.

In addition, the following additional fields can now be used as search filters:

- The job name that performed or scheduled a transfer
- The volume serial numbers used at the local and remote locations
- The local and remote file names as well as the option to do a case-sensitive search on the file names

See the following history search enhancements:

- New Configuration Parameters
- History Parameters Screen
- Transfer Request Display

New Configuration Parameters

Configuration parameters added for the enhanced history search.

OFILE

This parameter specifies the file name, local, or remote, to match for a history request.

OFILECASE

This parameter specifies whether the specified file name (OFILE parameter) search is case-sensitive.

OJOB

This parameter specifies the invoking job name to match for a history request.

OVOL

This parameter specifies the volser (local or remote) to match for a history request.

History Parameters Screen

The History Parameters screen now includes a refresh button that, when clicked, initiates an auto-refresh of the Transfer Request Display at the interval set (in seconds). When in auto-refresh mode, the Refresh button changes to a Stop button. Auto-refresh mode ends when the user clicks the Stop button or changes the interval to 0.

Fields added to the History Parameters screen for enhanced history search.

Vol Specifies the volser (local or remote) to match for a history request. File Specifies the file name, local, or remote, to match for a history request. Case Sensitive Specifies whether the specified file name search is case-sensitive. Job Name Specifies the invoking job name to match for a history request.

Transfer Request Display

The Req. No. (Request Number) column on the Transfer Request Display now includes an indicator that shows if the transfers listed are sorted in ascending or descending order. The order can be changed by clicking on the Req. No. column.

The Transfer Request Display now includes a Select History Table Columns expandable section which allows the user to select the columns shown in the display. The selection can then be saved for future use. The Last Message is now available as an optional column.

The Transfer Request Display now includes an Unpin button which, when clicked, opens the Transfer Request Display in a separate window. This allows the user to list more transfers than when the display is attached to the History Parameters screen. Closing the Transfer Request Display window or clicking the Pin button returns the display to the History Parameters screen.

Cross Platform Additional Parameters

CA XCOM Data Transport Service Pack 11.6.01 includes support for additional parameters used by other CA XCOM Data Transport partner systems. These parameters will only be honored if supported by the CA XCOM Data Transport partner system.

See the following cross platform additional parameters enhancements:

Modified Configuration Parameters

Configuration parameters modified to support other CA XCOM Data Transport partner systems.

ALLOCATION_TYPE

This parameter adds the following new options for transfers to an IBM mainframe:

REC

Record

DSNTYPE

This parameter adds the following new options for transfers to an IBM mainframe:

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.

NUM_OF_DIR_BLOCKS

The range is now between 0 and 16,777,215.

PRIMARY_ALLOC

The range is now between 0 and 16,777,215.

SECONDARY_ALLOC

The range is now between 0 and 16,777,215.

New Config Parameters

Configuration parameters added to support other CA XCOM Data Transport partner systems.

AVGREC

For a data set created on an IBM mainframe, this parameter specifies the multiplier for Primary and Secondary allocation units when allocating based on the number of records. The record size is based on the value of the LRECL parameter.

COMPRESS_PDS

This parameter controls if, and when, an IBM mainframe PDS dataset gets compressed.

CREATEDELETE

This parameter specifies whether an existing IBM mainframe data set can be deleted and a new data set allocated at the start ofa FILE_OPTION=CREATEtransfer.

EATTR

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

Edit Transfer Record Screen

Fields Modified on the Edit Transfer Record screen to support other CA XCOM Data Transport partner systems.

Remote System Identification and Parameters DSNTYPE

BASIC, LARGE,EXTREQ and EXTPREF have been added as valid options.

Remote System Identification and Parameters Space DIRBLK

The range is now between 0 and 16,777,215.

Remote System Identification and Parameters Space Primary

The range is now between 0 and 16,777,215.

Remote System Identification and Parameters Space Secondary

The range is now between 0 and 16,777,215. Remote System Identification and Parameters Space Unit REC has been added as a valid option.

Fields added to the Edit Transfer Record screen to support other CA XCOM Data Transport partner systems.

Options File Options Delete and Recreate

Specifies whether an existing IBM mainframe data set can be deleted and a new data set allocated at the start ofa FILE_OPTION=CREATE transfer.

Options PDS Compression

Controls if, and when, an IBM mainframe PDS dataset gets compressed.

Remote System Identification and Parameters Average Record Unit

For a data set created on an IBM mainframe, specifies the multiplier for Primary and Secondary allocation units when allocating based on the number of records. The record size is based on the value of the LRECL parameter.

Remote System Identification and Parameters Extended Attributes

Identifies if the dataset can have extended attributes when the dataset is allocated on an IBM mainframe Extended Address Volume (EAV).

Resolved Issues

Review the resolved issues to understand the key fixes in this release.

XCOM Data Transport Service Pack 11.6.01 fixes the following issues. To locate and download the issues, visit the Download Center and search for the fixes.

Fixes for Linux

- XCLX86 PROBLEM 124: SQL0433N WRITING ODBC HISTORY WHEN AVGREC SPECIFIED
- XCLX86 PROBLEM 125: INCOMPLETE HISTF/CSV FILE CREATED ON EXPORT FROM GUI
- XCLX86 PROBLEM 126: XCOMU3019E ON STARTUP OF GUI
- XCLX86 PROBLEM 127: UNICODE GUI TRANSFER WITH TRUNCATE=YES DOES NOT FAIL
- XCLX86 PROBLEM 128: XCOMU3643E ERROR WHEN LOADING CNF/XML FILES INTO THE GUI
- XCLX86 PROBLEM 130: UNICODE: DEFAULT_INPUTERROR IGNORED FOR RECEIVE FILE
- XCLX86 PROBLEM 131: PING: WRONG CIPHER NEGOTIATED
- XCLX86 PROBLEM 133: MISPLACED PARAMETER VALUES WHEN EDITING CNF FILES IN GUI
- XCLX86 PROBLEM 134: LICENSE - SUPPRESS CA LICENSE MESSAGES
- XCLX86 PROBLEM 135: GATEWAY BRIDGE SUPPORT FOR GATEWAY R11.6
- XCLX86 PROBLEM 136: GATEWAY FILE NOT DECRYPTED CORRECTLY
- XCLX86 PROBLEM 137: REMOVE XCOM GLOBAL PARAMETERS FROM XCOMAPI
- XCLX86 PROBLEM 138: API - RUN XCOM QUEUE=NO AS NON-ROOT
- XCLX86 PROBLEM 140: TRANSFER - AVGREC AND EATTR ALWAYS IN XCOM HEADER
- XCLX86 PROBLEM 141: XCOMPRE - LOCAL PREALLOCATION EXIT ERROR SEGMENTATION FAULT
- XCLX86 PROBLEM 142: XCOMTCP CRASHES WITH WRONG STARTDATE FORMAT
- XCLX86 PROBLEM 144: XCOM FAILS TO EXTRACT FILES WITH FILENAME >128 FROM GATEWAY
- XCLX86 PROBLEM 145: GATEWAY EXTRACTIONS AND XCOM ASCII TRANSFERS GET XCOMU1306E
- XCLX86 PROBLEM 146: HISTORY DETAILS COLLAPSE ON AUTO REFRESH OR PIN
- XCLX86 PROBLEM 148: TRANSFER FAILS WITH XCOMU1306E DIGEST DOES NOT MATCH
- XCLX86 PROBLEM 149: XCOM FAILS TO INSERT/EXTRACT FILES FROM GATEWAY ON LINUX
- XCLX86 PROBLEM 150: GATEWAY EXTRACTION FAILS WITH PASSWORD PROTECTED CERTIFICATE
- XCLX86 PROBLEM 151: ONWARD DELIVERY STATUS DISPLAYED AS QUERYHISTORYFAILED
- XCLX86 PROBLEM 156: GATEWAY - XCOMLIB TRANSFER WILL END ABNORMALLY
- XCLX86 PROBLEM 157: SOCKET ERROR IN HISTORY SUBMIT
- XCLX86 PROBLEM 159: FILE MISMATCH WITH CACHE WRITE PROCESSING NFS TIMEOUTS
- XCLX86 PROBLEM 160: CLEANLOG ERROR:PERMISSION DENIED
- XCLX86 PROBLEM 161: DEFUNCT ZOMBIE XCOM PROCESSES REMAIN IN THE PROCESS TABLE
- XCLX86 PROBLEM 162: RETRY INTERVAL NOT HONORED
- XCLX86 PROBLEM 171: GATEWAY BRIDGE SUPPORT FOR GATEWAY V12CA
- XCLX86 PROBLEM 173: OPENSSL SECURITY ADVISORY CVE-2014-0224
- XCLX86 PROBLEM 174: TRANSFER ENDS SUCCESSFULLY WHEN TARGET FILE SYSTEM IS FULL
- XCLX86 PROBLEM 175: XCOMLIB: TEMP FILE NOT RENAMED
- XCLX86 PROBLEM 179: XCOM - SSL 3.0 SECURITY ADVISORY CVE-2014-3566
- XCLX86 PROBLEM 181: RESTART_SUPPORTED=NO AND NUMBER_OF_RETRIES=0 NOT HONORED
- XCLX86 PROBLEM 187: PATCH OPENSSL VULNERABILITIES
- XCLX86 PROBLEM 188: PATCH OPENSSL VULNERABILITIES
- XCLX86 PROBLEM 189: 0 BYTE FILES WITH WILDCARD TRANSFERS
- XCLX86 PROBLEM 191: UPDATE XCOM TO USE LATEST JAVA VERSION JAVA 8
- XCLX86 PROBLEM 192: XCOM 32-BIT MAINTENANCE INSTALLER FAILS TO LAUNCH ON REDHAT5
- XCLX86 PROBLEM 193: ZERO BYTES TRANSFERRED WHEN FIPS_MODE=YES AND CIPHER=DES
- XCLX86 PROBLEM 194: META-TRANSFER HANGS WHEN NON-FIPS CIPHER USED IN FIPS MODE
- XCLX86 PROBLEM 196: REPORT FORMAT IS WRONG WHEN IT CONTAINS IBM PRINTER CODES
- XCLX86 PROBLEM 197: BROWSE BUTTON MISSING TO SAVE CONFIG FILES TO A DIRECTORY
- XCLX86 PROBLEM 198: ALLOW OPTION TO SET NUMBER OF RECORDS DISPLAYED PER PAGE
- XCLX86 PROBLEM 199: GUI HANGS AFTER SUSPENDING A TRANSFER WITH AUTO REFRESH ON
- XCLX86 PROBLEM 200: ERROR WHILE DELETING A SUSPENDED TRANSFER USING GUI
- XCLX86 PROBLEM 201: GUI: NO PROVISION TO REFRESH LOG ENTRIES IN LOG BROWSER
- XCLX86 PROBLEM 202: UTF-8 CODE CONVERSION ADDS AN EXTRA 0XC2 WHEN NO NL DELIM
- XCLX86 PROBLEM 203: UNABLE TO RECEIVE WILDCARD TRANSFERS FROM OLDER RELEASES

Fixes for AIX

- XCURS6 Problem 179: INCOMPLETE HISTF/CSV FILE CREATED ON EXPORT FROM GUI
- XCURS6 Problem 180: XCOMU3019E ON STARTUP OF GUI
- XCURS6 Problem 181: MISPLACED PARAMETER VALUES WHEN EDITING CNF FILES IN GUI
- XCURS6 Problem 182: REMOVE XCOM GLOBAL PARAMETERS FROM XCOMAPI
- XCURS6 Problem 183: API - RUN XCOM QUEUE=NO AS NON-ROOT
- XCURS6 Problem 184: TRANSFER - AVGREC AND EATTR ALWAYS IN XCOM HEADER
- XCURS6 Problem 185: XCOMPRE - LOCAL PREALLOCATION EXIT ERROR SEGMENTATION FAULT
- XCURS6 Problem 186: XCOMTCP CRASHES WITH WRONG STARTDATE FORMAT
- XCURS6 Problem 190: FILE MISMATCH WITH CACHE WRITE PROCESSING NFS TIMEOUTS
- XCURS6 Problem 191: CLEANLOG ERROR:PERMISSION DENIED
- XCURS6 Problem 192: DEFUNCT ZOMBIE XCOM PROCESSES REMAIN IN THE PROCESS TABLE
- XCURS6 Problem 193: RETRY INTERVAL NOT HONORED
- XCURS6 Problem 196: LSLPP -L COMMAND CAUSES COREDUMP DUE TO DESCRIPTION LENGTH
- XCURS6 Problem 204: OPENSSL SECURITY ADVISORY CVE-2014-0224
- XCURS6 Problem 205: MAINTENANCE INSTALLER FAILS WHILE COPYING FILES
- XCURS6 Problem 206: LSLPP -L COMMAND CAUSES COREDUMP DUE TO DESCRIPTION LENGTH
- XCURS6 Problem 208: INCOMPATIBLE WITH IBM COMMUNICATION SERVER 7
- XCURS6 Problem 210: XCOMLIB: TEMP FILE NOT RENAMED
- XCURS6 Problem 211: RESTART_SUPPORTED=NO AND NUMBER_OF_RETRIES=0 NOT HONORED
- XCURS6 Problem 212: XCOM - SSL 3.0 SECURITY ADVISORY CVE-2014-3566
- XCURS6 Problem 213: TRANSFER_NAME IS NOT PASSED TO EXITS WITH REMOTE TRANSFERS
- XCURS6 Problem 214: GATEWAY BRIDGE SUPPORT ON AIX FOR GATEWAY V12
- XCURS6 Problem 215: GATEWAY CLIENT SSL SCRIPTS NOT INSTALLED ON AIX
- XCURS6 Problem 216: ENCRYPTION AT REST TRANSFERS USING WILDCARDS ARE NOT WORKING
- XCURS6 Problem 219: PATCH OPENSSL VULNERABILITIES
- XCURS6 Problem 221: UPDATE XCOM TO USE LATEST JAVA VERSION JAVA 8
- XCURS6 Problem 222: MISLEADING ERROR WHEN ULIMIT NOT CONFIGURED PROPERLY
- XCURS6 Problem 223: ZERO BYTES TRANSFERRED WHEN FIPS_MODE=YES AND CIPHER=DES
- XCURS6 Problem 224: META-TRANSFER HANGS WHEN NON-FIPS CIPHER USED IN FIPS MODE
- XCURS6 Problem 225: CLEANLOG UTILITY RUNS CONTINUOUSLY ON AIX
- XCURS6 Problem 226: REPORT FORMAT IS WRONG WHEN IT CONTAINS IBM PRINTER CODES
- XCURS6 Problem 227: BROWSE BUTTON MISSING TO SAVE CONFIG FILES TO A DIRECTORY
- XCURS6 Problem 228: ALLOW OPTION TO SET NUMBER OF RECORDS DISPLAYED PER PAGE
- XCURS6 Problem 229: GUI HANGS AFTER SUSPENDING A TRANSFER WITH AUTO REFRESH ON
- XCURS6 Problem 230: ERROR WHILE DELETING A SUSPENDED TRANSFER USING GUI
- XCURS6 Problem 231: GUI: NO PROVISION TO REFRESH LOG ENTRIES IN LOG BROWSER
- XCURS6 Problem 232: XCOMU0436E ERROR OCCURS AFTER RESUMING A TRANSFER ON AIX
- XCURS6 Problem 233: UNABLE TO RECEIVE WILDCARD TRANSFERS FROM OLDER RELEASES
- XCURS6 Problem 234: VULNERABILITY - XLOGFILE COMMAND INJECTION

Fixes for Solaris Sparc

- XCUSOL Problem 146: API - RUN XCOM QUEUE=NO AS NON-ROOT
- XCUSOL Problem 147: TRANSFER - AVGREC AND EATTR ALWAYS IN XCOM HEADER
- XCUSOL Problem 148: XCOMPRE - LOCAL PREALLOCATION EXIT ERROR SEGMENTATION FAULT
- XCUSOL Problem 149: XCOMTCP CRASHES WITH WRONG STARTDATE FORMAT
- XCUSOL Problem 153: FILE MISMATCH WITH CACHE WRITE PROCESSING NFS TIMEOUTS
- XCUSOL Problem 156: CLEANLOG ERROR:PERMISSION DENIED
- XCUSOL Problem 157: DEFUNCT ZOMBIE XCOM PROCESSES REMAIN IN THE PROCESS TABLE
- XCUSOL Problem 158: RETRY INTERVAL NOT HONORED
- XCUSOL Problem 163: WILDCARD TRANSFERS OF LONG FILE NAMES ARE BEING TRUNCATED
- XCUSOL Problem 166: TRANSFER ENDS SUCCESSFULLY WHEN TARGET FILE SYSTEM IS FULL
- XCUSOL Problem 171: OPENSSL SECURITY ADVISORY CVE-2014-0224
- XCUSOL Problem 173: XCOM - SSL 3.0 SECURITY ADVISORY CVE-2014-3566
- XCUSOL Problem 174: XCOMLIB: TEMP FILE NOT RENAMED
- XCUSOL Problem 175: RESTART_SUPPORTED=NO AND NUMBER_OF_RETRIES=0 NOT HONORED
- XCUSOL Problem 178: PATCH OPENSSL VULNERABILITIES
- XCUSOL Problem 179: PATCH OPENSSL VULNERABILITIES
- XCUSOL Problem 180: VLR/VLR2 TRANSFERS FAIL WITH CA XCOM 64-BIT VERSION
- XCUSOL Problem 181: UPDATE XCOM TO USE LATEST JAVA VERSION JAVA 8
- XCUSOL Problem 182: REPORT FORMAT IS WRONG WHEN IT CONTAINS IBM PRINTER CODES
- XCUSOL Problem 183: BROWSE BUTTON MISSING TO SAVE CONFIG FILES TO A DIRECTORY
- XCUSOL Problem 184: ALLOW OPTION TO SET NUMBER OF RECORDS DISPLAYED PER PAGE
- XCUSOL Problem 185: GUI HANGS AFTER SUSPENDING A TRANSFER WITH AUTO REFRESH ON
- XCUSOL Problem 186: ERROR WHILE DELETING A SUSPENDED TRANSFER USING GUI
- XCUSOL Problem 187: GUI: NO PROVISION TO REFRESH LOG ENTRIES IN LOG BROWSER
- XCUSOL Problem 188: LICENSE - SUPPRESS CA LICENSE MESSAGES
- XCUSOL Problem 189: UTF-8 CODE CONVERSION ADDS AN EXTRA 0XC2 WHEN NO NL DELIM
- XCUSOL Problem 191: UNABLE TO RECEIVE WILDCARD TRANSFERS FROM OLDER RELEASES
- XCUSOL Problem 192: VULNERABILITY - XLOG FILE COMMAND INJECTION
- XCUSOL Problem 193: CASE SENSITIVITY OF USER ID NOT HONORED IN TRUSTED TRANSFERS

Fixes for Solaris x86

- XCSS86 Problem 37: OPENSSL SECURITY ADVISORY CVE-2014-0224
- XCSS86 Problem 38: XCOMLIB: TEMP FILE NOT RENAMED
- XCSS86 Problem 39: RESTART_SUPPORTED=NO AND NUMBER_OF_RETRIES=0 NOT HONORED
- XCSS86 Problem 40: XCOM - SSL 3.0 SECURITY ADVISORY CVE-2014-3566
- XCSS86 Problem 42: PATCH OPENSSL VULNERABILITIES
- XCSS86 Problem 43: UPDATE XCOM TO USE LATEST JAVA VERSION JAVA 8
- XCSS86 Problem 44: REPORT FORMAT IS WRONG WHEN IT CONTAINS IBM PRINTER CODES
- XCSS86 Problem 45: BROWSE BUTTON MISSING TO SAVE CONFIG FILES TO A DIRECTORY
- XCSS86 Problem 46: ALLOW OPTION TO SET NUMBER OF RECORDS DISPLAYED PER PAGE
- XCSS86 Problem 47: GUI HANGS AFTER SUSPENDING A TRANSFER WITH AUTO REFRESH ON
- XCSS86 Problem 48: ERROR WHILE DELETING A SUSPENDED TRANSFER USING GUI
- XCSS86 Problem 49: GUI: NO PROVISION TO REFRESH LOG ENTRIES IN LOG BROWSER
- XCSS86 Problem 50: LICENSE - SUPPRESS CA LICENSE MESSAGES
- XCSS86 Problem 51: UTF-8 CODE CONVERSION ADDS AN EXTRA 0XC2 WHEN NO NL DELIM
- XCSS86 Problem 52: UNABLE TO RECEIVE WILDCARD TRANSFERS FROM OLDER RELEASES
- XCSS86 Problem 53: VULNERABILITY - XLOG FILE COMMAND INJECTION
- XCSS86 Problem 54: CASE SENSITIVITY OF USER ID NOT HONORED IN TRUSTED TRANSFERS

Service Pack 11.6.01

The installation and upgrade process is simplified for XCOM Data Transport 11.6.01, as follows:

Install and Run CA XCOM Data Transport As a Docker Container on Linux Systems

You can now install and run CA XCOM Data Transport as a Docker container on Linux systems. This simplifies the installation or management process and service monitoring. The docker containers enable you to run multiple instances of XCOM on a single host that allows you to separate roles and responsibilities of each XCOM server. You can also configure XCOM server to read and write files only from the designated file systems. For more information about how to install and run CA XCOM Data Transport As a Docker Container on Linux Systems, see Install Using Docker Container.

Simplified Installation and Upgrade Using Conventional Method

The installation and upgrade process is more efficient. The same program lets you either install XCOM Data Transport for the first time or upgrade it from 11.6 to 11.6.01. Following changes are made to simplify the installation and upgrade process with the conventional method.

- Enhanced Silent Installation
- Improved Troubleshooting
- Reduced Installer Footprint
- Other Changes Related to Installation

The new installer runs faster than the previous ones. For more information, see Installing and Upgrading Using Conventional Method.

Enhanced Silent Installation

You no longer need to use the installer to generate a response file before you run a silent installation. Instead, you can now create and customize your response file and use it to complete the installation. For details, see Install or Upgrade in Silent Mode.

Improved Troubleshooting

The installer creates these log files:

xcominstaller.log
contains a detailed log of all custom actions that are performed by the installer.

- If pre-install checks fail:
  `/tmp/xcominstaller.log`
- In all other cases:
  `$XCOM_HOME/Uninstaller/Log`

CA_XCOM_Data_Transport__r11.6_SP01_(nn-bit)_Install_<timestamp>.log in the $XCOM_HOME/ Uninstaller/Logs
directory contains *all* the actions that are performed by the installer.

Reduced Installer Footprint

XCOM Data Transport binary files that are not related to the service pack have been removed from the Installer, reducing its size.

Other Changes Related to Installation

Other installation-related changes follow:

Installer Return Codes
Installer return codes are documented. This information helps you verify the installation status when you run silent mode from your script or other automation programs.

Java Runtime Environment (JRE) location
The new installer deploys JRE under XCOM_HOME. During an uninstallation, this folder is removed.

ALPKEYS are Discontinued
The ALPKEYS have been removed and no other physical license keys are required for the product.

Passphrase Support

The maximum length of the password fields PASSWORD and LPASSWORD is increased up to 100 characters. Means that the new range of password characters is 0-100 to support passphrase input. Use of passphrase increases system security by providing a bigger number of possible character combinations compared to using a standard password. The passphrase must conform to your site security standards.

To implement the passphrase support, apply the following PTFs to the corresponding platforms: SO05627 (Linux), SO05628 (AIX), SO05629 (Solaris Sparc), SO05630 (Solaris x86).

PASSWORD:
The higher input length for password has been increased from 31 to 100.

LPASSWORD:
The higher input length for meta-transfers has been increased from 31 to 100.

Trusted Transfers:
The input length for the password field has been increased to 100 characters.

Oracle Support

The database support has been extended to Oracle Database. Now you can configure Oracle Database for managing your history records and performing trusted transfers. The XCOMDB_SQLCONNECT_TIMEOUT parameter has been introduced for setting database connection time-out. To implement the Oracle support, apply the following PTFs to the corresponding platforms: SO05627 (Linux), SO05628 (AIX), SO05629 (Solaris Sparc), SO05630 (Solaris x86).

TLS v1.1 and TLS v1.2 Support

This release adds support for Transport Layer Security (TLS) v1.1 and v1.2. Based on the security requirements, you can enable these protocols for Secure Socket transfers. The following TLS/SSL methods are supported in configssl.cnf:ssl.cnf:

ALL

- Supports TLSv1.2, TLSv1.1, TLSv1.0 and SSLv3 protocols
- Maintains backward compatibility

TLSv1.2

- Supports TLSv1.2 protocol
- This option is new for this release.

TLSv1.1

- Supports TLSv1.1 protocol
- This option is new for this release.

TLSv1 or TLS

- Supports TLSv1.0 protocol

V3

- Supports SSLv3 protocol

For more details about these protocols, see Cryptographic Protocols.

The default TLS/SSL methods are set to ALL. As a result, the XCOM Data Transport transfer selects the newest protocol that is supported by the partner CA XCOM.

In the following example, the transfer runs over TLSv1.2. If the partner does not support TLSv1.2, then the implementation falls back to the next best version that the partner supports.

```
[SSL_METHOD]
INITIATE_SIDE = ALL
RECEIVE_SIDE = ALL
```

When a TLS or SSL connection is established, the client and server negotiate a cipher suite, exchanging cipher suite codes in the client hello and server hello messages. The cipher suite specifies a combination of cryptographic algorithms to be used for the connection. For more details, see Supported Cipher Suites.

Miscellaneous Changes

This release provides the following miscellaneous changes:

Default Values for Parameters

This release changes the default values for the following parameters:

Parameter	Previous Value	New Value
XCOMHIST	xcomhist	None
XCOMHIST_PASSWORD	root	None

XCOMHIST_USER	root	None
XCOMHIST_OWNER	xcomhist	None
TRUST_ODBC	xcomtrust	None
TRUST_PASS	pass	None
TRUST_USER	root	None
TRUST_OWNER	xcomtrst	None
CACHE_READ_SZ	1024	0
CACHE_WRITE_SZ	1024	0
GATEWAY_VERSION	r115	r120

NOTE

None indicates no value.

XCOM API and XCOM QAPI

You can optionally use the XCOM API and XCOM QAPI to write custom applications. Recompile your applications by using the following compilers:

- For Linux: gcc compiler 4.3.2 or above.
- For AIX: xlC compiler version 12.1 or above.

Support Deprecated for XCOM Data Transport Gateway

The support for XCOM Data Transport Gateway is deprecated. This deprecation includes the following XCOM parameters:

- GATEWAY_IP
- GATEWAY_PORT
- GATEWAY_VERSION
- GATEWAYCERT
- GATEWAYCPASS
- GATEWAYDPATH
- GATEWAYPKEY
- GATEWAYGUID
- GATEWAYPROTOCOL
- LGATEWAYGUID

Release Compatibility and Support

Release compatibility and support information let you see the tools that Broadcom offers to assist you in the product lifecycle.

The following resources are available from Broadcom Support online:

- Mainframe Installation and Maintenance Tools
- Security Advisories - Mainframe Software (login required)
- Broadcom Mainframe Products Solutions List
- Broadcom Mainframe Products Fix Category Solutions List
- Recommended Service for z/OS (CARS)
- Migrate SMP/E Environments into z/OSMF

NOTE
For migration assistance and access to z/OSMF trainings from Broadcom, see z/OSMF Migration.

- Mainframe Essentials: SYSVIEW Essentials, Software Toolkit Plug-in for z/OSMF, Mainframe Resource Intelligence
- Broadcom Mainframe Product Lifecycle Page
- XCOM Data Transport Release and Support Lifecycle Dates
- Mainframe Compatibilities
- Broadcom Support Network Details

For other technical insights and to consult your peers and product management, monitor our global communities:

- Broadcom Mainframe Software Division (MSD) Microsite
- Broadcom Mainframe Software Communities
- XCOM Data Transport Community

Product Names and Abbreviations

This list defines the acronyms and product names, in long and short form, that appear in this documentation.

This documentation references the following products and abbreviations:

- XCOM™ Data Transport® for UNIX/Linux (XCOM Data Transport)
- XCOM™ Data Transport® Gateway (XCOM Data Transport Gateway)
- XCOM™ Data Transport® Management Center

Installing

Learn how to install, upgrade, and configure XCOM Data Transport for UNIX/Linux.

Required roles: systems programmer, security administrator, database administrator

You can install the product using the following methods.

ISO Conventional Method

The ISO conventional method is a traditional way to install and run XCOM Data Transport on a host (bare-metal, virtual machine, cloud instance) server, and to connect partner servers to the listening ports. XCOM Data Transport uses shared memory to communicate between the processes. By default, the XCOM Data Transport application can access the host server file system. The application can read and write files to the location where the user has permission to read, write, and execute jobs.

Docker Method

Docker lets you package and deploy self-sufficient applications in Linux containers. Docker packages the application software into a single unit, which can be deployed anywhere as long as the Docker engine is installed. XCOM Data Transport is considered a single unit within the Docker container, and it serves only XCOM Data Transport connection requests on the assigned ports. XCOM Data Transport uses ports 8044 through 8047 inside the Docker container to listen incoming connections, but these ports are not the real port numbers for sending connections. Port numbers must be mapped to the selected ports on the host server to allow them to listen incoming connections over the network. To confine the file system scope of the container, assign only the selected storage locations as volumes to the container. The container can read and write files from its assigned storage locations. You can run multiple XCOM Data Transport containers on the same host server. You can assign separate ports and file storage locations to each container.

Ansible Method

Ansible is an open-source tool that you can use for XCOM Data Transport Windows and Linux software installation and configuration management. Ansible uses an agentless architecture that does not require you to install an agent on the target system. Ansible simplifies software management tasks like installing, upgrading, and maintaining the product on a group of systems because you are not required to manually log on to each system.

Install Using ISO Conventional Method

Review the steps to acquire, install, upgrade, and roll back XCOM Data Transport using the traditional ISO method.

Required roles: systems programmer

The traditional way to install and run XCOM Data Transport is to install the XCOM software on a host (bare-metal, virtual machine, cloud instance) server. The partner XCOM servers have the connection to the listening ports.

Installation Types

To install XCOM Data Transport, use one of the following ISO Conventional methods:

Silent Mode

The silent-mode installation is a non-interactive, or optionally limited interactive, method of installing XCOM Data Transport. To specify the custom installation options, you can use the prepared response file in advance. The silent-mode of installation lets you install the product from a script or from the command-line interface. The silent-mode installation allows you to define the installation configuration once with the response file and use the same configuration for installation on multiple systems.

Console Mode

The console-mode installation is an interactive, text-based method to install your software from the command-line interface.

Installation Roadmap

The following procedure provides the high-level tasks that you are required to complete to install XCOM Data Transport.

1. Download the package from Broadcom Support.
2. Complete the installation planning requirements and address the product installation requirements.
3. Determine the appropriate installation mode for your environment (silent mode or console mode).
4. To install XCOM Data Transport for UNIX/Linux, AIX, or Solaris, run the installer that is available in the mode you select. Both the installation modes provide the option to automatically generate a detailed installation log.
5. (Optional) As an extra in the silent mode, prepare a response file to enable the automated installation. Use the same installer to install a fresh instance and upgrade to the latest release
6. Perform the post-installation or post-upgrade tasks.

The product is ready to start.

Acquire the Product Files

Learn how to acquire the product files from the Broadcom Support website.

Required roles: systems programmer

To begin the product installation procedure, download the product file to your local machine.

> **NOTE**
> To optimize downloads from Broadcom Support, configure the downloads.broadcom.com URL for HTTPS and Secure FTP in your network security software, firewalls, or both. Sites that regulate access through an IP address are required to allow network access to 141.202.253.110.

1. Go to Broadcom Support and select **Software**, **Mainframe Software**, and **My Downloads**.
2. Select **XCOM** from the list or enter the product name in the **Search** field.
   The product page appears with two tabs under the product name: **Products** and **Solutions**.
3. Under **Products**, select **XCOM Data Transport for Linux (PC)**.
   A list by release, release level, and language appears.
4. Use the hypertext link to select a release.
   The **Primary Downloads** page appears. This page shows the product-specific software packages that you can select to download.
5. Review the packages that appear under **Primary Downloads** and complete the following steps:
   a. View the file information for the product that you want by selecting the right arrow key (>) or by selecting **Expand All**.
   b. Select the checkbox under **Downloads** for the files that you want to include in the product download.
   c. Select **HTTPS Download** or **Secure FTP Download**. Secure FTP Download is the preferred method.

To select a download method from the **Download Manager**, select **Download Selected**. This option lets you download multiple files at one time.

For download tips, see Product Download Help. For sample JCL and security requirements, see Download Methods and Locations.

The product files are downloaded.

6. Transfer the product package to your Linux system.

You are now ready to install the product using silent mode or console mode.

Prepare for Installation

Address the operating system support, system, software, and database requirements to install XCOM Data Transport.

Required roles: systems programmer, security administrator. database administrator

Before you install XCOM Data Transport, verify the following requirements:

- The system requirements for installing XCOM Data Transport are met.
- You have downloaded the package to install the product.
- Your UNIX/Linux PC is configured to use TCP/IP.
- Your system is connected to the appropriate network.
- Your system has been configured to use the SNA protocol.

Operating System Support

Linux

XCOM Data Transport requires both of the following platforms:

- Red Hat Enterprise Linux (6, 7, or 8) or SUSE Linux Enterprise (11, 12, or 15 SP02 or below) running on an Intel or compatible PC
- Kernel level at Version 2.6.32 or higher

IBM AIX

XCOM Data Transport supports the following platforms:

- IBM AIX 7.1 (7100-03)
- IBM AIX 7.2 (7200-00)
- IBM AIX 7.3 (7300-00)

SOLARIS

XCOM Data Transport supports the following platforms:

- Solaris Sparc 10, 11.1, 11.2, or 11.3
- Solaris x86 10, 11.1, 11.2, or 11.3

System Requirements

Ensure that the system meets the following minimum hardware, memory, and storage requirements:

Hardware

- A physical or logical connection to the remote system (for example, Ethernet)
- One of the following processors:

- Linux: Intel x64 or compatible processor
- AIX: IBM Power Systems platform
- SOLARIS: SPARC or Intel x86 compatible processor

Memory

Minimum 1 GB of memory (1.5 GB recommended)

Storage

A hard disk with at least 750 MB of free space. Avoid using mounted file systems or UNC paths as an installation directory.

Software Requirements

Ensure that the system meets the following software requirements:

TCP/IP Protocol Support Requirements

The TCP/IP release is packaged with your Unix/Linux operating system.

gcc Compiler Version 4.3.2 or higher

You can optionally use the XCOM API and XCOM QAPI to write custom applications for Linux. Recompile the applications with the gcc compiler, using version 4.3.2 or above.

cc Compiler Version 5.12 or higher

You can optionally use the XCOM API and XCOM QAPI to write custom applications for SOLARIS. Recompile the applications with the cc compiler, using version 5.12 or above.

xlC Compiler Version 12.1 or higher

You can optionally use the XCOM API and XCOM QAPI to write custom applications for AIX. Recompile the applications with the xlC compiler, using version 12.1 or above.

XINETD Service

The XINETD service must be available on the UNIX/Linux PC.

INETD Service

The INETD service must be available on the IBM AIX and Solaris.

Supported Internet Browser

To display help screens from the XCOM Data Transport GUI, install a supported Internet browser.

(Optional) Pluggable Authentication Modules (PAM) package

You can optionally install the Pluggable Authentication Modules (PAM) package that is provided by the OS vendor. This package enables XCOM Data Transport to use PAM-based authentication. If PAM is not used, traditional UNIX authentication is used by default.

Database Requirements

A relational database is required only when you use the History Records and Trusted Transfer features. XCOM Data Transport works with industry-standard relational databases. One of the following certificates is required:

- MySQL v5 or later
- IBM Db2 v9.5 or later
- Oracle v12.1 or later

NOTE

Service pack 11.6.01 supports Oracle when the appropriate platform-specific PTF is applied:

- SO05627 (Linux)
- SO05628 (AIX)
- SO05629 (Solaris Sparc)
- SO05630 (Solaris x86)

The database client (ODBC) must be on the same machine as XCOM Data Transport. However, the database can be on a different machine. The 64-bit application requires a 64-bit version of ODBC and a 64-bit version of the database driver.

Install Using Silent Mode

Learn how to install XCOM Data Transport using the silent mode.

Required roles: systems programmer

The silent mode installs XCOM Data Transport with minimal user interaction. Use silent mode when you plan to provide your installation settings in a response file.

To install the product in silent mode, complete the following steps:

1. Choose a unique system ID with up to four characters and a system name with up to eight characters for this XCOM Data Transport.
2. Create a response file.
3. Run the installer.

   The silent mode installer verifies that you have root permissions and ensures that no XCOM Data Transport processes are running. The installer installs the XCOM Data Transport binary files and services and CAPKI.
4. Perform the post-installation tasks.

After you complete the post-installation tasks, the product is ready to start.

Create a Response File

Prepare a response file to install the product in silent mode.

Required roles: systems programmer

When you use silent mode to install the product, a response file is required to enable the automated installation. The response file contains values for the variables that control the instance configurations. The file uses a simple key=value format. You can use the same file to install the product on multiple systems. Separate response files are required only when different systems require different configurations.

To create a response file, use the console mode or manually customize the sample response file that is supplied with the product package.

Create a Response File Using the Console Mode

The console mode provides an automated method for configuring a response file.

1. Navigate to the directory path that contains the XCOM installer.
2. Enter the following command to create a response file:

   ```
   sh XCOM.bin -r <response file path and name>
   ```

   Example:

   ```
   sh XCOM.bin -r /tmp/XCOM.properties
   ```

   If you specify a path name without a file name, then installer.properties is used as the file name.
   The command starts the installer.
3. Answer the installer prompts.
   For more information about prompts, see Install Using Console Mode.
4. When you reach the Preinstallation Summary page, type **quit** and press **Enter**.
   Your selections and changes are saved in the response file.

The response file is ready to execute to complete the installation process.

Create a Response File Manually

To create a response file manually, copy and customize a sample response file.

1. Review the sample files that are supplied with the product package and select the file that best suits your requirements.
2. Copy the text from the sample file into a text file.
3. Customize the file to meet your configuration requirements and save the changes.

The response file is ready to execute to complete the installation process.

Example: Response File for a New Installation

```
# Tue Jun 14 17:25:04 IST 2022
# Replay feature output
# ---------------------
# This file was built by the Replay feature of InstallAnywhere.
# It contains variables that were set by Panels, Consoles or Custom Code.



#Choose Install Folder
#---------------------
USER_INSTALL_DIR=/opt/CA/XCOM

#Set System ID and System Name
#-----------------------------
SYSID=LINX

#
#
SYSNAME=SYSNAME
```

Example: Response File for a New Installation

```
# Tue Nov 01 08:17:13 EDT 2016
# Replay feature output
# ---------------------
# This file was built by the Replay feature of InstallAnywhere.
# It contains variables that were set by Panels, Consoles or Custom Code.

#Upgrade Choice
#--------------
UPGRADE=YES
#Choose Backup Directory
#-----------------------
BACKUPDIR=/opt/CA/XCOM_R116BACKUP
```

```
#Trusted Database
#----------------
BACKUP_TRUSTED_DB_JARS=YES
#JDBC JAR PATH
#-------------
JDBC_DRIVER=/opt/CA/XCOM/lib/xyz.jar
#JDBC License JAR
#----------------
DB2_LICENSE_DRIVER=/opt/CA/XCOM/lib/abc.jar
#History DSN
#-----------
UPGRADE_HISTORY_DB=YES
```

Run the Installer in Silent Mode

Learn how to run the installer in silent mode.

Required roles: systems programmer

To run the silent mode installer, issue the following command:

```
sh XCOM.bin -i silent -f <response file path and name>
```

Example:

```
sh XCOM.bin -i silent -f </temp/response.properties>
```

To query the status of the installation, issue the following command:

```
echo $?
```

The command-line interface lets you override the predefined values for the following variables in the response file with the new values:

- For new installations:
  - SYSID
  - SYSNAME
  - USER_INSTALL_DIR
- For upgrades:
  - BACKUPDIR
  - JDBC_DRIVER
  - DB2_LICENSE_DRIVER
  - UPGRADE_HISTORY_DB
  - BACKUP_TRUSTED_DB_JARS

To change the values for these variables, use the **-D** attribute. For example, in a new installation, enter the following command to override the SYSID and SYSNAME values in the response file with new values:

```
sh XCOM.bin -i silent -f </temp/response.properties> -DSYSID=<xxxx> -DSYSNAME=<yyyyyyyy>
```

Use this example as a model to set the custom values for other variables from the command-line interface.

If you do not use the command line to specify a new variable value, the value from the response file is used. If the response file does not provide a value for the variable, a default value is used.

After you run the installer to install the product, perform the post-installation tasks.

Perform Post-Installation Tasks

After you use silent mode to install XCOM Data Transport, verify its installation status by using the log files and return codes.

Required roles: systems programmer

Review the Log Files

Review the log files to verify the XCOM Data Transport installation status.

- If an error occurred during installation, view the warnings, error messages, and exit codes in the following directories:
  - For preinstallation check failures:
    `/tmp/xcominstaller.log`
  - For all other failures:
    `$XCOM_HOME/Uninstaller/Logs/xcominstaller.log`
- To view the sequence of actions that the installer performed, view the following log:
  `$XCOM_HOME/Uninstaller/Logs/*install*.log`

Analyze the Return Codes

To verify that the product installed successfully, analyze the return codes. This section contains a list of errors and warning messages that the silent installer returns. The errors and warnings are organized into the following tables:

- Default InstallAnywhere Error Messages (Table 1)
- Product-Specific Installation Error Messages (Table 2)
- Product-Specific Uninstallation Error Messages (Table 3)

When you install the product in silent mode, the messages are not displayed. The messages are written to the installation log files. Map the error codes in the installation log files with the codes in the following tables to obtain their specific descriptions.

Table 1: Default InstallAnywhere Error Messages

Code	Description
0	Success: The installation completed successfully without any warnings or errors.
1	The installation completed successfully. However, one or more actions in the installation sequence caused a warning or a non-fatal error.
-1	One or more actions in the installation sequence caused a fatal error.
1000	The user canceled the installation.
1001	The installation included an invalid command-line option.
2000	An unhandled error occurred. **Note:** InstallAnywhere for the AIX platform returns 208 for a silent uninstallation, even though the product is uninstalled successfully. This issue is a known issue, IOJ-17322968, with InstallAnywhere 2015 for AIX. For a detailed description, see InstallAnywhere 2015 Known Issues.

Code	Description
2001	The installation failed the authorization check. This error can indicate an expired version.
2002	The installation failed a rules check. A rule that is placed on the installer itself failed.
2003	An unresolved dependency in silent mode caused the installer to exit.
2004	The installation failed because insufficient disk space was detected during the execution of the Install action.
2006	The installation failed because it was launched in a UI mode that this installer does not support.
2009	The user attempted to launch multiple instances of an installer at the same time even though the installer was configured to prevent multiple launches. The Prevent multiple launches of an installer at a given time check box was selected in the General Settings view on the Project page of this InstallAnywhere project.
3000	An unhandled error that is specific to a launcher occurred.
3001	The installation failed due to an error that is specific to the lax.main.class property.
3002	The installation failed due to an error that is specific to the lax.main.method property.
3003	The installation was unable to access the method that was specified in the lax.main.method property.
3004	The installation failed due to an exception error that was caused by the lax.main.method property.
3005	The installation failed because no value was assigned to the lax.application.name property.
3006	The installation was unable to access the value that is assigned to the lax.nl.java.launcher.main.class property.
3007	The installation failed due to an error that is specific to the lax.nl.java.launcher.main.class property.
3008	The installation failed due to an error that is specific to the lax.nl.java.launcher.main.method property.
3009	The installation could not access the method that was specified in the lax.nl.launcher.java.main.method property.
4000	A Java executable file could not be found at the directory that was specified by the java.home system property.
4001	An incorrect path to the installer jar caused the relauncher to launch incorrectly.
5000	The modification of the existing instance failed because the instance has not been uninstalled properly or because the product registry has been corrupted.
7000	The installation was rolled back due to a fatal exception.

Code	Description
8000	The upgrade was canceled because a newer version of the product is already installed on the target system.
8001	The user canceled the upgrade.
8002	The upgrade exited because the earlier version of the product could not be uninstalled.

Table 2: Product-Specific Installation Error Messages

Code	Description
1	Exit when any of the following installation procedures fail: • CAPKI Installation • Makelinks.sh script execution • Xinetd rpm package check (Linux only) • Xinetd rpm refresh (Linux only) • SYSID validation • SYSNAME validation **Note:** If the silent installation is done without a response file or a response file that has a blank SYSID and SYSNAME, a default SYSID and SYSNAME are used. To identify the XCOM server uniquely, update $XCOM_HOME/config/xcom.glb after installation is completed with unique SYSID and SYSNAME values.
11	Exit when a user does not have root group privileges.
12	Exit when a 64-bit platform is not detected (Intel/AMD 64-bit architecture) (Linux only).
13	Exit when any of the XCOM, xcomqm, xcomtcp, or xcomd processes are running.
14	• Exit when XCOM r11/r11.5/r11.6 SP00 32-bit/ r11.6 SP01 64-bit or any patch is already installed. or • Exit when the XCOM_HOME environment variable is not set correctly.
15	Exit when the installation directory path is read only or it is not valid.
16	Exit when the disk space is insufficient. (A minimum of 750 MB is required.)
17	Exit when the glibc version is < 2.9 (Linux only).
31	Exit when the user does not want to upgrade XCOM Data Transport from r116 to r116 SP01.
32	Exit when the backup directory path is blank.
33	Exit when the specified backup directory is invalid or inaccessible.
34	Exit when the installer fails to back up the existing product installation.
35	Exit when the backup directory is a subdirectory of the product installation directory.

Code	Description
36	Exit when the XCOM Data Transport shared memory is in use.

Table 3: Product-Specific Uninstallation Error Messages

Code	Description
21	Exit when the user does not have root group privileges.
22	Exit when any of the XCOM, xcomqm, xcomtcp, or xcomd processes are running.
23	Exit when the XCOM r11.6 SP01 patch is already installed.

Install Using Console Mode

Learn how to install XCOM Data Transport for UNIX/Linux using the console mode.

Required roles: systems programmer

This interactive and text-based mode installs the product from the command-line interface. Use console mode when you want to be prompted for each installation setting.

1. Choose a unique system ID with up to four characters and a system name with up to eight characters. These values are used to uniquely identify this XCOM Data Transport instance.
2. Log in as a superuser (root).
3. Mount the ISO file that you downloaded from Broadcom Support, or copy the extracted installation file to the UNIX/Linux system.
4. Navigate to the directory that contains the XCOM.bin file and issue the following command:

   ```
   sh XCOM.bin
   ```

   The installation process starts.
5. Respond to the prompts to select the default or custom options. Enter the number for your choice, or press Enter to accept the default. You can enter **quit** at any time to exit the installation.

   The installer displays a message indicating that the product is installed.
6. (Optional) Review the log files.
   – If an error occurred during installation, view the warnings, error messages, and exit codes in the following directories:
     - For preinstallation check failures:

       ```
       /tmp/xcominstaller.log
       ```
     - For all other failures:

       ```
       $XCOM_HOME/Uninstaller/Logs/xcominstaller.log
       ```
   – To view the sequence of actions that the installer performed, view the following log:

     ```
     $XCOM_HOME/Uninstaller/Logs/*install*.log
     ```

You can now perform the post-installation tasks.

Upgrade Using ISO Conventional Method

Upgrade XCOM Data Transport for UNIX/LINUX with the automated and manual methods.

Required roles: systems programmer

Review the following upgrade considerations:

- The installer automatically detects the Release 11.6 and allows you to upgrade to the latest Service Pack 11.6.1. You can use either silent mode or console mode to run the installer.
- Use the manual method to upgrade from the release 11.5 and older releases to the latest service pack release.
- Rollback of an upgraded instance is possible *only* manually and does not support auto-rollback.
- The silent and console mode installer programs act as both a new installer and an upgrade installer. If the installer does not find any installed XCOM Data Transport instance, the installer acts as a new installer. If the installer program finds an installed previous release instance, the installer acts as an upgrader installer.

Upgrade Using Silent or Console Mode

Upgrade XCOM Data Transport for UNIX/Linux 11.6 to the highest release with the silent or console mode.

Required roles: systems programmer

The following video guides you through the upgrade procedure.

Upgrade Considerations

Consider the following upgrade criteria before you start the process to upgrade XCOM Data Transport 11.6 to the latest versions of service pack:

- XcomAPI and XcomQAPI are updated in XCOM Data Transport Service Pack 11.6.01. If you use XcomAPI and XcomQAPI from an earlier release to build custom applications, perform the following tasks *before* you use them with Service Pack 11.6.01:
  - For XCOM Data Transport for Linux PC: Recompile your applications with the gcc compiler version 4.3.2 or above.
  - For XCOM Data Transport for AIX: Recompile your applications with the xlC compiler version 12.1 or above.
  - For XCOM Data Transport for Solaris: Recompile your applications with the cc compiler version 5.12 or above.
  - Link them with the new versions of xcomapitcp.so/xcomapisna.so from Service Pack 11.6.01.
- You can upgrade XCOM Data Transport from the release 11.6 64-bit to Service Pack 11.6.01 64 bit.
- The XCOM Data Transport Service Pack 11.6.01 installer acts as a new installer and an upgrade program. If the Installer does not find any installed XCOM instance on the system, the installer acts as a new installer. If the installer finds a release 11.6 XCOM instance, the installer acts as an upgrade installer.
- Rollback of upgrade is possible *only* manually and does not support any Auto-rollback.

Understand the Upgrade Process

When you run the installer to upgrade the product, the program performs the following tasks:

1. Performs the following pre-installation verifications:
   - The user who invokes the installer has root permissions.
   - No XCOM Data Transport processes are running.
   - The currently installed product version is a candidate for the upgrade process.
2. Creates a backup for the XCOM_HOME directory from the existing product environment.

3. Removes the existing XCOM Service Pack 11.6.00 and CAPKI software.
4. Installs XCOM Data Transport binary files and services and CAPKI
5. Restores the existing XCOM Data Transport user customized settings:
   - User customized configuration files, including configssl.cnf, history.inserts, log4j.properties, StandAloneGUIParameters.xml, xcom.cnf, xcom.glb, xcom.ses, xcom.tid, xcom.log, xcomlp, xcomntfy, txpi files (Linux only), and exit scripts
   - User customized configuration folders, including ssl, tmp, preferences, trace, and convtab
6. Copies the user specified JDBC driver jar files to the XCOM_HOME/lib directory.
7. Performs configuration file migration where the key value pair is compared and merged in the configuration files such as xcom.glb, xcom.cnf, and configssl.cnf.
8. Migrates the history database schema from the release 11.6 to Service Pack 11.6.01.
9. Migrates the Transfer Queue records from the release 11.6 to Service Pack 11.6.01.
10. Performs post-installation tasks.

Prepare for Upgrade with Silent or Console Mode

Before you start the upgrade procedure with the silent mode or console, review the queue, stop the active procedures and back up the configuration files.

Review the Queue

To review what is in the queue, issue the following command:

```
xcomqm -La
```

Stop XCOM Data Transport Procedures

Before you upgrade or reinstall, stop the XCOM Data Transport procedure. This procedure deletes all XCOM Data Transport queue entries. Coordinate with the other users for the best time to perform this procedure to make sure that none of the XCOM Data Transport processes are active.

Follow these steps:

1. Log in as a superuser (root), or as a member of the xcomadm group.

   NOTE
   The XCOM Data Transport daemon must still be running to start the upgrade process.
2. Issue the following command in the command line:

   ```
   xcomqm -Rf*
   ```

   Delete the entries that are in the queue.
3. Close xcomtool (Release 11.5), if it is active. To close xcomtool, click the main window and select **Quit** from the menu bar.
4. Issue the following command:

   ```
   xcomd -s
   ```

   XCOM Data Transport is stopped, and the following messages appears:

   ```
   XCOMU0079I xcomd: stop requested.XCOMU0082I xcomd: stop request accepted.
   ```

   A message similar to the following example might appear: Install Using Console Mode

   ```
   2009/02/22 17:08:49 PRG=XCOMD PID=22189 XCOMU0089I xcomd ended
   ```

Back Up Files

If you have edited any of the following files, save them after you stop the XCOM Data Transport for UNIX/Linux PC x64. The reinstallation or upgrade procedure overwrites the following files:

- /usr/include/xcom
- /usr/lib/xcom
- /usr/spool/xcom
- /usr/sbin/xcomd
- /usr/bin/xcomossl
- /usr/bin/xcomqm
- /usr/bin/xcomsmtp
- /usr/bin/xcomtcp
- /usr/bin/xcomtrst

Save any user files within those directories, before installing XCOM Data Transport for UNIX/ Linux PC x64. A link is also created in the /etc directory for xcomd (the Key defintion for "xcom" not found in the DITA map. for the UNIX/ Linux PC x64 daemon).

Upgrade Using Silent Mode

Upgrade XCOM Data Transport for UNIX/Linux from release 11.6 to the latest service pack releases using the silent mode.

Required roles: systems programmer

Before you start upgrading XCOM Data Transport for UNIX/Linux PC x64 Service Pack 11.6.01, log in as superuser (root) to perform this procedure.

Prepare Response File for Upgrade

Prepare a response file to define the custom settings for the XCOM Data Transport instance. The response file contains pre-defined values for the parameters that control the installation and settings of the instance. The response file uses a simple key=value format. Use the console mode or manual method to prepare the response file. For more information about how to prepare a response file, see Create a Response File.

The following sample provides an example of response that you can use for the upgrade procedure:

```
# Tue Nov 01 08:17:13 EDT 2016
# Replay feature output
# ---------------------
# This file was built by the Replay feature of InstallAnywhere.
# It contains variables that were set by Panels, Consoles or Custom Code.

#Upgrade Choice
#--------------
UPGRADE=YES
#Choose Backup Directory
#-----------------------
BACKUPDIR=/opt/CA/XCOM_R116BACKUP
#Trusted Database
#----------------
BACKUP_TRUSTED_DB_JARS=YES
#JDBC JAR PATH
#-------------
JDBC_DRIVER=/opt/CA/XCOM/lib/xyz.jar
#JDBC License JAR
```

```
#----------------
DB2_LICENSE_DRIVER=/opt/CA/XCOM/lib/abc.jar
#History DSN
#-----------
UPGRADE_HISTORY_DB=YES
```

Run the Installer Program to Upgrade

Run the installer program with the following command to upgrade XCOM Data Transport in the silent mode.

```
sh XCOM.bin -i silent -f <response file path and name>
```

For example:

```
sh XCOM.bin -i silent -f /temp/response.properties
```

Issue the following command to query the status of the upgrade process:

```
echo $?
```

Optionally, the installer program lets you override the pre-defined values for a parameter in the response file when you run the installer program for upgrade. Use the `-D` argument to override the values in the command prompt. The installer program lets you override the values for following parameters in the command prompt:

- BACKUPDIR
- JDBC_DRIVER
- DB2_LICENSE_DRIVER
- UPGRADE_HISTORY_DB_JARS

Example:

```
sh XCOM.bin - silent -f /temp/response.propeties -DBACKUPDIR=XXX -DJDBC_DRIVER=xxx
```

Use this example as a model to set custom values for other variables from the command line, if necessary.

Perform Post-Upgrade Tasks

After you complete the upgrade process, perform these post-upgrade tasks:

1. (Optional) Set the PATH variable to include the directory where XCOM Data Transport is installed.
2. Before you use the product, start the xcomd XCOM daemon. Issue the following command at the system prompt:

   ```
   $XCOM_HOME/sbin/xcomd
   ```

   NOTE
   Enter the complete path of xcomd. Make sure that the XCOM_HOME environment variable is set to the XCOM installed location.
3. If an error occurs, view the status information about the upgrade in the default xcominstaller.log file in the following directories:
   – If pre-upgrade checks are failed:

     ```
     /tmp/xcominstaller.log
     ```

   – In all other cases:

     ```
     $XCOM_HOME/Uninstaller/Logs/xcominstaller.log
     ```

   The xcominstaller.log file includes warnings, error messages, and exit codes that occurred during the installation.
4. To view the sequence of actions that the installer performed, see the following section:

   ```
   $XCOM_HOME/Uninstaller/Logs/*install*.log
   ```

To verify if the upgrade process completed successfully, analyze the return codes. For more information, see Perform Post-Installation Tasks.

Upgrade Using Console Mode

Upgrade XCOM Data Transport from the release 11.6 to the latest service packs with the console mode.

Required roles: systems programmer

Before you start upgrading XCOM Data Transport for UNIX/Linux PC x64 Service Pack 11.6.01, log in as superuser (root) to perform this procedure.

> **NOTE**
> If you are upgrading or reinstalling, the installation process overwrites the existing XCOM Data Transport for UNIX/ Linux PC files.

This procedure is prompt-driven and quick. Before you upgrade or reinstall, stop the XCOM Data Transport procedures. The stop procedure deletes all XCOM Data Transport queue entries. Coordinate with users for the best time to perform this procedure, so that none of the XCOM Data Transport processes are active.

Start Upgrade

Use the command line to upgrade or reinstall XCOM Data Transport for UNIX/ Linux PC x64.

1. Mount your ISO file that you downloaded from the Broadcom Support. Or copy the extracted installation file to the linux/UNIX system.
2. Change to the directory that contains the XCOM.bin file and issue the following command to start with the upgrade procedure:

   ```
   sh XCOM.bin
   ```

3. Respond to the prompt to select the default or the custom options:
   - Enter the number that is associated with your choice.
   - Press Enter to accept the default.

   > **NOTE**
   >
   > - During the upgrade process, enter the system ID and system name that you selected for this system.
   > - To exit the upgrade process at any time, enter quit.

   This upgrade program installs the XCOM Data Transport base product components and components that you selected.

   > **WARNING**
   >
   > - For the upgraded changes to take effect in your current upgrade session, the environment variable XCOM_HOME must be available.
   > - Before you can use XCOM Data Transport for UNIX/ Linux PC x64, start the xcomd XCOM daemon.

Perform Post-Upgrade Tasks

After you complete the upgrade process, perform these post-upgrade tasks:

1. (Optional) Set the PATH variable to include the directory where XCOM Data Transport is installed.
2. Before you use the product, start the xcomd XCOM daemon. Issue the following command at the system prompt:

   ```
   $XCOM_HOME/sbin/xcomd
   ```

   > **NOTE**
   > Enter the complete path of xcomd. Make sure that the XCOM_HOME environment variable is set to the XCOM installed location.

3. If an error occurs, view the status information about the upgrade in the default xcominstaller.log file in the following directories:
   - If pre-upgrade checks are failed:
     ```
     /tmp/xcominstaller.log
     ```
   - In all other cases:
     ```
     $XCOM_HOME/Uninstaller/Logs/xcominstaller.log
     ```

   The xcominstaller.log file includes warnings, error messages, and exit codes that occurred during the installation.
4. To view the sequence of actions that the installer performed, see the following section:
   ```
   $XCOM_HOME/Uninstaller/Logs/*install*.log
   ```

To verify whether the upgrade process completed successfully, analyze the return codes.

Upgrade History Database Schema

Learn how to manually upgrade the history database schema.

Required roles: systems programmer

The history database schema usually updates automatically when you upgrade XCOM Data Transport. In rare cases, this schema is not updated (for example, if the network fails), and the upgrade completes with warning messages.

Review the following log file for the upgrade status:

```
$XCOM_HOME/Uninstaller/Logs/xcominstaller.log
```

If the schema was not upgraded successfully, run the UPGRADEDB utility to upgrade it.

1. Stop all transfers.
2. Stop the XCOM Scheduler service.
3. Back up the history database.
4. Verify that your user ID has ALTER table privileges. The utility reads the history database configuration in the xcom.glb file and performs the upgrade operation. If the specified user in the xcom.glb file does not have ALTER table privileges, obtain a user ID that has these privileges, and specify credentials in the DBUSER and DBPASSWORD parameters in the next step.
5. Run UPGRADEDB from the command line, using the following syntax:
   ```
   UPGRADEDB [-v] [GLBFILE=xcom.glb file path] [LOGFILE=Log file path] [DBUSER=DatabaseUser] [DBPASSWORD/
   DBPASSWORD.ENCRYPTED=<Database Password /Encrypted Database Password>]
   ```

-v

(Optional) Prints logs on a standard output terminal. If you specify LOGFILE, then -v is ignored.

GLBFILE

(Optional) Specifies the complete path name of the xcom.glb file. The default name is $XCOM_HOME/config/xcom.glb.

LOGFILE

(Optional) Specifies the complete path name of the log file.

DBUSER

(Optional) Specifies the database user that has ALTER privileges. The default value is the XCOMHIST_USER value in xcom.glb.

DBPASSWORD

(Optional) Specifies the database password for the user. The default value is the XCOMHIST_PASSWORD value in xcom.glb.

DBPASSWORD.ENCRYPTED

(Optional) Specifies the encrypted database password for the user id. The default value is the XCOMHIST_PASSWORD.ENCRYPTED value in xcom.glb.

See the following example:

```
UPGRADEDB LOGFILE=/tmp/update.txt
```

The utility runs, performing the following actions in sequence:

a. Reads the xcom.glb file to obtain the required history configuration parameters.
b. Verifies that sufficient information is available in the xcom.glb file.
c. Performs basic validations to verify database connectivity.
d. Verifies whether the history database schema requires updates.
e. If schema updates are required, runs ALTER queries based on the type of DBMS in use. The utility is certified with supported Db2 LUW, Db2 z/OS, and MySQL DBMS versions.
f. Verifies that the schema is up to date.

Return Codes

Return Code	Descriptio
0	Successful
40	Xcom.glb file not found or failed to open
41	Failed to get required values (XCOMHIST, XCOMHIST_USER, XCOMHIST_TBL, XCOMHIST_PASSWORD,
42	Failed to allocate SQL handle
43	Failed to connect to database
44	Failed to get DBMS name
45	Failed to get database name, or database name is blank
46	Failed to obtain one of the ODBC function pointers
47	XCOM_HOME environment variable is not set and GLBFILE is not provided
255	Unhandled SQL exceptions

Upgrade from Release 11.5

You can upgrade XCOM Data Transport from Release 11.5 to Service Pack 11.6.01 manually. Automatic upgrades are not supported.

Required roles: systems programmer

Review the Considerations

Consider the following information before you upgrade XCOM Data Transport:

- Rollback of upgrade is possible *only* manually.
- XcomAPI and XcomQAPI are updated in XCOM Data Transport Service Pack 11.6.01. If you use XcomAPI and XcomQAPI from an earlier release to build custom applications, perform the following tasks *before* you use them with Service Pack 11.6.01:

  - For Linux: Recompile your applications with gcc compiler 4.3.2 or above.
  - For AIX: Recompile your applications with xlC compiler version 12.1 or above.
  - Link them with the new versions of xcomapitcp.lib/xcomapisna.lib from Service Pack 11.6.01.
- You can upgrade XCOM Data Transport from Release 11.6 32-bit to Service Pack 11.6.01 32-bit. But you cannot upgrade it from Release 11.6 32-bit to Service Pack 11.6.01 64-bit.
- Similarly, you can upgrade XCOM Data Transport from Release 11.6 64-bit to Service Pack 11.6.01 64 bit. But you cannot upgrade it from Release 11.6 64-bit to Service Pack 11.6.01 32-bit.
- The XCOM Data Transport Service Pack 11.6.01 installer acts as both a new and an upgrade installer. If the Installer does not find any installed XCOM instance on your computer, the installer acts as a new installer. If the installer finds a Release 11.6 XCOM Data Transport instance, the installer acts as an upgrade installer.
- You can upgrade in either console mode or in silent mode:
  - To upgrade in console mode, see Upgrade Using Console Mode.
  - To upgrade in silent mode, follow the instructions on Upgrade Using Silent Mode

Understand the Process

Understand the upgrade process, as follows:

1. Prepare for an upgrade, which includes uninstalling XCOM Data Transport Release 11.5.
2. Perform the upgrade tasks, which include running the installation program in either GUI mode or silent mode.
   The installation program performs the following tasks to install XCOM Data Transport Service Pack 11.6.01:
   a. Verifies the following pre-installation criteria:
      - The user who invokes the installer has administrative permissions.
      - No XCOM Data Transport processes are running.
   b. Installs the following entities:
      - XCOM Data Transport binary files and services
      - CAPKI and Broadcom License software
   c. Performs post-installation tasks.
3. Perform the post-upgrade tasks.

Prepare for Upgrade

Before you upgrade, complete the following steps:

1. Verify that you meet the Installation Prerequisites.
2. Coordinate with users for the best time to perform the upgrade.
3. Verify that no active transfers are running.
4. Stop all XCOM Data Transport applications.
5. Stop the Scheduler service.
6. Complete the following tasks:
   a. Delete the queue.
   b. Back up files.
   c. Uninstall your current product.

Delete the Queue

When performing an upgrade, remove existing queue entries by issuing the following command:

```
XCOMQM -R*
```

Back Up Files

Back up the following files to a different directory if you have edited them.

The following files are listed with their default directory locations. But your system might have them installed in directories other than the defaults.

- $XCOM_HOME/CONFIG/XCOM.CNF
- $XCOM_HOME/CONFIG/XCOM.GLB
- $XCOM_HOME/CONFIG/XCOM.SES
- $XCOM_HOME/CONFIG/XCOM.TID
- $XCOM_HOME/CONFIG/StandAloneGUIParameters.xml
- $XCOM_HOME/XCOMLP.BAT
- $XCOM_HOME/XCOMNTFY.BAT
- $XCOM_HOME/XCOMPP.BAT
- $XCOM_HOME/XCOMEND.BAT
- $XCOM_HOME/XCOMPRE.BAT
- $XCOM_HOME/CONFIG/CONFIGSSL.CNF
- $XCOM_HOME/SSL/*

If you have customized the following files, save them also:

- $XCOM_HOME/CONVTAB/ATOE.TAB
- $XCOM_HOME/CONVTAB/ETOA.TAB

Consider to back up your configuration files. After you upgrade XCOM Data Transport, compare the values in the-up files with the new files and revise the new files if necessary.

Uninstall Your Current Product

Uninstall XCOM Data Transport Release 11.5 before you install Service Pack 11.6.01. Before you uninstall Release 11.5, delete the Queue and back up the files.

Perform the Upgrade Tasks

To upgrade XCOM Data Transport to Service Pack 11.6.01, complete these tasks:

1. Review upgrade in Silent Mode.
2. Follow the instructions to:
   - Install Service Pack 11.6.01.
   - Perform the post-installation tasks.
   - Review the log files.

Perform the Post-Upgrade Tasks

1. (Optional) If you used a history database in Release 11.5, Upgrade History Database Schema to Service Pack 11.6.01.

   If you did not set up a history database for Release 11.5, ignore this step.
2. Compare your backup files (described earlier on this page) to the newly installed files. If necessary, customize the values in the newly installed files to match the values in your backed-up files.
3. (If applicable) If you used XcomAPI and XcomQAPI from a previous release to build custom applications, perform the following tasks *before* you use them with Service Pack 11.6.01:

- For XCOM Data Transport for Linux PC: Recompile your applications with the gcc compiler version 4.3.2 or above.
- For XCOM Data Transport for AIX: Recompile your applications with the xlC compiler version 12.1 or above.
- Link them with the new versions of xcomapitcp.so/xcomapisna.so from Service Pack 11.6.01.

Upgrade from Release 11.0

You can upgrade xcom from Release 11.0 to Service Pack 11.6.01 exclusively for AIX and Solaris. Automatic upgrades are not supported.

Required roles: systems programmer

Review the Considerations

Consider the following information before you upgrade XCOM Data Transport:

- Rollback of upgrade is possible *only* manually. Auto-rollback does not exist.
- The XCOM Data Transport Service Pack 11.6.01 installer acts as both a new installer program and an upgrade program. If the installer program does not find any installed XCOM instance on your computer, the installer acts as a new installer. If the installer finds a Release 11.6 XCOM instance, the installer acts as an upgrade installer.
- You can upgrade in either console mode or in silent mode:
  - To upgrade in console mode, see Upgrade in Console Mode.
  - To upgrade in silent mode, follow the instructions on Upgrade in Silent Mode.

Understand the Process

Understand the upgrade process, as follows:

1. Prepare for an upgrade, which includes uninstalling XCOM Data Transport Release 11.0.
2. Perform the upgrade tasks, which include running the installation program in either GUI mode or silent mode.
   The installation program performs the following tasks to install XCOM Data Transport Service Pack 11.6.01:
   a. Verifies the following pre-installation criteria:
      - The user who invokes the installer has administrative permissions.
      - No XCOM Data Transport processes are running.
   b. Installs the following entities:
      - XCOM Data Transport binary files and services
      - CAPKI
   c. Performs post-installation tasks.
3. Perform the post-upgrade tasks.

Prepare for Upgrade

Before you upgrade, complete the following steps:

1. Verify that you meet the Installation Prerequisites.
2. Coordinate with users for the best time to perform the upgrade.
3. Verify that no active transfers are running.
4. Stop all XCOM Data Transport applications.
5. Stop the Scheduler service.
6. Follow the instructions to complete these tasks:
   a. Delete the queue.

b. Back up files.
c. Uninstall your current product.

Delete the Queue

When performing an upgrade, remove existing queue entries by issuing the following command:

```
XCOMQM -R*
```

Back Up Files

Back up the following files to a different directory if you have edited them.

The following files are listed with their default directory locations. But your system might have them installed in directories other than the defaults.

- /var/spool/xcom/config/xcom.cnf
- /var/spool/xcom/config/xcom.glb
- /var/spool/xcom/config/xcom.ses
- /var/spool/xcom/config/xcom.tid
- /usr/lib/xcom/xcomlp
- /usr/lib/xcom/xcomntfy
- /usr/lib/xcom/xcompp
- /usr/lib/xcom/xcomend
- /usr/lib/xcom/xcompre
- /usr/spool/xcom/config/configssl.cnf
- /usr/spool/xcom/ssl/*

If you have customized the following files, save them also:

- /var/spool/xcom/convtab/atoe.tab
- /var/spool/xcom/convtab/etoa.tab

Consider backing up your configuration files. After you install XCOM Data Transport, compare the values in these backed-up files to the new files and revise the new files if necessary.

Uninstall Your Current Product

Uninstall XCOM Data Transport Release 11.0 before you install Service Pack 11.6.01. Before you uninstall Release 11.0, perform the steps in Back Up Files, the previous sections.

Perform Post-uninstallation Tasks

After you uninstall the XCOM Data Transport Release 11.0 on AIX, follow these steps:

1. Clear the txpi /txpis entries from /etc/inetd.conf file.
2. Refresh the inetd service < refresh -s inetd>.

After you uninstall the XCOM Data Transport Release 11.0 on Solaris, follow these steps:

1. Clear the txpi /txpis entries from /etc/inetd.conf file.
2. Refresh the inetd service <svcadm refresh network/txpi/tcp:default>.

Perform the Upgrade Tasks

To upgrade XCOM Data Transport to Service Pack 11.6.01, complete these tasks:

1. Review the Installation or Upgrade in Silent Mode.
2. Follow the instructions to:
   - Install Service Pack 11.6.01.
   - Perform the post-installation tasks.
   - Review the log files.

Uninstall or Roll Back

Use the following information to uninstall Service pack 11.6.01 or roll it back to Release 11.6 or 11.5.

Required roles: systems programmer

These procedures are applicable for both UNIX/Linux PC and AIX.

Before You Uninstall

Follow these steps:

1. Verify that no transfers are running.
2. Stop all the applications that access the $XCOM_HOME directory.
3. Stop the Scheduler service.

Uninstall in Console Mode

Follow these steps:

1. Before starting to uninstall XCOM Data Transport Service Pack 11.6.01 for Linux PC x64, log on as superuser (root).
2. Go to the Uninstaller subdirectory in the directory where you installed XCOM Data Transport for UNIX/Linux PC, the $XCOM_HOME/Uninstaller.
3. Issue the command:
   ```
   $ sh Uninstaller
   ```
4. Move through the installation process as follows. Respond to the prompt in each section in either of the following ways:
   - Enter the number that is associated with your choice.
   - Press Enter to accept the default.

   You have uninstalled XCOM Data Transport for UNIX/Linux PC x64 Service Pack 11.6.01.

   > **NOTE**
   >
   > For AIX,
   >
   > - After the uninstallation, clean the following directories if installation exits with 208 and if installer temp directories are not automatically cleaned up.
   >   - $INSTALLER_TEMP_DIR$/install.dir.*
   >   - $INSTALLER_TEMP_DIR$/ia_remove*
   >
   >   /tmp is the temporary directory for the UNIX system by default.
   > - If XCOM listeners are running, issue the command "refresh -s inetd" to refresh the inetd subsystem.

Uninstall in Silent Mode

To uninstall XCOM Data Transport in silent mode from the command line, follow these steps. Unlike the silent installation, the silent uninstallation does not require any response file.

1. Before starting to uninstall XCOM Data Transport for UNIX/Linux PC x64 Service Pack 11.6.01, log on as superuser (root).
2. Change to the Uninstaller subdirectory in the location where you installed XCOM Data Transport ($XCOM_HOME/Uninstaller).
3. Issue the command:

```
$ sh Uninstaller –i silent
```

4. To verify that the uninstallation occurred successfully, analyze the return codes.

NOTE

For AIX,

- After the uninstallation, clean the following directories if installation exits with 208 and if installer temp directories are not automatically cleaned up.
  - $INSTALLER_TEMP_DIR$/install.dir.*
  - $INSTALLER_TEMP_DIR$/ia_remove*

  /tmp is the temporary directory for the UNIX system by default.
- If XCOM listeners are running, issue the command "refresh -s inetd" to refresh the inetd subsystem.

Review the Log Files

To review the details of the uninstallation, read the following log files:

- $XCOM_HOME/Uninstaller/Logs/xcominstaller.log - contains warnings, error messages, and exit codes that occurred during the uninstallation.
- $XCOM_HOME/Uninstaller/Logs/*Uninstall*.log - contains the sequence of actions that the uninstaller performed.

Roll Back

To roll back XCOM Data Transport Service Pack 11.6.01 to your previous release 11.5 or 11.6, perform the following tasks manually. Automatic rollback is not supported.

NOTE

Rollback does not let you preserve suspended or scheduled transfers.

1. Uninstall the product, as described earlier.
2. Reinstall your previous release. For instructions, see XCOM Data Transport Release 11.6.
3. Reapply customizations from the files that you backed up from your previous release before you upgraded to Service Pack 11.6.01.
4. If you used XCOMAPI and XCOMQAPI for your custom applications that were compiled to work with Service Pack 11.6.01, recompile them to work with your previous release.

Install Using Docker

Learn how to install XCOM Data Transport for UNIX/Linux using Docker.

Required roles: systems programmer

As a system administrator, you can use Docker to install the product. Docker is used to package and deploy self-sufficient applications in Linux containers. Docker containers isolate the application and its dependencies from the underlying operating system and from other containers. The containerized apps share a single, common operating system, but they are separated from each other and from the host server.

Using Docker to install the product provides the following advantages:

Faster software delivery cycles

With the quicker software delivery cycles, Docker containers make it easy to install and update new versions of software into production quickly to take advantage of new features. Docker also allows you to roll back to a previous version.

Portable applications

Docker packages your application or software into a single unit that you can deploy anywhere that the Docker engine is installed.

Isolated applications and resources

Docker ensures that your applications and resources are isolated. Each container owns its resources, which are isolated from the other containers. You can run multiple XCOM containers on the same host server with separate ports to listen for incoming connections.

Security

Because Docker containers are isolated from each other, you have complete control over traffic flow and management. XCOM processes that are running in one container cannot be accessed from another container.

Installation Prerequisites

Before you install the XCOM Docker on Linux, ensure that these requirements are met.

Required roles: systems programmer

- You are logged in as a superuser (root).
- You have identified a system ID and name for the XCOM Data Transport system. The system ID can be one to four characters. The system name can be one to eight characters. These values are used as a unique identifier for the system.

Operating System Requirements

One of the following operating systems is required on the host machine:

- CentOS 7 or higher
- Red Hat 7 or higher
- Ubuntu 16.04 or higher

System Requirements

Confirm that the following software and hardware are installed on the host machine:

- Docker 1.13 or higher
- Docker Compose 1.16 or higher
- Unzip utility

NOTE

If SELinux is enabled in the enforcing or permissive mode, ensure that its policies let the XCOM container read and write files from the mapped volumes on the host machine. The installation script runs the command to change the context for the XCOM configuration files that are deployed onto the host machine.

Install XCOM Docker

Download and install the XCOM Docker package on the Linux host server. Use Docker to install XCOM Data Transport.

Required roles: systems programmer

When you install the XCOM Docker package, you have two options:

- If the Linux host server has Internet access, you can use the Broadcom Docker repository to install the XCOM Docker.
- If the host server does not have Internet access, use the TAR file.

Install XCOM Docker Using the Broadcom Docker Repository

1. Download the XCOM Docker package:
   a. In the Broadcom Support portal, go to **Mainframe Software** and select **My Downloads**.
   b. Use the **Search by Product Name** field to search for **XCOM**.
   c. Select **XCOM for Linux (PC)**.
   d. Select the **Token Download** icon for the desired product version. A token is required to use the Broadcom Docker repository for installation.

      A new window opens, with a link to the ISO file for the XCOM Data Transport for Linux PC Docker Install Package. The window also provides instructions to export the auto-generated user_name and token variables.

      NOTE
      The auto-generated token expires after two hours. If your token expires, repeat this step.
   e. Use the provided link to download the ISO file.
   f. Follow the instructions to export the variables. These variables are required to use the repository for installation.
2. Extract the contents of the ISO file to the Linux host server.

   Example:
   a. Create the mount point directory on Linux.
   ```
   sudo mkdir /mnt/iso
   ```
   b. Mount the ISO file on Linux.
   ```
   sudo mount -o loop /path/to/my-iso-image.iso /mnt/iso
   ```
   c. Create a destination directory.
   ```
   mkdir /your/destination
   ```
   d. Copy the files.
   ```
   mkdir /your/destination
   ```
   e. Unmount the ISO file.
   ```
   cp -r /mnt/iso/* /your/destination
   ```
3. Install XCOM Docker on the host server:
   a. Navigate to the directory containing the extracted ISO contents.
   b. (Optional) To install the software in silent mode, review the license agreement text file in the `/tmp` directory and issue the following command:
   ```
   export XCOM_ACCEPTEULA=1
   ```
   The license terms are accepted. When you run the `InstallXCOMDocker.sh` script, it skips the license agreement and proceeds with the installation, without requiring further intervention from you.
   c. Run the `InstallXCOMDocker.sh` script:
   - To install the software in the default location (/opt/CA/XCOMDocker), issue the following command:
   ```
   sh InstallXCOMDocker.sh install repo
   ```
   - To install the software in a custom location (for example, `/apps/XCOMDocker`), issue the following command:

```
sh InstallXCOMDocker.sh install repo /apps/XCOMDocker
```

If the specified directory does not exist, the script creates it. If the specified directory exists, but it contains files, the script fails.

The script installs XCOM Docker and displays a message when it is done. If the message indicates success, skip to Step 5. If the message indicates failure, go to Step 4.

4. Resolve the installation issues by reviewing the log file at `/tmp/xcomdockerinstall.log`. This log records your installation steps. When you have installed XCOM Docker successfully, go to Step 5.

   NOTE

   If you cannot resolve the issues, submit the log file to Broadcom Support.
5. Verify that the XCOM Docker image has been loaded to the local docker registry by issuing the following command:

   ```
   docker images | grep -i CAXCOM
   ```

XCOM Docker is installed. You can now configure and manage your Docker containers.

Install XCOM Docker Using the TAR File

1. Download the XCOM Docker package:
   a. In the Broadcom Support portal, go to **Mainframe Software** and select **My Downloads**.
   b. Use the **Search by Product Name** field to search for **XCOM**.
   c. Select **XCOM for Linux (PC)**.
   d. Select the link for the desired release.
   e. Download the following ISO files by selecting the **HTTPS Download** icon or the **Secure FTP Download** icon next to each one:

XCOM Data Transport for Linux PC Docker Install Package

Contains files that are needed for all XCOM Docker installations, regardless of the installation method.

XCOM Data Transport for Linux PC Docker Image in TAR Format

Contains the Docker image in TAR format.

2. Extract the files from the XCOM Data Transport for Linux PC Docker Install Package to a location of your choice.

   Example:

   a. Create the mount point directory on Linux.

   ```
   sudo mkdir /mnt/iso
   ```

   b. Mount the ISO file on Linux.

   ```
   sudo mount -o loop /path/to/my-iso-image.iso /mnt/iso
   ```

   c. Create a destination directory.

   ```
   mkdir /your/destination
   ```

   d. Copy the files.

   ```
   mkdir /your/destination
   ```

   e. Unmount the ISO file.

   ```
   cp -r /mnt/iso/* /your/destination
   ```
3. Extract the files from the XCOM Data Transport for Linux PC Docker Image in TAR Format to the `/tmp` directory.

   Example:

   a. Create the mount point directory on Linux.

   ```
   sudo mkdir /mnt/isotar
   ```

   b. Mount the docker tar ISO file on Linux.

   ```
   sudo mount -o loop /path/to/my-xcom-tar-iso-image.iso /mnt/isotar
   ```

   c. Copy the files.

   ```
   cp -r /mnt/isotar/* /tmp
   ```

   d. Unmount the ISO file.

```
sudo umount /mnt/isotar/
```

4. Install XCOM Docker on the host server:
   a. Navigate to the directory containing the extracted contents from the XCOM Data Transport for Linux PC Docker Install Package.
   b. (Optional) To install the software in silent mode, review the license agreement text file in the `/tmp` directory and issue the following command:

      ```
      export XCOM_ACCEPTEULA=1
      ```

      The license terms are accepted. When you run the `InstallXCOMDocker.sh` script, it skips the license agreement and proceeds with the installation, without requiring further intervention from you.
   c. Run the `InstallXCOMDocker.sh` script:
      - To install the software in the default location (`/opt/CA/XCOMDocker`), issue the following command:

        ```
        sh InstallXCOMDocker.sh install localtar /tmp/XCOMImage.tar
        ```

      - To install the software in a custom location (for example, /apps/XCOMDocker), issue the following command:

        ```
        sh InstallXCOMDocker.sh install localtar /tmp/XCOMImage.tar /apps/XCOMDocker
        ```

        Where:

XCOMImage.tar

The file name that is extracted from the XCOM Data Transport for Linux PC Docker Image in TAR Format

The script installs XCOM Docker and displays a message when it is done. If the message indicates success, skip to Step 6. If the message indicates failure, go to Step 5.

5. Resolve the installation issues by reviewing the log file at `/tmp/xcomdockerinstall.log`. This log records your installation steps. When you have installed XCOM Docker successfully, go to Step 6.

   NOTE

   If you cannot resolve the issues, submit the log file to Broadcom Support.
6. Verify that the XCOM Docker image has been loaded to the local docker registry by issuing the following command:

   ```
   docker images | grep -i CAXCOM
   ```

XCOM Docker is installed. You can now configure and manage your Docker containers.

Configure XCOM Docker Containers

Configure your XCOM Docker containers by customizing the `docker-compose.yml` file in the installation directory (`/opt/CA/XCOMDocker`). The path depends on your installation and deployment settings.

Required roles: systems programmer

The `docker-compose.yml` file lets you configure the following settings.

Customize SYSID and SYSNAME

The `docker-compose.yml` file contains SYSID and SYSNAME environment variables. Specify values for these variables to provide a unique identifier for the XCOM server. This identifier is required for Trusted Transfers and for retrieving history records from the history database. The SYSID value can be one to four characters. The SYSNAME value can be one to eight characters.

See the following example:

```
environment:
   - SYSID=LINX
   - SYSNAME=xcdocker
```

After you specify these values, restart the container. The XCOM global file is updated with the values, and your changes take effect.

Customize the Port Numbers

XCOM Data Transport uses ports 8044 through 8047 ports to listen for incoming connection requests. By default, these ports are exposed to the same port numbers on the host machine. You can expose these ports to a different set of port numbers on the host machine. For example, to expose the XCOM listener ports to ports 9044 through 9047, modify the ports section as shown in the following example:

```
ports:
     - 9044:8044
     - 9045:8045
     - 9046:8046
     - 9047:8047
```

Update the Volumes Section

To mount the physical storage from the host machine to the Docker container, update the volumes section. For example, to mount the `/var/data directory` from the host machine to the Docker container as the `/data` directory, add the following line to the volumes section:

```
volumes:
  - /var/data:/data
```

Manage User Accounts

Use Pluggable Authentication Modules (PAM) or a custom script to manage user accounts for the Docker container.

PAM

If you use PAM for authentication, configure SSSD to authenticate against host system user accounts. You can also configure PAM to authenticate against external sources like LDAP. The required PAM, pam_ldap, authConfig, and SSSD client libraries are packaged with the container. Configure the host machine to allow PAM to authenticate against host user accounts.

1. Install the `sssd-common` and `sssd-proxy` packages.
2. Create a PAM service for the container. See the following example:

   ```
   # cat /etc/pam.d/sss_proxy
   auth required pam_unix.so
   account required pam_unix.so
   password required pam_unix.so
   session required pam_unix.so
   ```

3. Create an SSSD config file `/etc/sssd/sssd.conf` . If the file is already available, review the content. Ensure that the permissions are set to 0600 and that they are owned by root: root. See the following example:

   ```
   # cat /etc/sssd/sssd.conf
   [sssd]
   services = nss, pam
   config_file_version = 2
   domains = proxy
   [nss]
   [pam]
   [domain/proxy]
   id_provider = proxy
   # The proxy provider will look into /etc/passwd for user info
   ```

```
proxy_lib_name = files
# The proxy provider will authenticate against /etc/pam.d/sss_proxy
proxy_pam_target = sss_proxy
```

4. Use the following command to start SSSD:

   ```
   systemctl start sssd
   ```

5. Use the following command to verify that a user can be retrieved with the SSSD:

   ```
   getent passwd -s sss <user>
   ```

6. Create a file with the name `xcomauth` at `/opt/CA/XCOMDocker` and update it with PAM configurations. See the following example:

   ```
   #%PAM-1.0
   auth required /opt/CA/XCOM/redistrib/pam_userpass/pam_userpass.so #nullok set_setrpc
   auth required pam_sss.so
   account required pam_sss.so
   password required pam_sss.so
   session required pam_sss.so
   ```

7. Add a volume mapping in the `docker-compose.yml` file to map the `xcomauth` file to the container:

   ```
   /opt/CA/XCOMDocker/xcomauth:/etc/pam.d/xcomauth
   ```

8. Add a volume mapping in the `docker-compose.yml` file to map the SSSD pipes file to the container:

   ```
   /var/lib/sss/pipes:/var/lib/sss/pipes
   ```

9. Enable PAM authentication by modifying the following global parameters:

   - AUTH_TYPE=PAM
   - PAM_PATH=/usr/lib64

   For more information about PAM authentication, see Pluggable Authentication Modules (PAM) Based Authentication.

10. Restart the container:

    ```
    sh XCOMAdmin.sh restart
    ```

    The configuration changes in the GLB file are applied.

Custom Script

To create user accounts that are specific to the container, you can create a custom script with the required commands that executes when the container starts up. Place the script inside the `/opt/CA/XCOMDocker/cmd` directory and give it permission to execute. A sample `customscript.sh` is available in this directory. By default, the XCOM container uses the `bin/start.sh` script as the entry point for the container. This script starts the XCOMD and XINETD services. If you use the custom script as the entry point, call `/opt/CA/XCOM/bin/start.sh` at the end of the custom script to allow XCOM to function properly. To execute the custom script with the default entry point, add a section in the `docker-compose.yml` as shown in the following example:

```
command: cmd/<customscript>.sh && bin/start.sh
```

Configure XCOM Data Transport to use the History or Trusted Database

You can configure XCOM Data Transport to store the history of transfers and configure the trusted database for passwordless transfers.

- Configure JDBC connectivity

  To perform trusted transfers, copy the appropriate database JDBC connection JAR files to `/opt/CA/XCOMDocker/jdbclib`. The standalone UI uses these files for connecting to the trusted database.
- Configure ODBC connectivity

  To store the `odbc.ini` and `odbcinst.ini` files, copy the appropriate database ODBC libraries to `/opt/CA/XCOMDocker/jdbclib`.

Manage XCOM Docker Container

Use the `XCOMAdmin.sh` script to start, stop, and restart your XCOM containers. This script is in the installation directory (default path `/opt/CA/XCOMDocker`).

Required roles: systems programmer

You can also use the script to verify the XCOM Data Transport version in the container and update it to another version. This script uses Docker Compose. Before you use this script, verify that Docker Compose is installed and is available in the path environment. Also, you must be a root user or a member of the docker user group to use the script.

Use the following commands with the provided script to manage the XCOM container:

- Start the XCOM container:
  ```
  sh XCOMAdmin.sh start
  ```
- Stop the XCOM container:
  ```
  sh XCOMAdmin.sh stop
  ```
- Restart the XCOM container to apply configuration changes:
  ```
  sh XCOMAdmin.sh restart
  ```
- Update the XCOM version to a specific maintenance level (for example, 11.6.01-20090) with the Docker repo:
  ```
  sh XCOMAdmin.sh update repo 11.6.01-20090
  ```
- Update the XCOM version to a specific maintenance level (for example, 11.6.01-20090) with the downloaded TAR file from Broadcom Support:
  ```
  sh XCOMAdmin.sh update localtar /tmp/XCOMImage-1160120090.tar 11.6.01-20090
  ```
- Get the XCOM version from the running container:
  ```
  sh XCOMAdmin.sh getversion
  ```
- Verify the status of the container:
  ```
  docker ps -a | grep -i CAXCOM
  ```
- Roll back the XCOM version in the container:
  a. Stop the container:
     ```
     sh XCOMAdmin.sh stop
     ```
  b. Open the `dockerimagetag.env` file in the `config` folder.
  c. Modify the XCOM_LINUX_IMAGE_TAG value to the previous image tag. To view a list of XCOM images and tags that use docker images, use the command `grep -i CAXCOM` .
  d. Start the container:
     ```
     sh XCOMAdmin.sh start
     ```
     The container is rolled back to a previous version.

Perform Health Check and Trace XCOM Container

Troubleshoot the XCOM Docker container installation and configuration with a health check. You can also capture information and send it to Broadcom Support.

Required roles: systems programmer

Before you follow these steps, ensure that Linux is running in the mode as mentioned under system requirements. Also, you must have read and write permissions to the installation directory.

Perform a Health Check on a Container

To perform a health check on a container, issue the following command:

```
docker ps -a|grep -i CAXCOM
```

The Status column displays one of the following values:

Starting
Indicates that the container is still starting.

Healthy
Indicates that the container is healthy.

Unhealthy
Indicates that the container is unhealthy. This status indicates that a single run of the container takes longer than the specified time. If a health check fails, the Docker daemon retries multiple times before declaring the container as unhealthy.

Exited
Indicates that the container is stopped or an error was encountered.

Capture Information about the Container

If you encounter issues that are related to starting up or exiting, capture the container information and share it with Broadcom Support.

Use the following command on the host machine to check the container logs that are created by the Docker daemon:

```
docker logs -f <xcomcontainerid>
```

xcomcontainerid
Specifies the XCOM container ID or name. To find the container ID or name, use the command `docker ps -a|grep -i CAXCOM`.

The output from the startup commands is displayed. This information is helpful when you encounter permissions issues with the directories or an unsuccessful execution of the entry point script.

Use the following command on the host machine to verify the details of a container:

```
docker inspect <xcomcontainerid>
```

The inspect command output is displayed. This output includes the following sections:

State Exit code
Identifies the reason for the container exits.

Mount
Provides details, read, and write access for the volume mappings that are defined in the `docker-compose.yml` file.

Config
Provides information about environment variables, health check details, and port mappings.

Network Settings
Provides Docker network details that are created for XCOM.

Uninstall XCOM Docker

To uninstall XCOM Docker, perform the following steps on the host system.

Required roles: systems programmer

1. Stop the active XCOM containers.
2. List the XCOM images that are available in the local docker repository:
   ```
   docker images|grep -i CAXCOM
   ```
3. Delete the images:
   ```
   docker rmi <ImageID>
   ```
   Repeat this command for all ImageID values that correspond to the XCOM images. If duplicate images use the same ID, use `<Repository:TAG>` instead of `<ImageID>` .
4. Delete the installation directory containing the XCOM Docker configuration files. The default path for the installation directory is `/opt/CA/XCOMDocker` .

Install Using Ansible

Learn how to install XCOM Data Transport using Ansible.

Required roles: system administrator

Ansible is an open-source tool with an agentless architecture that does not require you to install an agent on the target system. Ansible simplifies software management tasks such as installation, upgrade, and deployment of maintenance on a group of systems because you do not have to manually log on to each system.

To install, upgrade, and configure XCOM Data Transport with Ansible software, perform the following steps:

1. Familiarize yourself with the basic concepts of Ansible.
2. Prepare your environment.
3. Customize and run the playbooks.

Ansible Concepts

Before you use Ansible to install, upgrade, or configure XCOM Data Transport, familiarize yourself with its basic concepts.

Required roles: system administrator

Control System

The control system is a Linux system with Ansible software installed. The control system is the only system that hosts the XCOM Data Transport software installation packages that are used to deploy the XCOM Data Transport software on the target systems.

Target System

Target systems are any systems where you can deploy XCOM Data Transport from the control system. The target system can be a Linux or Windows system.

Inventory File

An inventory file provides a list of target systems on which you can install, upgrade, and deploy maintenance for XCOM Data Transport. The inventory file contains the IP addresses of the target systems. The file also contains the credentials that the control system uses to connect the target systems. The inventory file lets you specify the credentials in an encrypted format for security purposes. You can group the target systems and then define the common properties of target systems at a group level. For example, you can group target systems by platform, by department or by geographic location.

Playbook

A playbook is a workflow that describes a set of steps to perform on the target system. The playbook consists of variables and tasks.

Variable File

A variable file lets you define values for variables. You can reuse the same variable file in multiple playbooks. A variable file can be defined in the YAML key-value format. Keys that are defined in the playbook are substituted with the corresponding values at runtime.

Prerequisites to Install with Ansible

Prepare your environment to install XCOM Data Transport with the Ansible software.

Required roles: system administrator

Ensure that your environment meets the following prerequisites.

Control System Requirements

The control system must have the following software:

- Python version 2.73.6 or later
- Ansible version 2.9 or later

To use Windows servers as target systems, install the following software on the control system:

- Pywinrm package from Python on the CentOS or Red Hat server
  Issue the following commands to install Pywinrm:
  ```
  yum install python-pip pip
  ```
  ```
  pip install pywinrm
  ```
- `Ansible.windows` and `community.windows` collections from the Ansible galaxy
  Issue the following commands to install the collections:
  ```
  Ansible-galaxy collection install Ansible.windows
  ```
  ```
  Ansible-galaxy collection install community.windows
  ```

For more information about installing Ansible and the control system requirements, see the Ansible Documentation.

Target System Requirements

The control system uses the WinRM protocol to communicate with Windows targets and the SSH protocol to communicate with Linux targets.

Complete the following configuration on the target system:

- To allow key-based authentication, set up SSH keys on all Linux servers. You can also specify the user ID and password in the inventory file.

  NOTE

  `Ansible.cfg` is a configuration file that controls the Ansible behavior. To use the SSH username and password for connection, disable SSH key host checking in the file. To disable host checking, set `host_key_checking` to False in the file.

  Protect the `Ansible.cfg` file with appropriate file permissions to prevent other users from accessing or modifying the file.
- Set up WinRM-based authentication on Windows servers. To use the WinRM protocol, specify the user ID and password in the inventory file. Also specify the Python location for `Ansible_python_interpreter` in the inventory file. For more information about setting up Windows servers with WinRM, see Ansible Documents.

After you prepare the environment, customize and run the playbooks to install XCOM Data Transport.

Customize and Run Playbooks

Customize and run the sample Ansible playbooks to install and manage XCOM Data Transport.

Required roles: system administrator

Before you use the playbooks, ensure that your environment meets the installation prerequisites.

The playbooks are written in YAML and are easy to read, write, share, and understand. A playbook consists of variables and tasks. You can use sample playbooks that demonstrate how to use playbooks to automate various installation and configuration processes. The sample playbooks are in the `%XCOM_HOME%/ansible/playbooks` directory. You can customize the playbooks to match your needs or you can create your own playbooks. For more information about creating and editing playbooks, see the Ansible Documentation.

The following sample playbooks are provided:

Linux-new-install.yml
Installs XCOM Data Transport on Linux servers.

Linux-upgrade-install.yml
Upgrades the existing XCOM Data Transport installations to the latest release. The existing installations must be 11.6 or higher.

Linux-maintenance-install.yml
Installs maintenance on XCOM Data Transport Linux servers.

Windows-new-install.yml
Installs XCOM Data Transport on Windows servers.

Windows-upgrade-install.yml
Upgrades the existing XCOM Data Transport Windows installations to the latest release. The existing installation must be 11.6 or higher.

Windows-maintenance-install.yml
Installs maintenance on XCOM Data Transport Windows servers.

Linux-deploy-sslkeys.yml
Distributes a copy of SSL keys to all Linux servers.

Windows-deploy-sslkeys.yml
Distributes a copy of SSL keys to all Windows servers.

Linux-deploy-splunk.yml
Distributes the Splunk configuration script and updates `xcom.glb` on Linux servers.

Windows-deploy-splunk.yml
Distributes the Splunk configuration script and updates `xcom.glb` on Windows servers.

Before you run any playbooks, verify the system connectivity and define an inventory file.

Verify System Connectivity

Use the following commands to verify the connectivity between the control system and the target systems:

- To verify the connectivity with a Linux_Upgrade group of servers, issue the following command:

```
watch -n 30 'ansible -i inventory --vault-password-file vault_pwd -m ping Linux_Upgrade'
```

- To verify the connectivity with a Windows_NewInstall group of servers, issue the following command:

```
watch -n 30 'ansible -i inventory --vault-password-file vault_pwd -m win_ping Windows_NewInstall'
```

Define the Inventory File

You can define the inventory file in a YAML, JSON, or INI file format. Sample YAML `inventory.yml` and INI `hosts.ini` files are available in the `%XCOM_HOME%/Ansible/playbooks` directory for your reference.

You can group the set of target systems where you can run the installation, configuration, and maintenance tasks. The sample inventory file contains the system groups that are defined with the names Windows_Maintenance, Windows_New Installation, and Windows_Upgrade. These names correspond to the respective installation tasks. If you modify the group names, update the respective playbook to refer to the modified group name. This guidance applies to other groups that are defined for Linux targets.

The passwords in an inventory file can be specified in a plain text or encrypted format. To encrypt the passwords, use the `ansible-vault` command. You must specify a key in a `vault_pwd` text file or must type it when prompted. The key can be similar to other passwords that encrypt or decrypt passwords in the inventory. If you use the `vault_pwd` file, protect it with appropriate file permissions to prevent other users from accessing or modifying the file.

- To use the key in a `vault_pwd` text file, issue the following command:

```
ansible-vault your-encrypt-string --vault-password-file vault_pwd 'your-plain-text-password' --name 'your-ansible-password'
```

- To type the key at the prompt, issue the following command:

```
ansible-vault your-encrypt-string 'your-plain-text-password' --name 'your-ansible-password'
```

- To run the playbook with the `vault_pwd` file, issue the following command:

```
ansible-playbook -i inventory playbook.yml --vault-password-file vault_pwd
```

- To run the playbook with an interactive password, issue the following command:

```
ansible-playbook -i inventory playbook.yml -ask-vault-pass
```

Run the Playbooks

After you have tested the connectivity and defined the inventory file, you can run the Ansible playbooks to install and manage XCOM Data Transport. If you run the playbooks as a non-root user on a control system, ensure that the required modules are available for the current user. The software packages and configuration files should be accessible to the logged-in user.

See the following guidelines:

- To use the software management playbooks:

    - Download the XCOM Data Transport software packages to any location on the system. Prepare the response files for a fresh installation or upgrade. For more information about downloading software packages, see Download Product Package.
- To use the configuration task playbooks:
    - For SSL certificates deployment, create a directory with the required SSL certificates to distribute to the target system.
    - For Splunk configurations, prepare a customized post-processing script to enable communication with Splunk.

When the required software and configuration files are ready, update the variable file to specify the location of the installation packages and configuration files. This file, named `main.yml`, is available in the `%XCOM_HOME%/ansible/vars` directory. The parameters in this file are preceded with comments that describe them.

Getting Started

Provides an overview of the product.

XCOM Data Transport is a family of software products that operate on various operating systems. XCOM Data Transport operates under SNA using LU 6.2, or under TCP/IP, to provide high-speed data transfer between the supported systems. The supported systems are mainframes, midrange, PCs, servers, and workstations. You can send files from your local system to remote systems, and you can retrieve files from those remote systems. Local and remote systems have the same transfer capabilities.

Features

XCOM Data Transport provides peer-to-peer communications using LU6.2 or TCP/IP over a wider range of systems than any other product. All the major features of XCOM Data Transport are supported across the product line.

File Transfer

XCOM Data Transport supports high-speed file transfers between all supported operating systems. In some environments, you can start thousands of transfers resulting in hundreds of simultaneous transfers, all with a single operation. Parallel sessions are possible in varying degrees throughout the product line.

You can totally automate XCOM Data Transport transfers. On a PC, you can actively access other applications (for example, word processing) when receiving or transmitting files in the background. Comprehensive management tools allow for effective central-site control of XCOM Data Transport activity, including advanced problem determination features.

XCOM Data Transport supports transfers between any two systems in an SNA network or a TCP/IP network with one of the following methods:

- By using the z/OS, z/VM, or z/VSE mainframes for store-and-forward
- Through Independent Logical Unit (ILU) support over the SNA (Systems Network Architecture) backbone
- Through Dependent Logical Unit (DLU) support over the SNA (Systems Network Architecture) backbone
- Through use of the TCP/IP network (except for z/VM and Stratus)
- Through use of TLS (Transport Layer Security) or SSL (Secure Sockets Layer) connections over a TCP/IP network (except for z/VSE, z/VM and Stratus)

Types of Transfers

XCOM Data Transport performs the following transfers:

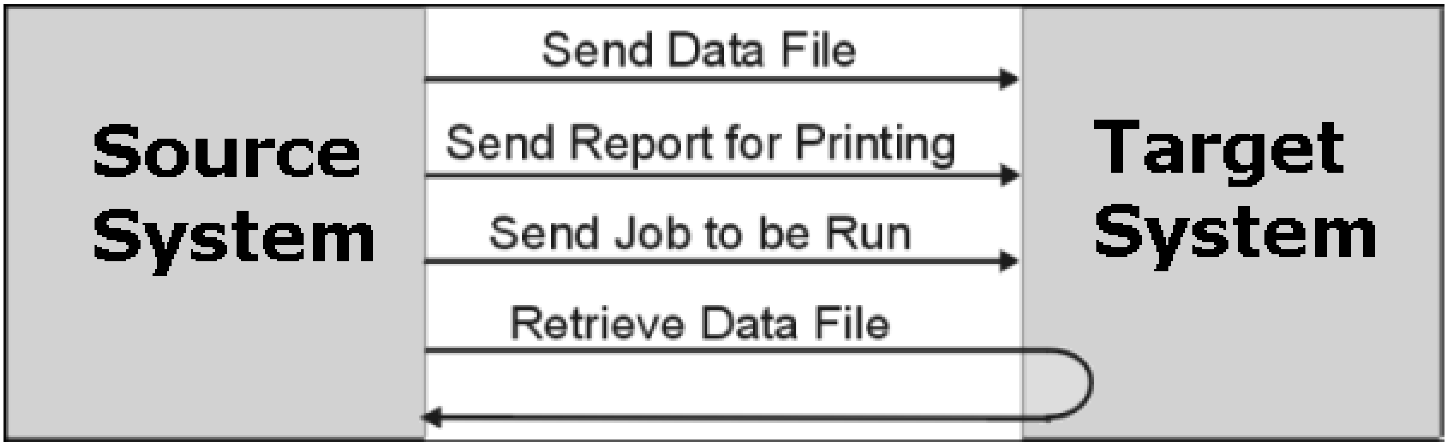
Sending Files
With XCOM Data Transport, a local system can send a data file to be stored on the remote system in a specified remote file.

Sending Reports
XCOM Data Transport can send a report to be printed on a remote system.

Sending batch jobs for execution
XCOM Data Transport can send a job to be executed on a remote system.

Retrieving files
When a system starts the transmission request, it can also retrieve a file from a remote system and store it in a specified local remote file.

The following flow chart illustrates the type of transfers supported by the product:



PU Type 2.1 Support

XCOM Data Transport supports PU Type 2.1 connections to allow direct interchange of files between the Windows operating environment, NetWare workstations, and others. Support for Independent Logical Units (ILUs) allows XCOM Data Transport delivering data in Advanced Peer to Peer Networking (APPN) and Low Entry Networking (LEN) networks. Means PCs and midrange that are attached to the same SNA or APPN network can exchange data even if they are not directly connected.

TCP/IP Support

XCOM Data Transport can use TCP/IP to perform transfers between XCOM Data Transport platforms that support TCP/IP and that are running r3.0, r3.1, r11, r11.5 or, r11.6. TCP/IP support is provided between the following platforms:

- i5/OS (AS/400)
- z/Linux
- Linux s390x
- Linux x86
- Linux x64
- NetWare
- Open VMS Alpha
- HP NonStop (Tandem)
- z/VSE
- Windows family
- z/OS
- Most common UNIX platforms

You can use the Transport Layer Security (TLS) or Secure Socket Layer (SSL) to perform secure TCP/IP transfers between platforms. The platforms must be running XCOM Data Transport r11 and above and must have this support enabled. XCOM Data Transport uses OpenSSL to encrypt the transmitted data and adds a digital signature to the encryption of the transmitted data. Secure TCP/IP support is provided between the following platforms.

- i5/OS (AS/400)
- z/Linux
- Linux s390x
- Linux x86
- Linux x64
- Windows family
- z/OS
- Most common UNIX platforms

Report Distribution

XCOM Data Transport allows z/OS, z/VM, z/VSE, I5/OS (AS/400), and OpenVMS Alpha users to take print output from any supported system and automatically transfer it to another system for printing. The application programs producing the reports do not require any modification to support XCOM Data Transport report distribution, and no operator intervention is required at either end.

RJE Replacement

Current Remote Job Entry (RJE) systems contain inherent limitations. Remote systems can submit work to the host for processing and receive print data, but the host cannot distribute processing tasks to idle processors residing on the network. A further concern for data processing managers is the requirement that users are familiar with Job Entry Subsystem (JES) commands to operate the system.

XCOM Data Transport avoids these limitations by taking advantage of the LU 6.2 and TCP/IP protocols, providing a peer-to-peer relationship between all supported systems. Any XCOM Data Transport system is able to send and receive batch jobs and print data from any other XCOM Data Transport system without formatting constraints.

For example, an i5/OS (AS/400) user can do the following:

- Automatically retrieve files from a number of attached PCs.
- Process the data.
- Generate a report.
- Send one copy of the report back to the source PC for printing.
- Send another to the z/OS mainframe for printing on a high speed printer.

You can easily implement XCOM Data Transport without any changes to your existing application programs. Data is transferred with greater integrity and higher efficiency.

Support of Most Operating Systems

Because XCOM Data Transport supports the TCIP/IP protocols, it can transfer data between various platforms. XCOM Data Transport is now available on the following systems:

- HP TRU64 UNIX (Digital UNIX)
- HP Non-Stop (Tandem)
- HP-UX PA RISC
- HP-UX IA64
- IBM AIX
- IBM AIX 64
- i5/OS (AS/400)
- z/Linux
- Linux s390x
- Linux x86
- Linux x64
- MS Windows
- NCR 3000 (AT&T)
- Novell NetWare
- Open VMS Alpha
- SCO Open Server
- SCO UnixWare
- Stratus VOS
- Sun Solaris
- Oracle Solaris SPARC 64
- Oracle Solaris x86 64
- z/OS
- z/VM
- z/VSE

Data Link Types

XCOM Data Transport supports the following data link types:

- SDLC
- X.25
- Local Area Network (such as Token Ring and Ethernet)
- All SNA data links, including channel-based links
- TCP/IP

Standard Features

The following features are standard to XCOM Data Transport:

- Simple installation
  You can install XCOM Data Transport without hardware changes to your system.
- Initiation by either system (any-to-any)
  Either system can send and retrieve data files.
- Low maintenance
  There are no hooks or patches into the operating system.
- Choice of interface
  You can choose from batch/command line, programming (on supported platforms), and menu interfaces.

Standard Functions

The following functions are offered over most of the XCOM Data Transport platforms:

- Compression
  XCOM Data Transport offers a wide range of compression options on most platforms.
- Packing
  XCOM Data Transport can pack records into fixed-size data transfer blocks as large as 31K, significantly improving performance and throughput.
- ASCII/EBCDIC translation
  XCOM Data Transport can translate data between ASCII and EBCDIC formats as needed. Translations occur on the ASCII-based platform.
- Checkpoint/Restart
  All components of XCOM Data Transport support checkpoint/restart. Transfers that are stopped or fail prior to completion automatically resume, continuing from the last checkpoint.
- Store-and-forward
  Users communicating through a common z/OS, z/VM, or z/VSE hub can perform data transfers even if the remote (target) machine is not communicating or turned on at the time of the initial transfer. XCOM Data Transport ensures that the data is sent as soon as the device is available.
- Remote spooling
  XCOM Data Transport allows z/OS, z/VM, z/VSE, i5/OS (AS/400), and Open VMS Alpha users the following reporting options:
  - XCOM Data Transport on all platforms can receive reports.
  - XCOM Data Transport on all platforms can send a file to a remote XCOM Data Transport partner, requesting that it be treated as a report.
  - Some XCOM Data Transport platforms can also take reports off the system spool and forward them to another XCOM Data Transport platform without operator action. This automatic report transfer facility is called Process SYSOUT on z/OS and z/VSE, and it is called XQUE on AS/400, HP NonStop (Tandem), and Open VMS Alpha. The z/VM platform does not allow automatic processing of spooled files. However, spooled files on z/VM can be manually received and redirected.

High Capacity and Performance

XCOM Data Transport is optimized for high-speed bulk data transfer. For instance, XCOM Data Transport for z/OS can allow hundreds of simultaneous file transfers from a single system, depending upon your hardware and software configuration. Comparatively, CICS-based products limit the user to a maximum of 34 simultaneous transfers, and many other VTAM file transfer products are faced with similar limitations.

Security

XCOM Data Transport interfaces with Pluggable Authentication Module (PAM) or the native security facility on all supported systems that are based on user preference.

When security is invoked, you are required to provide a valid user ID and password for the remote system. For example, in the z/OS environment, an interface is also provided to IBM RACF, ACF2, and Top Secret. Unlike most other communication facilities, XCOM Data Transport encrypts passwords. Encrypting passwords ensures that communications line tapping does not breach security.

XCOM Data Transport also has special security capabilities that can help data centers handle their individual needs. The security features of XCOM Data Transport allow installers to specify what can or cannot run under the privileges of someone other than the person requesting the transmission. These security features can also force user IDs from both remote systems to be the same or different. For otherwise unsatisfied security needs, XCOM Data Transport supplies various user exits, which enable user-written security packages to be fully integrated.

XCOM Data Transport can also use the Secure Socket Layer (SSL) or Transport Layer Security (TLS) to perform data transfers under TCP/IP. XCOM Data Transport provides certificate authentication, data encryption, and data integrity to ensure that all data transfers using TLS/SSL are secure.

Management

An important feature for any enterprise-wide information product is the ability to effectively control and manage the distribution of files and work throughout the network. XCOM Data Transport systems maintain a comprehensive log of all transfer activity. Utilities are provided to allow the system administrator to view the log online and modify the status of pending or currently active transfers.

Details of any transfer errors are also maintained in the log, allowing rapid problem determination and resolution. In addition, messages signaling the completion of any XCOM Data Transport event can be directed to a user in the network.

Process of Data Transport

To understand the data transport function in a very simplified and generalized way, consider a scenario.

For example, when a local system transfers a file to a remote (partner) system, following steps are performed:

1. Initiation
   The user submits a batch program, starts the menu (the menu interface) or a customer program written using the XCOM API (application programming interface) to initiate the transfer.
2. Information verification
   XCOM Data Transport verifies the information contained in the request. For example:
   - When requesting a send file transfer, XCOM Data Transport checks whether the local file exists on the local system.
   - When requesting a receive file transfer, XCOM Data Transport checks whether the file exists on the remote system.
3. Information confirmation
   If the information is confirmed, XCOM Data Transport starts the file transfer.
4. Completion
   The transfer completes and XCOM Data Transport logs the details of the transfer in a log.

   NOTE
   The previous example illustrates a general idea of how XCOM Data Transport works, please be aware that it is simplified; there are many more steps involved in the process.

Response to Remote Requests

Use XCOM Data Transport to monitor the network for incoming requests.

You can use XCOM Data Transport to monitor the network for incoming requests. Upon detecting one, XCOM Data Transport determines whether it is a request to send a file inbound (from the remote system to this system) or outbound (from this system to the remote system).

Ports and Memory

XCOM Data Transport uses the following four ports to listen for incoming connection requests:

- Port TCP 8044 for IPv4 non-SSL
- TCP 8045 for IPv4 SSL
- TCP 8045 for IPv6 non-SSL
- TCP 8047 for IPv6 SSL

XCOM Data Transport uses the shared memory to communicate between the processes. By default, the XCOM Data Transport application can access the file system of the host server. The application and reads and write the files to the location where the user has permission to read, write, and execute jobs.

File Transfers

You can use the file transfer feature to send or retrieve files from a remote system to a local system.

When XCOM Data Transport transfers a file from a remote system to your system, the following actions occur:

- The remote system requests XCOM Data Transport to send a file to your system.
- XCOM Data Transport allocates memory to the requesting process and opens the file.
- XCOM Data Transport then reads the data records from the file.
- XCOM Data Transport transfers the file to your system.
- Your system receives the file.

Job Transfers

When XCOM Data Transport transfers a job from a remote system to your system, the following actions occur:

- The remote system requests XCOM Data Transport to submit a job to your system.
- XCOM Data Transport submits the job to your system.
- Your system receives the job file.

Report Transfers

The report transfer feature allows a remote system to send a report to a local system. XCOM Data Transport provides a high degree of print redirection and spooling capabilities.

When XCOM Data Transport transfers a report from a remote system to your system, the following actions occur:

- The remote system requests XCOM Data Transport to send a report to your system.
- XCOM Data Transport writes the report to an output spool file.
- XCOM Data Transport transfers the file to your system.
- Your system retrieves the report from the spool file.

File Transfers

You can use the file transfer feature to send or retrieve files from a remote system to a local system.

When XCOM Data Transport transfers a file from a remote system to your system, the following actions occur:

- The remote system requests XCOM Data Transport to send a file to your system.
- XCOM Data Transport allocates memory to the requesting process and opens the file.
- XCOM Data Transport then reads the data records from the file.
- XCOM Data Transport transfers the file to your system.
- Your system receives the file.

Job Transfers

When XCOM Data Transport transfers a job from a remote system to your system, the following actions occur:

- The remote system requests XCOM Data Transport to submit a job to your system.
- XCOM Data Transport submits the job to your system.
- Your system receives the job file.

Report Transfers

The report transfer feature allows a remote system to send a report to a local system. XCOM Data Transport provides a high degree of print redirection and spooling capabilities.

When XCOM Data Transport transfers a report from a remote system to your system, the following actions occur:

- The remote system requests XCOM Data Transport to send a report to your system
- XCOM Data Transport writes the report to an output spool file.
- XCOM Data Transport transfers the file to your system.
- Your system retrieves the report from the spool file.

Remote System Information

This section contains information about important aspects of the operating systems supported by XCOM Data Transport that you should be aware of when performing transfers.

For more specific information about operating XCOM Data Transport on a specific platform, see the XCOM Data Transport guides for that platform and the manufacturer's guides.

The following topics are covered for each platform, as appropriate:

- Naming conventions
- Types of files supported
- Additional features
- Restrictions

HP NonStop (Tandem)

This article discusses the important aspects of the HP NonStop operating system.

Naming Conventions

Use the following format to name a HP NonStop file:

```
\<system>.<volume>.<subvolume>.<filename>
```

All of these components are restricted to eight characters, except as indicated below.

The following list describes the parts of an HP NonStop file name:

system
Specifies the system name. Up to seven characters.

volume
Specifies the disk name.

subvolume
Specifies a directory name.

filename
Specifies the name of your file.

Example:

The following example uses a volume of $CLX12, a subvolume of SCI, and a file name of FILE1:

```
$CLX12.SCI.FILE1
```

The HP NonStop file system is not a tree structure. Each volume.subvolume is independent, that is, it has no subvolumes above or below.

Types of Files Supported

XCOM Data Transport for HP NonStop supports the following file types through ENSCRIBE, Tandem's disk file architecture:

- Edit files
- Unstructured files
  Unstructured files are large-byte arrays. Data in these files is accessed by using the relative byte address and the READ-COUNT or WRITE-COUNT parameters in the system procedure calls. The application program determines the way in which they are used. An EDIT file is a type of unstructured file signified by the file code 101.
  For more information about ENSCRIBE and unstructured files, see the *ENSCRIBE Programmer's Guide*.
- Structured files
  XCOM Data Transport supports entry-sequenced, relative, and key-sequenced structured files:
  - Entry-sequenced files
    Entry-sequenced files are sequential files. Records are stored in the order in which they are entered. These records are variable in length and cannot be added or deleted. They are accessed by their record address.
  - Relative files
    Relative files are ordered by relative record number. The space allocated for each record is specified when the file is created. Records in these files can be deleted and added again in place.
  - Key-sequenced files
    Key-sequenced files are supported only for the Replace operation. The file must already exist for XCOM Data Transport to perform an action on it.

File Type Specification

File type specification differs for send requests and received requests, described as follows:

- Send Requests
  When you send a file from HP NonStop (locally initiated), the remote XCOM Data Transport determines the file type when it opens the file.
- Receive Requests
  For locally or remotely initiated receive requests, the file type must be specified by the GUARDIAN_FILE_TYPE parameter. Use one of the following values:
  - EDIT
  - UNSTRUCTURED
  - ENTRY_SEQ
  - RELATIVE

Remotely Initiated Send Requests

For remotely initiated transfer requests (for example, send a file, job, or report), use the following record formats, which create the indicated Guardian file types:

F

Relative

FB

Entry Sequence

VB

Edit

U

Unstructured

NOTE
Key sequence files are supported only if the file exists. You can do a replace but not a create.

i5/OS (AS/400)

This article contains information about important aspects of the i5/OS operating system.

Naming Conventions

Use the following format to specify an i5/OS file:

```
libraryname/filename(membername)
```

The following list describes the parts of an i5/OS file name:

- libraryname
  The name of the library that holds the file.
- filename
  The name of the file you wish to access. Periods are allowed within the file name.
- membername
  The name of the member in the file. If this component is omitted, it defaults to the file name.

Types of Files Supported

In addition to the standard file type discussed above, the Save File format is also supported. When you wish to send such a file to a System i5 from a z/OS or z/VSE system, the file must exist on the target system prior to your transmission.

Additional Features

XQUE is an XCOM Data Transport feature that allows the unattended transfer of reports from output queues to other XCOM Data Transport nodes.

XQUE can select specific classes of reports (based on the user, job name, form, and so on) from output queues. XQUE also allows user and workstation groups to be equated to printer destinations on remote XCOM Data Transport nodes. You can use XQUE, for example, to get reports back to your host system that are generated on a System i5 that you reach through IBM's HCF facility, or between multiple i5/OS (AS/400) systems connected within a pass-through environment.

Configuration Issues

If you are configuring the VTAM LU that represents the System i5 on a mainframe, make sure that the VTAM USS message 10 is not sent to that LU. IBM's APPC software cannot start a session when this message, commonly called the welcome message, is sent.

To prevent this problem, the VTAM or NCP USSTAB definition must be set to a table that does not have a USSMSG10. The table that IBM originally provided with VTAM is a good alternative because it does not include message 10.

Case Sensitivity

The IBM i5/OS is case-sensitive. Enter the user ID and password in uppercase.

Novell NetWare

This article discusses the important aspects of the Novell NetWare operating system.

Naming Conventions

Use the following format to name a Netware file:

> **NOTE**
> XCOM Data Transport for LAN Workstation accesses files from any Novell file server in a NetWare network.

```
[server\]volume:directory\subdirectory\...\filename
```

Types of Files Supported

XCOM Data Transport for NetWare LAN supports standard NetWare file types.

Destination Printer Information

When sending a report to a NetWare system, specify the Destination parameter value or the Destination Printer field in the following form:

```
\\server name\printer queue name
```

XCOM Data Transport limits the length of this field to 21 characters. The actual name on the destination system can be longer.

Restriction

XCOM Data Transport for NetWare LAN does not support library transfers to Novell NetWare systems.

OpenVMS

This article discusses the important aspects of the OpenVMS operating system.

Naming Conventions

Use the following format to name an OpenVMS Alpha file:

```
device[directory]filename.type;version
```

The entire file specification can be a maximum of 255 characters. The file type can be a maximum of 31 characters.

The following list describes the parts of an OpenVMS file name:

device

Specifies the disk drive name. If the device is not specified, the default provided in the SYSUAF (as defined on the DEC system) for that user is used.
Range: 1 to 15 characters.
Note: The XCOM Data Transport remote USERID field determines the SYSUAF USERID.

directory

Specifies the directory and subdirectory information. If this information is not provided, defaults are selected as described under “device” above.
Note: XCOM Data Transport accepts angle brackets (< >) in OpenVMS file names, which are converted to square brackets on the DEC system.

Example:
PLAYERS1:<BRIDGES>CARD.DAT
is treated as equivalent to
PLAYERS1:[BRIDGES]CARD.DAT

filename.type

Specifies the specific file within the directory. OpenVMS null file names are used if the file name and type are not provided.

version

Specifies the version of the file. The OpenVMS operating system can keep multiple versions of a file each time that file is saved. It is normal to omit this number to indicate that you want the most recent version of a file, the highest version number.

For more information about OpenVMS file specifications, see the OpenVMS documentation.

Restrictions

The following restrictions apply to XCOM Data Transport for OpenVMS Alpha:

- Specifying transfer type
  All transfers must be TYPE=SCHEDULE (for batch) or QUEUED (from ISPF).
- Non-queued host transfers
  Due to restrictions in the DEC SNA software, the z/OS or z/VSE TYPE=EXECUTE (non-queued) transfer feature fails with an 8003 sense code. It is not supported by XCOM Data Transport to an OpenVMS system.
- Operating system
  XCOM Data Transport currently supports the OpenVMS Alpha operating system.
- Connectivity
  Specifies the DECNET/SNA software is based on the Physical Unit 2.0 standard and not on the more flexible 2.1. This means that the system must be connected to a VTAM (PU 5) system in an SNA network. XCOM Data Transport uses the store-and-forward function (described previously as an additional z/OS, z/VM, and z/VSE feature) to transfer files with other XCOM Data Transport partners.
- Multiple session configuration
  Digital does not support parallel sessions with a z/OS or z/VSE system. However, if three file transfers are needed concurrently with an OpenVMS system, it does allow you to define three APPC logical units as a group. A group name can be from one to eight characters. The first character must be alphabetic, while the rest can be any combination of alphanumeric or national characters. Try to use mnemonic names. This feature is useful for assigning nicknames as well.

Example:

The following example calls three logical units, LUD1, LUD2, and LUD3, and assigns them a group name of LAVAX. Code the GROUP and LU parameters for this #PSOTAB entry as follows:

```
GROUP=LAVAX,
LU=(LUD1,LUD2,LUD3)
Type LAVAX as the remote system name to schedule transmissions to this VAX through the menu interface. When
 using the batch interface, use the GROUP parameter instead of the LU parameter. Use the following code:
GROUP=LAVAX
```

Groups can be used with all the XCOM Data Transport interfaces, including the Process SYSOUT Interface.

- Initiating the session bind request
  Although separate VTAM LU names can be used for XCOM Data Transport sessions, you should not LOGAPPL these LUs to XCOM Data Transport when configuring on z/OS, z/VSE, or z/VM. This fails with an 0801 or 8003 sense code, because the DEC software must initiate the session bind request.
- Compression options

Large packing is supported as well as a number of different compression algorithms.

- ASCII-based
  The OpenVMS platform is an ASCII-based system.

Stratus

This article discusses the important aspects of the Stratus operating system.

Naming Conventions

Use the following format to name the Stratus files:

```
#top_directory>group_directory>home_directory>filename.suffix
```

All names must be unique to that level.

The following list describes the parts of a Stratus file name:

top_directory
Specifies the physical disks.
Range: 1 to 32 characters.

group_directory
Specifies a group of user home directories.
Range: 1 to 32 characters.

home_directory
Specifies the user's home directory. This directory resides in a group directory.
Range: 1 to 32 characters.

filename
Specifies the name of the Stratus file. Required.
Range: 1 to 32 characters.

suffix
Specifies a file classification. You can have multiple suffixes at the end of a file name. Each suffix starts with a period. The following list describes some common Stratus suffixes for different file types:

source
Suffixes: .pl1, .cobol, .c
Examples: payroll.c, application.cobol

object
Suffixes: .obj
Examples: payroll.obj, application.obj

list
Suffix: .list
Examples: payroll.list, application.list

error
Suffix: .error
Examples: payroll.error, application.error

program module
Suffix: .pm
Examples: payroll.pm, application.pm

command macro
Suffix: .cm

Examples: start_up.cm, compile_and_bind.cm

back up

Suffix: .backup
Examples: payroll.c.backup

Types of Files Supported

Stratus supports the following file types for remotely initiated transfers:

Fixed

This type of file contains records of the same size. Each record is stored in a disk or tape region holding a number of bytes that is the same for all the records in the file.

Sequential

This type of file contains records of varying sizes in a disk or tape region holding approximately the same number of bytes as the record (for example, the record storage regions vary from record to record). Records can only be accessed on a record-by-record basis.

Additional Features

Know the following additional features of XCOM Data Transport for Stratus:

Security option

XCOM Data Transport for Stratus can use its own account file to verify the user ID and password and to map the XCOM Data Transport user ID to a VOS user ID to check for file access. If this option is turned on and the remote user ID/password combination is invalid, XCOM Data Transport for Stratus rejects the request.

Restart/Recovery facility

XCOM Data Transport for Stratus can attempt periodic data transmissions after the initial file transfer has failed. A certain number of retries can be specified through the xcom_ser.pm file.

Restrictions

The following restriction applies to XCOM Data Transport for Stratus:

No library transfers

XCOM Data Transport for Stratus does not support the transfer of libraries from the mainframe.

UNIX or Linux

This article discusses the important aspects of the UNIX or Linux operating systems.

Naming Conventions

Use the following format to name a UNIX or Linux file:

```
/directory/subdirectory/.../filename
```

Use up to 256 characters for the entire path of the file; there are no restrictions on size for the individual parts of the path.

The following list describes the parts of a UNIX or Linux path:

/ (slash)

The root directory when it is in the first position: otherwise, the slash separates directories and file names in the path.

directory

Specifies the directory that contains the file. You can specify more than one directory in a path.

filename

Specifies the name of the UNIX or Linux file.

Types of Files Supported

XCOM Data Transport for UNIX or Linux supports standard UNIX or Linux file types.

Restriction

XCOM Data Transport does not support library transfers to UNIX or Linux systems.

Trusted Access

XCOM Data Transport supports Trusted Access. To use the Trusted Access feature when transferring to UNIX- or Linux-based platforms, note the following:

- The USEROVR and USERPRO default table parameters for XCOM Data Transport for z/OS or z/VSE must be set to YES.
- Specify USERID=' ', and no passwords in the parameters for the transfer.
- The user ID must be configured for Trusted Access on the UNIX or Linux partner. For more information, see the XCOM Data Transport for UNIX and Linux documentation.

Windows

This article discusses the important aspects of the Windows operating systems.

Naming Conventions

XCOM Data Transport supports the standard Windows file names and the Universal Naming Convention (UNC). Some of the file naming conventions are outlined in this article.

Use the following format to name files when using standard Windows file names:

```
d:[\][directory name\..\]filename[.ext]
```

NOTE
This format can be used only when the drive is a local drive on the Windows system. Do not use this format for mapped or redirected drives. Use UNC conventions only for mapped or redirected drives.

Use the following format to name files when using UNC file names:

```
\\server name\share name\directory\filename
```

The following list describes the parts of the file names and UNC file names:

d

Specifies a particular device, indicated as a drive letter. This value is used for local drives only. This value is required.

directory name

Specifies one or more optional directories and subdirectories. This value is required.
Subdirectories can take the form of *name[.ext].* The form of the directory name and the file name depend on the operating system running on the server.

filename

Specifies the name of the data file. This value is required.
For FAT file systems, *filename* is 1 through 8 characters.
NTFS and HPFS file systems support long file names that are up to 256 characters, including the extension.

Names may or may not be case-sensitive, depending on the file system on the server. For FAT, NTFS, and HPFS, names are not case-sensitive. You can use uppercase and lowercase when creating a name, and they display as typed, but internally Windows makes no distinction. For example, MYFILE and MyFiLe are considered to be the same file.

Windows also creates an MS-DOS-style name based on the long name for compatibility with environments where long file names are not always supported.

ext

Specifies the file extension.

For FAT file systems, the extension is up to 3 characters.

For NTFS and HPFS, the extension is included in the long file name limit of 256 characters.

If you do not specify an extension, XCOM Data Transport does not supply a default.

server name

Specifies the name of the server.

share name

Specifies the share name. This name is network provider dependent.

For Microsoft Windows networks, this name is the name of the share.

Supported File Types

XCOM Data Transport supports standard Windows file types.

Additional Features

File Systems

The standard file systems are:

- File Allocation Table format (FAT)
- Windows File System format (NTFS)
- High-performance File System format (HPFS)

File Access

XCOM Data Transport accesses files locally or from any file server on the Microsoft Windows Network or the NetWare or Compatible Network, or any other network provider installed on the Windows system.

Security

For transfers to Windows systems that are running any release of XCOM Data Transport:

- Windows is a secured system. Windows requires a valid user ID and password (as defined when setting up a user account) to log in to or connect to a server. User IDs and passwords are case-sensitive. XCOM Data Transport uses the underlying security system to log in to the server as the user defined in the XCOM_USERID parameter. The authority to log on locally is required because there is no facility in XCOM Data Transport for Windows 2000, XP, 2003, 2008, or 7 to allow for another domain to be specified.
- When a transfer is sent from another system (such as z/OS or z/VSE), USERID and PASSWORD must be supplied. The following methods model sending from an XCOM Data Transport z/OS or z/VSE system to a local Windows drive.

Methods of handling Windows systems security from XCOM Data Transport:

Employing some security

All users employ the same user ID.

Set XCOM_USERID and XCOM_PASSWORD to a valid user ID and password that has local logon authority in the xcom.glb file on the Windows side. On the z/OS or z/VSE side, send a transfer with parameter USERID=' ' (blank between two single quotes). This transfer uses the user ID and password from the xcom.glb file.

Employing user level security

Users employ their own user ID and password.
Set XCOM_USERID and XCOM_PASSWORD to an INVALID user ID and password in the xcom.glb file on the Windows side. On the z/OS or z/VSE side, send a transfer with parameters USERID= and PASSWORD= with a valid Windows user ID that has local logon authority. These values cause XCOM Data Transport to use the supplied user ID and password. If a password is not supplied and xcom.glb is checked, the transfer fails due to the invalid ID and password in the xcom.glb file on the Windows side

NOTE
If either of these methods is to be successful, in the case where XCOM Data Transport for z/OS or z/VSE is sending to XCOM Data Transport for Windows 2000, XP, 2003, 2008, or 7 the XCOM Data Transport for z/OS or z/VSE default *tabl*e must have USEROVR=YES. USEROVR=YES is the default. This value allows the user ID in the MVS JCL to override the batch job ID. For more information about the USEROVR parameter, see the Reference section of this documentation.

Trusted Access

XCOM Data Transport for Windows supports Trusted Access. To use the Trusted Access feature when transferring to Windows platforms, note the following items:

- The USEROVR and USERPRO default table parameters for XCOM Data Transport for z/OS or z/VSE must be set to YES.
- Specify USERID=' ' (single quotes without any blanks in between the quotes) and no passwords in the parameters for the transfer.
- The user ID must be configured for Trusted Access on the Windows partner. For more information, see the XCOM Data Transport for Windows documentation.

Home Directory

A Windows user can have a default home directory that is assigned by the Windows administrator.

Destination Printer Information

When sending a report to a Windows system, specify the Destination parameter value or the Destination Printer field in the following form:

```
\\server name\printer queue name
```

XCOM Data Transport limits the length of this field to 21 characters. The actual name on the destination system can be longer.

Restrictions

Access to directories and files on drives that are formatted for NTFS can be controlled with the security features of Windows 2000, XP, 2003, 2008, or 7.

Access to all files on a Windows system can be controlled by the permissions set on a directory or file. The access rights of the user ID on the remote system determine the actions permitted for the transfer. Users cannot use a directory or file unless they have been granted the appropriate permissions.

z/OS

This article discusses the important aspects of the z/OS operating system.

Naming Conventions

Use the following format to name a z/OS file (data set):

```
[level1.level2.level3...level7].level8[(membername)]
```

The following table describes the parts of a z/OS file name:

level

Specifies the level of a file name. This value is required.
A file name can consist of multiple levels. The levels are separated by a period. Each level has the following characteristics:

- The level can be up to eight uppercase characters long.
- The level starts with either an alphabetic character or a national character ($ # @ + - : _).

The file name is limited to eight levels with a total of 44 characters, including the separating periods.
In most z/OS environments, a data set name is further restricted by security rules that are created by the installation. Contact the appropriate personnel within your organization for details. Typically, the high-level name (first-level name) must match your z/OS user ID or some other predefined index.

membername

Specifies the particular member in a z/OS partitioned data set (PDS). A PDS is a library containing members that are each a separate sequential file. The member name is appended to the end of the file name in parentheses. This value is required for z/OS partitioned data sets only.
Range: 1 to 8 alphanumeric or national characters

NOTE
Most sites catalog all files through a system catalog which means that the system can locate a file that you specify by name only. If a file is uncataloged, you must specify the volume and unit information for the device that holds the file.

Example:

The following examples show valid z/OS data set names:

```
SYS1.VTAMLST
C54684.UTILITY.CNTL(JOBCARD)
PROD.PAYROLL.SEPT90.TIMECARD.DATA
TESTDATA
A.$DDD.LOAD
```

Supported File Types

Sequential files are the most common forms of data that are transferred. Individual members of PDS files can also be sent as sequential files. Entire PDS libraries or selected members can be transferred between two z/OS systems or to other systems that are running XCOM Data Transport. PDSE and entire PDSE program libraries are supported in XCOM Data Transport. PDSE program libraries do not support wildcarding.

All three VSAM file types (KSDS, ESDS, and RRDS) can be transferred between z/OS systems. These VSAM files must be preallocated, or they can be sent to non-z/OS systems as sequential files.

UNIX System Services (USS) files are also supported where an entire file system is stored in a single z/OS data set.

Extended Attribute data sets and Extended Addressability Volumes (EAV) are supported.

ISAM, BDAM, IMS, FDR, and DFDSS data sets are not directly supported, but they can be put into a sequential format using native utilities before transmission.

DCB Information

The file characteristics for z/OS must be predefined when creating a file. Collectively, the following characteristics are known as Data Control Block (DCB) parameters:

- Block size
- Logical record length
- Record format
- Volume
- Unit

For more information regarding any of these fields, see the IBM documentation.

CICS Interface

Enable the CICS interface by indicating that you want to notify CICS in the appropriate "remote system notify" field in your version of XCOM Data Transport. Provide the VTAM APPLID of the CICS system in the related ID field. Your z/OS or CICS application development team can provide you with this information. Invoking this interface should start a CICS transaction program following the successful completion of a transfer when one has been provided at the host.

Store-and-Forward

Perform transfers between two nodes that are connected to an intermediate z/OS system by invoking the indirect transfer feature in your version of XCOM Data Transport. You are prompted for the final destination LU name and the transfer is sent in two stages. The first stage goes to the z/OS JES spool, where it waits for the final destination to be connected. Upon connection, the second stage goes to the final destination.

z/VM

This article discusses the important aspects of the z/VM operating system.

Naming Conventions

Use the following format to name z/VM files under the CMS operating system:

```
filename.filetype
```

The two parts can be a maximum of eight characters in length. They can consist of letters, numbers, and/or national characters ($, #, @, +, -, :, _). In general, lowercase letters are not allowed. In the XCOM Data Transport for z/VM parameters FILE and LFILE, the file name and file type are specified as one string with a period as a separator.

For minidisk specifications:

- CP OWNER is taken from the volume field, if present. Otherwise, the userid field is used.
- CP address is taken from the unit specification. The default is 191.
- You can have two files with the same file name and file type, but they cannot reside on the same minidisk.

Types of Files Supported

The XCOM Data Transport Service Virtual Machine runs IBM's Group Control System (GCS) operating system. Due to the limitations of this environment, XCOM Data Transport for z/VM only supports the CMS extended file system format. This covers CMS files on minidisks formatted with 512 KB, 1,024 KB, 2,048 KB, and 4,096 KB block sizes.

> **NOTE**
> It does *not* support the following: CMS Shared File System, minidisks formatted with 800-byte blocks, or tape I/O.

DCB Information

CMS file characteristics must be predetermined when creating a new file. You must specify the following parameters:

- Record format
  This can be fixed (F) or variable (V).
- Logical record length
  This is the number of characters in the longest line of the file.

Restriction

The maximum logical record lengths for different file types are as follows:

Disk file
32767 bytes

Job (RDR file)
80 bytes

Report (PRT file)
133 bytes

z/VSE

This article discusses the important aspects of the z/VSE operating system.

VSAM Naming Conventions

When accessing a file on a z/VSE system, the Remote file name field indicates the file ID as it would be specified on the DLBL (an indicator of whether the file is VSAM or SAM) and, optionally, additional information needed for locating the file.

Format for VSAM File Names

Use the following format to name a VSAM file:

```
file-id,V[,catalog-id]
```

The following list describes the parts of a VSAM file name:

file-id
Specifies the name given to the data set when it was defined using IDCAMS by including the following line in the JCL:

```
DEFINE CLUSTER (NAME   (file-id)...
```

V
Indicates that this is a VSAM file.

catalog-id
Optional.
The name of the user catalog that owns the VSAM data set as defined using IDCAMS by including the following line in the JCL:

```
DEFINE USERCATALOG (NAME (catalog-id)...
```

Leave this field blank if the data set is owned by the master catalog.

Format for SAM File Names

Use the following format to name a SAM file:

```
file-id,S,[unit],[location],[size],[override]
```

The following list describes the parts of a SAM file name:

file-id

The name that identifies this data set in the VTOC of the specific DASD volume. This is the file ID you specify on the DLBL JCL statement. Range: 1 to 44 characters.
Note: Do not enclose it in quotes.

S

Indicates that this is a SAM file.

unit

The physical device address as defined by the CUU parameter on the ASSGN JCL statement. It identifies the disk drive on which this file resides. This parameter can be omitted if the UNIT or VOL parameters are specified, or if a DASD manager is in use.

location

Optional for output files.
The starting location of the file on the disk, as defined on the EXTENT JCL statement. If a DASD manager is in use, specify a value of 1.

size

Optional for output files.
Indicates how much space this data set is to use, as defined on the EXTENT JCL statement. For CKD devices, this is the number of tracks. For FBA devices, this is the number of blocks.

override

Optional for output files.
The following override parameters can be specified. The equivalent DFLTAB option is included to describe each override parameter:

- DMYES to force DASDM=YES for this file
- DMNO to force DASDM=NO for this file
- DMEPIC to force DASDM=EPIC for this file

> **NOTE**
> If you are running with a DASD manager, the DASD manager's STRTTRK or Trigger value would be placed in the location field. DASD manager pools should be indicated by putting the pool name in the Volume parameter.

For EPIC/VSE users, you can omit the following:

- The location if you want EPIC to default to its STRTTRK value.
- The size if you want EPIC to default to its DEFEXT value.
- The Volume information if you want EPIC to default to its DEFPOL value.

For Dynam/T users who want to access Dynam catalog controlled files (included GDG data sets), no extent information should be entered. (No *cuu*, location, size, or override information and no Volume or Unit parameters for the files you are referencing.)

TAPE Naming Conventions

Use the following format to name a TAPE file:

```
file-id,T,[unit],[unit],[unit],[override]
```

The following list describes the parts of a TAPE file name:

file-id

Specifies the name that identifies this data set in the tape manager catalog or in the HDR1 label on the tape. This is the file ID you specify on the TLBL JCL statement. Range: 1 to 44 characters.

NOTE
When the file ID contains imbedded spaces or commas, it should be enclosed in quotes. IBM only supports a 17-character file ID in a tape header label. If you have a tape manager, 44-character tape file IDs can be supported. XCOM Data Transport does not validate your file ID, but takes whatever you put on the statement and passes it to IBM's OPEN routine or to your tape manager as you have entered it.

T

Indicates that this is a TAPE file.

NOTE
If you enter a transfer request from a platform that has not yet implemented the extended tape processing parameters or menu interface fields for controlling tape processing to an XCOM Data Transport z/VSE server, you must use the T option. You are restricted to standard label tape processing.

unit

The physical device address as defined by the CUU parameter on the ASSGN JCL statement. If you are using TAPEM=YES|EPIC, XCOM Data Transport ignores any units coded and the tape manager does the tape AVR and assignment. If you are not using the tape manager, the primary assignment is made to the first unit XCOM Data Transport finds. Other units found are assigned as temporary alternates.
This parameter can be omitted if you prefer to use the UNIT parameter to specify a unit or two units (primary and alternate). This parameter can be used in conjunction with the UNIT parameter to specify a primary unit and up to four alternate units that are to be assigned by XCOM Data Transport prior to open. Units specified on the statement containing the file ID are assigned before units specified on the UNIT parameter. The unit parameter is ignored because tape processing is only supported when you have a tape manager on your z/VSE system.

override

Optional for output files.
The following override parameters can be specified. The equivalent DFLTAB option is included to describe each override parameter:

- TMYES to force TAPEM=YES for this file
- TMNO to force TAPEM=NO for this file
- TMEPIC to force TAPEM=EPIC for this file

NOTE
The override applies only to the processing for the file whose data set name is on the statement that the override appears on. It is in effect for this transfer only.

VSAM Managed SAM Naming Conventions

Use the following format to name a VSAM managed SAM file:

```
file-id,M,prim#recs, sec#recs,catalog-id
```

The following list describes the parts of a VSAM managed SAM file name:

file-id

The name that identifies this data set, which is implicitly defined to VSAM at open time.
Range: 1 to 44 characters.

M

Indicates that this is a VSAM managed SAM file.
Note: If you enter a transfer request from a platform that has not yet implemented the extended tape processing parameters or menu interface fields for controlling tape processing to an XCOM Data Transport z/VSE server, you must use the T option. You are restricted to standard label tape processing.

prim#recs

Used for output files only. This indicates the number of blocks (of the size defined by the BLKSIZE parameter) for the primary data set allocation.

sec#recs

Used for output files only. This indicates the number of blocks for the secondary data set allocation. If no secondary allocation is coded, VSAM defaults to 20% of the primary allocation. Zero can be specified if you do not want any secondary allocation.

catalog-id

Optional for output files.
Defines the name of the user catalog that will own the data set. You can leave this field blank if the master catalog owns the data set.

> **NOTE**
> The use of VSAM managed SAM files requires IBM's IDCAMS program to be dynamically loaded in the partition. This requires an additional 130 KB partition GETVIS storage.

DTF Information

z/VSE file characteristics must be predetermined when creating the files. If sending to or receiving from a z/VSE system you must specify the following:

- The record format (RECFM), which can be either fixed (F), fixed blocked (FB), variable (V), or variable blocked (VB).
- The logical record length (LRECL) indicates the number of characters in the longest record in the file.
- The block size (BLKSIZE), which must be one of the following:
  - The LRECL for fixed files
  - A multiple of the LRECL for fixed blocked files
  - The LRECL +4 for variable files
  - The BLKSIZE +4 for variable blocked files

Types of Files Supported

IBM z/VSE supports VSAM (RRDS, KSDS, and ESDS) and SAM files.

Restrictions

The following restrictions apply to XCOM Data Transport for z/VSE:

- No FILEOPT=ADD for receiving z/VSE
  XCOM Data Transport for z/VSE does not support FILEOPT=ADD if the z/VSE is receiving the file.
- No Checkpoint/Restart for SAM
  XCOM Data Transport for z/VSE does not support checkpoint/restart for SAM jobs.

About SNA Logical Units

This section explains the various logical unit (LU) types and discusses independent logical units (ILUs) and other pertinent issues.

LU Connections

An LU is the addressable connection point into an SNA network with which an end-user can send and receive messages. An LU is a set of rules and responsibilities. LUs can be either dependent or independent, and each LU type is associated with a protocol (for example, LU 0, LU 2, LU 3, or LU 6.2). XCOM Data Transport only supports LU type 6.2.

The LU provides a connection into SNA for the end-user, which may be either an individual or a transaction program (for example, XCOM Data Transport). It allows end-users to communicate with each other and with other network addressable units (NAUs) in the network.

Components of Logical and Physical Network

An SNA network is divided into physical and logical components.

The physical network consists of the following:

- Actual processors called nodes
- Data links between the nodes

The logical network consists of a set of software components called NAUs that include the following:

- Logical units (LUs)
- Physical units (PUs)
- System services control points (SSCPs)

Sessions

A session is a logical connection between two NAUs. Although several types of sessions exist, the end-user is aware of only one type that is LU-to-LU. Sessions are established when one LU sends another LU an SNA request known as a BIND. Each session has its own procedure correlation identifier (PCID).

PCIDs

A PCID is an eight-byte field placed in the BIND, UNBIND, and other SNA requests to help an LU distinguish one session from another. It is required when you are running parallel sessions.

A PCID is also known as a session identifier (SID) in VTAM displays. For each session, VTAM prompts you to note the primary or secondary node and displays the Session ID (SID) in hex. This SID is the PCID. If a trace of the BIND is taken, the PCID vector is towards the end.

The following VTAM operator command lists all sessions generated for that LU:

```
D NET,ID=<luname>,E
```

LUs

IBM Strategic LUs

IBM classifies LUs into roughly seven different types. LU 6.2 is a subset of LU 6. LU 6.2 is the only LU type that is crucial to IBM long-term strategy. The products that support each of these LU types will continue to be supported in future and the 3270 data stream will also be preserved, but not in its current form. They will be moved on top of LU 6.2. None of the other LU types are strategic, for example, they are not considered in IBM's long-term plans.

LU Types

The following list shows all of the LU types that are in use today to differing degrees.

0

Denotes a flexible protocol, which eliminates standardization beyond layers of SNA. This was commonly used in the late 1970s (before the advent of LU 6.2).

1

Specifies the protocol used as early as the 1960s by remote job entry (RJE) devices such as the 3770 RJE terminal. Designed for use with printers and card readers, this protocol is most typically used in asymmetrical links where one node is a agent to the host.

2

Specifies the protocol for 3270 video display stations. It defines the data streams used by dumb terminals to communicate with the host.

3

Specifies a variant subset of 3270 protocol that was used to drive printers attached to 3274 cluster controllers. Today it is still used to support old hardware.

4

Specifies a protocol that was intended for use on word processors attached to a host network. You can still see it on old IBM word processors.

6.1

Specifies SNA's prototype protocol defined for program-to-program communication that was developed during the late 1970s. It was a first attempt to provide a standardized mechanism for communication between intelligent peer systems.

6.2

Alternatively referred to by the marketing title Advanced Program-to-Program Communications (APPC), this used to be called the Convergent LU, or the LU type around which the entire IBM product line would converge. LU 6.2 defines standard functions or verbs such as SEND, RECEIVE, and CONFIRM that simplify the work of making two different programs on two different kinds of system talk to each other. This is the only LU type supported by XCOM Data Transport.

7

Specifies the data stream of the 5250 video display stations commonly used with the IBM midrange systems.

ILUs

XCOM Data Transport supports IUs. An IU is a logical unit that can generate sessions independent of the host. An IU also meets the following criteria:

- It utilizes LU 6.2.
- It works on top of PU 2.1.
- It functions as a primary logical unit and therefore can send a BIND.
- It supports an extended BIND (one that contains a PCID) and works with the Network Control Program (NCP) PU 2.1 support.

Systems that currently support IUs include the following:

- i5/OS(AS/400)
- z/OS
- z/VSE
- MS Windows
- VM (all versions)
- UNIX or Linux
- Netware

LU 6.2 Independent Implementations

Only Type 6 logical units can be independent. All other LU types are dependent. However, not all LU 6.2 implementations are independent.

Not every LU 6.2/PU 2.1 implementation can work with independent LUs. There are some aspects of PU 2.1 that NCP requires with which not all PU 2.1 implementations will work correctly. This reflects the fact that not all midrange and PC SNA Gateway vendors had the latest NCP and VTAM for testing.

PU 2.1 support can be enhanced to work with ILUs without changes to XCOM Data Transport. NCP supports ILUs over SDLC, the most common configuration using ILUs. A local area network gateway attached through an SDLC link to a host can also use ILUs. NCP also supports ILUs over a token ring through the TIC.

Direct Sessions with a Dependent Logical Unit

An independent logical unit can have an LU 6.2 session with a dependent LU. This session allows for direct sessions from an i5/OS(AS/400) to an OpenVMS over the SNA background network, even though the VTAM is PU Type 2.0. In this environment, the VTAM LOGAPPL parameter and the VTAM VARY NET LOGON command does not work.

> **NOTE**
> The ILU must initiate the session; it must send to the BIND.

PU Types

When using ILUs with VTAM and NetView displays, VTAM shows the PU type in its status display (PU Type 2 or PU Type 2.1). All PUs originally appears as PU 2.0. Once they become active, they display as PU 2.1.

About XCOMD XCOM Scheduler service Service

The XCOMD XCOM Scheduler service runs as a background process to control file transfers and manage XCOM Data Transport resources. The XCOMD XCOM Scheduler service performs the following activities:

- Schedules and synchronizes transfer requests
- Controls shared memory for transfers
- Establishes the default parameter values by reading the parameter file XCOM.GLB
- Controls the automatic restart of locally initiated transfers
- Writes queue information out to disk periodically
- Deletes aged entries from the queue
- Notifies a local user by executing the XCOMNTFY script when LOCAL_NOTIFY is required
- Communicates with active or pending transfers to terminate a transfer

Start XCOMD XCOM Scheduler service Service

Use the following command to start XCOMD XCOM Scheduler service:

```
$XCOM_HOME/sbin/xcomd
```

Using

Provides instructions for using XCOM Data Transport for UNIX/Linux.

Using XCOM Data Transport GUI

This section introduces XCOM Data Transport GUI. Read this section before installing or configuring XCOM Data Transport.

The XCOM Data Transport GUI is a desktop-based interface. This interface allows you to create, load, edit, delete, save, and submit data transfers from your local computer. The GUI supports the legacy format of transfer records, the configuration text files (.CNF), for reading. However, it saves the transfer records in XML file format *only.*

The XCOM Data Transport GUI supports the following functions:

- Scheduling transfers
- Getting history records
- Log browsing
- Administering global parameters
- Setting up systems for trusted transfers

Features

The XCOM Data Transport GUI facilitates data transfer between all platforms that XCOM Data Transportsupports. The following list provides a summary of the XCOM Data Transport GUI features:

- Building transfer records
- Saving transfer records in the configuration file, or submitting the records for scheduling, or both
- Scheduling data transfer among XCOM Data Transport partner systems
- Retrieving history records from XCOM Data Transport partner systems
- Processing history records and generating reports
- Saving configuration files and storing them on the server for later use
- Loading an already saved configuration file and processing it
- Suspending active transfers and resuming suspended transfers
- Browsing xcom.log with a log browser
- Performing Trusted Transfer configuration (by user with administrative privileges)
- Updating the Global Parameters (by user with administrative privileges)

Audience

The XCOM Data Transport GUI help is written for the following types of users:

Normal User

A user without administrator authority. This user can view and use all the tabs except Trusted Transfer and Global Parameters.

Admin

A user with administrator authority. This user can view and use all the tabs and features.

Login Process

To invoke XCOM Data Transport

1. Set properly the X-windows.
2. Go to the directory /opt/CA/XCOM/bin.
3. Issue the command sh StandaloneUI.sh.

Before you launch the XCOM Data Transport GUI, confirm that the Scheduler service is running. If the Scheduler service is not running, an error message appears.

The first page that appears is the Home page. Home is the first tab in a set of tabs with different functions of the interface. You can also navigate to each function through the left pane of the Home page. To navigate, you can click either the tab or the link.

Each page of the interface displays three additional links:

- Log Browser at the top right
- Help at the top right
- About at the bottom right

User Considerations

As a normal user, consider the following points:

- Before you execute a file transfer request, properly set up your directories and path variables.
- XCOM Data Transport accesses files locally or from any file server on the network.
- The Universal Naming Convention (UNC) is supported.
- Standard path names are supported.
- A path name that contains a blank or special character, has double quotes (“”) around it.
- Standard file types are supported (FAT, NTFS, HPFS).

Administrator Considerations

As an administrator, consider the following points:

- Insure that the XCOM_HOME environment variable is set to the directory where XCOM Data Transportis installed.
- Your PATH variable can include the directory where XCOM Data Transport is installed. The default directories for various Operating Systems are:
  - For Windows: C:\Program Files (x86)\CA\XCOM (32-Bit) and C:\Program Files\CA\XCOM (64-Bit)
  - For UNIX/Linux: /opt/CA/XCOM
- XCOM Data Transport uses C:\Program Files (x86)\CA\XCOM or /opt/CA/XCOM as default repository for the subdirectories for log and configuration files.

Default Directories and Drives

You can have installed XCOM Data Transport under a different path or on a different drive. If the default directories change during your installation, the default *values in the XCOM.GLB file reflect this change.*

Schedule Transfers

From the Schedule Transfer page, you can build transfer records and can submit them for scheduling.

The XCOM Data Transport GUI allows you to build transfer records and schedule them for processing on an assigned date. Typically, a transfer record holds several parameters that are related to the following items:

- The remote recipient system properties
- The intended date for transfer
- The data that is sent

You can use the schedule transfer page to do the following items:

- Build transfer records and save as configuration files in XML format
- Submit transfer records for scheduling
- Edit configuration files
- Load configuration files (supports XML and CNF formats) and process them

Actions on the Schedule Transfer Page

On the Schedule Transfer page, use the following user elements to perform corresponding tasks:

New

Use the New button to add and save transfer records. A dialog box is displayed:

- If you choose Yes, then a save window appears for the listed transfer records.
- If you choose No, then a blank row is added on the Schedule Transfer page.

Load

Use the Load button to load and display transfer records from a previously saved configuration file. The records appear in either an xml or cnf file format.

Add

Use the Add link to add a new, uninitialized transfer record at the end of the table of transfer records.
You can then edit this transfer record to specify required transfer parameters and then submit or save the record.

Edit

Use the Edit button to display the Edit Transfer screen, for a particular transfer record row. You can submit the transfer record or can save it in an xml file after editing.

Delete

Use this link to delete the selected transfer records from the list of transfer records.

Copy

Use this link to copy the selected transfer records and create new transfer records. Each new transfer record is added to the list immediately following the record from which it was copied.

SelectAll

Use the SelectAll link to select all listed transfer records.

> **NOTE**
> If ShowAll has not been selected, then only those transfer records on the current page of transfer records are selected.

ClearAll

Use the ClearAll link to deselect all selected transfer records.

Save,

SaveAll

- Use the Save link to save selected transfer records in a TRANSFERCONTAINER XML file.
- Use the SaveAll button to save all listed transfer records in a TRANSFERCONTAINER XML file.

When you click Save or SaveAll without selecting any records, an error message appears for you to specify the file name.

Submit, SubmitAll

- Use the Submit link to submit selected transfer records for scheduling.
- Use the SubmitAll button to submit all listed transfer records without needing to select them.

Click the Submit link or the SubmitAll button and the schedule status of each submitted transfer record gets refreshed and shown in the Schedule Status column.
Each entry is checked for completeness, builds a transfer scheduling request for the transfer records, and sends them to XCOM Data Transport for immediate scheduling.

NOTE

If a transfer record cannot be processed as entered, an error message is issued.

Rows Per Page

Limit the number of transfer items to be displayed per page. Use the text box on left side of this button to specify the number of records to be displayed per page. By default, 20 rows are displayed per page.

Show All

Display all transfer items in the same page.

Browse

Limit the number of transfer items to be displayed per page. Use the text box on left side of this button to specify the number of records to be displayed per page. By default, 20 rows are displayed per page.

To perform each of these actions, complete the following steps:

1. Click the appropriate button or link that is provided on the Schedule Transfer page.
2. Use the select checkboxes to the left of each transfer record and then click the appropriate link to perform the action required.

Edit a Transfer Record

This article describes how to use the XCOM Data Transport GUI to edit transfer records.

Edit a Transfer Record

Follow these instructions to edit a transfer record.

1. On the Schedule Transfer page, click the Edit button in the Actions column for the transfer edited record.
   The Edit Transfer Record page appears, showing the parameter values for the selected transfer.
   For a new added transfer record, the default values of the parameters are shown.
2. In the Select Action pane, select *one* of the following actions:
   – Send File
   – Send Job
   – Send Report
   – Receive File
3. Update the details as required for the items appearing in the following panes:
   – Local System Parameters for Server
   – Options
   – Remote System Identification and Parameters
   – Misc Options
   – XCOM Transfer Control (XTC) Parameters
4. Click one of the following buttons:

Update

Validates and saves the changed parameter values and returns you to the Schedule Transfer page.

Submit

Validates and saves the changed parameter values and submits the changed transfer record. A confirmation window appears on the Edit Transfer Window page.

If you click Cancel, you return to the Schedule Transfer page without saving any changed parameter values.

Local System Parameters

This section describes how to edit the parameters that appear on the Local System Parameters for the Server pane of the Edit Transfer Record page.

1. Click the arrow in the Local System Parameters for the Server pane to expand the pane.
   The pane expands to display the items in the following fields and sections:
   - File Name
   - File Access (Windows only)
   - Gateway GUID
   - Gateway Destination Path
   - Notify
   - File Storage Encryption
   - Unicode

   The values of the following fields determine what fields are displayed in this pane.
2. For the **File Name** field, click the Browse button and select the file that you want for the transfer record.
   Range: 0 to 256 characters
   Default: None
3. In the **Gateway GUID** field, enter a value for the GATEWAYGUID attribute of the local file. GATEWAYGUID defines the unique instance (GUID) for a file that resides in the XCOM Data Transport Gateway. GATEWAYGUID is an XML attribute in the TRANSFERITEM and HISTORYITEM XML elements.
   Range: 0 to 36 characters
   Default: Null
4. In the **Gateway Destination Path** field, enter a value for the GATEWAYDPATH attribute. GATEWAYDPATH is an XML attribute in the TRANSFERITEM and HISTORYITEM XML elements. GATEWAYDPATH appears when you select Receive File in the Select Action pane, and you set the Gateway Version as R120 or above in Global Parameters. GATEWAYDPATH specifies the destination path that XCOM Data Transport Gateway uses when the local file is a XCOM Data Transport Gateway file. The Gateway makes the destination path available as a symbolic parameter &GWDPATH for onward delivery.
   Range: 0 to 254 characters
   Default: None
5. Complete the following fields in the **Notify** section:

User

Identifies the user on the local system who receives a transfer completion notification.
Range: 0 to 12 characters
Default: Null

Level

Specifies the local notification level for transfers that are initiated from the local server:

All

Notifies after transfer completion.

Warn

Notifies only when the transfer received a warning or an error.

Error

Notifies only when the transfer received an error.

Default: ALL

Method

Specifies the notification method for completed transfers:

ALL

Displays a message ion the local system console.

WRITE

Displays a message on the screen.

MAIL

Sends a mail message to the user.

NONE

Does not send any notifications.

Default: NONE

6. Complete the following fields in the **Encryption At Rest** section:

Cipher

Specifies the encryption algorithm that is used for encrypting data on the local server. Select a value from the drop-down list.
Default: None

Cipher Key

Specifies the key that is used to encrypt or decrypt data.
Range: 0 to 128 characters
Default: None

Hash

Specifies the hash algorithm to calculate the digest on the local server. Select a value from the drop-down list.
Range: SHA1, MD5, MD
Default: None

Digest

Specifies a digest value that determines the integrity of the data.
Range: 0 to 128 characters
Default: None

7. Complete the following fields in the **Unicode** section. The Unicode fields appear only for the Encoding value of UTF8 or UTF16 that is specified in the Options section.

Character-set

Specifies the local character set that the XCOM Data Transport uses for Unicode transfers (Encoding=UTF8 or Encoding=UTF16). Use the following format:

```
CCSID#nnnnn
```

The *nnnnnn* specifies the CCSID number that corresponds to the character set. Valid values are from 1 to 65535. Alternatively, the character set can be specified as an IANA character set name, or as an (ICU) acceptable alias name.
Range: 0 to 60 characters
Default: The DEFAULT_CHARSET global parameter determines the default.

Record Delimiter

Specifies an optional encoding for which the specified Local Character-set is based. The encoding can be ASCII or EBCDIC. If the encoding is specified, it must be the first option in the list.
This field also specifies a colon-separated list of record delimiters. These record delimiters let you detect and mark the end of a record.
Use the record delimiters only for UNICODE transfers (Encoding=UTF8 or Encoding=UTF16).
Range: 0 to 60 characters
Default: The DEFAULT_DELIM global parameter determines the default.

Options

This section describes how to edit the parameters on the Options pane of the Edit Transfer Record page.

1. Click the arrow in the Options pane to expand the pane.
   The following options specify how to handle the file that is sent to the remote system or received on the local system.
2. Select a value for **File Option**:

Create

Creates (allocates) a new file on the remote system for send file transfers, or on the local system for receive file transfers. If you are transferring a partitioned data set, specify Create only if the PDS itself is being created. If a new member is being sent to an existing PDS, specify Replace. If you do not override the default value of CREATE and the file exists on the target system, the transfer terminates with an error.

Replace

Replaces the contents of a file on the remote system with the data being transferred.

Append

Adds the records being transferred to the end of an existing file on the remote system.

Delete and Recreate

Sets the File Option to Create and sets the CREATEDELETE parameter to YES. This option is valid for z/OS only.

For more information on the CREATEDELETE parameter, see CREATEDELETE.

Default: Create

3. Select the required value for the **Encoding** field from the drop-down list. This field indicates the type of data being transferred. The system from which the data are retrieved is responsible for performing any necessary conversion.

BINARY

Indicates binary data.

EBCDIC

Indicates that the data is in EBCDIC code.

ASCII

Indicates that the data is in ASCII code.

UTF8

Indicates that a Unicode file which is based on the UTF8 encoding system is being transferred. When this option is set, LOCAL_CHARSET/ REMOTE_CHARSET parameters are used to identify the character encoding of the input file. When this option is set, it also identifies the required encoding for the output file.

UTF16

Indicates that a Unicode file which is based on the UTF16 encoding system is being transferred. When this option is set, LOCAL_CHARSET/ REMOTE_CHARSET parameters are used to identify the character encoding of the input file. When this option is set, it also identifies the required encoding for the output file.

The encoding options can also include the following carriage flag values.

31-K Pack

Indicates a text file with record packing and uses a 31-K pack buffer.

VLR

Indicates a binary file of variable-length records with a field of 4 bytes preceding each record. This value applies to locally initiated transfers only.

Default: ASCII

4. Enter a value for the **Codetable** field. This field specifies the translation table that the remote partner uses for data conversion. A one-character to three-character prefix to the file names specifies, atoe.tab, and etoa.tab that contain the external ASCII-to-EBCDIC and EBCDIC-to-ASCII custom character conversion tables on the XCOM Data Transport for Windows and the XCOM Data Transport for UNIX platforms.
   Range: One to three alphanumeric characters
   Default: None

5. Enter a value for the **Max Record Length** field. This field specifies the maximum logical record length for the transfer records.
   Range: 0 to 32767
   Default: 1024
6. Select the required value for the **Truncate** field from the drop-down list. This field specifies how to handle records exceeding the maximum logical record length.

Yes

Truncates the records to the maximum record length. The truncated data is lost. Truncation is not applicable for binary data, and it is not valid for UTF8 and UTF16 encodings.

No

Terminates the file transfer.

Default: No

7. Select the required value for the **Compress** field from the drop-down list. This field specifies whether to compress the data being transferred and decompress it on the remote system. Compressing the data decreases the transmission time on lower-speed lines.

YES

Provides Run-Length Encoding (RLE) for blanks and binary zeros only.

NO

Performs no data compression

RLE

Provides complete Run-Length Encoding for all characters.

COMPACT

Provides full RLE plus a byte compaction scheme that is suitable for uppercase English text.

COMPACTL

Provides the same compression as COMPACT, but the compaction scheme is most beneficial for lowercase English text.

LZSMALL

Provides Lempel-Ziv 77 compression with a small memory allocation scheme.

LZMEDIUM

Provides Lempel-Ziv 77 compression with a medium memory allocation scheme.

LZLARGE

Provides Lempel-Ziv 77 compression with a large memory allocation scheme.

LZRW3

Uses a general-purpose algorithm that runs fast and gives reasonable compression.

ZLIB(

*n***)**
Provides greater compression than LZRW3 but less than LZSMALL, LZMEDIUM, and LZLARGE. The *n* value can be 0 through 3.

> **NOTE**
> Not all compression modes are supported on all platforms.

Default: YES

8. Enter a value for the **Checkpoint Count** field. this field specifies the interval at which XCOM Data Transport takes a checkpoint. Checkpoints can be used to restart a suspended or failed file transfer. The length of the checkpoint interval is measured in terms of a number of blocks.

0

Does not take any checkpoints.

1 to 9999

Specifies the number of blocks that form a checkpoint interval. A checkpoint is taken whenever the specified number of blocks has been transferred.

If the record packing is not used, one record = 1 block.

Each time a checkpoint is taken, the output buffers on the receiving system are written to the disk. Making the checkpoint interval too short slows down file transfers. Making the interval too long increases the risk of starting a long running transfer from the beginning, if it encounters an error. We recommend setting the Checkpoint Count to at least 1000. On Token Ring, Ethernet, and other high-speed networks, the Checkpoint Count could be set to the highest allowable value or turned off.

If the receiving system is z/OS or z/VSE, the Checkpoint Count must be a multiple of the blocking factor. For example, if the DCB attributes are RECFM=FB LRECL=80 BLKSIZE=8000, the Checkpoint Count must be a multiple of 100.

Default: 1000

9. Enter a value for the **Number of Retries**. This field specifies the number of times to retry a failed attempt to contact the local XCOM Data Transport server.
   Range: 0 to 255
   Default: 1
10. Enter a value for the **Retry Interval** field. This field specifies the time (in seconds) to wait before retrying a failed attempt to contact the local XCOM Data Transport server. If this parameter is omitted, a system default value of 1 second is used.
    Range: 0 to 99999
    Default: 1
11. Enter descriptive text in the **Transfer Comment** field.
    Range: 0 to 256 characters
    Default: None
12. Select the required option for the **Remove Trailing Blanks** field from the drop-down list. This field specifies whether to remove the blank spaces at the end of the transfer records. This field displays only when the Receive File action has been selected.

    Default: No
13. Specify the **Gateway Protocol**. This field identifies the protocol that the Gateway uses to access the external server. The value must be ‘FTP’ or ‘SFTP’ or ‘FTPS’ for Transfers between associated Gateway and External Server.
    In all other cases, the value is XCOM by default. This field is used only when you are extracting a file from the Gateway environment to an external server or when you are inserting a file from an external server to the Gateway environment. You can only use this field when the associated Gateway version is Release 11.6 or later. This field is not valid for SEND JOB and SEND REPORT actions.

FTP

Is used for transfers between Gateway and the registered FTP Server.

SFTP

Is used for transfers between Gateway and the registered SFTP Server.

FTPS

is used for transfers between Gateway and the registered FTPS Server.

XCOM

Is used for transfers between Gateway and the XCOM server.

Default: XCOM

14. Specify the **PDS Compression**. If your administrator has enabled the programmatic PDS compression feature in a XCOM Data Transport region, the COMPRESS_PDS option controls the output when the PDS data sets get compressed. COMPRESS_PDS applies only to PDS data sets that are opened for output as the target of a XCOM Data Transport transfer. This field appears only when the remote system type is set to z/OS.

NONE

Does not compress the output PDS data set.

BEFORE

Compresses the PDS before the transfer begins.

AFTER

Compresses the PDS after the transfer completes.

BOTH

Compresses the PDS both before and after the transfer.

Default: NONE

Remote System Identification and Parameters

This section describes how to edit parameters that are shown on the Remote System Identification and Parameters pane of the Edit Transfer Record page.

1. Click the arrow in the Remote System Identification and Parameters pane to expand the pane.
   The pane expands to display the items in the following fields and sections:
   - System Identification
   - Indirect Transfer
   - Credential (for the remote user)
   - File Name
   - SMS
   - Gateway GUID
   - Gateway Destination Path
   - UMASK
   - DCB
   - Report
   - Notify
   - File Storage Encryption
   - Unicode

   The values of the following fields determine what fields are displayed in this pane.
2. Complete the following fields in the **Remote System Identification** section:
   a. Select the required **System Type** from the drop-down list.
   Range: iSeries (AS400), STRATUS, TANDEM, Unix, Windows, z/OS, z/VM, z/VSE, and others.
   Default: Unix
   b. Select **IP Address** or **LU Name** from the drop-down list and enter the actual remote system IP address or LU name to match the selection.
   Default: IP Address
   c. Select YES or NO from the **SSL** drop-down list.
   Default: NO
   d. Enter a value in the **Port** field.
   Range: 0 to 65535
   Default: 8044
3. Check the **Indirect Transfer** field if you are performing store-and-forward transfers to z/OS, z/VSE, or z/VM systems. When this field has been checked, the **Indirect Destination Member** field appears so that a destination member can be specified. This field can be one to eight characters.
4. Complete the following fields in the **Credential** section:

User ID

Specifies the user ID under whose set of resource access privileges the transfer is to execute on the remote system.
Range: 0 to 12 characters

Default: None

Password

Identifies the password that is associated with the remote user ID.
Range: 0 to 31 characters
Default: None

TRUSTED

Indicates whether the transfer is a trusted transfer. This field is applicable only to those remote systems that support trusted transfers.
Range: YES or NO
Default: NO

5. In the **File Name** section, enter values for the following fields:

File Name

Indicates the file to which the transferred data is being written on the remote system.
Range: 0 to 256 characters
Default: None

Unit Name

Specifies the unit on which a data set is created on the remote system. This field is displayed only when the remote system type field is set to z/OS, z/VSE, or, z/VM.
Range: 0 to 8 characters
Default: None

Unit Count

Specifies the number of units that are allocated for the tape data set on the remote system. This field is displayed only when the field TAPE=YES is specified in the DCB section.
Range: 1 to 20
Default: None

Volume Count

Specifies the maximum number of volumes that is used in processing a multivolume output tape data set on the remote system. This field is displayed only when the field TAPE=YES is specified in the DCB section.
Range: 1 to 225
Default: None

Volume Sequence

Specifies the sequence number of the first volume of a multivolume remote data set to be used. This field is displayed only when the field TAPE=YES is specified in the DCB section.
Range: 1 to 225
Default: None

Volume Serial

Specifies the volume on which a data set that is created on the remote system. This field is displayed only when the remote system type field is set to z/OS, z/VSE, or, z/VM.
Range: 0 to 6 characters
Default: None

6. In the **SMS** section, enter values for the following fields. The SMS parameters are displayed only when the remote system type is set to z/OS and the action is Send File or Receive File.

DATACLAS

Specifies the data class to use when allocating a new SMS-managed data set.
Range: 0 to 8 characters
Default: None

MGMTCLAS

Specifies the management class to use when allocating a new SMS-managed data set.

Range: 0 to 8 characters
Default: None

STORCLAS

Specifies the storage class to use when allocating a new SMS-managed data set.
Range: 0 to 8 characters
Default: None

7. In the **Gateway GUID** field, enter a value for the GATEWAYGUID attribute of the remote file. GATEWAYGUID is an XML attribute in the TRANSFERITEM and HISTORYITEM XML elements. This attribute defines the unique instance (GUID) for a file that resides in the XCOM Data Transport Gateway.
   Range: 0 to 36 characters
   Default: Null
   For more information about GATEWAYGUID, see the *XCOM Data Transport Gateway Product Guide*.
8. In the **Gateway Destination Path** field, enter a value for the GATEWAYDPATH attribute. GATEWAYDPATH is an XML attribute in the TRANSFERITEM and HISTORYITEM XML elements. This field is displayed only when the Send File action is selected in Select Action pane and Gateway Version in Global Parameters is set to R120 or above. GATEWAYDPATH specifies the destination path that the XCOM Data Transport Gateway uses when the remote file is a XCOM Data Transport Gateway file. The Gateway makes the destination path available as symbolic parameter &GWDPATH for onward delivery.
   Range: 0 to 254 characters
   Default: None
9. Enter a value in the **Umask** field. This field is displayed only if the remote system type is set to Unix or z/OS and the action is Send File. Umask defines what file permissions that are set on a newly created file on a UNIX system.
   Range: 000 to 777
   Default: 022
10. Select a value for **DSNTYPE** from the drop-down list. This field specifies the data set definition. This field is displayed only when the remote system type is set to z/OS or z/VM and the action is Send.

LIBRARY

Defines a PDSE.

PDS

Defines a partitioned data set.

BASIC

Defines a legacy sequential data set.

LARGE

Defines a large format sequential data set.

EXTREQ

Defines an extended format data set.

EXTPREF

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

<blank>

Defines a partitioned or sequential data set based on the data set characteristics that are entered.

Default: None

11. Complete the fields in the **DCB** section. If the remote system type is set to Tandem, z/OS, z/VM, or z/VSE, only the relevant fields in the DCB section are displayed.
12. Complete the following fields in the **Report** section when a Send Report transfer has been selected:
    a. Enter a value for **Class**. This field specifies the print class that is assigned to a report that is transferred to a remote system.
       Range: One character
       Default: None

b. Enter a value for **Chars**. This field indicates the character set that JES uses when the report is sent to a remote system.
Range: 1 to 4 characters
Default: None

c. Select a value for **Control** from the drop-down list. This field indicates the type of printer carriage-control codes, if any, that are included in the report file.
Range: ASA, IBM, BYPASSASA, OTHER
Default: OTHER

d. Enter a value for **Copies**. This field specifies the number of report copies that are sent.
Range: 1 to 999
Default: 1

e. Enter a value for **Destination**. This field identifies the printer or other device on the remote system where the report is sent.
Range: 0 to 21 characters for Version 2 transfers
Default: None

f. Select a value for **Disposition** from the drop-down list. This field indicates what the remote system does with the report file after the report has been printed.
Range: Delete, Keep, or, Hold
Default: Delete

g. Enter a value for **FCB**. This field identifies the FCB JCL parameter when the report file is sent to a remote system, defining print density, lines per page, and so on.
Range: 0 to 4 characters
Default: None

h. Enter a value for **Form**. This field specifies the type of form that is used to print the report.
Range: 0 to 10 characters
Default: None

i. Select a value for **Hold** from the drop-down list. This field indicates whether a transferred report file is placed on HOLD on the remote system or it is printed immediately.
Range: YES or NO
Default: NO

j. Enter a value for **Name**. The value in this field is used on remote systems in the following ways.

AS/400

Uses the REPORT_TITLE as the printer file name.

z/OS

Uses the REPORT_TITLE to interpret a non-blank value in this field as specifying the generation of a separator (banner) page for this value.

VAX/VMS

Uses the REPORT_TITLE to print with the report.

UNIX

Uses the REPORT_TITLE to let XCOM Data Transport pass this field to the LP spooler as a title field.

Other systems

Uses the REPORT_TITLE as a descriptive comment only and does not print it as part of the report.
Range: 0 to 21 alphanumeric or blank characters
Default: None

k. Select a value for **Spool** from the drop-down list. This field indicates whether to spool the report to disk or print it immediately.
Range: YES or NO
Default: YES

l. Enter a value for **Writer**. This field specifies the name of the external writer that is to process the report on the remote system.
Range: 0 to 8 characters
Default: None

13. Complete the following fields in the **Notify** section:

User

Identifies the user on the remote system to receive a notification of the completion of the transfer.

Level

Specifies the remote notification level for transfers that are initiated from the local server:

All

Notifies after transfer completion.

Warn

Notifies only when the transfer received a warning or an error.

Error

Notifies only when the transfer received an error.

Default: All

Method

Specifies the notification method for completed transfers:

LOG

Causes the XCOM Data Transport transfer log to be written.

TSO

Identifies the TSO user who is sent a broadcast message when the file transfer completes.

CICS

Causes XCOM Data Transport to begin an LU 6.2 process to CICS when the file transfer concludes.

LU

Defines the local system LU name and associated VTAM logical unit name.

ROSCOE

Sends notification messages to users of the Roscoe timesharing product when a file transfer begins, ends, or, fails.

NONE

Does not send any notifications.

Default: LOG

14. Complete the following fields in the **File Storage Encryption** section. These fields are displayed only for those remote systems that support this feature.

Cipher

Specifies the encryption algorithm that is used for encrypting data on the local server. Select a value from the drop-down list.
Default: None

Cipher Key

Specifies the key that is used to encrypt or decrypt data.
Range: 0 to 128 characters
Default: None

Hash

Specifies the hash algorithm to calculate the digest on the local server. Select a value from the drop-down list.
Range: SHA1, MD5, MD
Default: None

Digest

Specifies a digest value that determines the integrity of the data.
Range: 0 to 128 characters
Default: None

15. Complete the following fields in the **Unicode** section. These fields are displayed only for Encoding value of UTF8 or UTF16 in the Options section.

Character-set

Specifies the remote character set that XCOM Data Transport uses for Unicode transfers (Encoding=UTF8 or Encoding=UTF16). Use the following format:

```
CCSID#nnnnnn
```

The *nnnnnn* specifies the CCSID number that corresponds to the character set. Valid values are from 1 to 65535. Alternatively, the character set can be specified as an IANA character set name, or as an (ICU) acceptable alias name.
Range: 0 to 60 characters
Default: The DEFAULT_CHARSET global parameter determines the default.

Record Delimiter

Specifies an optional encoding for which the specified Remote Character-set is based. The encoding can be ASCII or EBCDIC. If the encoding is specified, it must be the first option in the list.
This field also specifies a colon-separated list of record delimiters that are used to mark and detect the end of a record.
This field is used only for UNICODE transfers (Encoding=UTF8 or Encoding=UTF16).
Range: 0 to 60 characters
Default: The DEFAULT_DELIM global parameter determines the default.

Miscellaneous Options

This section describes how to edit the parameters on the Misc Options pane of the Edit Transfer Record page.

1. Click the arrow in the Misc Options pane to expand the pane.
2. Complete the fields in the Misc Options pane.
   a. Enter a value for **Age**. This field denotes the queue-purging interval in number of days for the transfer requests initiated locally.
   Range: 1 to 999
   Default: 10
   b. Enter a value for **Transfer User Data**. This field is an open field where a user can specify any text that is associated with the transfer.
   Range: 1 to 10 characters
   Default: None
   c. Enter a value for **Log File Name**. This field specifies the name of the file where XCOM Data Transport logs activity.
   Range: 0 to 256 characters
   Default:
   - %XCOM_HOME%\xcom.log (Windows)
   - $XCOM_HOME/xcom.log (UNIX and Linux)

   d. Enter a value for **System User Data**. This field is an open field where you can specify any text that is associated with the transfer.
   Range: 1 to 10 characters
   Default: None
   e. Select a value for START from the calendar. This field specifies the start date and time for the scheduled transfer.
   Range: *mm/dd/yy hh:mm:ss*
   Default: Current date and time
   f. Enter a value for **SSLConfig FileName**. This field specifies the configssl.cnf file path and file name.

Range: 0 to 256 characters
Default:

- %XCOM_HOME%\configssl.cnf (Windows)
- $XCOM_HOME/config/configssl.cnf (UNIX and Linux)

g. Enter a value for **Transfer ID**. This value identifies the file transfer request.
Range: 1 to 10 characters
Default: None

h. Select a value for **XCOM Header Version** from the drop-down list. This value indicates the version of the XCOM Data Transport protocol to use for a transfer. When you are using TCP/IP, only a value of 2 is valid. Currently, only version 2 is supported.
Default: 2

i. Enter descriptive text in the **Name** field.
Range: 0 to 256 characters
Default: None

j. Enter a value for **SPRTY**. This field indicates the scheduling priority of the transfer.
Range: 1 to 255
Default: 16

k. Select a value for **Write EOF** from the drop-down list. This field specifies whether to write the end of file character (CTRL + Z) at the end of the file.
Range: YES or NO
Default: NO

l. Enter a value for **XTrace**. This field indicates the level of desired execution tracing.
Range: 0 to 10
Default: 0

m. Enter a value for **Character-set Input Error**. This field is displayed only for an Encoding value of UTF8 or UTF16 in the Options section. This field identifies the appropriate action when the input file contains data that is not consistent with the specified input character set.
Range: FAIL, REPLACE, SKIP, or REPLACE#*nnnnnnn*, where *nnnnnnn* is a value from 0 to 1114111
Default: FAIL

n. Enter a value for **Character-set Convert Error**. This field is displayed only for an Encoding value of UTF8 or UTF16 in the Options section. This field identifies the appropriate action when the input file contains characters that cannot be converted because they are not included within the output character sets character repertoire.
Range: FAIL, REPLACE, SKIP, or REPLACE#*nnnnnnn*, where *nnnnnnn* is a value from 0 to 1114111
Default: FAIL

o. Enter a value for **Local Cipher List**. This field specifies the requested list of ciphers that are used to encrypt the password fields for locally initiated transfers or meta-transfers.
Range: 0 to 256
Default: COMPAT

XCOM Transfer Control (XTC) Parameters

This section describes how to edit the parameters shown on the XCOM Transfer Control (XTC) Parameters pane of the Edit Transfer Record page.

1. Click the arrow in the XCOM Transfer Control (XTC) Parameters pane to expand the pane.
2. Complete the fields in the XCOM Transfer Control (XTC) Parameters pane.

Network

Specifies the name of the XTC network (XTCNET) that is running this transfer.
Range: 1 to 8 alphanumeric characters
Default: None

Job

Specifies the name of a transfer request (XTCJOB) in a group of interrelated transfer requests named through the Network (XTCNET) parameter.
Range: 1 to 8 alphanumeric characters
Default: None

Hold

Specifies whether to prevent a schedule transfer from starting until it is explicitly released.

Yes

Does not initiate the transfer until it is released.

No

Does not hold the transfer.
Default: No

Hold Count

Specifies a value that controls the holding/releasing of a transfer request. The transfer is released when the value of the parameter reaches 0.
Limits: 0 to 255.
Default: 0

3. Complete the fields in the **Dependencies** panel.

Submit, Update, or Cancel the Edit Transfer Record

When you have finished editing a transfer record, you have the following options on the Schedule Transfer page:

- Click Submit
  The transfer record that you created is sent to XCOM Data Transport for scheduling.
- Click the Update button.
  The transfer record that you created is updated with any pending changes made.
- Click the Cancel button
  The page refreshes to display the Schedule Transfer page. Any pending changes that were made to the transfer record are lost.

Edit a Saved Configuration File

Follow these steps to edit a saved configuration file.

1. Click Schedule Transfer on the Home page.
2. Click the Load button on the Schedule Transfer page.
   A File Open dialog box opens, through which you can select and load the configuration file.
3. Click Edit action on any row.
   The transfer records belonging to the selected configuration file are displayed on the Edit Transfer page.
4. Edit the configuration details as required.
5. To save any editing changes made, click the Update button, or to cancel any editing changes made, click the Cancel button.
   Both of these actions return you to the Schedule Transfer main page.

Submit an Existing Configuration File

Follow these steps to submit an existing configuration file to XCOM Data Transport for scheduling.

1. Click Schedule Transfer on the Home page.
2. Click the Load button on the Schedule Transfer page.
   A File open dialog box opens, through which you can select and load the configuration file.

3. Perform *one* of the following actions:
   - Select one or more rows and click the Submit link.
   - Click the SubmitAll button.
4. The transfer records belonging to the selected files are submitted.

Check the Status of the Submitted Transfer Records

You can see the schedule status of the submitted records in the Schedule Status column of the Process Transfer Record table.

Get History Records

The Get History Records page lets you extract and display history record for one or more XCOM Data Transport systems. You can also use different search criteria to limit the number of records displayed.

This page allows you to do the following actions:

- Get the history of transfer records
- Filter transfer records
- View transfer details
- Auto Refresh History table
- Dynamically change the columns displayed in the History table
- Save in tab-delimited form for Easytrieve reports
- Export to Excel

History Search Considerations

Each XCOM Data Transport base system has a set of XCOMHIST*xxxxx* parameters in the xcom.glb file. These parameters, when defined, determine the database where history records are written for each transfer run from that XCOM Data Transport system. So, for a transfer run from system PC1 to system PC2, two history records are created, as follows:

- PC1 writes a history record to its associated database.
- PC2 writes a history record to its associated database.

With this setup, you must run a query on the PC1 system to access the database for its history records, and then run a separate query on the PC2 system to access the database for its history records. To be able to see the records from both the PC1 and PC2 systems when you run a query on either of these PC systems, you would need to set up your ODBC connections and set the XCOMHIST*xxxxx* parameters to write history records to the same database and table so that it would be shared across one or more systems.

To determine what users are then authorized to access the history records from either a shared or non-shared database, two local groups would need to be created, XCOMADM and XCOMSADM. The XCOMADM group allows a user to see history records from any user on the same XCOM Data Transport system, but the XCOMSADM group goes a step further and allows a user to see history records in the database from any XCOM Data Transport system.

- The XCOMSADM and XCOMADM groups need to be defined as local groups.
- Only history records are written to the history database. Inactive transfers (that is, transfers scheduled for the future) and active transfers are in the XCOM Data Transport queue. Each XCOM Data Transport system has its own queue, which cannot be shared.

Specify Search Criteria for History Records

To display the History Parameters page, click one of the following:

- The Get History Records tab
- The Get History Records link on the Home page of the XCOM Data Transport GUI

This is the default screen for the Get History Records Tab.

To display short history information for the local system

1. Specify values for the parameters on the Get History Records page.
2. Click the Submit button to get a list of history records matching the search criteria specified.

Export History Parameters

From the Get History Records tab, you can use the Export button to export history parameters to an XML file. You can then use this XML file to retrieve history records, using xcomtcp -c7 transfers.

For more information about using -c7 transfers.

> **NOTE**
> The XML Schema Definition (XSD) of the history parameters is contained in the TransportInterfaceSchema file.

To export history parameters to an XML file

1. On the History Parameters pane of the Get History Records tab, enter values for the history parameters that you want to use.
2. Click the Export button.
3. Enter the XML file name to save the history parameters.
4. Click the Save button.
   The history parameters XML file is created in the selected location.

Display History Records

The Get History Records page displays history records for one or more XCOM Data Transport servers, depending on your setup. The display of records also depends on your criteria and search on transfer status to help you taking further action.

Display Brief History Records

To display brief history records

1. Log in to the XCOM Data Transport GUI.
   The Home page appears.
2. Click Get History Records.
   The Get History records page appears to display the History Parameters pane.
3. Enter values for the history parameters and click Submit.

   > **NOTE**
   > These history parameters are not case-sensitive except for the File parameter which has a separate Case-Sensitive parameter. If this parameter is set to yes, then the history records returned are case-sensitive for the File parameter only. Otherwise, the filtering of the history records based on these parameters return case-insensitive results. For example, specifying USER01 as the Requesting User ID returns the same results as specifying user01.
4. The CA XCOM Transfer Request Display Request Display pane appears displaying brief history for the local computer based on the specified parameters.

Display Detailed History Records

To display detailed history records

On the CA XCOM Transfer Request Display Transfer Request Display, click the Req. No. link for a transfer record.

The Detail History Record page appears to display history details for the specified record in the following collapsible panes:

- General Information
- Notification
- Misc Information

All fields on this page are display-only.

> **NOTE**
>
> When a transfer is active, two additional buttons appear on the detailed history record screen:
>
> - Refresh - Click this button to get the latest update for the active transfer.
> - Watch - Click this button to get constant updates for the active transfer.

Auto Refresh the Table of History Records

The GUI has been enhanced to refresh the table of History records after a specified interval time. By default auto-refresh is disabled on the Get History Records page. Enabled auto-refresh as documented.

To enable auto-refresh

To enable auto-refresh, select any non-zero value from the Refresh parameter drop-down list. The list contains the following refresh values in seconds, 0, 10, 20, 30, 45, 60, 90, and 120.

> **NOTE**
>
> When auto-refresh is enabled, the Refresh button is renamed as Stop button and the Submit button is renamed as Update button.

- When auto-refresh is enabled, if the history parameters fields are changed, click Update to retrieve and display the history records that match these new history parameter values.

  > **NOTE**
  >
  > When auto-refresh is enabled, during each refresh, the history parameter values that were set during the last click of the Update/Submit button are used.

- To disable/stop the auto-refresh, select 0 (zero) from the Refresh drop-down box or Stop.

  > **NOTE**
  >
  > When auto-refresh is disabled to update the table of displayed History records, perform either of the following actions:
  >
  > - Click Refresh to refresh the history records so that they display up-to-date information.
  > - Change the history parameters and click Submit to rebuild the history table.

CA XCOM Transfer Request Display Transfer Request Display Pane

The XCOM Data Transport Transfer Request Display pane lets you dynamically change the columns in the displayed History Table and to save the preferences.

- Unpin the history record table display.
- Select options to hide or display brief details of the history records and to sort the history records.
- Perform actions on the history records
- Export the table of history records in CSV or Easytrieve format.

Dynamically Update the History Table

Follow these steps:

1. Select History Table Columns, from the drop-down box, to open the list of columns present in the History table.
2. Select the columns that you want to display in the History table, and click Submit to update the History table.
   Every logged-in user can save the preferences of the display of columns in the History table by clicking Save.

   NOTE
   After clicking Save if a folder named Preferences is not present in the XCOM_HOME folder, it will be created first. After this the list of selected columns is saved to the file selection_LOGGED_IN_USERID. If an unautorized user creates the folder, an error message appears and the preferences fail to save.

Unpin the Table of History Records

The GUI allows a user to unpin the table of History records into a separate window by clicking Unpin. This action lets you view the table of history records separate from the GUI. To pin the table of history records back to the GUI, close the window containing the table of history records.

Options Available on the Table of History Records

The table of History records provides the following options:

Show option in the Show Details column

Click Show Record Detail to view brief details of the selected transfer record.

- Hide option in the Show Details column
  After brief details of a transfer are displayed, click Hide Record Detail to hide the brief details of the selected transfer record.

  NOTE

  The brief details of the transfer records are hidden by default.
- Show All Details link
  Click Show All Details to view the brief details of all the history records displayed in the table.
- Hide All Details link
  After you click Show All Details, click Hide All Details to hide the details of all the history records displayed in the table.

  NOTE

  The brief details of all the transfer records are hidden by default.
- Sort Rows by the Request Number.
  Click column header to sort rows in the table of history records based on the Req No (request number).

Actions Available on the Table of History Records

The following actions are available on the drop-down list at the right of each row of the CA XCOM Transfer Request Display Request Display for those transfer records where an action can be performed:

NOTE
Some of these actions are not available for particular transfer records.

Cancel
Cancels a queued or active transfer.

Hold
Holds a queued transfer.

Release
Releases a queued transfer.

Suspend
Suspends an active transfer.

Resume
Resumes a suspended transfer.

Update
Changes execution priority, scheduling priority, or scheduled start time for a queued transfer.

Purge
Removes a queued transfer from the queue.

Export History Reports

Use the Export drop-down list on the CA XCOM Transfer Request Display Transfer Request Display to export information to an external file, as follows:

CSV Format
Creates a file containing the history record information in comma-separated value format.

Easytrieve
Creates a file containing the history record information in a format that an Easytrieve report can be generated from.

Trusted Transfer

The XCOM Data Transport Trusted Facility allows to initiate a transfer without having to supply a password for the remote computer.

The Trusted Facility works with the credentials of the user who initiates the transfer and passes it to the remote computer. The remote computer looks up in a database whether the incoming transfer is from a trusted partner and a trusted user. If these checks pass, the transfer is allowed; otherwise the transfer fails.

Manage Remote System

You can manage remote systems by adding, updating, and deleting remote system information.

To manage a remote system

1. Click the Trusted Transfer tab.
   The Trusted Transfer page appears
2. Click Remote System in the Contents column at the left of the page.
3. The Remote System page appears.

For information about adding, updating, and deleting remote systems, see the following sections.

Add a Remote System

You can use the Remote System page (from the Trusted Transfer page) to add remote systems.

To add a remote system

1. On the Remote System page, do *one* of the following:

- Enter a remote system name in the Remote System Name text box.
- Enter values in the Remote SYSID and Remote SYSNAME fields.

2. Click Add.
   The specified remote system is added to the database.

   > **NOTE**
   > If you specify SYSID and SYSNAME values, the Remote System value is saved as the combination of these two values.

Update a Remote System

You can use the Remote System page (from the Trusted Transfer page) to update remote systems.

To update a remote system

1. On the Remote System page, select a remote system name from the Remote System Names list and do *one* of the following:
   - Update the values in the Remote SYSID and/or Remote SYSNAME fields.
   - Update the value in the Remote System field.
2. Update the information in the Notes text box.
3. Click the Update button.
   The specified remote system is saved to the database with the updated values.

Delete a Remote System

You can use the Remote System page (from the Trusted Transfer page) to delete remote systems.

To delete a remote system

1. On the Remote System page, select a remote system name from the Remote System Names list.
2. Click the Remove button.
   The specified remote system is deleted from the database.

Manage Groups

You can manage groups by adding and deleting groups.

To manage groups

1. Click the Trusted Transfer tab.
   The Trusted Transfer page appears
2. Click Manage Groups in the Contents column at the left of the page.
3. The Manage Groups page appears.

For information about how to add, update, or delete groups, see the following sections.

Add a Group

You can use the Manage Groups page (from the Trusted Transfer page) to add groups to the database.

To add a group

1. On the Manage Groups page, select a remote system name from the list in the Remote System Names text box.
2. Enter a value in the Group Name field.
3. Click the Add button.
   The specified group is added to the database.

Delete a Group

You can use the Manage Groups page (from the Trusted Transfer page) to delete groups from the database.

To delete a group

1. On the Manage Groups page, select a remote system name from the list in the Remote System Names text box.
2. Select a group from the list in the Group Names text box.
   The selected group is displayed.
3. Click the Remove button.
   The group is deleted from the database.

Manage Users

You can manage users by adding and deleting users.

To manage users

1. Click the Trusted Transfer tab.
   The Trusted Transfer page appears
2. Click Manage Users in the Contents column at the left of the page.
3. The Manage Users page appears.

For information about how to add, update, or delete users, see the following sections.

Add a User

You can use the Manage Users page (from the Trusted Transfer page) to add users to the database.

To add a user

1. On the Manage Users page, select a remote system name from the list in the Remote System Names text box.
2. To add a user who will be a member of a group, select a group from the Group Name drop-down list, and enter a value in the User Name field.
3. To add a user as an individual member, leave the Group Name field blank, and enter a value in the User Name field.
4. Click the Add button.
   The specified user is added to the database.

> **NOTE**
> Asterisk (*) is used as a generic user for the particular system. If an asterisk is defined as a user, the target XCOM Data Transport system finds the corresponding group name for that particular system (source system).

Delete a User

You can use the Manage Users page (from the Trusted Transfer page) to delete users from the database.

To delete a user

1. On the Manage Users page, select a remote system name from the list in the Remote System Names text box.
2. To delete a user who is a member of a group, select the group from the Group Name drop-down list, and select a user from the User Names list.
   The selected user is displayed.
3. To delete an individual user, leave the Group Name field blank, and select a user from the User Names list.
   The selected user appears.
4. Click the Remove button.
   The database removes the user.

Manage Global Parameters

This article describes how you can add or modify global parameters. The following global parameters are manipulated using the xcom.glb text file.

XCOM Data Transport uses these parameters as default values to process transfers and other features. XCOM Data Transport validates each parameter against expected valid values.

> **NOTE**
> For any changes made on this page to take effect, restart the XCOMD XCOM Scheduler service Service.

Authentication Parameters

1. Authentication Type
   Authentication type specifies the type of authentication that you use for transfers.

PAM

PAM enables Pluggable Authentication Modules authentication.

SYSTEM

SYSTEM enables traditional UNIX authentication.

2. PAM Path
   PAM path specifies the path to your PAM library for your current UNIX or Linux platform. Do not specify the library name in the path. For example, if the library is available at `/usr/lib64`, specify `PAM_PATH=/usr/lib64`. XCOM appends the library name to the path internally.
   Range: 1 to 256 characters

   > **NOTE**
   >
   > This parameter is valid only if AUTHENTICATION TYPE=PAM.

Character Conversion Parameters

You can manage character conversion parameters by updating their default values.

Follow these steps

1. Click the arrow in the Character Conversion pane to expand it, if it is not already expanded.
   The current default values of the character conversion parameters appear, in the following sections:
   – Character Conversion Identification
   – Internal Conversion Table
2. To update the Character Conversion Identification section, perform the following actions:
   a. Click the Browse button to select a value for the **ATOE Filename** field.
      The name of the file containing the ASCII-to-EBCDIC character conversion table
      Range: 0 to 256 characters
      Default: $XCOM_HOME/convtab/atoe.tab
   b. Click the Browse button to select a value for the **ETOA Filename** field.
      The name of the file containing the EBCDIC-to-ASCII character conversion table
      Range: 0 to 256 characters
      Default: $XCOM_HOME/convtab/etoa.tab
   c. Enter a value in the **Convert Classes** field.
      A character string containing print classes for which EBCDIC-to-ASCII conversions are performed. For the incoming report transfers only.
      Range: 1 to 64 characters
      Default: None

3. To update the Internal Conversion Table section, select or clear the Use internal conversion tables for the character conversions field.
   The field Indicates whether internal or external conversion tables are used for ASCII-to-EBCDIC conversion and EBCDIC-to-ASCII conversion.
4. Click the Update button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

Gateway Parameters

You can manage Gateway parameters by updating their default values for Gateway parameters.

Follow these steps

1. Click the arrow the Gateway pane to expand it, if it is not already expanded.
   The current default values of the Gateway parameters appear.
2. To update the Gateway parameters:
   a. Enter values in the following fields:

IP Address

Specifies the Gateway IP address that the XCOM Data Transport server is allied to. For Gateway transfers only.
Default: None

Port Number

Specifies the Gateway port that the XCOM Data Transport server is allied to. For Gateway transfers only.
Default: None

b. Click the Browse button to select a value for the **Certificate File Path** field.
   Certificate File Path specifies the file name of the certificate that is passed to the Gateway server. It is used to confirm that the request is from an authorized XCOM Data Transport server. (For Gateway transfers only.)
   Default: None

c. Click the Browse button to select a value for the **Private Key File Path** field.
   Private Key File Path specifies the file name of the certificates private Key. (For Gateway transfers only.)
   Default: None

d. Enter a value in the **Pass Phrase** field.
   Pass Phrase specifies the password that was set when the certificate was created. This is required only if a password was defined at the time of certificate creation. (For Gateway transfers only.)
   Default: None

e. Click the Browse button to select a value for the **CAPKI Home Path** field.
   CAPKI Home Path specifies the CAPKI/ETPKI library path. Set this path before you perform any of the following actions:

   - Encryption at rest
   - TLS/SSL transfers
   - Transfers using XCOM Data Transport Gateway

   Range: 0 to 256 characters
   Default: Varies according to the version (32-bit or 64-bit) of XCOM Data Transport:
   For 32-bit: /opt/CA/SharedComponents/CAPKI/CAPKI5/Linux/x86/32/lib
   For 64-bit: /opt/CA/SharedComponents/CAPKI/CAPKI5/Linux/amd64/64/lib

f. Select a value from the drop-down list for the **Fips Mode** field.
   Fips Mode specifies the mode that you use for encryption/decryption.
   Default: NO

g. Select a value from the drop-down list for the Gateway Version field.
   Gateway Version specifies the XCOM Data Transport Gateway version to which this instance of the XCOM Data Transport server is allied.
   For XCOM Data Transport Gateway transfers only.

R116

Indicates that the XCOM Data Transport server is allied to the Release 11.6 Gateway

R120

Indicates that the XCOM Data Transport server is allied to the Release 12.0 Gateway

Default: R120

3. Click the Update button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

History Parameters

You can manage history parameters by updating their default values.

Follow these steps

1. Click the arrow in the History pane to expand it, if it is not already expanded.
   The current default values of the history parameters appear.

   NOTE
   Where appropriate, the range of valid values is shown for each field.
2. To update the history parameters:
   a. Enter values in the following fields:

History ODBC Name

Specifies the name of the ODBC connection that is used to connect to the relational database.
Default: No default value

History Table

The name of the table that was created for the XCOM Data Transport history records.
Default: xcom_history_tbl

History User

Specifies the user ID that has complete access to the history database.
Default: No default value

History Password

The password of History User (XCOMHIST_USER)
Default: No default value

b. Select values from the drop-down lists for the following fields:

History Split File

Indicates when an insert fails whether the query is written out as 72-byte records.
Default: YES

History Backslash

Treat a backslash in a file name as a single backslash. Number of backslashes depends on the target ODBC system.

NOTE

If the ODBC is z/OS or DB2, you need a single \ to display the data correctly. If the ODBC is PC-based, for example MySQL, \\ is treated as a single \.

Default: No

c. Click the Browse button to select a value for the **History File** field.
   History File is the name of the flat file that contains insert records when the database computer is not connected and available. Records are also written into this file as a result of an SQL failure.
   Range: 0 to 256 characters
   Default: $XCOM_HOME/config/history.inserts

d. Enter a value in the **History Owner** field.
   History Owner specifies the ID of the creator of the History Table. You can omit if History Owner and the History User (XCOMHIST_USER) are the same.
   Default: No default value

3. (Optional) To test the connection to the history table, click the Test button next to the History Table field. A pop-up window indicates whether the connection to the database is successful and number of rows that exist in the table.
4. Click the Update button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

Incoming Transfer Parameters

You can manage incoming transfer parameters by updating their default parameters.

Follow these steps

1. Click the arrow in the Incoming Transfers pane to expand it, if it is not already expanded.
   The current default values of the incoming transfer parameters appear, in the following sections:
   - Incoming Transfer Identification
   - Create Directories
   - Remove Trail Blanks
   - Add EOF Marker
2. To update the **Incoming Transfer Identification** section, enter values in the following fields:

EOL Classes

A character string containing print classes for which an ASCII new line character is appended to each record. (For the incoming report transfers only.)
Range: 0 to 64 characters
Default: None

Metacode Classes

Classes of print jobs that are saved in metacode format, a variable length record format. (For the incoming report transfers only.)
Range: 0 to 64 characters
Default: None

3. To update the Create Directories section, select or clear the Create a nonexistent Directory check box.
   This check box indicates whether a new directory is created to accommodate incoming file transfers.
4. To update the **Remove Trail Blanks** section, select or clear the Remove the blanks at the end of each record check box.
   This check box indicates whether to remove the blanks at the end of each record when receiving a text file.
5. To update the **Add EOF Marker** section, select or clear the Add an EOF marker to the end of the file check box.
   This check box indicates, when receiving a text file, whether you need the end-of-file character (CTRL + Z) written at the end of the file.
6. Click the Update button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

Library Parameters

Use library parameters to specify the value of parameters and paths to the ODBC and JVM libraries.

1. **ODBC Library**
   - This path specifies the value of parameter XCOM_ODBC in xcom.glb.
   - This path specifies the full path to the ODBC library (libodbc.so).
     Range: 0 to 256 characters
     Default: None

2. **JVM Library**
   - This path specifies the value of parameter XCOM_JVM in xcom.glb.
   - This path specifies the full path and file name for the Java JVM sharedlibrary.
     Range: 1 to 256 characters
     Default:
     Linux: $XCOM_HOME/JRE/1.8.0_77/lib/amd64/server/libjvm.so

Mail Parameters

You can manage mail parameters by updating their default values.

Follow these steps

1. Click the arrow in the Mail pane to expand it, if it is not already expanded.
   The current default values of the mail parameters appear.
2. To update the mail parameters, perform the following actions:
   a. Select a value from the drop-down list for the **Mail Type** field.
      The mail type specifies the type of MAIL server used for sending mail notifications.
   b. If the Mail Type field is set to SMTP, enter a value for the **SMTP Server** field.
      Range: 0 to 64 characters
      Default: None
3. Click the Update button to save your change.
   The default value of the updated parameter is changed in the xcom.glb file.

Queue Parameters

You can use manage queue parameters by updating their default values.

Follow these steps

1. Click the arrow in the Queue pane to expand it, if it is not already expanded.
   The current default values of the queue parameters appear, in the following sections:
   - Queue Identification
   - Expiration Time
   - Max Queue Entries
   - Max Session Entries
   - Age Time

   NOTE
   Where appropriate, the range of valid values is shown for each field.
2. To update the **Queue Identification** section, perform the following functions:
   a. Click the Browse button to select a value for the **Queue Path** field.
      The queue path defines the path name for the transfer queue data files.
      Range: 0 to 256 characters
      Default: $XCOM_HOME\Q
   b. Click the Browse button to select a value for the **Sessions File** field.
      The session file defines the path name of the XCOM.SES file. This file tells XCOM Data Transport the number of parallel sessions available for locally initiated transfers for each remote system.
      Range: 0 to 256 characters
      Default: $XCOM_HOME\config\xcom.ses
3. To update the **Expiration Time** section, enter a value in the **Secs** field.
   The maximum time, in seconds, that a transaction takes in the transfer queue after the transaction is executed. When transaction reaches the maximum time, all references to the transaction are removed from the queue. The references include trace files and temporary files.

Range: 0 to 32767
Default: 6000

4. To update the **Max Queue Entries** section, enter a value in the **Entries** field.
   The maximum number of entries allowed in the transfer queue. Once the maximum number of queue entries is reached, XCOM Data Transport rejects subsequent transfer attempts.
   Range: 0 to *n*, where *n* is the maximum allowed memory segment divided by 512.
   Default: 50
5. To update the **Max Session Entries** section, enter a value in the **Partners** field.
   The maximum number of partners that can be described in the XCOM.SES file.
   Range: 1 to 999
   Default: 15
6. To update the **Age Time** section, enter a value in the **Secs** field.
   The number of seconds before waiting queue entries are removed from the queue. If the value is 0, the waiting queue entries never age and are never removed from the queue.
   Range: 0 to 86313600 (999 days)
   Default: 432000 (Five days)
7. Click the Update button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

System Parameters

You can manage system parameters by updating their default values.

Follow these steps

1. Click the arrow in the System pane to expand it, if it is not already expanded.
   The current default values of the system parameters appear, in the following sections:
   - System Identification
   - Stat Frequency
   - Daemon
   - Commands
   - Cache

   NOTE
   Where appropriate, the range of valid values is shown for each field.
2. Update the **System Identification** section. To do so, follow these steps:
   a. Enter values for the following fields:

SYSID

Specifies the system ID for the local system.

NOTE

This initially gets set during the installation of XCOM.

This value is used for Trusted Transfers and for getting history records.
SYSID and SYSNAME together provide a unique system identifier.
Range: 1 to 4 characters
Default: None

SYSNAME

Specifies the system name for the local system.

NOTE

This initially gets set during the installation of XCOM.

This value is used for Trusted Transfers and for getting history records.
SYSID and SYSNAME together provide a unique system identifier.
Range: 1 to 8 characters
Default: None

Shell Cmd

The name of the command that runs jobs, reports, notification scripts, and post processing scripts on the local system.
Range: 1 to 256 characters
Default: /bin/bash

b. Click the Browse button to select a value for the **TempDir** field.
Indicates the directory in which temporary files for jobs and reports can be created.
Range: 0 to 256 characters
Default: /tmp

c. Click the Browse button to select a value for the **Xlogfile** field.
The name of the file where XCOM Data Transport logs activity.
Range: 0 to 256 characters
Default: $XCOM_HOME/xcom.log

d. Update the **Log Connect Message** field by selecting a value (YES or NO) from the drop-down list. The selected value determines whether or not the informational connection messages to be written to the log.

e. Use **UMASK** to set the permissions those are assigned to a file when creating and receiving it on the 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 umask command.
Range: 000 to 777
Default: 022

> **NOTE**
>
> For directories: XCOM Data Transport sets permissions for a created directory to 7xx, no matter what owner UMASK value is specified. Group and other permissions, represented by xx, represent the permissions with the specified UMASK removed.
>
> For files: When the file is being transferred, XCOM Data Transport sets permissions for a created file to 6xx. Here xx represents the permissions with the specified UMASK removed. After the transfer is complete, XCOM Data Transport sets the owner permission with the specified UMASK removed.

3. Update the **Stat Frequency** section by entering a value in the **Transfer stats refresh** field.
The value indicates the frequency with which transfer statistics are made available to XCOMQM and the GUI. Intended for tuning high-speed links. Longer values help performance but byte/record counts in XCOMQM and the GUI may be slightly behind the actual counts.
Range: 1 to 9999
Default: 10

4. Update the **Daemon** section by entering a value in the **Timeout** field.
The value specifies the number of minutes that XCOM Data Transport will wait for a response from a partner before aborting a session. This ensures a transfer does not hang indefinitely waiting for a response.
Range: 10 to 1440
Default: 60

5. Update the **Commands** section. To do so, follow these steps:

a. Select one of the following commands from the drop-down list.

TCP_CMD

Path and name of the XCOM Data Transport program started by the XCOMD XCOM Scheduler service that is used for transfers that use TCP/IP protocols.
Range: 0 to 256 characters
Default: $XCOM_HOME/bin/xcomtcp

TP_CMD

Path and name of the XCOM Data Transport program started by the XCOMD XCOM Scheduler service that is used for transfers that use SNA/APPC protocols.
Range: 0 to 256 characters
Default: $XCOM_HOME/bin/xcomtp (UNIX)
None (Linux)

XEND_CMD

Name of the post-processing command file optionally invoked by the XCOM Data Transport transfer program after any type of transfer is finished, whether successful or not. A sample command file is provided in $XCOM_HOME/cmd/xcomend.
Range: 0 to 256 characters
Default: None

XLPCMD

Name of the post processing command file used to send print jobs to the spooler. For incoming reports only.
Range: 0 to 256 characters
Default: $XCOM_HOME/cmd/xcomlp

XNOTIFYCMD

Name of the command file that XCOM Data Transport uses to notify users on the local system of the completion of a transfer. This is normally a shell script that composes a message and invokes mail or write as appropriate.
Range: 0 to 256 characters
Default: $XCOM_HOME/cmd/xcomntfy

XPPCMD

Name of the command file used for user-defined post processing. A sample command file is provided in $XCOM_HOME/cmd/xcompp.
Range: 0 to 256 characters
Default: None

b. Enter the path of the selected command in the Details box.

6. Update the **Cache** section. To do so, enter appropriate values in the following fields:

Cache Read Size

Specifies the size (in KB) of the cache for reading files.
Range: 0 to 9999
Default: 0

Cache Write Size

Specifies the size (in KB) of the cache for writing files.
Range: 0 to 9999
Default: 0

7. Click the Update Button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

TCP/IP Parameters

You can manage TCP/IP parameters by updating their default values.

Follow these steps

1. Click the arrow in the TCP/IP pane to expand it, if it is not already expanded.
   The current default values of the TCP/IP parameters appear, in the following sections. The range of valid values is also shown for each field, where appropriate.

- Client Transfer Settings
- Protocol
- Secure Socket Server Settings

NOTE
Some of the parameters in the above sections are not editable within the GUI since they either do not apply to this platform or can only be configured outside of the GUI. During the installation of XCOM, default values are automatically configured for those applicable parameters.

2. To update the **TCP/IP Identification** section, click the Browse button to select a path value for the **Executable Program Path** field.
Range: 0 to 256 characters
Default:
3. To update the **Client Transfer Setting**s section, do the following:
a. Enter values in the following fields:

Port Number
The number of the TCP/IP port on the remote XCOM Data Transport server. Used for TCP/IP transfers only.
Range: 1 to 65535
Default: 8044

Sock Rcv Bufsize
This is the TCP/IP Socket option SO_RCVBUF and is the buffer size used for receive transfers. Use zero for the default size provided by the socket implementation. The value for SOCK_RCV_BUF_SIZE can be smaller than the value for TXPI_BUF_SIZE. Used for TCP/IP transfers only.

NOTE

Socket options affect the way XCOM Data Transport uses the TCP/IP stack implementation.

Range: 0 to 65536
Default: 0

Sock Send Bufsize
This is the TCP/IP Socket option SO_SNDBUF and is the buffer size used for send transfers. Use zero for the default size provided by the socket implementation. The value for SOCK_SEND_BUF_SIZE can be smaller than the value for TXPI_BUF_SIZE. Used for TCP/IP transfers only.

NOTE

Socket options affect the way XCOM Data Transport uses the TCP/IP stack implementation.

Range: 0 to 65536
Default: 0

b. Check or uncheck the **Sock Delay** field.
This is the TCP/IP Socket option SO_RCVBUF and is the buffer size used for receive transfers. Use zero for the default size provided by the socket implementation. The value for SOCK_RCV_BUF_SIZE can be smaller than the value for TXPI_BUF_SIZE. Used for TCP/IP transfers only.

NOTE
Socket options affect the way XCOM Data Transport uses the TCP/IP stack implementation.

Range: 0 to 65536
Default: 0

c. Enter values in the following fields:

Txpi Term Timeout
Maximum wait time, in seconds, for the 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 occurs only during an error in the termination of the connection. Used for TCP/IP transfers only.

Range: 0 to 999 seconds
Default: 20 seconds

Txpi Rcv Timeout

Maximum wait time, in seconds, after a TCP/IP failure is detected before a socket error is generated. This defaults to zero and should remain at zero unless it becomes necessary for a broken connection to generate a retryable error.
Changing this parameter from zero invokes the TCP/IP select function, which adds a measure of connection detection, but sacrifices some performance. This parameter is for TCP/IP only.
Range: 0 to 1200 seconds
Default: 0 seconds

Txpi Buf Size

For TCP/IP transfers, the internal buffer size for sends and receives. The default size allows multiple XCOM Data Transport records to be received in a single socket call. With this default, if your XCOM Data Transport record size is less than 32K, XCOM Data Transport attempts to receive multiple records in a single socket call. Used for TCP/IP transfers only.
Range: 0 to 65536
Default: 32768

Txpi Send Check Freq

For TCP/IP transfers, the internal buffer size for sends and receives. The default size allows multiple XCOM Data Transport records to be received in a single socket call. With this default, if your XCOM Data Transport record size is less than 32K, XCOM Data Transport attempts to receive multiple records in a single socket call. Used for TCP/IP transfers only.
Range: 0 to 65536
Default: 32768

4. To update the **Server Settings** section, enter values in the following fields:
   a. **Port Number**
      The number of the TCP/IP port on the local XCOM Data Transport server. Used for TCP/IP transfers only.
      Range: 1 to 65535
      Default: 8044
   b. **Maximum Clients**
      Range: 0 to 9999
      Default: 16
   c. **Termination Loop**
      Range: 0 to 9999
      Default: 5
   d. **Termination Delay**
      Range: 0 to 9999
      Default: 5
5. To update the **Implicit Packing** section, check or uncheck the **Default to use big packing 31K data buffer size** field.
   Indicates whether TCP/IP should always use packing.
6. To update the **Protocol** section, select a value from the drop-down list of the **Default protocol** field.
   The type of communication protocol to use. This is dependent on what protocols are supported on the local system and if the SNA component (where applicable) was selected to be installed during the installation process.
   Range: SNA or TCPIP
   Defaults: SNA and TCP/IP
7. To update the **Secure Socket Server Settings** section, do the following:
   a. Check or uncheck the **Display Cipher in Queue Details** field.
      Specifies whether to display encryption algorithms in the XCOM Data Transport queue detailed information for transfers.
   b. Click the Browse button to select a value for the **Secure Socket Fil**e field.

Specifies the value of the configssl.cnf file path and file name.
Range: 1 to 256 characters
Default: $XCOM_HOME/config/configssl.cnf

8. To update the **Server Ipv6 Port Numbe**r section, do the following:
   a. Enter values in the following fields:

Ipv6 Port Number
The number of the TCP/IP port on the remote XCOM Data Transport server used for IPv6 transfers. Used for TCP/IP transfers only.
Range: 1 to 65535
Default: 8046

Ipv6 TLS/SSL Port Number
The number of the TCP/IP port on the remote XCOM Data Transport server used for IPv6 TLS/SSL transfers. Used for TCP/IP transfers only.
Range: 1 to 65535
Default: 8047

   b. Select a value from the drop-down list of the **Choose Listeners** field.
   Determines which TCP/IP listeners are started when the XCOMD daemon is started.
   Default: IPv4 listeners

9. Click the Update button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

Trace Parameters

You can manage trace parameters by updating their default values.

Follow these steps

1. Click the arrow in the Trace pane to expand it, if it is not already expanded.
   The current default values of the trace parameters appear, in the following sections:
   - Xtrace
   - Debug Flag
   - System Level
   - Trace Path

   NOTE
   Where appropriate, the range of valid values is shown for each field.
2. To update the **Xtrace** section, enter a value in the **Trace level** field.
   This field indicates the level of desired execution tracing.
   Range: 0 to 10
   Default: 0
3. To update the **Debug Flag** section, check or uncheck the **Print trace to STDERR** field.
   This field indicates whether to print trace data to STDERR.
4. To update the **System Level** section, perform the following actions:
   a. Select from the **Component** drop-down list the components that need these trace settings: Trace Off, Trace On, or Trace Detail.
   b. Click the Browse button to select a value for **Component Trace File**.
   This value specifies the path and file name for the Component Trace File.
   Range: 1 to 256 characters
   Default: %XCOM_HOME%\trace.trc
   c. Click one of the following:

- Trace Off
- Trace On
- Trace Detail

5. To update the **Trace Path** section, click Browse to select a value for **Trace Path**.
   This value specifies the path where trace files will be written to.
   Range: 1 to 256 characters
   Default: $XCOM_HOME/trace
6. Click the Update button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

Trusted Parameters

You can manage trusted parameters by updating their default values.

Follow these steps

1. Click the arrow in the Trusted pane to expand it, if it is not already expanded.
   The current default values of the trusted parameters appear.

   NOTE
   Where appropriate, the range of valid values is shown for each field.
2. To update the trusted parameters, do the following:
   a. Enter values in the following fields:

Trusted Database Server
Specifies the name of the database server where the trusted database was created.
Range: 1 to 256 characters
Default: None

Trusted Database Port
Specifies the port of the database server where the trusted database was created.
Range: 1 to 65535
Default: 50000

Trusted Database Name
Specifies the name of the database where the trusted tables were created.
Range: 1 to 256 characters
Default: None

   b. Select a value from the drop-down list for the **Trusted Database Type** field.
   Specifies the database type that resides on the database server.
   Range: DB2 or MySQL
   Default: DB2
   c. Enter values in the following fields:

Trusted Table Prefix
Specifies the prefix to use for the names of the Trusted Tables.
Range: 1 to 16 characters
Default: XCOM_TRUSTED

Trusted Database Owner
Specifies the ID of the creator of the Trusted Tables. May be omitted if it is the same as the Trusted Database User (TRUST_USER).
Range: 1 to 32 characters
Default: No default value

Trusted Database User
Specifies a generic user ID that has been defined to the RDMS for the Trusted Tables.

Range: 1 to 64 characters
Default: No default value

Trusted Database Password
Specifies the password of the Trusted Database User for the Trusted Tables.
Range: 1 to 128 characters
Default: No default value

Trusted ODBC Data Source
Specifies the ODBC Data Source Name for the Trusted Tables.
Range: 1 to 16 characters
Default: No default value

3. If you want to test the trusted database values, click the Test button next to the Trusted Database Port field. A pop-up window appears, indicating whether the JDBC and ODBC connections to the database were successful and how many rows exist in the trusted tables.
4. Click the Update button to save your changes.
   The default values of the updated parameters are changed in the xcom.glb file.

Unicode Parameters

You can manage Unicode Parameters.

- **To update default values for Unicode parameters:**
- Click the arrow button in the Unicode pane to expand it, if it is not already expanded. The current default values of the Unicode parameters appear.
- Enter a value in the Action to Take On the Input Character Error field.
  Action to Take On Input Character Error
  Specifies the default action when the input file contains data that is not consistent with the specified input character set.
  Range: FAIL, REPLACE, SKIP, or REPLACE#nnnnnnn where nnnnnnn is in the range of 0 through 1114111.
  Default: FAIL
- Enter a value in the Action to Take On the Convert Character Error field.
  Action to Take On Convert Character Error
  Specifies the default action when the input file contains characters that cannot be converted as they are not included within the output character sets character repertoire.
  Range: FAIL, REPLACE, SKIP, or REPLACE#nnnnnnn where nnnnnnn is in the range of 0 through 1114111.
  Default: FAIL
- Enter a value in the Default Character set field.
  Default Character set
  Specifies the default character set XCOM Data Transport uses for Unicode transfers (Encoding=UTF8 or Encoding=UTF16).
  CCSID#nnnnn
  nnnnnn - specifies the CCSID number that corresponds to the character set. Valid values are 1 - 65535.
  Alternatively, can be specified as an IANA character set name, or (ICU) acceptable alias name.
  Range: 0 to 60 characters
  Default: ISO-8859-1
- Enter a value in the Default Character set field.
  Default Delimiter
  Specifies default encoding for which the specified Character-set is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list.
  Additionally, specifies a colon separated list of record delimiters that are used to mark and detect the end of a record.
  Range: 0 to 60 characters
  Default: CRLF:LF
- **ICU Path**

Specifies the path to your ICU libraries for your current UNIX platform.
Range: 0 to 256 characters
Default: $XCOM_HOME/bin

Transmission Password Encryption

You can manage Transmission Password Encryption Parameters by updating their default values.

Follow these steps

1. Click the arrow button in the Transmission Password Encryption pane to expand it, if it is not already expanded. The current default values of the Transmission Password Encryption parameters appear.
2. Enter a value in the Default Local Cipher List field.

Default Local Cipher List
Specifies the default list of ciphers which are used to encrypt the password fields for locally initiated transfers or meta-transfers.
Range: 0 to 256 characters
Default: COMPAT

3. Enter a value in the Remote Permitted Cipher List field.

Remote Permitted Cipher List
Specifies the permitted list of ciphers that are used to encrypt the password fields for remotely initiated transfers or meta-transfers.
Range: 0 to 256 characters
Default: XCOM:ALL:COMPAT

4. Select a value for the Remote DH Prime Number Size (in bits)field.

Remote DH Prime Number Size (in bits)
Specifies the size (in bits) of the prime number that is used during DH (Diffie-Hellman) exchange for remotely initiated transfers or meta-transfers. The exchanged value generates the key that is used to encrypt the password fields transmitted in the XCOM Data Transport header.
Range: 256, 512, 1024, 2048, 4096
Default: 1024

Creating and Using Configuration Files

Use configuration files to set default parameters for XCOM Data Transport transfers. Use xcom.cnf to change default settings or to set transfer-specific variables for an individual locally initiated transfer. You can have any number of .cnf files with different names.

XCOM Data Transport reads the specified .cnf parameter file, or the $XCOM_HOME/config/xcom.cnf if no configuration file is specified. In such cases, the Data Transport reads any parameters that are specified on the command line, and then executes the specified file transfer. The Data Transport automatically uses the defaults from the xcom.glb file to supply any unspecified parameter values. For each transfer, the Data Transport compiles a full set of parameter values. To collect these values, the program follows a search sequence. The values are collected in the following order:

1. The XCOM Data Transport program defaults
2. Parameters that were defined in xcom.glb when the XCOM Data Transport daemon was started
3. Parameters that are defined in the specified .cnf file
   If you do not find a copy of the configuration file in the current directory, use $XCOM_HOME/config/xcom.cnf. If a .cnf file is specified, the parameters are not picked up from the xcom.cnf.
4. Parameters that are defined in the command line following the -f option

NOTE
Use a text editor to modify the values in xcom.cnf. All modifications go into effect the next time a transfer is initiated.

For a description of how parameter values take precedence, see Using XCOM Data Transport Parameters in "Operating Environment." For a complete list of XCOM Data Transport parameters, see the "Parameters."

Creating Configuration Files

Create configuration files of your own for different types of transfers.

Follow these steps

From the command line, copy xcom.cnf to a new file.

1. cp $XCOM_HOME/config/xcom.cnf *configfilename.cnf*
   The file xcom.cnf gets renamed to *configfilename.cnf*, and resides in your local directory.

   NOTE

   The file name *configfilename.cnf* is used only as an example here.
2. From your local directory, load *configfilename.cnf* into vi or another editor.
   The *configfilename.cnf* file gets loaded in the editor.
3. Modify the parameters in the new file to your specific needs.
   It is not necessary to list all available parameters in a configuration file. Any unspecified parameters use the defaults that are set in xcom.glb or the XCOM Data Transport daemon.
   If you specify a configuration file, then xcom.cnf is not used. If you do not specify a configuration file, then XCOM Data Transport looks in xcom.cnf for parameter values.
4. Save the file and exit the editor.

Example 1

In the following example, XCOM Data Transport is invoked from the command line and a configuration file is specified.

```
xcom62 [ -cnumber] [ -f [configfilename]
[PARAMETER_NAME=value ...] ]
```

Replace *number*, *configfilename*, and *PARAMETER_NAME=value* with the desired transfer value, configuration file name, and/or parameter name and values.

For the complete syntax and options for using xcom62 or xcomtcp, see Transferring Files.

Example 2

In the following example, a regional office transfers its accounting records daily to the mainframe headquarters.

For this transfer, create a configuration file named *account.cnf* that contains the following parameter values:

```
FILE_OPTION=REPLACE
LOCAL_FILE=/usr/bills/july
REMOTE_FILE=BILLS.JULY
```

Then specify this configuration file to use for the transfer as follows:

```
xcom62 -c1 -f account.cnf
```

Using Symbolic Parameters

You can place symbolic variables in any combination in SYSIN01 control statements to create composite parameter values. You can use a period (.) as a terminating character for the symbolic variable, but this is not required. If a period is present, it is removed from the resultant field content.

Example

```
REMOTE_FILE=C:\REMOTE_FILE-&TIME(HH)-&TIME(MM)-&TIME(SS)
```

> **WARNING**
> The symbolic variable expression cannot exceed the maximum length allowed by the parameter. In the case of REMOTE_FILE, the maximum length is 256 characters.

This results in the value for REMOTE_FILE being set to:

```
C:\REMOTE_FILE-15-31-28
```

> **WARNING**
> If you specify a symbolic parameter on the command line, you need to enclose it in double quotes, so that Windows does not interpret the ampersand character (&) as a special command line character. So this example would become *one* of the following:

- REMOTE_FILE=C:\REMOTE_FILE-"&TIME(HH)"-"&TIME(MM)"-"&TIME(SS)"
- REMOTE_FILE=”C:\REMOTE_FILE-&TIME(HH)-&TIME(MM)-&TIME(SS)"

If you specify a symbolic parameter in a configuration file, there is no need for double quotes.

Assigning Values

If the symbolic variable does not have a system value, you must assign a value before using the symbolic variable. You can also specify subscripts on symbolic variables that do not have a format-code. There are two formats for subscripted symbolic variables. If only one numeric value is present, a starting position of 1 is assumed. A second subscript format allows for a starting position as well as a length to be entered:

Example (one subscript)

```
LU=L784000
ID=LU#&LU(4)
```

This results in the value for ID being set to:

```
LU#L784
```

Example (two subscripts)

```
LU=LU250021
ID=LU&LU(5,4)
```

This results in the value for ID being set to:

```
LU0021
```

Parameters Supporting Symbolic Variables

You can use symbolic parameters with the following parameters:

- DESTINATION
- FORM
- LOCAL_FILE, LOCAL_FILE_RF, LOCAL_FILE_SJ, LOCAL_FILE_SR, LOCAL_NOTIFY, LUSER
- NOTIFY_NAME
- OEDATE, OETIME, OFLMAX, OFLMIN, OID, OLMSG, OREQ, OSDATE, OSTIME, OTNAME, OUSER
- REMOTE_FILE, REMOTE_FILE_RF, REMOTE_SYSTEM, REMOTE_SYSTEM_RF, REMOTE_SYSTEM_SJ, REMOTE_SYSTEM_SR, REPORT_TITLE
- TRANSFER_ID, TRANSFER_USR_DATA
- UNIT, UNIT_RF, USER_DATA, USERID
- VOLUME, VOLUME_RF
- XCOM_CONFIG_SSL, XTCERRDECR, XTCERRINCR, XTCERRPURGE, XTCERRREL, XTCGOODDECR, XTCGOODINCR, XTCGOODPURGE, XTCGOODREL, XTCJOB, XTCNET

Transferring Files

This section contains information about performing file transfers from the command line using xcom62 or xcomtcp. It describes the syntax and options for these commands, and also contains information about multiple transfers, using semicolons, and wildcard characters.

- Use xcom62 or xcomtcp to initiate file transfers from the command line.
- Use xcom62 for transfers that use SNA/APPC protocols.

Use xcomtcp for transfers that use TCP/IP protocols.

Specifying Protocols

When using the xcom62 or xcomtcp commands, the choice of protocol to use is indicated by the PROTOCOL parameter. This can be specified at the command line, in a configuration file, or in the xcom.glb file, depending upon your installation's needs. If the protocol is not specified at the command line, the defaults specified in the configuration file or in xcom.glb are used.

Queuing Transfers

Transfers can be queued or not queued. When queued, failed transfers are retried automatically, depending on the parameters set in the xcom.glb. Multiple transfers can be sent to a partner simultaneously using SNA or TCP/IP.

When not queued, transfers execute immediately, single-threaded and are not retried if they fail.

Using xcom62

For performing transfers using SNA/APPC protocols, xcom62 allows you to initiate the transfer of a file, a job or a report from the command line, and to specify the parameters to be used for that transfer.

Using xcomtcp

For performing transfers using TCP/IP, xcomtcp allows you to initiate the transfer of a file, a job or a report from the command prompt, and to specify the parameters to be used for that transfer

Multiple Transfers

You may initiate multiple transfers in two ways. You can use the NEWXFER option of the CONTROL parameter or you can use separate commands from the command line by placing a semicolon in between each transfer request. For information on the CONTROL parameter, see the appendix "Parameters."

Using Semicolons

You can do several transfers from the command line by using the semicolon (;), the general shell command separator.

Example

In the following example, two files are sent to the remote system, using semicolons to separate the transfers. The file *test1* is sent into file *testA* and file *test2* is sent into file *testB*.

```
xcom62 -c1 -f LOCAL_FILE=test1 REMOTE_FILE=testA; xcom62 -c1
-f LOCAL_FILE=test2 REMOTE_FILE=testB
```

Wildcard Characters

XCOM Data Transport for UNIX and Linux supports the use of wildcards for local files and remote files.

Syntax

The syntax for using xcom62 and xcomtcp is identical, except for the choice of command. The syntax for each is shown below.

Syntax for xcom62

The syntax for using xcom62 is as follows:

```
xcom62 [-cnumber] [option] [-f [configfilename]
[PARAMETER_NAME=value ...] ]
```

Syntax for xcomtcp

The syntax for using xcomtcp is as follows:

```
xcomtcp [-cnumber] [option] [-f [configfilename]
[PARAMETER_NAME=value ...] ]
```

Options

The following list explains the options available for xcom62 and xcomtcp:

NOTE
When the shell encounters an option that performs an action and exits xcom62 or xcomtcp, it will perform the command and exit xcom62 or xcomtcp without reading the rest of the command line.

-c*number*

The type of transfer to be attempted. Valid values for number are:
1 -- Send file to remote system (default).
2 -- Send report to be printed on remote system.
3 -- Send job to be executed on remote system.
4 -- Receive file from remote system.
5 -- Metatransfer to a remote system.
6 -- Metatransfer to inquire on the status of metatransfers.
7 -- Metatransfer to retrieve history records.

-ping

XCOM Data Transport tests reachability of the remote XCOM Data Transport Server and displays information about XCOM Data Transport Server.

NOTE
All other options are ignored.

-h

XCOM Data Transport displays help. The usage message is displayed and exits xcom62 or xcomtcp.

NOTE
All other options are ignored.

-r

XCOM Data Transport displays the release level. The release level of the software is displayed and exits xcom62 or xcomtcp.

-q

XCOM Data Transport queues the transfer. This option works like the QUEUE=YES parameter in xcom.cnf.

-s

XCOM Data Transport sets silent mode, which turns off all output to stderr.

-t

XCOM Data Transport sends trace output to stderr.

-i

XCOM Data Transport ignores configuration errors.

-f `configfilename`

XCOM Data Transport reads *configfilename* to set the parameters necessary for the transfer. The syntax for setting parameters within the configuration file is as follows:

```
PARAMETER1=value
PARAMETER2=value
```

NOTE
Though you can have XCOM Data Transport read parameters from a file as well as from the command line, the parameters on the command line take precedence and must come after the file name.

If you specify -f, you must specify *configfilename* or PARAMETERNAME=value, or you can specify both.

PARAMETERNAME=*value...*

Instead of setting parameters in a file, you can set them on the command line. The syntax is as follows:

```
PARAMETER1=value PARAMETER2=value
```

The value of the parameters must be in uppercase where appropriate, but the parameter names themselves, with an underscore character where indicated, are not required to be in uppercase.

NOTE
Parameters entered on the command line take precedence over those in the parameter file.

no options

If you invoke XCOM Data Transport without options, the defaults that are in xcom.cnf are used for the transfer.

-v

Sets the transfer version type. Valid values are 1 or 2.

-x

Sets the trace level. Valid values are 0 to 10.

Transfer Files Using SNA/APPC Protocols

Learn how to use the `xcom62` command to transfer files using SNA/APPC protocols.

The `xcom62` syntax and options are identical to the `xcomtcp` syntax and options.

To use SNA/APPC protocols with XCOM Data Transport, your computer and the remote XCOM Data Transport system must be configured for the appropriate SNA/APPC.

The PROTOCOL parameter specifies the protocol to use. This parameter can be specified at the command line, in a configuration file, or in the `XCOM.GLB` file. If you do not specify the protocol at the command line, the defaults in the configuration file or in `XCOM.GLB` are used.

The following example queues a file for a send file transfer, using SNA/APPC protocols. The configuration file is `myconfig.cnf`, the local file is `testa.aaa`, and the remote file is `testb.bbb`. The remote system is XCOMSYS2, as specified in the REMOTE_SYSTEM parameter. QUEUE is set to YES, so the transfer is queued. The PROTOCOL parameter in the `XCOM.GLB` file controls the protocol that is used; in this case, it should be set to SNA.

```
xcom62 -c1 -f myconfig.cnf LOCAL_FILE=testa.aaa
REMOTE_FILE=testb.bbb REMOTE_SYSTEM=XCOMSYS2
QUEUE=YES
```

The following example sends a file directly, without sending it to the queue, using SNA/APPC protocols. The configuration file is `myconfig.cnf`, the local file is `testa.aaa`, and the remote file is `testb.bbb`. The remote system is XCOMSYS2, as specified in the REMOTE_SYSTEM parameter. QUEUE is set to NO, so the transfer starts immediately after you press **Enter**. PROTOCOL is set to SNA, so the SNA protocol is used. If the configuration file or the `XCOM.GLB` file specifies PROTOCOL=SNA, you do not need to specify this value here.

```
xcom62 -c1 -f myconfig.cnf LOCAL_FILE=testa.aaa
REMOTE_FILE=testb.bbb REMOTE_SYSTEM=XCOMSYS2
QUEUE=NO PROTOCOL=SNA
```

Transfer Files Using TCP/IP Protocols

Use IP addresses, host names, or domain names when you perform XCOM data transfers using TCP/IP protocols.

> **NOTE**
> The syntax and options for `xcomtcp` are identical to the syntax and options for `xcom62` . For more information, see Transferring Files.

Using IP Addresses and Names

Your system and the remote XCOM Data Transport system must be configured for TCP/IP for you to use TCP/IP protocols with XCOM Data Transport.

Before you perform a file transfer, you must know the IP address, the host name, or the domain name of the remote system. Check with the network administrator of the remote system for these values.

If your system is not configured for TCP/IP, check with your network administrator for further information.

The formats of the IP address, host name, and domain name are as follows:

IP address
Specifies a unique number for a particular system, to identify the system on the TCP/IP network. IP addresses are in the dotted decimal notation format (###.###.##.##).

Host Name
Specifies the host name of a particular system.

Domain Name
Specifies the Domain Name Service (DNS) name, which identifies the system group in the DNS hierarchy.
The host name and the domain name make up the fully qualified domain name of the system. For example, in `example.yourdomain.com`, `example` is the computer name, and it is in the `yourdomain.com` domain.

Specifying the Remote System

When you use TCP/IP, you can specify the remote system in different ways. For example, you can use the following forms:

By host name:
REMOTE_SYSTEM=example

By fully qualified domain name:
REMOTE_SYSTEM=example.yourdomain.com

By IP address:

REMOTE_SYSTEM=###.###.##.##

The IP address is the most efficient method to use when specifying a remote system location.

> **NOTE**
> The preceding examples use the REMOTE_SYSTEM parameter, but this usage also applies to REMOTE_SYSTEM_RF, REMOTE_SYSTEM_SJ, and REMOTE_SYSTEM_SR.

TCP/IP Name Resolution

If you are using a hostname or domain name, your system must have a way to resolve that name to an IP address.

You can use any symbolic name that can be mapped to an IP address, such as a hostname or a domain name. Your system must be set up to resolve the name to an IP address. Check with your network administrator for further information about the use of names in your system.

If you are relying on name resolution to resolve names to IP addresses, this capability must be installed and configured on your system. You can implement name resolution in many ways, including the Domain Name Service (DNS), and the use of host files.

Using TCP/IP Protocols with XCOM Data Transport

To use TCP/IP protocols for an XCOM Data Transport transfer, XCOM Data Transport must have the port specification for the remote system and an indication that you want to use TCP/IP protocols.

The PORT parameter specifies the port. The default value in the `xcom.glb` file should be valid for most remote hosts. However, for transfers that specify an IPv6 remote address and/or the parameter SECURE_SOCKET=YES for secure transfers using OpenSSL, then you must change the PORT parameter accordingly. If you must change the port value of the local system, see the Installing section of this documentation.

The PROTOCOL parameter specifies the protocol to use. Specify this parameter at the command line, in a configuration file, or in the `xcom.glb` file. If the protocol is not specified at the command line, the defaults in the configuration file or in `xcom.glb` are used.

XCOM Data Transport also uses other parameters for TCP/IP functionality. Systems administrators can use these parameters for performance tuning.

Example 1:

This example uses the `xcomtcp` command to queue a file for a send file transfer, using TCP/IP. The configuration file is `myconfig.cnf`, the local file is `testa.aaa`, and the remote file is `testb.bbb`. The remote system is indicated by the host name `goodsys` in the REMOTE_SYSTEM parameter. QUEUE is set to YES, to queue the transfer. TCP/IP is indicated by the TCPIP value in the PROTOCOL parameter. The port is determined by the value of the PORT parameter in the `xcom.glb` file.

```
xcomtcp -c1 -f myconfig.cnf LOCAL_FILE=testa.aaa
REMOTE_FILE=testb.bbb REMOTE_SYSTEM=goodsys
QUEUE=YES PROTOCOL=TCPIP
```

Example 2:

This example uses the `xcomtcp` command to submit a transfer directly, without sending it to the queue, using TCP/IP. The configuration file is `myconfig.cnf`, the local file is `testa.aaa`, and the remote file is `testb.bbb`. The remote system is indicated by the fully qualified domain name `goodsys.goodsite.com` in the REMOTE_SYSTEM parameter. QUEUE is set to NO, so the transfer starts immediately after you press **Enter**. TCP/IP is indicated by the TCPIP value in the PROTOCOL parameter. If the default value in the configuration file or the xcom.glb file is TCPIP, you do not need to specify this value here. The port is indicated by the value 8044 in the PORT parameter.

NOTE

The default value of PORT should be valid for most hosts. Only specify a value for this parameter when needed.

```
xcomtcp -c1 -f myconfig.cnf LOCAL_FILE=testa.aaa
REMOTE_FILE=testb.bbb REMOTE_SYSTEM=goodsys.goodsite.com
QUEUE=NO PROTOCOL=TCPIP PORT=8044
```

Setting Data Transfer Attribute Parameters

Use the parameters in the xcom.cnf file or in any XCOM Data Transport configuration files to configure XCOM Data Transport data transfer attribute parameters. These parameters do not change for the different types of file transfers. We recommend modifying these values in xcom.cnf during the initial configuration of XCOM Data Transport.

The data transfer attribute configuration parameters are shown in this section.

XCOM--COMPRESS

Indicates the compression type. When communicating with an IBM mainframe, if the data file contains any empty lines, COMPRESS can only be set to YES, LZSMALL, LZMEDIUM, or LZLARGE.

NOTE

- Not all compression types are supported on all platforms. For supported compression types, see the partner platform documentation.
- The LZ values enable LZ (Lempel-Ziv) compression to replace sequences of data bytes that occur more than once in a data stream with a code value.

COMPACT

RJE compaction algorithm optimized for uppercase English text.

LCOMPACT

RJE compaction algorithm optimized for lowercase English text.

LZLARGE

Activates LZ compression to search back 32K in the datastream for a matching string.

LZMEDIUM

Activates LZ compression to search back 16K in the datastream for a matching string.

LZRW3

General-purpose algorithm that runs fast and gives reasonable compression.

LZSMALL

Activates LZ compression to search back 4K in the datastream for a matching string.

NO

Indicates no compression.

RLE

Run length encoding.

YES

Indicates run length encoding of binary zeros and blanks only.

ZLIB*n*

Greater compression than LZRW3 but less than LZSMALL, LZMEDIUM, and LZLARGE. The *n* value can be 1 through 9.
If COMPRESS not equal to NO, PACK=LENGTH with a MAXPACK greater-than 31744 then COMPRESS will default to ZLIB2

Default: YES

MAXRECLEN

For Windows, UNIX, and Linux systems, the locally initiating XCOM Data Transport system determines the values for MAXRECLEN, TRUNCATION, and LRECL, for send and receive operations. When the local XCOM Data Transport system initiates a transfer of a text file, this parameter designates the length, in bytes, of the largest record that can be transferred. If a record length is longer than this value, XCOM Data Transport uses the value in the TRUNCATION parameter on the initiating side to determine whether to terminate the transfer or to truncate the record and continue the transfer. When XCOM Data Transport transfers binary files, this value indicates the length of the records that are transferred. On a receive operation, MAXRECLEN is set to whatever the LRECL value is on the initiating side.

Range: 1 to 32767

Default: 1024

TRUNCATION

Indicates whether XCOM Data Transport truncates excess characters in the source file if the record exceeds the maximum record length as indicated by the MAXRECLEN parameter. If NO is selected, and the maximum record length is exceeded, XCOM Data Transport aborts the transfer. This parameter is ignored if CARRIAGE_FLAG=NO.

> **NOTE**
> Truncation is not valid for BINARY data or for non-text data received on the UNIX or Linux platform.

Range: YES or NO

Default: NO

VERSION

Indicates the version of the XCOM Data Transport protocol to be used for this transfer. For TCP/IP, only a value of 2 is valid.

Range: 1 or 2

Default: 2

> **NOTE**
> This is a Version 2 parameter.

Using the Send File Command

Use the Send File command to send a copy of a file on the local system to a file on a remote system.

In the Send File command, the -c1 option specifies that this is a Send File transfer. The local file is indicated by the LOCAL_FILE parameter. The file on the remote system is indicated by the REMOTE_FILE parameter.

> **NOTE**
> Ensure that your user ID complies with the LUSERID limit to send a copy of a file from the local system to a remote system with the **SEND File** command. For more information, see LUSERID.

Example 1

In the following example, the xcom62 command is used. The configuration file is */myconfig.cnf*, the local file is named */testa.aaa*, the remote file is named */testb.bbb*, and the transferred file replaces a file on the remote system that already exists. The PROTOCOL parameter specifies the protocol as SNA. All other necessary parameters are read from the default configuration file, XCOM.CNF.

```
xcom62 -c1 -f /myconfig.cnf LOCAL_FILE=/testa.aaa
```

```
REMOTE_FILE=/testb.bbb FILE_OPTION=REPLACE PROTOCOL=SNA
```

Example 2

In the following example, the xcomtcp command is used. The configuration file is */myconfig.cnf*, the local file is named /*testa.aaa*, the remote file is named /*testb.bbb*, and the transferred file replaces a file on the remote system that already exists. The PROTOCOL parameter specifies the protocol as TCP/IP. All other necessary parameters are read from the default configuration file, XCOM.CNF.

```
xcomtcp -c1 -f /myconfig.cnf LOCAL_FILE=/testa.aaa
REMOTE_FILE=/testb.bbb FILE_OPTION=REPLACE
PROTOCOL=TCPIP
```

The Send File parameters are in the following sections:

CREATE_DIR

Indicates whether to create the specified directory, if the directory does not exist.

YES
Create the directory if it does not exist.

NO
Do not create the directory if it does not exist.

Default: YES

FILE_OPTION

Indicates how the transferred data is to be processed by the receiving system. For file transfers only.

For most file transfers, specify the values for the following parameters:

CREATE
Create a new file on the receiving system.

APPEND
Append the transferred data to an existing file on the receiving system.

REPLACE
Replace an existing file on the receiving system.

For wildcard transfers, the parameter values are as follows:

CREATE
Creates a PDS/Directory and adds the transferred members. If the PDS/Directory already exists, the transfer fails with an error.

APPEND
Adds the transferred members/files. If the PDS/Directory does not exist or the member/file already exists, the transfer fails with an error.

REPLACE
Adds or replaces transferred members/files. If the PDS/Directory does not exist, the transfer fails with error XCOMN0403E Cannot open output file -- No such file or directory.

> **NOTE**
> When creating a file on an IBM mainframe system, some additional information may be necessary. For more information, see the explanations for RECORD_FORMAT, BLKSIZE, VOLUME, and UNIT parameters.

Default: CREATE

LOCAL_FILE

The name of the file on the local system that is being transferred. At the command prompt or in a script, if this variable is null or unset, standard input is read. In this manner, XCOM Data Transport commands can be used in a pipeline or with redirection. All UNIX or Linux file naming conventions apply.

For wildcard transfers, use an asterisk (*) as a file name to indicate that all files within the specified directory are to be transferred. For example, the statement LOCAL_FILE=/NAMES/* indicates that all files under the NAMES directory are to be transferred.

When a prefix is followed by an asterisk, all members beginning with a specific prefix are to be transferred. For example, LOCAL_FILE=/NAMES/AL* requests that files AL, ALEX, and ALICE are all to be transferred. The same rules apply if an asterisk is followed by a suffix.

The actual file name range (not including its path) for wildcard transfers can be between 0 and 71 characters. This also includes the file extension where applicable. File names over 71 characters are truncated. However, when sending files to a mainframe PDS, any file name over eight characters in length is truncated. These systems do not recognize file extensions. For example, a file that is called *longfilename.txt* is truncated to *longfile* and a file called *file.txt* will be truncated to *file*.

> **NOTE**
> If QUEUE=YES, the full path name must be specified.

Range: 0 to 256 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_FILE

Indicates the file on the remote computer to which the transferred data is being written. If you are creating the file (FILE_OPTION=CREATE), the file name must be consistent with the file naming conventions of the remote system. The local XCOM Data Transport system does not validate this name. The remote I/O system determines whether the file name is valid.

For wildcard transfers, use an asterisk (*) as a file name to indicate and to inform the receiving partner that multiple files will be sent.

Example

```
REMOTE_FILE=/PAYROLL/*.
```

If multiple files are sent and the user specifies a file name, all files that are received by the partner are written to that specified file as one single file. An asterisk is used to send files to an IBM mainframe system shows that all files are to be transferred to a partitioned data set (PDS). For platforms that support this functionality, specify a common file extension to append to each file name.

Example

```
REMOTE_FILE=/PAYROLL/*.TXT.
```

> **NOTE**
> For send file transfers only.

Range: 1 to 256 characters

Default: None

REMOTE_SYSTEM

The name of the remote system that receives a file, job, report, or ping request.

For SNA/APPC protocols, the name is as specified in the SNA/APPC configuration of the local system. For RS/6000, the name is the LU6.2 Side Information record.

For TCP/IP protocols, the value can be the remote systems IP address, hostname, or domain name.

Range: 1 to 256 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

Creating a File on an IBM Mainframe

Learn about the parameters to use when you are creating a file from a local system and FILE_OPTION=CREATE.

The IBM mainframe file system, unlike the UNIX or Linux system, may require you to specify various attributes when you to create a file on the mainframe. To determine the appropriate values for these parameters, check with your IBM mainframe applications group.

ALLOCATION_TYPE

The ALLOCATION _TYPE parameter specifies the unit of storage allocation for a data set that is created on an IBM mainframe. This parameter is a Version 2 parameter.

CYL
Specifies cylinders.

TRK
Specifies tracks.

BLK
Specifies blocks.

REC
Specifies records.

Default: CYL

AVGREC

The AVGREC parameter specifies the multiplier to apply to primary and secondary allocation units for a data set that is created on an IBM mainframe, based on the number of records. The record size is based on the value of the LRECL parameter.

This parameter applies only when the SPACE parameter specifies a value of REC. REC indicates that a file is being allocated based on a specific number of records.

The following values are valid:

U
Indicates that the PRIMARY_ALLOC and SECONDARY_ALLOC parameters specify the number of records to allocate for.

K

Indicates that the PRIMARY_ALLOC and SECONDARY_ALLOC parameters specify the number of records in thousands. That is, the value you specify here is multiplied by 1024. For example, specify 3 to indicate 3 K or 3072 records.

M

Indicates that the PRIMARY_ALLOC and SECONDARY_ALLOC parameters specify the number of records in millions. That is, the value you specify here is multiplied by 1048576. For example, specify 2 to indicate 2 MB or 2097152 records.

Default: None

BLKSIZE

The BLKSIZE parameter specifies the block size of a data set that is created on an IBM mainframe. BLKSIZE is used when FILE_OPTION=CREATE.

If the format is…	Then the block size must be…
Fixed or fixed block record	A multiple of the record length
Variable record	Four bytes larger than the record length
Undefined record	Larger than the largest record length

Limits: 0 to 32767

Default: 800

COMPRESS_PDS

The COMPRESS_PDS parameter controls when to compress a partitioned data set (PDS) that is opened for output as the target of an XCOM Data Transport transfer. The XCOM Data Transport for z/OS partner must support this functionality. The value of the CMPRS_PDS_ALLOW parameter in the z/OS default table (XCOMDFLT) or in the destination member (XCOMDFLT) determines whether PDS compression is allowed. When this parameter is set to YES, then PDS compression can occur on the XCOM Data Transport z/OS partner.

NONE

Suppresses the compression.

BEFORE

Compresses the output PDS data set before the transfer begins.

AFTER

Compresses the data set after the transfer has completed.

BOTH

Compresses the data set both before and after the transfer.

Default: NONE

CREATEDELETE

The CREATEDELETE parameter specifies whether an existing data set can be deleted and a new data set can be allocated at the start of a FILEOPT=CREATE transfer. Te XCOM Data Transport z/OS partners must support CREATEDELETE. The CREATEDELETE parameter in the z/OS default table (XCOMDFLT) or in the destination member (XCOMCNTL) that is specified by the z/OS XCOM Data Transport Administrator affects the functionality of this parameter.

- When z/OS has CREATEDELETE=ALLOW:

YES

If FILEOPT=CREATE and the data set exists, then the data set is deleted and a new data set is allocated at the start of the transfer.

NO

If FILEOPT=CREATE and the data set exists, then the transfer fail with a catalog/file error.

- When z/OS has CREATEDELETE=YES:

YES or NO

If FILEOPT=CREATE and the data set exists, then the data set is deleted and a new data set is allocated at the start of the transfer.

- When z/OS has CREATEDELETE=NO:

YES or NO

If FILEOPT=CREATE and the data set exists, then the transfer fails with a catalog/file error.

Default: NO

NOTE

- The attributes of the existing data set are deleted and the new data set is allocated with the attributes that are specified in the transfer when CREATEDELETE=YES.
- CREATEDELETE applies only if the target data set is a sequential data set or an entire PDS/PDSE. CREATEDELETE is ignored for other types of data sets (such as PDS members, PDSE members, VSAM, and USS files).
- CREATEDELETE applies to relative GDGs when the data set is specified using the fully qualified GxxxxVxx name.

DATACLAS

The DATACLAS parameter specifies the data class to use when allocating a new mainframe SMS-managed data set.

Limits: One to eight characters

Default: None

DSNTYPE

The DSNTYPE parameter specifies the data set definition. This parameter applies only to mainframe SMS data sets.

LIBRARY

Defines a PDSE.

PDS

Defines a partitioned data set.

BASIC

Defines a legacy sequential data set.

LARGE

Defines a large format sequential data set.

EXTREQ

Defines an extended format data set.

EXTPREF

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

<blank>

Defines a partitioned or sequential data set based on the data set characteristics entered.

NOTE
These values are IBM standards for SMS processing.

Limits: One to eight characters

Default: None

EATTR

The EATTR parameter specifies whether the data set can have extended attributes when it is allocated on an Extended Address Volume (EAV). This parameter applies only to data sets that are created on an IBM mainframe.

OPT

Specifies that a data set can optionally have extended attributes.

NO

Specifies that a data set cannot have extended attributes.

LRECL

The LRECL parameter specifies the actual or maximum length, in bytes, of a logical record. This parameter corresponds to the JCL LRECL subparameter.

Limits: 0 to 32767

If the format is…	Then the maximum length of a logical record must be equal to the…
Variable blocked record	Maximum record length plus four
Fixed or fixed block record	Constant record length

Default: 160

MGMTCLAS

The MGMTCLAS parameter specifies the management class to use when allocating a new mainframe SMS-managed data set.

Limits: One to eight characters

Default: None

NUM_OF_DIR_BLOCKS

The NUM_OF_DIR_BLOCKS parameter specifies the number of directory blocks to allocate for a data set that is created on an IBM mainframe. This parameter is a Version 2 parameter.

Limits: 0 to 16,777,215

Default: 0

PRIMARY_ALLOC

The PRIMARY_ALLOC parameter specifies the primary storage allocation for a data set that is created on an IBM mainframe. This parameter is a Version 2 parameter.

Limits: 0 to 16,777,215

Default: 1

RECORD_FORMAT

The RECORD_FORMAT parameter specifies the record format of a data set that is created on an IBM mainframe. This parameter corresponds to the JCL RECFM subparameter.

Limits: See the following table.

Value	Description	Record Length	Comment
F	Fixed unblocked	The same length as the data set	
FA	Fixed unblocked ANSI	The same length as the data set	Contains ISO/ANSI/FIPS control characters
FB	Fixed blocked	Fixed	Fixed record length with multiple records per block
FBA	Fixed blocked ANSI	Fixed	Multiple records per block where these records contain ISO/ANSI/FIPS control characters
FBS	Fixed blocked spanned	Fixed	Multiple records per block, written as standard blocks
FM	Fixed unblocked machine	The same length as the data set	Contains machine code control characters
FS	Fixed unblocked spanned	The same length as the data set	Written as standard blocks where these records do not contain any truncated blocks or unfilled tracks
U	Undefined	Undefined	
V	Variable unblocked	Variable	
VA	Variable unblocked ANSI	Variable	Contains ISO/ANSI/FIPS control characters
VB	Variable blocked	Variable	Multiple records per block
VBA	Variable blocked ANSI	Variable	Multiple records per block where these records contain ISO/ANSI/FIPS control characters
VBM	Variable blocked machine	Variable	Multiple records per block where these records contain machine code control characters
VBS	Variable blocked spanned	Variable	May have multiple records per block where these records can span more than one block
VM	Variable unblocked machine	Variable	Contains machine code control characters
VS	Variable unblocked spanned	Variable	A record can span more than one block

Default: VB

SECONDARY_ALLOC

The SECONDARY_ALLOC parameter specifies the secondary storage allocation for a data set that is created on an IBM mainframe. This parameter is a Version 2 parameter.

Limits: 0 to 16,777,215

Default: 0

SPACE

The SPACE parameter specifies the unit of storage allocation for the remote file.

CYL

Specifies cylinders.

TRK

Specifies tracks.

BLK

Specifies blocks.

REC

Specifies records.

Specify by:

- Primary allocation space for the remote file
- Secondary allocation for the remote file
- Directory blocks for the remote file

Default: CYL

STORCLAS

The STORCLAS parameter specifies the storage class to use when creating a mainframe SMS-managed data set.

Limits: One to eight characters

Default: None

UNIT

The UNIT parameter specifies the unit on which to create a data set on an IBM mainframe. This parameter is used when FILE_OPTION=CREATE.

Limits: Zero to six characters

Default: None

VOLUME

The VOLUME parameter specifies the volume on which to create a data set on an IBM mainframe.

Limits: Zero to six characters

Default: None

Tape Parameters for an IBM Mainframe

When IBM Mainframe tape drives are involved in the transfer, use the following parameters at the command prompt and in configuration files. The allowable values for these parameters are the same as for their IBM JCL counterparts except where noted. You can also refer to this section for parameter information when using the xcomtool.

When tape drives are involved in the transfer, some of the preceding parameters, like UNIT and VOLUME, may be required.

XCOM Data Transport (Linux x64, z/Linux, AIX 64, Solaris Sparc 64, Solaris x86 64, and HP-UX IA64) no longer support xcomtool. The xcomtool utility is not shipped with the product.

DEN

The DEN parameter specifies the density to use when creating a tape on the remote system. Valid values are the same as the values for the DEN parameter in JCL.

Limits: 1 to 4

Default: None

EXPDT

The EXPDT parameter specifies the expiration date for the tape data set being created.

yyddd

Specifies the expiration date with a two-digit designation for the year and a three-digit designation for the day of the year. For example, in the expiration date 21021, 11 is the year (namely, 2021) and 021 is the 21st day of that year. The tape data set will expire on January 21, 2021.

yyyy/*ddd*

Specifies an expiration date with a four-digit designation for the year and a three-digit designation for the day of the year. For example, in the expiration date 2021/021, 2021 is the year and 021 is the 21st day of that year. The tape data set will expire on January 21, 2021.

NOTE
EXPDT and RETPD are mutually exclusive; specify one or the other.

LABEL

The LABEL parameter specifies the type of processing to apply to a tape data set.

The valid processing types are AL, AUL, BLP, LTM, NL, NSL, SL, and SUL.

NOTE
XCOM Data Transport for z/OS supports only standard label tapes.

Default: SL

LABELNUM

The LABELNUM parameter specifies the sequence number of the data set on the tape.

Sequence number (0001 through 9999)

Identifies the sequence number of a data set on tape.

Example:

LABELNUM=2

This specification refers to the second data set on the tape.

Default: 0001

RETPD

The RETPD parameter specifies the number of days to retain the tape data set that is being created.

Limits: 1 to 9999

Default: None

> **NOTE**
> RETPD and EXPDT are mutually exclusive; specify one or the other.

TAPE

The TAPE parameter specifies whether the transfer is to a tape volume or to a disk file.

YES

Indicates that the transfer is to a tape volume. Mounts are allowed when dynamic allocation is performed.

NO

Indicates that the transfer is to a disk file.

Default: NO

TAPEDISP

The TAPEDISP parameter specifies the disposition value for MVS tape data sets.

1

Specifies a disposition of new.

2

Specifies a disposition of old.

3

Specifies a disposition of mod.

Default: 1

UNITCT

The UNITCT parameter specifies the number of units to allocate on the remote system. This tape parameter is used when the partner is an IBM mainframe.

Limits: 1 to 20

Default: None

VOLCT

The VOLCT parameter specifies the maximum number of volumes to use when processing a multi-volume output tape data set on the remote system.

Limits: 1 to 255

Default: None

VOLSQ

The VOLSQ parameter specifies the sequence number of the first volume of a multi-volume remote data set to be used.

Limits: 1 to 255

Default: None

Using the Retrieve File Command

Use the Retrieve File command to retrieve a copy of a file from a remote system and write it to a specified file on the local system.

The -c4 option specifies that this is a Retrieve File transfer. The remote file to be copied is indicated by the REMOTE_FILE_RF parameter. The file on the local system is indicated by the LOCAL_FILE_RF parameter.

Example 1

In the following example, the xcom62 command is used. The remote file *customer* is retrieved into the local file */accounts*. All other necessary parameters are read from the default configuration file, xcom.cnf.

```
xcom62 -c4 -f REMOTE_FILE_RF=customer LOCAL_FILE_RF=/accounts
```

Example 2

In the following example, the xcomtcp command is used. The remote file *customer* is retrieved into the local file */accounts*. All other necessary parameters are read from the default configuration file, xcom.cnf.

```
xcomtcp -c4 -f REMOTE_FILE_RF=customer LOCAL_FILE_RF=/accounts
```

The Retrieve File parameters are as follows:

FILE_OPTION_RF

Indicates how the transferred data is to be processed by the receiving system (that is, the local system). Used when the transfer type is Retrieve File. If a value is not specified, then the value of FILE_OPTION is used. If no default is specified in FILE_OPTION or FILE_OPTION_RF, then the value defaults to CREATE.

For most file transfers, the parameter values are as follows:

CREATE
Create a new file on the receiving system.

APPEND
Append the transferred data to an existing file on the receiving system.

REPLACE
Replace an existing file on the receiving system.

For wildcard transfers, the parameter values are as follows:

CREATE
Create the PDS/Directory and add the transferred members. If the PDS/Directory already exists, the transfer fails with an error.

APPEND
Add transferred members/files. If the PDS/Directory does not exist or the member/file already exists, the transfer fails with an error.

REPLACE
Add or replace transferred members/files. If the PDS/Directory does not exist, the transfer fails with error XCOMN0403E Cannot open output file-No such file or directory.

Default: CREATE

LOCAL_FILE_RF

The file name that is created, appended, or replaced on the local system when it receives a file. At the command prompt or in a script, if this value is null or unset, then XCOM Data Transport writes to the stdout. All UNIX or Linux file naming conventions apply.

For wildcard transfers, use an asterisk (*) as a file name to indicate that multiple files will be received.

Example

```
LOCAL_FILE_RF=/PAYROLL/*.
```

If multiple files are received and the user specifies a file name, all files received by the partner are written to that specified file as one single file.

For platforms that support it, you can specify a common file extension to be appended to each file name.

Example

```
LOCAL_FILE_RF=/PAYROLL/*.TXT.
```

NOTE
If QUEUE=YES, the full path name must be specified.

Range: 0 to 256 characters

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_FILE_RF

Indicates the name of the file to be retrieved from the remote system.

For wildcard transfers, use an asterisk (*) as a file name to indicate that all files within the specified PDS/Directory should be transferred. For example, the statement REMOTE_FILE_RF=/NAMES/* indicates that all files under the NAMES directory should be transferred.

When a prefix is followed by an asterisk, all members beginning with a specific prefix are transferred. For example, REMOTE_FILE_RF=/NAMES/AL* requests that files AL, ALEX, and ALICE should be transferred. The same rules apply if an asterisk is followed by a suffix.

The actual file name range (not including its path) for wildcard transfers can be between 0 and 71 characters. This also includes the file extension where applicable. File names over 71 characters are truncated.

NOTE
For retrieve file transfers only.

Range: 1 to 256 characters

Default: None

REMOTE_SYSTEM_RF

The name of the remote system that sends a file on a receive file operation. If no value is specified here, the value in REMOTE_SYSTEM is used.

For SNA/APPC protocols, the name is as specified in the SNA/APPC configuration of the local system. For RS/6000, this is the LU6.2 Side Information record.

For TCP/IP protocols, the value can be the remote system's IP address, host name, or domain name.

Range: 1 to 256 characters

Default: None

UNIT_RF

Specifies the unit from which a data set is to be retrieved from an IBM mainframe.

Range: Zero to six characters

Default: None

VOLUME_RF

Specifies the volume from which a data set is to be retrieved from an IBM mainframe.

Range: Zero to six characters

Default: None

Using the Send Report Command

Use the Send Report command to send a copy of a report file from a local system to a remote system for printing.

The -c2 option specifies that this is a Send Report transfer. The report file on the local system is specified by the LOCAL_FILE_SR parameter.

Example 1:

In the following example, the xcom62 command is used. The configuration file is */myconfig.cnf*, the local report file sent is named */tmp/myfile.rpt*, and it is placed on HOLD status on the remote system until released by the remote system.

```
xcom62 -c2 -f /myconfig.cnf LOCAL_FILE_SR=/tmp/myfile.rpt
HOLDFLAG=YES
```

Example 2:

In the following example, the xcomtcp command is used. The configuration file is */myconfig.cnf*, the local report file sent is named*/tmp/myfile.rpt*, and it is placed on HOLD status on the remote system until released by the remote system.

```
xcomtcp -c2 -f /myconfig.cnf LOCAL_FILE_SR=/tmp/myfile.rpt
HOLDFLAG=YES
```

The Send Report parameters are as follows:

CARRIAGE_CONTROL_CHARACTERS

Indicates the type of printer carriage-control codes, if any, that are included in the report file.

> **NOTE**
> For report transfers only.

ASA

ASA control codes in column 1.

> **NOTE**
> If CARRIAGE_CONTROL_CHARACTERS=ASA, then digest generation is not supported.

IBM

IBM Machine Characters (valid only for IBM mainframes).

BYPASSASA
If data is already in ASA format, bypass conversion.

OTHER
No carriage-control codes are used.

Default: OTHER

CLASS

The print class assigned to a report transferred to a remote system.

If the remote system is an IBM mainframe, this field designates the JES SYSOUT class.

> **NOTE**
> For report transfers only.

Example:

Enter **B** to print the report through SYSOUT=B.

Range: 1 character

Default: None

COPIES

The number of copies that are to be sent. If this parameter is not specified, the remote system queues one copy of the report to the system's default printer. For report transfers only.

Range: 1 to 999

Default: 1

DESTINATION

Identifies the printer or other device on the remote system where the report is to be sent. If this parameter is not specified, the remote system sends the report to the system's default printer. For report transfers only.

0 to 16 characters
For indirect transfers and for Version 1.

0 to 21 characters
For transfers that are not indirect and for Version 2.

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

DISPOSITION

Indicates what the remote system does with the report file after the report has been printed. For report transfers only.

> **NOTE**
> This field is not used when the remote system is an IBM mainframe.

DELETE
After printing the report is deleted.

KEEP
After printing the report is kept.

HOLD

After printing the report is held.

Default: DELETE

FCB

Identifies the FCB JCL parameter when sending the report file to an IBM mainframe, defining print density, lines per page, and so on. For report transfers only.

Range: Zero to four characters

Default: None

FORM

The type of form that should be used to print the report. Because XCOM Data Transport places the print job in the remote system's print queue, the print control functions depend on the remote system. The user must verify beforehand that the requested form is available at the remote site. For report transfers only.

> **NOTE**
> When sending a report to a VAX computer, leave this parameter blank unless you are certain that this is a valid form type. VMS interprets this to mean that no special form is being requested.

Range: 0 to 10 characters

Default: None

HOLDFLAG

Indicates whether a transferred report file is to be placed on HOLD on the remote system or is to be printed immediately. For report transfers only.

Range: YES or NO

Default: NO

LOCAL_FILE_SR

Local file name to be sent as a report to the remote system. If this value is null or unset, then XCOM Data Transport reads the standard input file. For report transfers only.

> **NOTE**
> If QUEUE=YES, user must specify full path name.

Range: 0 to 256 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_SYSTEM_SR

The name of the remote system to which a report is sent. If no value is specified here, the value in REMOTE_SYSTEM is used.

For SNA/APPC protocols, the name is as specified in the SNA/APPC configuration of the local system. For RS/6000, this is the LU6.2 Side Information record.

For TCP/IP protocols, the value can be the remote system's IP address, host name, or domain name.

Range: 1 to 256 characters

Default: None

REPORT_TITLE

This field contains the report title and job number that will be printed on the report. The field has the following format:

8 characters	2 characters	8 characters
Job Name	Blanks	Job Number

The job is optional and can be skipped. The job name can also be skipped, but if you skip the job name and want to use the job number, you must pad the number with 10 blanks.

> **NOTE**
> For report transfers only.

Examples

```
REPORT_TITLE="Salary94  Job12345"
REPORT_TITLE="          Job23456"
```

Non-example

```
REPORT_TITLE="     Job34567"
```

This is not a valid REPORT_TITLE because the job number spans both subfields.

This parameter is used by XCOM Data Transport on remote systems in the following ways:

System	Uses the REPORT_TITLE…
z/OS	To interpret a non-blank value in this field as specifying the generation of a separator (banner) page for this value.
VAX/VMS	To print with the report.
UNIX/Linux	To allow XCOM Data Transport to pass this field to the LP spooler as a title field.
Other systems	As a descriptive comment only and does not print it as part of the report.

Range: 0 to 21 alphanumeric or blank characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

SPOOL_FLAG

Indicates whether the report is to be spooled to disk or printed immediately. For report transfers only.

> **NOTE**
> If the remote system is an IBM mainframe, this parameter has no effect on the transfer.

Range: YES or NO

Default: YES

Support for IBM Machine Code Characters

IBM Machine Code characters will be converted to the proper combination of form feed and line feed characters when the report is received from a machine code data set on the mainframe, such as FBM, or the partner specifies that the report has machine code controls.

Using the Send Job Command

Use the Send Job Command to send an executable file to a remote system to perform a job.

The -c3 option specifies that this is a Send Job transfer. The local job file is specified by the LOCAL_FILE_SJ parameter.

The executable file or command script that contains the job must have in it the control statements that are needed to execute the job on the remote system. These control statements must be in a form that is recognized by the remote system. For example, if you are sending to an IBM z/OS system, this file could contain JCL statements. If you are sending to a UNIX or Linux system, this file could contain a shell script. For sending to a Windows system, the file would be a DOS .bat file.

Example 1

In the following example, the xcom62 command is used. The job sent for execution on the remote system is in the file */tmp/myfile*.

```
xcom62 -c3 -f LOCAL_FILE_SJ=/tmp/myfile
```

All other necessary parameters are read from the default configuration file xcom.cnf.

Example 2

In the following example, the xcomtcp command is used. The job sent for execution on the remote system is in the file */tmp/myfile*.

```
xcomtcp -c3 -f LOCAL_FILE_SJ=/tmp/myfile
```

All other necessary parameters are read from the default configuration file xcom.cnf.

The Send Job parameters are as follows:

LOCAL_FILE_SJ

Indicates the name of the file on the local system to be sent as a job. All the UNIX or Linux file naming conventions apply. If this value is null or unset, then XCOM Data Transport reads the standard input file.

> **NOTE**
> If QUEUE=YES, user must specify full path name.

Range: 0 to 256 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_SYSTEM_SJ

The name of the remote system to which a job is sent. If no value is specified here, the value in REMOTE_SYSTEM is used.

For SNA/APPC protocols, the name is as specified in the SNA/APPC configuration of the local system. For RS/6000, this is the LU6.2 Side Information record.

For TCP/IP protocols, the value can be the remote system's IP address, host name, or domain name.

Range: 1 to 256 characters

Default: None

Sending a Metatransfer

Use this command at the command prompt to send a metatransfer from a local system to a remote system.

> **NOTE**
> If the remote system is z/OS, it can be running release r11 or higher of XCOM Data Transport..

The **-c5** option specifies a metatransfer. The INQ_FILE command line parameter specifies the file containing information about the transfers being performed in this request.

Example:

```
XCOMTCP -c5 -f <CNF or XML configuration file> INQ_FILE=MYSCHEDULE STCIP=?? STCPORT=?? LUSERID=?? LPASSWORD=??
 LDOMAIN=?? SECURE_SCHEDULE=<Y or N> STCTRNENCRL_CIPHER=??
```

This following are the metatransfer (-c5) parameters:

-f <configfilename>

The name of the CNF or XML configuration file name for a metatransfer.

Range: 1 to 256 characters

Default: None

INQ_FILE

Specifies the complete path information of the file to contain the information required when doing an inquire metatransfer (-c6) about a metatransfer (-c5).

Range: 1 to 256 characters

Default: None

LDOMAIN

The domain associated with LUSERID and LPASSWORD, if the target system is Windows, when handling a metatransfer (-c5).

Range: 1 to 15 characters

Default: None

LPASSWORD

Specifies the password of the local user to be validated on the XCOM Data Transport server handling a metatransfer (-c5).

Range: 1 to 31 characters

Default: None

LUSERID

The user ID to use on the XCOM Data Transport system receiving a -c5 transfer.

Range: 1 to 12 characters

Default: None

SECURE_SCHEDULE

Specifies whether the -c5, -c6, or -c7 metatransfer uses TLS/SSL.

Y

The metatransfer uses TLS/SSL.

N

The metatransfer does not use TLS/SSL.

Default: N

STCIP

Specifies the IP address or name of the XCOM Data Transport server to handle the -c5, -c6, or -c7 metatransfer request.

Range: 1 to 64 characters

Default: None

STCPORT

Specifies the TCP/IP port number of the XCOM Data Transport server to handle the -c5, -c6, or -c7 metatransfer request.

Range: 1 to 65535

Default: 8044

STCTRNENCRL_CIPHER

For more information, see the Parameters section.

Sending a Ping Request

The ping command determines whether an XCOM Data Transport server is available.

Use the ping command at the command prompt of your local system to determine the availability of the remote system.

When this command is issued, information about the remote system is returned. See the following example:

```
XCOMTCP -ping REMOTE_SYSTEM=?? {PORT=??} {SECURE_SOCKET=YES|NO} {TRNENCRL_CIPHER=xx}
```

The ping command has the following parameters:

REMOTE_SYSTEM

Specifies the name of the remote system to receive the ping request.
For SNA/APPC protocols, the name is the symbolic destination name from the information record on the CPI C side.
For TCP/IP protocols, the name can be the IP address, host name, or domain name of the remote system.
Range: 1 through 64 characters
Default: None

PORT

Specifies the TCP/IP port number of the remote XCOM Data Transport server. This parameter is used only for TCP/IP transfers.
Range: 1 through 65535
Default: 8044

SECURE_SOCKET

Specifies whether to perform a secure transfer with an OpenSSL socket:

YES

Performs a secure transfer. The transfer uses an OpenSSL socket and connects to a TLS or an SSL listener on the remote system.

NO

Performs a non-secure transfer. The transfer uses a non-OpenSSL socket.

Default: NO

TRNENCRL_CIPHER

Specifies the list of ciphers to use when encrypting the password fields for locally initiated transfers. Separate the ciphers with colons.
An exclamation point (!) or hyphen (-) can precede each cipher in the list. If an exclamation point is used, the ciphers are permanently deleted from the list. The deleted ciphers can never reappear in the list even when they are explicitly stated. If a hyphen is used, the ciphers are deleted from the list, but some or all ciphers can be added again using later options.
Default: COMPAT
The following table shows the valid values for this parameter:

Value	**Comments**
ALL	ALL ciphers: `AES:3DES:RC4:RC2:DES:XCOM` ALL does NOT include the COMPAT value.
DES	All DES ciphers: `DES-CBC:DES-ECB:DES-CFB:DES-OFB`
DES-CBC	DES cipher with cipher-block chaining
DES-ECB	DES cipher with electronic codebook
DES-CFB	DES cipher with cipher feedback
DEC-OFB	DES cipher with output feedback
3DES	All 3DES ciphers: `3DES-CBC:3DES-ECB:3DES-CFB:3DES-OFB`
3DES-CBC	3DES cipher with cipher-block chaining
3DES-ECB	3DES cipher with electronic codebook
3DES-CFB	3DES cipher with cipher feedback
3DES-OFB	3DES cipher with output feedback
AES	All AES ciphers: `AES256-CBC:AES256-ECB:AES256-CFB:AES256-OFB:AES192-CBC:AES192-ECB:AES192-CFB:AES192-OFB:AES128-CBC:AES128-ECB:AES128-CFB:AES128-OFB`
AES128	All AES 128-bit ciphers: `AES128-CBC:AES128-ECB:AES128-CFB:AES128-OFB`
AES128-CBC	AES 128-bit cipher with cipher-block chaining
AES128-ECB	AES 128-bit cipher with electronic codebook
AES128-CFB	AES 128-bit cipher with cipher feedback

AES128-OFB	AES 128-bit cipher with output feedback
AES192	All AES 192-bit ciphers: `AES192-CBC:AES192-ECB:AES192-CFB:AES192-OFB`
AES192-CBC	AES 192-bit cipher with cipher-block chaining
AES192-ECB	AES 192-bit cipher with electronic codebook
AES192-CFB	AES 192-bit cipher with cipher feedback
AES192-OFB	AES 192-bit cipher with output feedback
AES256	All AES 256-bit ciphers: `AES256-CBC:AES256-ECB:AES256-CFB:AES256-OFB`
AES256-CBC	AES 256-bit cipher with cipher-block chaining
AES256-ECB	AES 256-bit cipher with electronic codebook
AES256-CFB	AES 256-bit cipher with cipher feedback
AES256-OFB	AES 256-bit cipher with output feedback
RC2	All RC2 ciphers: `RC2-CBC:RC2-ECB:RC2-CFB:RC2-OFB`
RC2-CBC	RC2 cipher with cipher-block chaining
RC2-ECB	RC2 cipher with electronic codebook
RC2-CFB	RC2 cipher with cipher feedback
RC2-OFB	RC2 cipher with output feedback
RC4	RC4 cipher
XCOM	XCOM Data Transport proprietary cipher.
COMPAT	This value is required for transfers that are sent to earlier XCOM Data Transport for z/OS releases. This value also permits the XCOM Data Transport proprietary cipher without the cipher negotiation that is required for backward password compatibility with XCOM Data Transport releases before 11.6.

Examples:

To request all ciphers except for any of the DES ciphers, use the following command:

```
TRNENCRL_CIPHER=ALL:!DES
```

To request only a 3DES or AES cipher, use the following command:

```
TRNENCRL_CIPHER=3DES:AES
```

To disable the cipher negotiation or remain backward compatible with the earlier releases of XCOM Data Transport, use the following command:

```
TRNENCRL_CIPHER=COMPAT
```

Example of Ping Command and Results

The following example shows a sample ping command and the results that are returned.

```
xcomtcp -ping REMOTE_SYSTEM=XX PORT=XX TRNENCRL_CIPHER=ALL
XCOMN0882I PING INFO FOR <SYSTEMNAME>
XCOMN0882I RELEASE=<RELEASE> SP00 GEN LEVEL <LEVEL> SYSTEM NAME=<SYSNAME> SYSTEM ID=<SYSID>
XCOMU0882I NEGOTIATED CIPHER=XCOM
```

Inquiring the Status of Metatransfers

Use this command at the command prompt to inquire on the status of metatransfers.

The **-c6** option specifies that this is a metatransfer status inquiry. The file containing information about the transfers being performed by the - c5 metatransfer request is specified by the INQ_FILE command line parameter.

Example

```
XCOMTCP -C6 STCIP=?? STCPORT=?? ING_FILE=?? ING_WAIT=?? HIST_FILE=??
```

The following sections describe the parameters for inquiring on the status of metatransfers.

HIST_FILE

Specifies the complete path information of the file to contain the history records returned by an inquire metatransfer (-c6) or a get history record retrieval metatransfer (-c7).

Range: 0 to 256 characters

Default: None

INQ_FILE

Specifies the complete path information of the file to contain the information required when doing an inquire metatransfer (-c6) about a metatransfer (-c5).

Range: 1 to 256 characters

Default: None

INQ_WAIT

Specifies how long XCOM Data Transport should wait for the metatransfer (-c5) to complete when doing an inquire metatransfer (-c6).

`hhmmss`

Specifies in hours (*hh*), minutes (*mm*), and seconds (*ss*) the length of the time that XCOM Data Transport should wait for the metatransfer request to complete.

Default: 001000 (10 minutes)

NOTE
This parameter's value is expressed as a number of up to six digits (for example, 010000 for 1 hour).

STCIP

Specifies the IP address or name of the XCOM Data Transport server to handle the -c5, -c6, or -c7 metatransfer request.

Range: 1 to 64 characters

Default: None

STCIP

Specifies the IP address or name of the XCOM Data Transport server to handle the -c5, -c6, or -c7 metatransfer request.

Range: 1 to 64 characters

Default: None

STCPORT

Specifies the TCP/IP port number of the XCOM Data Transport server to handle the -c5, -c6, or -c7 metatransfer request.

Range: 1 to 65535

Default: 8044

Return Codes

The inquire return codes are as follows:

0

All transfers in INQ_FILE have been completed.

4

INQ_FILE is empty.

8

OPEN failed for INQ_FILE or HIST_FILE.

36

At least one transfer in the INQ_FILE has not been completed.

40

One or more transfers specified in the INQ_FILE did not complete successfully.

44

At least one transfer in the INQ_FILE has not been completed *and* one or more transfers in the INQ_FILE did not complete successfully.

48

No history records were returned.
This condition can be caused by the STCIP= parameter on the - c6 (inquire) metatransfer pointing to a different XCOM Data Transport system than the STCIP= on the - c5 metatransfer request. In other words that XCOM Data Transport system is the wrong XCOM Data Transport system.

Retrieving History Records

Use the command, as shown in the example, at the command prompt to retrieve history records.

This command accepts a history filter file in XML format or CNF format and creates a file of the matching history records in a format that a [set the easy variable for your book] report can be generated from.

The **-c7** option specifies that this is a history transaction. The history filter file is generated in XML format by using the Export button on the Get History Records page of the GUI or is created in a CNF configuration file format by using the supported history record filter parameters. For a list of the parameters that can be used in a CNF configuration file for a -c7 metatransfer, see .

NOTE
For more information about the history filter fields in the GUI, see the *XCOM Data Transport GUI Online Help*.

Example:

```
XCOMTCP -c7 -f <configfilename> STCIP=<ip-value> STCPORT=<port-value> SECURE_SCHEDULE=<YES | NO>
 HIST_FILE=<history-file-name>
```

This following sections describe the history retrieval (-c7) parameters:

-f <configfilename>

The name of the CNF or XML configuration file name for a metatransfer.

Range: 1 to 256 characters

Default: None

SECURE_SCHEDULE

Specifies whether the -c5, -c6, or -c7 metatransfer uses TLS/SSL.

Y

The metatransfer uses TLS/SSL.

N

The metatransfer does not use TLS/SSL.

Default: N

STCIP

Specifies the IP address or name of the XCOM Data Transport server to handle the -c5, -c6, or -c7 metatransfer request.

Range: 1 to 64 characters

Default: None

STCPORT

Specifies the TCP/IP port number of the XCOM Data Transport server to handle the -c5, -c6, or -c7 metatransfer request.

Range: 1 to 65535

Default: 8044

History Record Filter Parameters

The following history record filter parameters can be used with a - c7 metatransfer in XCOM Data Transport.

OEDATE

Limits the history request to file transfers that were scheduled or completed on or before the specified end date.

YYYYMMDD

Specifies the end date:

YYYY

Specifies the four-digit year.

MM

Specifies the two-digit number of the month, as follows:

01 = January	02 = February	03 = March
04 = April	05 = May	06 = June
07 = July	08 = August	09 = September
10 = October	11 = November	12 = December

DD

Specifies the two-digit day of the month (01 through 31).

Default: Current date

NOTE

- OEDATE and OETIME specify an end date and time. These values limit the history request to file transfers that were scheduled or completed on or before the specified date and time.
- See OSDATE and OSTIME for the start date and time.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OETIME

Limits the history request to file transfers that were scheduled or completed on or before the specified end time.

HHMMSS

Specifies the end time.

HH

Specifies the two-digit hour (00 through 23).

MM

Specifies the two-digit minute (00 through 59).

SS

Specifies the two-digit second (00 through 59).

Default: 235959

NOTE

- OEDATE and OETIME specify an end date and time. These values limit the history request to file transfers that were scheduled or completed on or before the specified date and time.
- See OSDATE and OSTIME for the start date and time.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OFILE

Specifies the file name, local, or remote, to match for a history request.

You can use the following wildcard characters when you specify the file name:

*** or %**

Represents a string of zero or more characters.

_

Represents any single character.

Range: 1 to 256

Default: None

Example: An OFILE value of %MASTER.FIL_.G* locates a file with the following attributes:

- Starting with anything
  - Ending with anything
  - With the characters MASTER.FIL in the name, followed by any single character and .G.

NOTE

- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OFILECASE

Specifies whether the specified file name (OFILE parameter) search is case-sensitive.

YES

Specifies that the value is case-sensitive.

NO

Specifies that the value is not case-sensitive.

Default: NO

NOTE

- For a case-sensitive search, case-sensitive collation is set to the history database table. Such collation can also be set to the ‘file’ and ‘lfile’ columns in the history database table.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OFILETYPE

Limits the history request to transfers with the specified FILETYPE.

JOB

Restricts the search to only FILETYPE JOB transfers.

REPORT

Restricts the search to only FILETYPE REPORT transfers.

FILE

Restricts the search to only FILETYPE FILE transfers.

Default: All file types

NOTE

- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer are authorized to access.

OFLMAX

Limits the history request to file transfers where the number of bytes transferred is equal to or less than the specified value.

NNNNNNNNN(*N*|*X*)

Specifies a 1-digit to 10-digit number. The last digit can be another numeric digit or a 1-character qualifier. This parameter restricts the search to file transfers where the number of bytes transferred is equal to or less than the specified value.

X

Specifies one of the following qualifiers (default B):

- B = Bytes
- K = Kilobytes
- M = Megabytes
- G = Gigabytes
- T = Terabytes (maximum allowed value is 8388607T)
- P = Petabytes (maximum allowed value is 8191P)
- E = Exabytes (maximum allowed value is 7E)

Default: 7E

NOTE

- Use OFLMIN and OFLMAX to specify a range that limits the history request by the number of bytes transferred.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OFLMIN

Limits the history request to file transfers where the number of bytes transferred is equal to or greater than the specified value.

NNNNNNNNNN(*N*|*X*)

Specifies a 1-digit to 10-digit number. The last digit can be another numeric digit or a 1-character qualifier. This parameter restricts the search to file transfers where the number of bytes transferred is equal to or less than the specified value.

X

Specifies one of the following qualifiers (default B):

- B = Bytes
- K = Kilobytes
- M = Megabytes
- G = Gigabytes
- T = Terabytes (maximum allowed value is 8388607T)
- P = Petabytes (maximum allowed value is 8191P)
- E = Exabytes (maximum allowed value is 7E)

Default: None

NOTE

- Use OFLMIN and OFLMAX to specify a range that limits the history request by the number of bytes transferred.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OID

Limits the history request to file transfers with a specific transfer ID. The transfer ID is a user-defined identifier for file transfer requests.

xxxxxxxxxx

Specifies a 1-character to 10-character transfer ID.

Default: None

NOTE

- The wildcard character, *, can be used for this parameter only when it is specified as the last character.
- This parameter is not case-sensitive. Using this parameter to filter history records returns case-insensitive results. For example, specifying **TRANSFER01** returns the same results as specifying **transfer01**.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OJOB

Specifies the invoking job name to match for a history request.

Range: 1 to 8 characters

Default: None

NOTE

- Supported in - c7 CNF metatransfers only.
- The wildcard character, *, can be used for this parameter only when it is specified as the last character.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OINIT

Limits the history request to locally initiated transfers or remotely initiated transfers.

LOCAL or L
Restricts the search to locally initiated transfers.

REMOTE or R
Restricts the search to remotely initiated transfers.

Default: All local and remote transfers

NOTE

- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OLIMIT

Sets the maximum number of history records that can be returned.

Range: 0 to 9999

NOTE

- OLIMIT=0 means that all records are returned.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OLMSG

Limits the history request by the last message of the transfer. Use the following format for XCOM Data Transport messages:

XCOM*XNNNNS*
Specifies a 1-character to 10-character name that restricts the search to file transfers where the last message matches the specified value.

XCOM
Indicates that the message is from XCOM Data Transport.

X
Identifies the system.

NNNN

Is the message number.

S

Is the message severity:

- I = Informational
- W = Warning
- E = Error

Default: None

NOTE

- The wildcard character, *, can be used for this parameter only when it is specified as the last character.
- This parameter is case-insensitive.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OLOCATN

Limits the history request by transfer location.

DATABASE

Restricts the search to file transfers that are stored in the database. This option applies only when a history database has been set up at your site.

QUEUE

Restricts the search to file transfers that are still in the queue.

Default: Transfers in both the QUEUE and DATABASE

NOTE

- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OLU

Limits the history request to file transfers with a specific remote LU name.

xxxxxxxx

Specifies a one-character to eight-character LU name.

Default: None

NOTE

- The wildcard character (*) can be used for this parameter only when it is specified as the last character.
- This parameter is not case-sensitive. Using this parameter to filter history records returns case-insensitive results. For example, specifying **LU01** returns the same results as specifying **lu01**.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OREQ

Limits the history request to file transfers that contain the specified request number.

NNNNNN

Specifies a one-character to six-character request number.

Default: All request numbers

NOTE

- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OSDATE

Limits the history request to file transfers that were scheduled or completed on or after the specified start date.

YYYYMMDD

Specifies the start date.

YYYY

Specifies the four-digit year.

MM

Specifies the two-digit number of the month, as follows:

```
01 = January        02 = February         03 = March
04 = April          05 = May              06 = June
07 = July           08 = August           09 = September
10 = October        11 = November         12 = December
```

DD

Specifies the two-digit day of the month (01 through 31).

Default: Current date

NOTE

- OSDATE and OSTIME specify a start date and time. These values limit the history request to file transfers that were scheduled or completed on or after the specified date and time.
- See OEDATE and OETIME for the end date and time.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OSTIME

Limits the history request to file transfers that were scheduled or completed on or after the start time.

HHMMSS

Specifies the start time.

HH

Specifies the two-digit hour (00 through 23).

MM

Specifies the two-digit minute (00 through 59).

SS

Specifies the two-digit second (00 through 59).

Default: 000000

NOTE

- OSDATE and OSTIME specify a start date and time. These values limit the history request to file transfers that were scheduled or completed on or after the specified date and time.
- See OEDATE and OETIME for the end date and time.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OSYSID

Limits the history request to file transfers that were executed on the specified system ID.

NOTE
OSYSNAME and OSYSID together uniquely identify an XCOM Data Transport server that is r11.5 or higher.

Range: 1 to 4 characters

Default: None

NOTE

- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OSYSNAME

Limits the history request to file transfers that were executed on the specified system name.

NOTE
OSYSNAME and OSYSID together uniquely identify an XCOM Data Transport server that is r11.5 or higher.

Range: 1 to 8 characters

Default: None

NOTE

- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OTNAME

Limits the history request to file transfers with the specified remote TCP/IP name or TCP/IP address.

xxxxxxxx...x
Specifies a 1-character to 64-character TCP/IP name or address.

Default: None

NOTE

- The wildcard character (*) can be used for this parameter only when it is specified as the last character.
- This parameter is not case-sensitive. Using this parameter to filter history records returns case-insensitive results. For example, specifying **TCPIP01** returns the same results as specifying **tcpip01**.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OTYPE

Specifies the type of history records to retrieve:

I

Retrieves history records for file transfers whose execution is still pending.

A

Retrieves history records for file transfers that are in progress.

C

Retrieves history records for file transfers that have been successfully or unsuccessfully completed.

ALL|AIC|*

Retrieves history records for all file transfers, independent of their status.

Default: AIC

NOTE

- You can also specify values in combination; for example, specify AI to request history records for file transfers whose execution status is inactive and active.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OTYPEREQ

Limits the history request to send transfers or receive transfers.

SEND or S

Restricts the search to send transfers.

RECEIVE or R

Restricts the search to receive transfers.

Default: All send and receive transfers

NOTE

- This parameter is case-sensitive.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OUSER

Limits the history request to file transfers that are submitted by a specific user.

xxxxxxxxxxxx

Specifies a 1-character to 12-character user name.

Default: None

NOTE

- The wildcard character (*) can be used for this parameter only when it is specified as the last character.
- This parameter is not case-sensitive. Using this parameter to filter history records returns case-insensitive results. For example, specifying **USER01** returns the same results as specifying **user01**.
- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OVOL

Specifies the volser (local or remote) to match for a history request.

Range: 1 to 6 characters

Default: None

NOTE

- Supported in - c7 CNF metatransfers only.
- The defined users in groups XCOMADM and XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.
- The wildcard character, *, can be used for this parameter only when it is specified as the last character.

Tracing Problems

Learn how to use the trace facility to trace transfer problems.

The trace facility collects information about a transfer from the point where it is queued or submitted and a connection is established. The generated trace data is stored in a file that is named *tidnumber*.tra, where *tidnumber* is the transfer ID.

The tracing facility is used to collect information at the request of Broadcom Support to diagnose a problem. Broadcom Support requests the trace level that you should use. The `-t` option of the `xcom62` or `xcomtcp` command directs the trace output to STDERR, the standard error output.

Example 1

In the following example, the xcom62 command is used. The configuration file is */myconfig*, the XTRACE parameter sets the trace level to 10 to provide full tracing information, and the trace information is written to a trace file. The protocol is specified by the PROTOCOL parameter.

```
xcom62  - c1 -f /myconfig XTRACE=10 PROTOCOL=SNA
```

Example 2

In the following example, the xcomtcp command is used. The configuration file is */myconfig*, the XTRACE parameter sets the trace level to 10 to provide full tracing information, and the trace information is written to a trace file. The protocol is specified by the PROTOCOL parameter.

```
xcomtcp  - c1 -f /myconfig XTRACE=10 PROTOCOL=TCPIP
```

Return Codes

For queued transfers, the return code indicates success or failure in queuing the transfer.

For non-queued transfers the return codes from XCOMTCP and XCOM62 are as follows:

0

Indicates a successful transfer.

Non-zero

Indicates a failed transfer.

The XCOM Data Transport return code is the same as the error message number, which is usually a three-digit number. To pass back the error message number, XCOM Data Transport must make sure that the error code is 256 or less, because of system restrictions. To do so, XCOM Data Transport subtracts 256 from the error message number if it is greater than 256. In other words, the return code is modulo 256.

Example:

Message XCOMU0298E would give 42 as the return code (298 - 256 = 42).

Refer to $XCOM_HOME/api/include/xcomerr.h. These return codes are not modulo 256.

> **NOTE**
> $XCOM_HOME is an environment variable.

Setting Up Log Files

XCOM Data Transport automatically collects information about transfers and stores it in a log file. Read the log file to see information about transfers. The log indicates information such as when a transfer was started, stopped, completed, deleted, suspended, or aged off of the queue. For each transfer, it shows the time, date, transfer ID and number of blocks and bytes transmitted.

Example 1

In the following example, the xcom62 command is used. The configuration file is /tmp/myconfig and the log file is */home/phil/xcom/mylog*.

```
xcom62  - c1 -f /tmp/myconfig XLOGFILE=/home/phil/xcom/mylog
```

Example 2

In the following example, the xcomtcp command is used. The configuration file is */tmp/myconfig* and the log file is */home/phil/xcom/mylog*.

```
xcomtcp  - c1 -f /tmp/myconfig XLOGFILE=/home/phil/xcom/mylog
```

XLOGFILE

The name of the file where XCOM Data Transport logs activity. If you do not specify this parameter, the systemwide log file $XCOM_HOME/xcom.log is used. If you specify this parameter with a different file name, the logging information is only sent to the specified file.

> **NOTE**
> - If QUEUE=YES, specify the full path name.
> - $XCOM_HOME is an environment variable.

Range: 0 to 256 characters

Setting Up File Type Conversion

XCOM Data Transport uses internal character conversion sets to handle different file formats when files are sent or received. These default conversion sets are stored in files as conversion tables. When executing a non-binary file transfer that requires an ASCII/EBCDIC conversion, XCOM Data Transport uses these tables by default (INTERNAL_CONVERSION_TABLES=YES).

The default conversion tables are as follows:

ASCII to EBCDIC

$XCOM_HOME/convtab/atoe.tab

EBCDIC to ASCII

$XCOM_HOME/convtab/etoa.tab

> **NOTE**
> $XCOM_HOME is an environment variable.

For conversions, when transferring text files for transfers that are initiated on a UNIX or Linux system, CARRIAGE_FLAG is set to YES and CODE_FLAG is set to EBCDIC.

Example 1

In the following example, the xcom62 command is used. An EBCDIC text file is being transferred:

```
xcom62  - c1 -f CARRIAGE_FLAG=YES CODE_FLAG=EBCDIC
```

All other necessary parameters are read from the default configuration file, xcom.cnf.

Example 2

In the following example, the xcomtcp command is used. An EBCDIC text file is being transferred:

```
xcomtcp  - c1 -f CARRIAGE_FLAG=YES CODE_FLAG=EBCDIC
```

All other necessary parameters are read from the default configuration file, xcom.cnf.

Parameters

Use the parameters shown below to specify the characteristics of the file and the data involved in the transfer. For information on using custom character conversion sets rather than the XCOM Data Transport defaults, see the topic, Creating Custom Character Sets for File Conversion.

The file type parameters are listed next.

CARRIAGE_FLAG

Specifies the type of file being transferred and some special characteristics of the conversion done during the transfer.

YES

Indicates that the transferred file is a text file and a newline character should be added to the end of incoming records. Also, newline characters are removed from the ends of lines before an outgoing record is sent.

NO

Indicates no special processing.

MPACK

Indicates a text file with record packing. Uses 2K pack buffer.

VLR

Indicates a binary file of variable-length records with a field of four bytes preceding each record. Applies to locally initiated transfers only.

XPACK

Indicates a text file with record packing. Uses 31K pack buffer.

> **NOTE**
> MPACK does not support a MAXRECLEN (actual record length) over 2K. XPACK does not support a MAXRECLEN (actual record length) over 31K.

Default: YES

CODE_FLAG

Used to identify the type of data being transferred.

ASCII

An ASCII file is being transferred. This transfer indicates that the incoming file is assumed to be ASCII format, and is not translated. Therefore the file on the remote system has to be in ASCII format before it is transferred.

BINARY

A binary file, such as an executable file, is being transferred. This file indicates to a remote system that it is not to translate the data it is exchanging with your system.

EBCDIC

An EBCDIC file is being transferred. The data is translated from EBCDIC to ASCII when the local system receives the data. From ASCII to EBCDIC, when the local system sends the data.

UTF8

A Unicode file which is based on the UTF8 encoding system is being transferred. When this option is set, the LOCAL_CHARSET/ REMOTE_CHARSET parameters are used to identify the character encoding of the input file and the required encoding for the output file.

UTF16

A Unicode file which is based on the UTF16 encoding system is being transferred. When this option is set, the LOCAL_CHARSET/ REMOTE_CHARSET parameters are used to identify the character encoding of the input file and the required encoding for the output file.

Default: ASCII

Creating Custom Character Sets for File Conversion

The custom character sets are stored in standard ASCII files as conversion tables. Each original character and the character to which it is translated are defined by a single line in the file. The line number of the file itself indicates the decimal code value of the original character.

The character conversion table must be the only contents of the conversion file. Do not insert any comments or additional numbers. Each conversion file must contain 256 lines. If your text editor or word processor provides a line counter, you can use it to help you keep track of the line numbers. The procedures and examples below assume that the editor's line counter begins at 1.

When to Use Character Conversion

Use conversion processing when the following situations occur:

- When a non-USA character set is being used.
- When the transferred file is to be converted to all uppercase or lowercase characters.
- When XCOM Data Transport's default character conversion sets would not be appropriate.

> **NOTE**
> For information on using XCOM Data Transport's default character conversion sets, see Setting File Type Conversion.

Create a custom character translation table

F**ollow these steps:**

1. From the command line, use vi or another editor to create and open a file.
2. Define each character on a separate line.

XCOM Data Transport for UNIX or Linux performs the EBCDIC to ASCII translation as follows:

- When an EBCDIC code 0 is received, XCOM Data Transport substitutes the ASCII value from the first line (line 1) of the table.
- For EBCDIC code 1, XCOM Data Transport takes the value from the second line (line 2), and so on.

Because the first line is for code 0, you need to add 1 to the line number when determining which line of the table to modify.

Example 1

A question mark (?) must be sent to the mainframe in EBCDIC. Set up the translation table to translate it from ASCII to EBCDIC.

The ASCII code for ? is 63 (decimal). The EBCDIC code for ? is 111 (decimal). Take the ASCII code and add 1. This is 64. Go to line 64 of the ATOE table and type the value 111.

Example 2

An A is received from the mainframe in EBCDIC. Set up the translation table to translate it to ASCII.

The EBCDIC code for A is 193 (decimal). The ASCII code for A is 65. Take the EBCDIC code for A and add 1. This is 194. Go to line 194 of the ETOA table and type the value 65.

Hexadecimal Numbers in Conversion Tables

An EBCDIC character can be represented by either its decimal code or its hexadecimal equivalent. Hexadecimal numbers should be prefixed by 0*x*.

Example

In the following example, to translate the EBCDIC character A to the equivalent ASCII code using a hexadecimal number, line 194 of the conversion file would contain only the information below:

```
0x41
```

Customizing Default Character Sets

You can build a conversion file by customizing a copy of the atoe.tab or etoa.tab character sets and changing the character translations only where necessary.

To specify external character conversions tables for use by transfers, the CODETABL parameter (one to three characters with no default value). The CODETABL parameter is a prefix to the name of the files containing the external ASCII-to-EBCDIC and EBCDIC-to-ASCII character conversion tables.

> **NOTE**
> The CODETABL parameter is valid (resulting with the user-defined conversion tables loaded)
> only if the xcom.glb parameter INTERNAL_CONVERSION_TABLES=NO. Failure to specify INTERNAL_CONVERSION_TABLES=NO results in the source or target system not loading or using the defined conversion tables.

When creating external custom ASCII-to-EBCDIC and EBCDIC-to-ASCII translation tables, create the ATOE/ETOA tables using the *xxx*atoe.tab and *xxx*etoa.tab naming convention.

All custom translation tables must be in the $XCOM_HOME/convtab directory. The following files are supplied on your distribution media and contain the tables that are used to map ASCII-to-EBCDIC and EBCDIC-to-ASCII translations:
$XCOM_HOME/convtab/atoe.tab

- $XCOM_HOME/convtab/etoa.tab

> **NOTE**
> $XCOM_HOME is an environment variable.

Procedure

To translate all lower case ASCII characters to upper case EBCDIC characters

1. a. To create a custom character set, do the following:
   From the command line, type
   ```
   cp $XCOM_HOME/convtab/atoe.tab conversionfilename
   ```
   NOTE
   - *conversionfilename* can be any name you choose.
   - $XCOM_HOME is an environment variable.

   $XCOM_HOME/convtab/atoe.tab is copied to a new file and named conversionfilename.
   or
   b. To create a customer character set in order to use the CODETABL parameter, do the following:
   From the command line, type
   ```
   cp $XCOM_HOME/convtab/atoe.tab $XCOM_HOME/convtab/tstatoe.tab
   ```
   NOTE
   - The prefix used above is tst but it can be any 1 to 3 character prefix you choose.
   - $XCOM_HOME is an environment variable.

   $XCOM_HOME/convtab/atoe.tab is copied to a new file and named tstatoe.tab.
2. Use vi or another editor to open the file that was created in Step 1 and copy the values from lines 66 to 91 to lines 98 to 123.

   NOTE
   The decimal ASCII code for a is 98 and the decimal ASCII code for z is 123.

   The values in lines 98-123 are changed to represent uppercase EBCDIC characters.
3. Save the file and exit the editor.

Specifying a Custom Character Set

To use a custom character set you must change the parameter values in the xcom.glb file for the INTERNAL_CONVERSION_TABLES parameter and then either the CODETABL value, or the ETOA_FILENAME and/or ATOE_FILENAME values, depending on the custom character set changes that were made. To activate the changes in xcom.glb, you must restart xcomd.

To specify a custom character set in xcom.glb

1. From the command line, enter the following:
   ```
   vi xcom.glb
   ```
   The xcom.glb file is opened for editing.
2. Set INTERNAL_CONVERSION_TABLES=NO.
3. a. Change the value of CODETABL to the 1 to 3 character prefix specified for the xxxatoe.tab/xxxetoa.tab file names you created.
   or
   b. Change the values for ETOA_FILENAME and/or for ATOE_FILENAME to the file names containing the customized files you created.
4. Save xcom.glb and exit the editor.
5. Restart xcomd to activate the changes made.

Supporting Unicode and Multi-Byte Character Sets for Data Transfer

Before the advent of Unicode, a significant number of character sets were devised to permit the representation of symbols used in the Chinese, Japanese, Korean, and Taiwanese (CJK) languages. Today, Unicode is favored and there is an ongoing transition from this legacy character sets to Unicode encodings, most notably UTF-8 and UTF-16. Many CJK legacy Multi-Byte character sets are ASCII based, as is the case for the most commonly used Unicode encodings (i.e.UTF-8, UTF-16).

In the IBM mainframe (predominantly EBCDIC) world however composite character sets are commonly employed, involving a Shift-in/Shift-out encoding method. This encoding mechanism enables a single byte ASCII or EBCDIC character set to be used for the representation of Latin characters, in tandem with a multi byte character set for the representation of non-Latin characters. Shift-in and shift-out control characters are then inserted in the data stream to signal a switch between the two embedded character sets. For example, the CCSID 937 character set combines an EBCDIC single byte character set with a ‘Traditional Chinese’ multi-byte character set, whilst the CCSID 938 character set combines an ASCII single byte character set with the same ‘Traditional Chinese’ multi-byte character set.

XCOM Data Transport allows for transmission of text files that are encoded using Multi-Byte characters sets, including in-flight conversion of data between different character sets.

XCOM Data Transport utilizes the ICU (International Components for Unicode) toolkit to perform data conversion functions.

> **NOTE**
> For information about the ICU toolkit, see the ICU website http://site.icu-project.org/.

Using Unicode Transfer

XCOM Data Transport is capable of transmitting data using either the UTF-8 or UTF-16 Unicode encodings. They can be specified with the CODE_FLAG parameter to allow for conversion of files from one character encoding to Unicode and back.

When a file’s data comprises a low ratio of non-Latin characters versus Latin characters, UTF-8 encoding will consume fewer bytes and therefore result in a faster transfer. In contrast, when a file’s data comprises a high ratio of non-Latin characters versus Latin characters, UTF-16 encoding will produce the best result.

The Unicode transfer Data Formats parameter:

CODE_FLAG

Specifying Charset

The XCOM Data Transport sending server converts the input encoding to UTF-8 or UTF-16, while the XCOM Data Transport receiving server converts to the required output encoding. This divides the conversion workload between the two XCOM Data Transport servers.

Use the LOCAL_CHARSET and REMOTE_CHARSET parameters in order to choose the local file and remote file character encoding. If not specified for the transfer, they default to the value specified for the DEFAULT_CHARSET global parameter in the XCOM.GLB file.

For a list of supported Charsets see Appendix C.

The Unicode transfer Charset parameters:

DEFAULT_CHARSET

- LOCAL_CHARSET
- REMOTE_CHARSET

Handling Conversion Errors

Not all characters can be converted between Unicode and other charsets or vice versa. In most cases, Unicode is a superset of the characters supported by any given charset.

Use MBCS_INPUTERROR and MBCS_CONVERROR to specify what action XCOM Data Transport should take in the event of a character being encountered cannot be converted.

When erroneous data is encountered during conversion then the following actions are possible:

Skip the erroneous data and continue.

- Replace the erroneous data with the charsets default substitution character.
- Replace the erroneous data with the supplied Unicode character.
- Fail the transfer.

The XCOM Data Transport sending server uses MBCS_INPUTERROR whereas the XCOM Data Transport receiving server uses MBCS_CONVERROR to specify the action. If not specified the value of the DEFAULT_INPUTERROR and DEFAULT_CONVERROR global parameters in XCOM.GLB will be used.

The Unicode transfer Conversion error handling parameters:

DEFAULT_INPUTERROR

- DEFAULT_CONVERROR
- MBCS_INPUTERROR
- MBCS_CONVERROR

Record Processing

XCOM Data Transport uses a newline also known as line break or end-of-line (EOL) marker special character to identify records in the text files. With the advent of Unicode, all the end of the line delimiters are supported irrespective of the platform.

Use the LOCAL_DELIM or REMOTE_DELIM parameters to choose the delimiter depending on the charset used in a Unicode transfer. If these parameters are not specified then the value of the DEFAULT_DELIM global parameter in the xcom.glb file will be used.

The Unicode transfer Delimiter handling parameters:

DEFAULT_DELIM

- LOCAL_DELIM
- REMOTE_DELIM

Examples

Example1: In the following example, the XCOMTCP command is used to run a Unicode transfer by specifying CODE_FLAG=UTF8. The local file input.txt encoded in CP949 is converted to EUC-KR.

```
XCOMTCP  - c1 -f MYCONFIG.CNF LOCAL_FILE=input.txt CODE_FLAG=UTF8 LOCAL_CHARSET=CP949
 REMOTE_CHARSET=EUC-KR
```

Example2: In the following example, the conversion errors are handled. If any erroneous character is found while reading the input file, the transfer will be FAILED. If any erroneous character is found while converting to the remote charset, the transfer will continue by substituting the malformed character with a default substitution character.

```
XCOMTCP  - c1 -f MYCONFIG.CNF LOCAL_FILE=input.txt CODE_FLAG=UTF16 LOCAL_CHARSET=CP949
 REMOTE_CHARSET=EUC-KR MBCS_INPUTERROR=FAIL MBCS_CONVERROR=REPLACE
```

Example3: In the following example, the EBCDIC conversion is handled. The ASCII based text file is converted to EBCDIC encoding. The ASCII LF (Line Feed) delimiter is used to detect the end of a record. The NL is added as a line delimiter in the output file.

```
XCOMTCP  - c1 -f MYCONFIG.CNF LOCAL_FILE=input.txt CODE_FLAG=UTF8 LOCAL_CHARSET=ISO-8859-1
 REMOTE_CHARSET=CCSID#37 LOCAL_DELIM=ASCII:LF REMOTE_DELIM=EBCDIC:NL
```

Running Reports on History Records

You can run Easytrieve reports on XCOM Data Transport history records. Sample code is distributed with XCOM Data Transport.

The Easytrieve execution programs `xcomehrp` and `xcomesrp` are installed in the `$XCOM_HOME/ezt` directory.

- On an AIX 64-bit machine, add the `$XCOM_HOME/ezt` path to LIBPATH as follows to resolve the dependencies of executable programs xcomehrp and xcomesrp:

  ```
  export LIBPATH=$LIBPATH:$XCOM_HOME/ezt
  ```

- On a Linux s390x machine, add the `$XCOM_HOME/ezt` path to LD_LIBRARY_PATH as follows to resolve the dependencies of executable programs xcomehrp and xcomesrp:

  ```
  export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$XCOM_HOME/ezt
  ```

You can run the following reports:

- Transfer Information Report
- Statistical Data Transfer Report

Generate a Transfer Information Report

Follow these steps to generate a Transfer Information Report.

1. Go to the `$XCOM_HOME/ezt` directory.
2. Run *one* of the following commands:
   – To display the output report on the screen:

     ```
     xcomehrp {input_filename}
     Example:
     cd $XCOM_HOME/ezt
     xcomehrp
     ```

     Use this command to produce a report from xcomhist.histf (located in the $XCOM_HOME/ezt directory) and display the output report on the screen.

   – To place the output report into a file, where it can be reviewed with a text editor:

     ```
     xcomehrp {input_filename} > {PATH}/xcomehrp.out
     Example:
     xcomehrp /home/tester/xcom_history_record/XCOMHIST > /home/tester/ xcom_history_record/output.rpt
     ```

     Use this command to produce a report from the XCOMHIST file (located somewhere other than $XCOM_HOME/ezt) and generate an output report file.

   – (Optional) **{input_filename}**
     Use this parameter to pass an input file of XCOM Data Transport history records (other than xcomhist.histf) that is in the current directory.

   Default: xcomhist.histf

Generate a Statistical Data Transfer Report

Follow these steps to generate a Statistical Data Transfer Report.

1. Go to the `$XCOM_HOME/ezt` directory.
2. Run the following command:

   ```
   xcomesrp [time_interval] [flag_of_calculated_base] [input_filename]
   ```

`time_interval`

Specifies the display data interval:

1

Specifies a one-hour interval.

0.5

Specifies a half-hour interval.

Default: None

(Optional) `flag_of_calculated_base`

Specifies the calculation base for the start time or the end time.

S

Specifies the calculation base for the start time.

E

Specifies the calculation base for the end time.

Default: E

(Optional) `input_filename`

Passes an input file of XCOM Data Transport history records to the command. By default, XCOM Data Transport looks for the xcomhist.histf file in the current directory. This parameter lets you specify an input file with a different path or file name.

Default: xcomhist.histf

Example 1:

```
cd $XCOM_HOME/ezt
xcomesrp 1
```

Example 2:

```
cd $XCOM_HOME/ezt
xcomesrp 0.5 S
```

Example 3:

```
cd $XCOM_HOME/ezt
xcomesrp 0.5 /home/tester/xcom_history_record/XCOMHIST1
```

Example 4:

```
cd $XCOM_HOME/ezt
xcomesrp 1 E /home/tester/xcom_history_record/XCOMHIST2
```

Using Store and Forward Transfers

A store and forward transfer can be used to transfer a file indirectly when a destination system is not available. You can send a file to the mainframe, where it is stored temporarily. When the destination system becomes available, the file is automatically forwarded from the mainframe to the destination system.

This method is also used when you have two PU 2.0 devices that cannot communicate directly. They must use the mainframe as an intermediate device, because a UNIX or Linux system is not an intermediate device.

NOTE
A PU 2.1 device can handle direct transfers with another PU 2.1 device.

Example 1

In the following example, the xcom62 command is used. The file *mytest.tst* is sent to the intermediate z/OS node XCOMMVS2, which forwards the transfer to the remote Windows system defined in the *endest* destination member and into the file named *MYTEST.TST*.

```
xcom62 -c1 -f XIDEST=XCOMMVS2 REMOTE_SYSTEM=endest
LOCAL_FILE=mytest.tst REMOTE_FILE=MYTEST.TST
```

There must be a destination member created and enabled for the remote system in the z/OS Dynamic Control Library in order for XCOM Data Transport to forward the transfer successfully.

Example 2

In the following example, the xcomtcp command is used. The file *mytest.tst* is sent to the intermediate z/OS node XCOMMVS2, which forwards the transfer to the remote Windows system defined in the *endest* destination member and into the file named *MYTEST.TST.*.

```
xcomtcp -c1 -f XIDEST=XCOMMVS2 REMOTE_SYSTEM=endest
LOCAL_FILE=mytest.tst REMOTE_FILE=MYTEST.TST
```

There must be a destination member created and enabled for the remote system in the z/OS Dynamic Control Library in order for XCOM Data Transport to forward the transfer successfully.

When to Use

Use the XIDEST parameter with the REMOTE_SYSTEM parameter for store and forward transfers through an IBM mainframe system. Use store and forward to send an indirect transfer or to transfer files between two PU 2.0 devices.

The store and forward parameter is as follows:

XIDEST

Specifies the name of the remote system on the intermediate destination that is designated for store-and-forward transfers. If this variable is null or unset, then a direct connection to a remote system is attempted.

> **NOTE**
> For store-and-forward transfers only.

Range: 0 to 14 characters

Default: None

Set Password and User ID Security

A remote system requires a password and user ID before XCOM Data Transport can perform transfers. The access rights of the user ID determine the permitted transfer actions.

Example 1

In the following example, the `xcom62` command is used. *myname* is the USERID and *mypassword* is the PASSWORD.

```
xcom62 -c1 -f USERID=myname PASSWORD=mypassword
```

All other parameters are read from the default configuration file, `xcom.cnf` .

Example 2

In the following example, the `xcomtcp` command is used. *myname* is the USERID and *mypassword* is the PASSWORD.

```
xcomtcp -c1 -f USERID=myname PASSWORD=mypassword
```

All other parameters are read from the default configuration file, `xcom.cnf` .

The security parameters are as follows:

DOMAIN

Specifies the Windows domain name for use in authenticating the user ID and password when accessing a Windows-based machine that has shareable disks and drives that belong to that domain. Users can then access these shareable drives without having a local user ID or password that is defined to the machine.
Range: 1 to 15 characters
Default: None

PASSWORD

Specifies the password that is associated with the user ID on a remote system.
Range: 0 to 31 characters

> **NOTE**
>
> Service pack 11.6.01 with corresponding PTFs supports passphrase; hence, the accepted range is from 1 to 100 characters. To implement the passphrase support, apply the following PTFs to the corresponding platforms: SO05627(Linux), SO05628(AIX), SO05629(Solaris Sparc), SO05630(Solaris x86).

Default: None

TRUSTED

Allows the user to request a trusted transfer. When a trusted transfer is requested, the TRUSTED database on the partner XCOM Data Transport is searched to verify the user credentials. Trusted transfers eliminate the need for the user to specify a USERID and PASSWORD. If XCOM_TRUSTED_OVR is set to NO or no USERID is specified, the USERID of the process that initiated the transfer is used.
TRUSTED=YES cannot be specified with indirect transfers.
Range: YES, NO, Y, N
Default: NO

USERID

Specifies the user ID that the security system on the remote system checks before granting access for the file transfer.
Range: 0 to 12 characters
Default: None

Encrypt Parameter Values in Existing Configuration Files

You can encrypt selected parameter values in existing configuration files.

Syntax

The syntax for using XCOMENCR is as follows:

```
xcomencr input_file output_file
```

Options

The options for XCOMENCR are as follows:

- (minus sign)

Send output to stdout.
Example: xcomencr *input_file* -

+ (plus sign)

Replace input_file.
Example: xcomencr *input_file* +

no options

Displays help text.

Procedure

To encrypt a parameter value using XCOMENCR

1. Using a text editor, open the configuration file you want to modify, go to the parameter you want to encrypt, create a blank line above it, and enter the following:

   ```
   #!ENCRYPT
   ```

   Repeat as necessary for each parameter you want to encrypt.

   > **NOTE**
   > Because # denotes a comment line in the .cnf file, any line beginning with #!ENCRYPT is ignored by XCOM Data Transport and is used only by XCOMENCR.

2. Save the configuration file as an ASCII text file.
3. At the command prompt, enter the following:

   ```
   xcomencr input_file output_file
   ```

input_file

Is the name of the configuration file from Step 1.

The parameter value in the first non-comment line after each occurrence of the #!ENCRYPT statement is changed to the encrypted parameter value format.

> **NOTE**
> If you open an encrypted configuration file with a text editor, you would not be able to see the values of the encrypted parameters. In the GUI, when you open an encrypted configuration file, you can see the encrypted values, except for the user ID and password security parameters.

Change an Encrypted Value

Learn how to change an encrypted value that is already specified.

To change an encrypted value:

1. Delete all text after the parameter name and specify it again.

   For example, to change the following line, delete the values in italics:

   ```
   PASSWORD.ENCRYPTED=encrypted_value
   ```

2. Type an equal sign and the new parameter value in an unencrypted form, as follows:

   ```
   PASSWORD=new_value
   ```

3. Save the file as ASCII text and encrypt it.

Queue Locally Initiated Transfers

The queue contains the information that XCOM Data Transport uses to perform a locally initiated transfer.

> **NOTE**
> The xcomqm command provides access to information about the transfers in the queue. For information about using xcomqm, see Managing the XCOM Data Transport Queue in the "Operating Environment."

Example 1

In the following example, the xcom62 command is used. The transfer request goes into the queue to execute on July 28, 2012 at one minute after noon.

```
xcom62 -c1 -f QUEUE=YES START_DATE=07/28/12 START_TIME=12:01:00
```

All other necessary parameters are read from the default configuration file, xcom.cnf.

Example 2

In the following example, the xcomtcp command is used. The transfer request goes into the queue to execute on July 28, 2012 at one minute after noon.

```
xcomtcp -c1 -f QUEUE=YES START_DATE=07/28/12 START_TIME=12:01:00
```

All other necessary parameters are read from the default configuration file, xcom.cnf.

The queue management parameters are as follows:

PRIORITY

Indicates the priority that XCOM Data Transport uses for scheduling a transfer. If two transfers are scheduled for the same time, the one with the high priority is processed before one with a normal or low priority.

HIGH
Set high priority.

NORMAL
Set medium priority.

LOW
Set low priority.

Default: NORMAL

QUEUE

Indicates whether to execute the transmission request immediately or to allow the request to be queued. If the user does not specify a .cnf file, and has not changed a .cnf file, the default value is YES.

> **NOTE**
> If NO is specified and the remote system is unavailable, the request aborts. If YES is specified, START_TIME and START_DATE are read.

YES
The transfer request goes into a queue and executes depending on the traffic in the queue and START_DATE and START_TIME.

NO
The transfer starts immediately.

Default: YES

> **WARNING**
> If you load a .cnf file into the XCOM Data Transport GUI, the value (QUEUE=YES or QUEUE=NO) specified in the .cnf file is shown correctly in the generated xml file. However, when you submit the transfer from the XCOM Data Transport GUI, it is always treated as though QUEUE=YES.

START_DATE

Indicates the date on which the transfer becomes eligible for execution. The format is mm/dd/yy. If this field is blank, the current date is used.

Example:

A value of 02/28/13 indicates February 28, 2013 as the start date.

Format: mm/dd/yy

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

START_TIME

Indicates the time when the transfer becomes eligible for execution. The military format of *hh*:*mm*:*ss* is used. If this field is blank, then the current time is used.

Example

A value of 14:00:00 indicates 2 p.m. as the start time.

Format: *hh*:*mm*:*ss*

Default: None

Notify Transfer Completion

Using notification parameters, XCOM Data Transport automatically notifies a local system, a remote system, or a particular user that a transfer is complete without the user having to monitor the queue manually. Notification parameters allow you to specify one of the following levels of notification: Complete, Warning, or Error.

Example 1

The following example uses the xcom62 command. When the transfer completes, local user user1 is notified with a mail message, and remote user *USER2* is notified with a TSO message if the transfer receives an error.

```
xcom62 -c1 -f LOCAL_NOTIFY=user1 NOTIFYL=MAIL
NOTIFY_NAME=USER2 NOTIFYR=TSO RMTNTFYL=E LCLNTFYL=A
```

All other necessary parameters are read from the default configuration file, xcom.cnf.

Example 2

The following example uses the xcomtcp command. When the transfer completes, local user *user1* is notified with a mail message, and remote user *USER2* is notified with a TSO message if the transfer receives an error.

```
xcomtcp -c1 -f LOCAL_NOTIFY=user1 NOTIFYL=MAIL
NOTIFY_NAME=USER2 NOTIFYR=TSO RMTNTFYL=E LCLNTFYL=A
```

All other necessary parameters are read from the default configuration file, xcom.cnf.

The notification parameters are as follows:

LOCAL_NOTIFY

Specifies the user on the local system who is to be notified that XCOM Data Transport has completed a transfer. XCOM Data Transport uses the NOTIFYL parameter to determine the type of notification to use.

Range: 0 to 64 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

LCLNTFYL

Specifies the local user notification level.

A or ALL
Notify on transfer completion.

W or WARN
Notify only if the transfer received a warning or error.

E or ERROR
Notify only if the transfer received an error.

Default: ALL

NOTIFYL

The local user notification flag.

WRITE
A message is displayed on the workstation where the user is logged in.

MAIL
A mail message will be sent to the user.

NONE
No notification is sent.

ALL
A message is displayed on all workstations attached to the server.

Default: None

> **NOTE**
> The L in NOTIFYL indicates that the local system governs the processing of the resulting notification on that system.

NOTIFY_NAME

The user on the remote system who is to be notified when XCOM Data Transport completes a transfer.

> **NOTE**
> If the remote system is an IBM mainframe, XCOM Data Transport uses the value of NOTIFYR to determine the type of notification to deliver.

If the remote system is a UNIX or Linux system, the user receives a mail message.

Range: 0 to 12 characters

Default: None

NOTIFY_TERM

Specifies which terminals to write to if NOTIFYL=WRITE. If NOTIFY_TERM is not set, all users specified in LOCAL_NOTIFY are notified at the first terminal where they are logged in, as found in the system table.

Range: 0 to 256 characters

Default: None

NOTIFYR

Specifies the remote user notification type when sending data to a remote system.

WRITE
A message is displayed on the screen.

MAIL
A mail message is sent to the user.

TSO
The specified TSO user is notified.

WTO
XCOM Data Transport writes to the log only (WTO).

CICS
The specified CICS user is notified.

LU
The specified Logical Unit is notified.

ROSCOE
Notify Roscoe user.

NONE
No notification is sent.

ALL
Write to all users.

Default: None

NOTE
The R in NOTIFYR indicates that the remote system governs the processing of the resulting notification on that system.

RMTNTFYL

Specifies the remote user notification level when sending data to a remote system.

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: ALL

Checkpoint and Restart Transfers

By setting the parameters described in this section, a transfer is marked for checkpointing and automatic restart if an interruption occurs during the transfer attempt. After an interruption the daemon scans its table of transfers and automatically attempts to restart eligible transfers.

The CHECKPOINT_COUNT parameter indicates where to restart the transfer after it has been interrupted. If a transfer is interrupted, it is resumed from a checkpoint rather than starting over from the beginning of the file. When CHECKPOINT_COUNT=1000, XCOM Data Transport performs a check after every 1000 records sent, to ensure that the records are stored on the destination system. Then the next 1000 records are transmitted.

The restart parameters automatically restart a transfer after a transmission error, based on the number of retries and the retry time specified. Transmission conditions that permit restarts generally include most transmission errors. Restarts will also be attempted when a transfer is suspended by the remote system, and when a link goes down. Conditions that are not considered for restart include errors such as invalid passwords and "data set not cataloged" messages.

> **NOTE**
> Restart parameters can be specified without setting a checkpoint.

> **WARNING**
> Checkpointing impacts performance. Set the CHECKPOINT_COUNT as high as possible, or turn off checkpointing if it is not needed.

Example 1

In the following example, the xcom62 command is used. If the transfer is interrupted, it restarts from the most recent checkpoint of every 2000 records, and up to three attempts are made to retry the transfer at intervals of 30 seconds.

```
xcom62 -f CHECKPOINT_COUNT=2000 NUMBER_OF_RETRIES=3
RESTART_SUPPORTED=YES RETRY_TIME=30
```

All other necessary parameters are read from the default configuration file, xcom.cnf.

Example 2

In the following example, the xcomtcp command is used. If the transfer is interrupted, it restarts from the most recent checkpoint of every 2000 records, and up to three attempts are made to retry the transfer at intervals of 30 seconds.

```
xcomtcp -f CHECKPOINT_COUNT=2000 NUMBER_OF_RETRIES=3
RESTART_SUPPORTED=YES RETRY_TIME=30
```

All other necessary parameters are read from the default configuration file, xcom.cnf.

The checkpoint and restart parameters are as follows:

CHECKPOINT_COUNT

Defines how often (based on record count) the sending system requests a checkpoint to be taken. The value 0000 indicates no checkpointing.

Range: 0 to 9999

Default: 1000

> **NOTE**
>
> - XCOM Data Transport automatically sets CHECKPOINT_COUNT to 0 if any of the following parameter values are set:
>   - (L)EAR_CIPHER=XXX-OFB
>   - (L)EAR_CIPHER=RC4
>   - (L)EAR_HASH=XXX
> - This is a Version 2 parameter.

NUMBER_OF_RETRIES

Maximum number of retries before a transfer is logged as failed and taken out of the transfer queue. If the value is 0, no retries are attempted.

Range: 0 to 255

Default: 1

> **NOTE**
> This is a Version 2 parameter.

RESTART_SUPPORTED

Specifies whether automatic restart is to be supported on a transfer.

Range: YES or NO

Default: YES

> **NOTE**
> This is a Version 2 parameter.

RETRY_TIME

The number of seconds between retries of unsuccessful transfers.

Range: 0 to 99999

Default: 1

> **NOTE**
> This is a Version 2 parameter.

Generating TLS/SSL Certificates

This section describes how to generate certificates that can be used with XCOM Data Transport.

For more information on using OpenSSL, see *Network Security with OpenSSL* by John Vega, Matt Messier, and Pravir Chandra (O'Reilly & Associates).

Use TLS/SSL Mode

XCOM Data Transport uses TLS/SSL in client/server mode. In client/server mode, certificates are required for both the local (initiating) and remote (receiving) XCOM Data Transport partners. TLS/SSL considers the local XCOM Data Transport partner to be the client and the remote XCOM Data Transport partner to be the server.

When establishing the TLS/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 CA certificate from the other.

Setting up TLS/SSL for XCOM Data Transport involves the following tasks:

1. Set the expiration for the CA Certificate.
2. Create the CA Certificate.
3. Create the server certificate.
4. Create the client certificate.
5. Configure the XCOM Data Transport SSL server.
6. Configure the XCOM Data Transport client.

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 CA Certificate

To create 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 XCOM Data Transport installation directory for UNIX and Windows. For z/OS, it is 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 TLS/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.

   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.

2. 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 Server Certificate

To create 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.

   NOTE
   For UNIX, you must have 'root' authority to perform this task.

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.
   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 Client Certificate

To create 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.

   NOTE
   For UNIX, you must have 'root' authority to perform this task.
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 TLS/SSL Server and Client

Configure XCOM Data Transport to use the CA and server certificates for establishing SSL connections.

Configure XCOM Data Transport TLS/SSL Server

The following procedure configures XCOM Data Transport to use the CA and server certificates for establishing server (remote) SSL connections:

1. Review and modify the XCOM Data Transport TLS/SSL configuration file, configssl.cnf, so that the settings meet your site standards. Server connections use the RECEIVE_SIDE values. Also, ensure that the XCOM_HOME environment variable is set correctly to the XCOM installed location since it is used within this file.

   NOTE
   Make sure that you escape special characters. See Special Characters in Configuration File for more information.
2. Set the XCOM_CONFIG_SSL parameter in your default options table/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 XCOM Data Transport to receive remote TLS/SSL connections:
   a. For z/OS, specify the TCP/IP port that will accept TLS/SSL connection requests using the SSLPORT and/or SSLPORTV6 default options table parameters. In addition, the default options table parameter, SSL, must also be set to one of the following values:
      - ONLY -- To allow incoming TLS/SSL transfers only
      - ALLOW -- To allow both incoming TLS/SSL and incoming non-TLS/non-SSL transfers to this server

   b. For UNIX, during installation, the txpis and txpis6 (where applicable) services along with the default TCP/IP port values that will accept TLS/SSL connection requests are automatically added to the inetd configuration files. If different TCP/IP port values are needed from the default values then the txpis and/or txpis6 entries in the /etc/services file will have to be manually changed.
   c. For Windows, if the default TCP/IP port values that accept TLS/SSL connection requests need to be changed then they can be modified by using the TLS/SSL Port Number and/or the Ipv6 Port Number fields on the TCP/IP tab

in the Global Parameters GUI. In addition, the Choose Listeners field may need to be updated from the default of IPv4 Listeners depending on what listeners the site needs to have started.

4. Verify that the port that receives incoming TLS/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 TLS/SSL connections. If XCOM Data Transport will be receiving both incoming TCP/IP connections and incoming TLS/SSL connections, then two ports are required.
   a. For z/OS, reassemble the default options table and restart the XCOM Data Transport server (started task).
   b. For UNIX and Windows, restart the XCOM Data Transport service.

Configure XCOM Data Transport TLS/SSL Client

The following procedure configures the XCOM Data Transport TLS/SSL client to use the CA certificate and the server certificate when establishing client (local) TLS/SSL connections:

1. Review and modify the settings of the XCOM Data Transport TLS/SSL configuration file, configssl.cnf, as appropriate for your system. Client connections use the INITIATE_SIDE values. Also, ensure that the XCOM_HOME environment variable is set correctly to the XCOM installed location since it is used within this file.

   NOTE
   Make sure that you escape special characters. See Special Characters in Configuration File for more information.
2. Point the XCOM_CONFIG_SSL parameter in your default options table/global file to your customized configssl.cnf file.

   NOTE
   For z/OS, the path and file name must be an HFS file.
3. Set the SECURE_SOCKET parameter to YES to indicate a TLS/SSL connection.
   a. For z/OS, specify the SECURE_SOCKET parameter in the SYSIN01, the destination member, or the default options table.
   b. For UNIX and Windows, specify the SECURE_SOCKET parameter in the configuration (cnf) file.
4. Specify the port through which the remote XCOM Data Transport partner accepts SSL connections. Use one of the following parameters:
   a. PORT for UNIX and Windows
   b. IPPORT for z/OS
5. Initiate the transfer request.

Special Characters in Configuration File

You must pay attention to special characters when you edit the `configssl.cnf` file. If any value contains a special character, you must escape the special character. If you do not escape special characters, the OpenSSL libraries that are used by XCOM Data Transport may fail to properly read the values. In such a case the SSL transfers fails with the following error message:

```
TxpiSSLConfig Failed msg = ConfigSSL: Missing or invalid configssl.cnf file.
```

You can escape the special characters by using any kind of quote or the backslash (****) character.

Examples:

If a value contains the dollar (**$**) special character, you can use one of the following examples to escape it:

- `INITIATE_SIDE = "SAmp123XX$#ss45"`
- `INITIATE_SIDE = 'SAmp123XX$#ss45'`
- `INITIATE_SIDE = SAmp123XX\$#ss45`
- `INITIATE_SIDE = SAmp123XX"$"#ss45`
- `INITIATE_SIDE = SAmp123XX'$'#ss45`

Default Sample Scripts

By default, the XCOM Data Transport setup deploys sample scripts to create RSA certificates in your $XCOM_HOME\ssl directory. The type of certificate depends on the cipher suite used in your SSL communication. You can control this setting by using the $XCOM_HOME\config\configssl.cnf file.

Your security administrator needs to set the appropriate cipher suite for SSL communication.

Cipher Suites

When a TLS or SSL connection is established, the client and server negotiate a cipher suite, exchanging cipher suite codes in the client hello and server hello messages. The cipher suite specifies a combination of cryptographic algorithms to be used for the connection.

By default, a strong cipher suite is set in configssl.cnf as follows:

```
[SSL_METHOD]
INITIATE_SIDE = v3
RECEIVE_SIDE  = v3
# Optional
[CIPHER]
INITIATE_SIDE = ALL:!ADH:!LOW:!EXP:MD5:@STRENGTH
RECEIVE_SIDE  = ALL:!ADH:!LOW:!EXP:!MD5:@STRENGTH
```

This cipher suite uses the v3 protocol. By default, the following are excluded from the negotiation:

- Anonymous DH cipher suites
- Low encryption cipher suites (currently those using 64-bit or 56-bit encryption algorithms)
- Cipher suites using MD5

Ciphers are sorted according to their strength.

Cryptographic Protocols

XCOM Data Transport supports the TLSv1.2, TLSv1.1, TLSv1.0, and SSLv3 protocols. We recommend using TLSv1.2 or TLSv1.1 for the greatest security.

The protocol is negotiated as part of the TLS/SSL handshake. If the initiating side and receiving side do not agree on the protocol to use, the transfer fails.

You specify the protocol to use for TSL/SSL communications in the `configssl.cnf` file, under SSL_METHOD. The following values are supported. For 11.6.01 and higher, the protocol names are case-insensitive.

TLSV1.2
Specifies the TLSv1.2 protocol.

TLSV1.1
Specifies the TLSv1.1 protocol.

TLSV1 OR TLS
Specifies the TLSv1.0 protocol.

V3
Specifies the SSLv3 protocol. This protocol is applicable only when FIPS_MODE is OFF.

IMPORTANT
We do not recommend using SSLv3 because it is less secure.

ALL

Specifies all supported protocols (TLSv1.2, TLSv1.1, TLSv1.0, and SSLv3).
This option is provided for backward compatibility. If you require extensions such as server name, a client sends out TLSv1.0 "hello" messages that include the extensions. The client also indicates that it understands TLSv1.1 and TLSv1.2, and it permits a fallback to SSLv3. A server supports TLSv1.2, TLSv1.1, TLSv1.0, and SSLv3.

Examples

To enable TLSV1.2 on the initiating side and the receiving side, specify the following values:

```
[SSL_METHOD]
INITIATE_SIDE = TLSV1.2
RECEIVE_SIDE = TLSV1.2
```

To enable all protocols on the initiating side and the receiving side, specify the following values:

```
[SSL_METHOD]
INITIATE_SIDE = ALL
RECEIVE_SIDE = ALL
```

When you specify ALL, you can limit the available protocols by specifying SSL_OP_NO_SSLv3, SSL_OP_NO_TLSv1, SSL_OP_NO_TLSv1_1 and SSL_OP_NO_TLSv1_2 in the SSL_OPTION section. For example, to enable all protocols except SSLv3, specify SSL_OP_NO_SSLv3.

```
[SSL_OPTION]
INITIATE_SIDE = SSL_OP_ALL|SSL_OP_NO_SSLv2| SSL_OP_NO_SSLv3
RECEIVE_SIDE = SSL_OP_ALL|SSL_OP_NO_SSLv2|SSL_OP_SINGLE_DH_USE | SSL_OP_NO_SSLv3
[SSL_METHOD]
INITIATE_SIDE = ALL
RECEIVE_SIDE = ALL
```

Supported Cipher Suites

When a TLS or SSL connection is established, the client and server negotiate a cipher suite. The client and server exchange cipher suite codes in the client hello and server hello messages. The cipher suite specifies a combination of cryptographic algorithms to be used for the connection.

By default, a strong cipher suite is set in configssl.cnf as follows:

```
[SSL_METHOD]
INITIATE_SIDE = all
RECEIVE_SIDE = all
# Optional
[CIPHER]
INITIATE_SIDE = ALL:!ADH:!LOW:!EXP:MD5:@STRENGTH
RECEIVE_SIDE = ALL:!ADH:!LOW:!EXP:!MD5:@STRENGTH
```

This cipher suite uses the *ALL* TLS or SSL method to have backwards compatibility. By default, the following cipher suites are excluded from negotiation:

- Anonymous DH cipher suites
- Low encryption cipher suites (currently those using 64-bit or 56-bit encryption algorithms)
- Cipher suites using MD5

Ciphers are sorted according to their strength:

> **NOTE**
>
> For TLS v1.2 and later, we recommend SSL_METHOD.

Cipher Suites Supported in TLSV1 when FIPS_MODE=YES

The following cipher suites are supported in TLSV1 when FIPS_MODE=YES:

Cipher Suite Name	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm	Message Digest Algorithm
DHE-RSA-AES256-SHA	DH	RSA	AES(256)	SHA1
DHE-DSS-AES256-SHA	DH	DSS	AES(256)	SHA1
AES256-SHA	RSA	RSA	AES(256)	SHA1
EDH-RSA-DES-CBC3-SHA	DH	RSA	3DES(168)	SHA1
EDH-DSS-DES-CBC3-SHA	DH	DSS	3DES(168)	SHA1
DES-CBC3-SHA	RSA	RSA	3DES(168)	SHA1
DHE-RSA-AES128-SHA	DH	RSA	AES(128)	SHA1
DHE-DSS-AES128-SHA	DH	DSS	AES(128)	SHA1
AES128-SHA	RSA	RSA	AES(128)	SHA1

Cipher Suites Supported in TLSV1 when FIPS_MODE=NO

The following cipher suites are supported in TLSV1 when FIPS_MODE=NO:

Cipher Suite Name	Key Exchange Algorithm	Authentication Algorithm	Encryption Algorithm	Message Digest Algorithm
DHE-RSA-AES256-SHA	DH	RSA	AES(256)	SHA1
DHE-DSS-AES256-SHA	DH	DSS	AES(256)	SHA1
AES256-SHA	RSA	RSA	AES(256)	SHA1
EDH-RSA-DES-CBC3-SHA	DH	RSA	3DES(168)	SHA1
EDH-DSS-DES-CBC3-SHA	DH	DSS	3DES(168)	SHA1
DES-CBC3-SHA	RSA	RSA	3DES(168)	SHA1
DHE-RSA-AES128-SHA	DH	RSA	AES(128)	SHA1
DHE-DSS-AES128-SHA	DH	DSS	AES(128)	SHA1
AES128-SHA	RSA	RSA	AES(128)	SHA1
RC4-SHA	RSA	RSA	RC4(128)	SHA1
RC4-MD5	RSA	RSA	RC4(128)	MD5
EDH-RSA-DES-CBC-SHA	RSA	RSA	DES(56)	SHA1
EDH-DSS-DES-CBC-SHA	DH	DSS	DES(56)	SHA1
DES-CBC-SHA	RSA	RSA	DES(56)	SHA1
EXP-EDH-RSA-DES-CBC-SHA	DH(512)	RSA	DES(40)	SHA1

EXP-EDH-DSS-DES-CBC-SHA	DH(512)	DSS	DES(40)	SHA1
EXP-DES-CBC-SHA	RSA(512)	RSA	DES(40)	SHA1
EXP-RC2-CBC-MD5	RSA(512)	RSA	RC2(40)	MD5
EXP-RC4-MD5	RSA(512)	RSA	RC4(40)	MD5

Cipher Suite Table for TLS v1.1 and TLS v1.2

The following table provides information about the TLS v1.1 and TLS v1.2 Cipher Suites:

Cipher Suite Name	FIPS Mode	Protocol (SSL_METHOD)	Key Exchange	Authentication	Encryption	Message Digest
AES256-GCM-SHA384	YES	TLSv1.2	RSA	RSA	AESGCM(256)	AEAD
AES256-SHA256	YES	TLSv1.2	RSA	RSA	AES(256)	SHA256
AES256-SHA	YES	TLSv1.2, TLSv1.1	RSA	RSA	AES(256)	SHA1
AES128-GCM-SHA256	YES	TLSv1.2	RSA	RSA	AESGCM(128)	AEAD
AES128-SHA256	YES	TLSv1.2	RSA	RSA	AES(128)	SHA256
AES128-SHA	YES	TLSv1.2, TLSv1.1	RSA	RSA	AES(128)	SHA1
DES-CBC3-SHA	YES	TLSv1.2 TLSv1.1	RSA	RSA	3DES(168)	SHA1
DHE-RSA-AES128-SHA256	NO	TLSv1.2	DH	RSA	AES(128)	SHA256
NULL-SHA256	NO	TLSv1.2	RSA	RSA	None	SHA256

Cipher Suites Supported in v3

SSLv3 can be used only with FIPS_MODE=NO. It uses the same cipher suites as TLSv1 with FIPS_MODE=NO.

Integrate with Splunk Dashboards

Learn how administrators can use the `xcomanalytics.cnf` configuration file to send transfer details to the Splunk platform and view them on Splunk dashboards.

You can combine the transfer events with events from other applications to gain insights on the overall workflow. The Splunk dashboards can also serve as a centralized monitoring facility for your transfers.

XCOM Data Transport provides a configuration file, `xcomanalytics.cnf`, that you use to identify the Splunk server and to specify the transfer information to send. XCOM Data Transport also provides an application containing sample dashboards that you can install in Splunk.

Configure Splunk to Receive Transfer Events from XCOM

The Splunk administrator usually performs this task.

1. Create a source type with an appropriate name, such as `xcom-source`.
2. Create an HTTP event token:

- Specify the source type that you created in Step 1.
- Specify an appropriate index from the available choices, such as `main`.
- Uncheck the **Enable indexer acknowledgement** option.

For information about generating the HEC token in Splunk with SSL enabled, see the Splunk documentation.

3. Enable the HTTP event token with SSL.

Splunk is configured to receive transfer events from XCOM.

Configure XCOM to Send Transfer Events to Splunk

An XCOM administrator typically performs this task.

XCOM uses an analytics configuration file (`xcomanalytics.cnf`) to connect to Splunk. For XCOM to send transfer information to Splunk, you must provide the appropriate values in this file. You must also update the global parameters file (`XCOM.GLB`) to enable the analytics feature and specify the path for the analytics configuration file.

Review the parameters that are listed in the following procedure and gather the necessary values from the Splunk administrator. For detailed information about these parameters, see the Reference section of this documentation. Then follow the procedure to configure XCOM.

1. Update the `XCOM.GLB` global parameters file:
   a. Open the file in a text editor.
   b. Ensure that the XENDCMD parameter does not specify a path to a Splunk file.
   c. Add the following parameters and values to the file:

ANALYTICS_CNFFILE
Specifies the path for the `xcomanalytics.cnf` file.
Default: `$XCOM_HOME/config/xcomanalytics.cnf`

ENABLE_ANALYTICS
Specifies whether to send XCOM events to an external analytics platform. Specify `YES`.

   d. Save the `XCOM.GLB` file.
   Your changes are saved.
   e. Restart the XCOM Scheduler service.
   Your changes take effect.

2. Update the `xcomanalytics.cnf` file:
   a. Open the file in a text editor.
   b. Specify values for the following parameters.

ANALYTICS_HOST
Specifies the host name of the Splunk server.

ANALYTICS_PORT
Specifies the port number where the HTTP Event Collector (HEC) is running.

ANALYTICS_SCHEME
Specifies whether to use HTTP or HTTPS to connect to the Splunk server.

ANALYTICS_API
Specifies the REST API endpoint for the Splunk HEC.

ANALYTICS_HTTP_AUTH_HEADER
Specifies the authorization header for using the Splunk HEC.

ANALYTICS_HTTP_CUST_HEADER
(Optional) Specifies a custom header.

ANALYTICS_FIELDS

Specifies the transfer fields to send to Splunk when a transfer completes. Separate the fields with commas.

c. Save the file.

Your changes are saved and they take effect immediately.

XCOM is configured to send events to Splunk.

Install the XCOM Application in Splunk

The Splunk administrator usually performs this task.

This task is optional; installing the XCOM application installs the sample XCOM dashboards. If your site does not plan to use the XCOM samples, you do not need to install the application. You can create your own dashboards and can view them independently of the XCOM application.

To install the XCOM application in your local Splunk instance:

1. Open the Splunk home page in a web browser.
2. Upload the following application: `$XCOM_HOME\splunk\xcom.spl`. For upload instructions, see the Splunk documentation.
3. Restart your Splunk instance by issuing the following command:

   ```
   /opt/splunk/bin/splunk restart
   ```

   Your changes are applied.

The XCOM application is installed in your local Splunk instance. Users can now use the sample XCOM dashboards to view transfer events. For more information, see View the Sample XCOM Dashboards in Splunk.

Uninstall the XCOM Application

The Splunk administrator usually performs this task.

To remove the XCOM application from Splunk, issue the following command:

```
/opt/splunk/bin/splunk remove app xcom
```

The XCOM application is uninstalled. The sample XCOM dashboards are removed from Splunk.

View the Sample XCOM Dashboards in Splunk

Learn how to use the sample XCOM dashboards to view transfer events in Splunk.

The Splunk platform enables you to monitor and analyze system data. If your administrator has integrated XCOM with Splunk and installed the sample XCOM dashboards, you can use them to view transfer events.

The following sample dashboards are provided:

XCOM Service Critical Alerts

Displays the error messages from all XCOM UNIX/Linux and Windows servers. You can use time range filters to filter error messages from a given time range. Error messages are grouped into the following sections based on whether they are generated from the XCOM Scheduler service or from the XCOM transfer:

- XCOM Scheduler Service Alerts
- Failed Transfer Alerts

XCOM Data Transport Overall Summary

Displays the transfers of all XCOM servers, grouped by transfer status in a pie chart. Selecting a transfer status in the chart displays another pie chart for the selected transfer status, grouped by XCOM server. Selecting an

XCOM server in the chart displays a table with the associated transfer details. You can use time range filters to modify the time range of transfers that are displayed.

XCOM Data Transport Summary by XCOM Server

Displays a time range filter and a drop-down list of all XCOM servers. Selecting a time range and an XCOM server displays a pie chart that summarizes the transfer activity between the selected XCOM server and any remote XCOM servers. Selecting a remote server displays a pie chart that summarizes the transfers between this server and the selected remote server by status. Selecting a transfer status displays the transfer details in a table format.

To view the dashboards:

1. Access Splunk through your web browser. Your Splunk administrator can provide the web address.
2. Select **Dashboards**.

   The sample dashboards appear. If your Splunk administrator has created custom XCOM dashboards, they also appear here.
3. Select an XCOM dashboard.
4. Select items in the dashboard to drill down and view transfer event details.

Set Up Trusted Access

The Trusted Access feature allows XCOM Data Transport to transfer data to a remote partner actually specifying the user ID and password in the transfer.

Trusted Access transfers can be sent to and from Windows and various other partners. Any XCOM Data Transport platform that is r11 or higher can send and receive trusted transfers to and from each other.

Trusted Access Database

The Trusted Access feature works the same way across platforms. Two relational database tables are used to store trusted user authentication information:

XCOM_TRUSTED_SYS table

Contains the name of the remote SYSTEM_NAME that initiates the trusted transfer.

- Each row in this table can contain a NOTE and an optional GROUP name, group PASSWORD, and group DOMAIN. Each system can have multiple rows; that is, each system can have multiple groups defined.
- However, one user cannot belong to more than one group on a SYSTEM_NAME.

XCOM_TRUSTED_USERS table

Contains the remote SYSTEM_NAME and the USER_NAME.

- If the user is a member of a group, then the GROUP_NAME must also be specified.
- If the user is not a member of a group, then a USER_PASSWORD is also required. A USER_DOMAIN is optional.

Start the Trusted Access Interface

To start the Trusted Access interface, a user with root privileges starts the GUI and then selects the Trusted Transfer tab or link.

For more information about using the Trusted Access page, see Trusted Transfer.

Trusted Access Parameters

This section describes the parameters in the Trusted Access interface.

Group Name

Specifies the group that is associated with the remote system. This field is optional when groups are not being used for the remote system.
Range: 1 to 256 characters
Default: None

Notes

Describes the defined remote system. This field is optional.
Range: 1 to 256 characters
Default: None

System Name

Specifies the remote system for which you want to give trusted access to initiate transfers.

> **NOTE**
> This field is mutually exclusive with the Sysid and Sysname parameters.

Range: 1 to 256 characters
Default: None

Sysid

Specifies the system identifier of a remote system for which you want to give trusted access to initiate transfers. This value would be defined in the XCOM.GLB file of the remote system, using the parameter SYSID. This field and Sysname uniquely identify the remote system; they are mutually exclusive with the Remote System parameter.
Range: 1 to 4 characters
Default: None

Sysname

Specifies the system name of a remote system for which you want to give trusted access to initiate transfers. This value would be defined in the XCOM.GLB file of the remote system, using the parameter SYSNAME. This field and Sysid uniquely identify the remote system; they are mutually exclusive with the Remote System parameter.
Range: 1 to 8 characters
Default: None

User Name

Specifies the user ID of the user to associate with the remote system or group.
Range: 1 to 20 characters; cannot contain the following characters: " / \ [ ] : ; | = , + * ? < >
Default: None

Set Up Trusted Databases

This section describes how to set up a trusted database for XCOM Data Transport on your MySQL Database server.

Step 1. Install a Relational Database Management System

Install the relational database management system (RDBMS) that you will use to manage your trusted systems and user records.

Follow the procedures in the RDBMS documentation (MySQL for this example).

Step 2. Create a Database

A sample SQL file with the statements to create the database and tables is provided in the following file. Modify this file to match the RDBMS used at your site:

```
$XCOM_HOME/config/xcomtrstDB.sql
```

Then, enter the following command to create the database:

```
CREATE DATABASE XCOMTRST;
```

Step 3. Create the XCOM Trusted Systems Table

Create a Trusted Systems table to store your trusted system details.

1. Set up the Trusted Systems table by using the following code (for MySQL):

   ```
   USE XCOMTRST;
   CREATE TABLE XCOM_TRUSTED_SYS (
       SYSTEM_NAME              VARCHAR(255)         NOT NULL,
       IS_SYSNAME_ENTERED       CHAR(1),
       NOTES                    VARCHAR(255),
       SYSNAME                  CHAR(8)              NOT NULL,
       SYSID                    CHAR(4)              NOT NULL,
       GROUP_NAME               VARCHAR(255)         NOT NULL,
       USER_NAME                VARCHAR(255)         NOT NULL,
       USER_PASSWORD            VARCHAR(126),
       USER_DOMAIN              VARCHAR(15),
   PRIMARY KEY (SYSNAME, SYSID, SYSTEM_NAME, GROUP_NAME, USER_NAME));
   ```

   The Trusted Systems table is set up.
2. To create a unique index on the table to ensure that no duplicate record gets inserted, run the following code from the MySQL prompt:

   ```
   CREATE UNIQUE INDEX XCOM_TRUST_GPUX ON XCOM_TRUSTED_SYS (SYSNAME, SYSID, SYSTEM_NAME, GROUP_NAME);
   CREATE INDEX XCOM_TRUST_GPX ON XCOM_TRUSTED_SYS (SYSNAME, SYSID, SYSTEM_NAME);
   ```

   The Trusted Systems table now has a unique index.

Step 4. Create the XCOM Trusted Users Table

Create a Trusted Users table to store your trusted user details.

1. Set up the Trusted Users table using the following code (for MySQL):

   ```
   CREATE TABLE XCOM_TRUSTED_USERS (
       SYSTEM_NAME          VARCHAR(255)        NOT NULL,
       USER_NAME            VARCHAR(20)         NOT NULL,
       USER_PASSWORD        VARCHAR(126),
       USER_DOMAIN          VARCHAR(15),
       GROUP_NAME           VARCHAR(255),
       SYSNAME              CHAR(8)             NOT NULL,
       SYSID                CHAR(4)             NOT NULL);
   ```

   The Trusted Users table is set up.
2. To create a unique index on the table to ensure that no duplicate record gets inserted, run the following code from the MySQL prompt:

   ```
   CREATE UNIQUE INDEX XCOM_TRUST_UX on XCOM_TRUSTED_USERS (SYSNAME, SYSID, SYSTEM_NAME, USER_NAME,
    GROUP_NAME);
   ```

   The Trusted Users table now has a unique index.

Step 5. Install the Appropriate ODBC Driver

Multiple Windows PCs and multiple Linux/UNIX workstations can share the same XCOM Trusted tables. The full MySQL is installed only once, but the MySQL ODBC driver must be installed on every platform that shares the database.

To install the appropriate ODBC driver:

1. For MySQL, download the MysqlConnector/ODBC from Oracle.com.

2. Install the connector.
3. Follow the installation instructions.

Step 6. Configure ODBC

> **NOTE**
> Configuration is different for MySQL and Db2 and Oracle.

On Linux/Unix platforms, we recommend using UnixODBC software to create and manage DSN. Configuration to create DSN may vary between MySQL, Db2, and Oracle. Refer to the UnixODBC documentation and documentation from the respective DB vendor to configure DSN. Once DSN is configured, use the isql utility program to verify the connection.

For example, if you use XcomTrust as the DSN name, the MySQL DB server is hosted on your local machine on port 3306, and the root is the username to connect to the database, then testing ODBC connection with isql displays a "success" message.

1. If you establish a single remote database server instead of installing multiple copies of MySQL database server, perform the following steps on the database server machine so that it can be accessed from different machines:
   a. Enter the following command:
      ```
      CREATE user xcomuser@'%';
      ```
      The % indicates that you accept xcomuser from any host.
      To be more restrictive, code the following syntax (for example):
      ```
      CREATE user xcomuser@'user01-xp.ca.com';
      ```
   b. Grant each user accessing your database certain privileges:
      ```
      GRANT {all/select,insert} on xcomtrst.* to xcomuser@'{%/machine}';
      ```

Step 7. Install the Appropriate JDBC Driver

To install the appropriate JDBC driver:

1. For MySQL, download the appropriate MysqlConnector/J from the MySQL Downloads page.
2. Extract the zip file and copy the mysql-connector-java-*x.x.x*-bin.jar file to $XCOM_HOME/lib folder (or place the file path in the CLASSPATH environment variable).

Step 8. Modify the Default Global Parameters Table

Modify the XCOM Data Transport default Global Parameters table, XCOM.GLB, to include the following parameters:

TRUST_ODBC
Specifies the ODBC Data Source name that you created in a previous step.

TRUST_USER
Specifies the user ID that is used to establish a connection to the database.

TRUST_PASS
Specifies the password of XCOMTRUST_USER, in plain text.

> **WARNING**
> Plain text passwords are *not* recommended for production use. Encrypt this value as explained in the next parameter, TRUST_PASS[.ENCRYPTED].

TRUST_PASS[.ENCRYPTED]
Specifies the password of XCOMTRUST_USER, in encrypted form. Do not specify this value manually. Instead, run %XCOM_HOME%\XCOMENCR.EXE against the XCOM.GLB file after you have finished modifying it. XCOMENCR.EXE automatically supplies the encrypted value for TRUST_PASS[.ENCRYPTED]. For details about running XCOMENCR.EXE, see Encrypt Parameter Values in Configuration Files.

TRUST_OWNER

Specifies the creator ID of the Trusted tables. If you are using MySQL or if the ID is the same as XCOMHIST_USER when using Db2, you can omit this parameter.

TRUST_TABLE_PFX

Specifies the prefix to use for the names of the Trusted tables.
Default: XCOM_TRUSTED

TRUST_DATABASE_NAME

Specifies database that was created on the database server.

TRUST_SERVER

Specifies the database server where the trusted database was created.

TRUST_PORT

Specifies the port of the database server where the trusted database was created.

TRUST_DATABASE_TYPE

Specifies the type of database that was created on the database server.

Additional Information

You can set the Trusted parameters by editing the XCOM.GLB file in any text editor or you can set them on the GUI.

After you have set the Trusted parameters, restart the XCOM Data Transport daemon and the XCOM Data Transport GUI for your changes to take effect.

Configuring JDBC Database Access

The standalone UI in XCOM Data Transport is a Java-based application that uses JDBC to communicate with the database for accessing the trusted transfer tables. If you do not plan on using trusted transfers, then you can skip configuring the JDBC drivers.

Configure DB2 JDBC drivers

Follow these steps:

1. Locate the DB2 database driver jar files db2jcc.jar and db2jcc_license_cu.jar. In most installations these files are in the DB2 java directory.
2. Copy the database driver jar files to the XCOM_HOME/lib directory.
3. Add the directory path of the database driver jar files as an external directory (-java.ext.dirs) in the XCOM_HOME/bin/StandaloneUI.sh file.

Example:

```
-Djava.ext.dirs=$XCOM_HOME/lib:/opt/DB2path/java
```

Configure MySQL JDBC drivers

Follow these steps:

1. Locate the MySQL database driver jar file mysql-connector-java-x.x.x-bin.jar, where x.x.x is the release of MySQL. If the file is not part of your MySQL installation, it can be found on the mysql.com site in the mysql connector/j feature.
2. Manually copy the database driver jar file to the XCOM_HOME/lib directory.
3. Add the directory path of the database driver jar file as an external directory (-Djava.ext.dirs) in the XCOM_HOME/bin/StandaloneUI.sh file.

Example:

```
-Djava.ext.dirs=$XCOM_HOME/lib:/opt/mysqlpath/java
```

Interdependent Transfers

XCOM Transfer Control, referred to as XTC processing, makes use of transfer parameters allowing interdependent transfers to be defined as a single group. Transfers in the same group can be held, purged, or released depending on the successful or unsuccessful completion of other transfers in the group.

For important information about XTC processing, see the following sections:

- Transfer Control (XTC) Parameters
- Coding Interdependent Transfers

Coding Interdependent Transfers

Code a set of interdependent transfers for CNF configuration files.

NEWXFER

The parameter NEWXFER must be used at the beginning of each of the multiple transfer requests, as shown in the following diagram:

```
              ...
               ...
               ...
transfer request 1 parameters
               ...
               ...
               ...
#==============================
CONTROL=NEWXFER
#==============================
               ...
               ...
               ...
transfer request 2 parameters
               ...
               ...
               ...
#==============================
CONTROL=NEWXFER
#==============================
               ...
               ...
               ...
transfer request 3 parameters
               ...
               ...
               ...
```

XTCNET

The parameter XTCNET is used to indicate which transfer requests belong to the same group (are dependent on each other). Suppose, for example, that the transfers XFER1, XFER2 and XFER3 need to be grouped together. This can be accomplished simply by assigning the name of the group (say, GROUP1) to XTCNET and including XTCNET=GROUP1 in the definition of each of the three transfer requests.

XTCJOB

Each interdependent transfer request must have a unique name, which is assigned to the XTCJOB parameter. Thus, if XFER1, XFER2 and XFER3 are the names of three interdependent transfer requests, their transfer definitions must contain the parameter XTCJOB=XFER1, XTCJOB=XFER2, and XTCJOB=XFER3, respectively.

Other XTC Parameters

XTC parameters other than XTCNET and XTCJOB are used to indicate the dependencies obtaining between the transfer requests. The outcome of a transfer request can affect as many as eight other transfer requests. In addition to the XTC parameters, other configuration parameters can (and some, for example, the local file name, must) be used when defining interdependent transfer requests.

HOLD_TRANSFER

HOLD_TRANSFER=YES must be coded in each transfer request that is dependent on the successful/unsuccessful completion of another transfer request (see Example 1).

HOLDCOUNT

However, HOLD_TRANSFER=YES need not be coded if the holding/releasing of the transfer request is controlled by the HOLDCOUNT parameter (see Example 2).

Multiple Jobs

You can also define multiple jobs (see Example 3).

Transfer Control (XTC) Parameters

The parameters that are described in this section form a feature of XCOM Data Transport that is known as XCOM Data Transport Transfer Control (XTC). These parameters are used to handle dependencies between multiple transfers. For example, one transfer must complete in a certain way before another can start. They provide the means by which interdependent transfer requests can be defined and processed as a single group. A transfer request belonging to such a group can cause another transfer request in the same group to be held, purged, or released (either conditionally or unconditionally) from the transfer request queue.

For further discussion of interdependent transfers, see Coding Interdependent Transfers.

- HOLD_TRANSFER
- HOLDCOUNT
- XTCERRDECR
- XTCERRINCR
- XTCERRPURGE
- XTCERRREL
- XTCGOODDECR
- XTCGOODINCR
- XTCERRPURGE
- XTCGOODREL
- XTCHOLD_COUNT
- XTCJOB
- XTCNET
- XTRACE

Password Protection by Encryption

Learn how XCOM Data Transport protects your password using the encryption .

XCOM Data Transport always protects the password by using encryption. XCOM Data Transport encrypts the password in the configuration file and during transmission. A parameter is used to select the cipher that is used for encrypting the password.

How Transmission Password Encryption Works

The cipher that is used to encrypt the password during transmission is controlled by using the TRNENCRL_CIPHER/ STCTRNENCRL_CIPHER and TRNENCRR_CIPHER parameters. Each of these parameters provides a list of ciphers. TRNENCRL_CIPHER/STCTRNENCRL_CIPHER provides the list of requested ciphers for locally initiated connections. TRNENCRR_CIPHER provides a ranked list of permitted ciphers for remotely initiated connections. See the description of the TRNENCRL_CIPHER and TRNENCRR_CIPHER parameters in the XCOM.GLB Parameters section for the full list of the supported ciphers. See the description of the TRNENCRL_CIPHER and STCTRNENCRL_CIPHER parameters in Appendix A: Parameters for the full list of supported ciphers.

When a connection is started to the remote system, the requested list of ciphers for the (STC)TRNENCRL_CIPHER parameter is sent to the remote system. The remote system compares that list with its list of permitted ciphers from the TRNENCRR_CIPHER parameter. The common cipher with the highest ranking on the permitted list is selected to encrypt the password fields during transmission.

Once the cipher is selected a key is generated, the generated key is valid for the current connection only. For the XCOM proprietary cipher, a proprietary technique is used to exchange the encryption key. For all other ciphers, the key exchange is done using a DH (Diffie-Hellman) key exchange.

When using DH key exchange, the TRNENCRR_DHBITS parameter on the remote system controls the size of the prime number that is used during the key exchange. Consider the following factors when determining the value for the TRNENCRR_DHBITS parameter:

- The more bits used for the prime number the more secure the key exchange.
- The more bits used for the prime number the more CPU overhead is required to negotiate a secret value that is based on the prime number.
- Each XCOM Data Transport connection generates a unique secret value.

When using the DH key exchange, the secret value that is negotiated is then hashed using the SHA1 hash function to generate the encryption key that is used for the XCOM Data Transport connection.

NOTE
The CA XCOM proprietary cipher is compatible with releases of XCOM Data Transport before r11.6. However, releases of XCOM Data Transport before r11.6 do not have the capability to negotiate the cipher that is used for password encryption and assumes the use of the CA XCOM proprietary cipher. To allow incoming connections from XCOM Data Transport releases before r11.6, include the COMPAT option on the TRNENCRR_CIPHER parameter.

The cipher negotiation can be enabled or disabled by using the COMPAT option of (STC)TRNENCRL_CIPHER.

Security of Passwords in Traces and XCOM Files

XCOM Data Transport automatically encrypts passwords in transfers, traces, and queue files.

Create the History Database

Learn how to create the required relational database to store the XCOM Data Transport for UNIX/Linux history records.

Possible database engines include the following ones:

- Db2
- Db2/UDB
- MySQL
- Oracle

Use the following procedures as a model to help you create your history database. These procedures use MySQL. MySQL is an Oracle product that can be downloaded and installed on a wide variety of platforms.

Step 1. Install a Relational Database Management System

Install a relational database management system (RDBMS) to manage your history records.

To install an RDMS, follow the procedures in the RDBMS documentation.

Step 2. Create a Database

Sample SQL files with the commands to create the database and tables are provided in the following files:

For MYSQL/Db2:

```
$XCOM_HOME\config\historyDB.sql
```

For Oracle:

```
$XCOM_HOME\config\HistoryDB_Oracle.sql
```

Follow the instructions in these files to create the database and history tables.

NOTE
Update these files to match the database management system being used at your site. $XCOM_HOME is an environment variable referring to the XCOM installation directory.

Step 3. Modify the XCOM Default Global Table

Modify the XCOM global defaults file (`XCOM.GLB`) to include the following parameters:

SYSNAME and SYSID
Uniquely identify a XCOM Data Transport processor. Specify a 1 through 8 character SYSNAME and a 1 through 4 character SYSID.

NOTE
These values should have been defined during the installation process but can be changed as required.

These parameters are needed because history records can be accumulated from many different systems and system types.
In this example, SYSNAME is set to CAXCOM01 and SYSID is set to LINX.

XCOMHIST
The name that XCOM Data Transport uses to connect to ODBC.
In this example, XCOMHIST is set to xcomhist for MySQL.

XCOMHIST_TBL
The name of the table that was created.
The default value is xcom_history_tbl.

XCOMHIST_FILE
The name of a file to contain SQL insert statements that could not be executed due to inaccessibility to the Database server. These records are also written when XCOM Data Transport tries but fails to insert the record.
The default value is set to $XCOM_HOME/config/history.inserts where $XCOM_HOME is an environment variable referring to the XCOM installation directory.
The records may be used to insert history records directly into the database.

```
mysql-> use database xcomhist
mysql-> source $XCOM_HOME/config/history_records (file name pointed to by the global parameter
 XCOMHIST_FILE)
```

XCOMHIST_SPLIT_FILE= Y/N
Specifies whether to split the INSERT statements that are created when ODBC fails into 72-byte segments. IBM SPUFI, a Db2 utility, takes only 80-byte records (72 usable) as input, whereas MySQL takes as input the complete statement and does not handle 72-byte checks; Db2 does not handle complete statements (unless they are less than 72 bytes).

XCOMHIST_BACKSLASH=Y/N
Inserts to the named table include the name of the local file on the PC. Files are typically named c:\folder1\folder2\file.
MySQL interprets \\ sequences as a single backslash character both in the actual insert and in the statement that is created when an insert fails. IBM treats the characters differently. If your target system is Db2 then XCOMHIST_BACKSLASH=Y creates file names as c:\folder1\folder2\file.

NOTE
If your target system is MySQL and XCOMHIST_BACKSLASH=Y, then the sql-mode option must also have NO_BACKSLASH_ESCAPES enabled.

If N is specified, then XCOM creates the path as c:\\folder1\\folder2\\file.

NOTE
When the data is inserted into the table, the \\ is interpreted as a single slash.

XCOMHIST_USER
Specifies the user ID that connects to the database.

XCOMHIST_PASSWORD
Specifies the password of XCOMTRUST_USER, in plain text.

WARNING
Plain text passwords are not recommended for production use. Encrypt this value, as explained in the next parameter, XCOMHIST_PASSWORD[.ENCRYPTED].

XCOMHIST_PASSWORD[.ENCRYPTED]
Specifies the password of XCOMHIST_USER, in encrypted form. You do not specify this value manually. Instead, you run $XCOM_HOME\XCOMENCR against the XCOM.GLB file after you have finished modifying it.

XCOMENCR automatically supplies the encrypted value for XCOMHIST_PASS[.ENCRYPTED]. For details about running XCOMENCR, see Encrypt Parameter Values in Configuration Files.

XCOMHIST_OWNER

The ID of the creator of the History Table. This value can be omitted when using MySQL or if the owner is the same as the value defined in XCOMHIST_USER when using Db2.

Step 4. Install the Appropriate ODBC Driver

Multiple Windows PCs and multiple Linux/UNIX workstations can share the XCOM History Table. So you install the full version of MySQL once, but you must install the MySQL ODBC driver on every platform that shares the database.

1. For MySQL, download the MySQLConnector/ODBC from Oracle.
2. Install the connector.
3. Follow the installation instructions.

Step 5. Configure ODBC

> **NOTE**
> The configuration is different for MySQL and Db2 and Oracle.

On Linux/Unix platforms, it is recommended to use UnixODBC software to create and manage DSN. Configuration to create DSN may vary between MySQL, Db2, and Oracle. Refer to the UnixODBC documentation and documentation from the respective DB vendor to configure DSN. Once DSN is configured, use the isql utility program to verify the connection.

1. If you choose to establish a single remote database server, perform the following steps on the database server machine:
   a. Enter the following command:
   ```
   CREATE user 'xcomuser'@'%';
   ```
   The % indicates that you accept xcomuser from any host.
   To be more restrictive, code the following command (for example):
   ```
   CREATE user 'xcomuser'@'user01-xp.ca.com';
   ```
   b. Grant privileges to each user accessing your database:
   ```
   GRANT {all/select,insert}  on xcomhist.* to 'xcomuser'@'{%/machine}';
   ```

Security Considerations for History Records

XCOM Data Transport checks security by verifying that the user requesting history records is a member of the XCOMADM group and/or the XCOMSADM group. XCOM Data Transport returns records as follows:

- If the user is not a member of either XCOMADM or XCOMSADM, then the returned transfers are limited to the ones containing that user ID.
- If the user is a member of XCOMADM, then XCOM Data Transport performs *both* of the following actions:
  - Returns transfers in the queue for *all* users and allows operator functions against those transfers.
  - Returns history records for *all* transfers for the target machine (SYSNAME, SYSID), regardless of the owner.
- If the user is also a member of XCOMSADM, then XCOM Data Transport returns history records for *any* machine that is included in the history table.

For instructions to create local user groups and add users to them, see your Windows operating system documentation.

Set Up a Procedure to Generate Reports on History Records

You can run an Easytrieve® report on the XCOM Data Transport history records. A sample job is available to generate reports that have a similar look and feel to the GUI.

Consideration to Use XCOM Docker Containers

Unlike traditional XCOM Data Transport where xcomtcp and other XCOM Data Transport binaries are available on the host machine, you must access these commands from the running container. To perform transfers, before you start the container, map the directories that contain configuration files (CNF/XML) and other input files as volumes to the container. Once completed, you can use the Docker exec to execute transfers and other XCOM commands.

Examples:

Use the interactive mode to submit transfers and capture the submission details:

```
docker exec -it <xcomcontainerid> /opt/CA/XCOM/bin/xcomtcp -c1 -f /tmp/sample.cnf
```

Use the non-interactive mode to submit jobs to a container and return:

```
docker exec -d <xcomcontainerid> /opt/CA/XCOM/bin/xcomtcp -c1 -f /tmp/sample.cnf
```

Check Queue:

```
docker exec -it <xcomcontainerid> /opt/CA/XCOM/bin/xcomqm -La
```

You can use other XCOM Data Transport commands in the similar approach.

Messages

Provides information about product messages.

XCOM Data Transport for UNIX/Linux generates messages that describe normal processing, warning situations, and error conditions that can occur.

To browse all XCOM messages, go to XCOM Data Transport Messages.

Reference

Provides reference information about the product features and functions.

Operating Environment

The components that are used for scheduling transfers, handling files, and managing XCOM Data Transport resources are the xcomd command and the daemon process, the global parameters, the session control parameters, the queue, and the use of post processing scripts.

A substantial portion of a user's ability to control XCOM Data Transport comes from the use of files to set variables that govern the system's behavior. These files are xcom.glb and xcom.cnf.

PATH and Directory Considerations

- Be sure that you have properly set up your directories and path variables before executing a file transfer request.
- The XCOM Data Transport daemon, xcomd, is installed in the $XCOM_HOME/sbin directory.
- Be sure that $XCOM_HOME/bin is in your PATH variable and is set so that your shell can find xcom62 or xcomtcp, or both.
- The $XCOM_HOME directory is the default repository that XCOM Data Transport uses for logging and for configuration information.
- For further information about PATH and directory defaults, see Global Parameters.

Transfer Protocols

XCOM Data Transport supports transfer protocols, such as Ethernet, SDLC, Token Ring, and TCP/IP.

SNA/APPC Protocols

For SNA/APPC protocols, support is based on the capabilities that are provided by the particular APPC vendor. For SNA/APPC, XCOM Data Transport requires an active LU 6.2 session to transfer files. When an LU 6.2 session is established, local users can begin using XCOM Data Transport to initiate transfers.

TCP/IP Protocols

The TCP/IP component of XCOM Data Transport supports TCP/IP protocols for performing transfers between XCOM Data Transport r11.x platforms using TCP/IP.

About Parameters

Use parameter values to control the variables that govern the behavior of XCOM Data Transport. For a full list of XCOM Data Transport for UNIX and Linux parameters, see Global Parameters.

Parameter Values

Parameter values can be defined to XCOM Data Transport as indicated in the following table. The values take precedence depending on where they are specified.

The order in which parameter values take precedence is as follows:

Order of Precedence	Parameter Value Specified in	Explanation
1	Command Line	If the user specifies a value for a parameter on the command line, this overrides every other specification in the files or the program.
2	xcom.cnf or *filename.cnf* or *filename.xml*	If the value is specified in the xcom.cnf file, or in a user-customized configuration file, *filename.cnf, or filename.xml*, it overrides the value in the xcom.glb file for locally initiated transfers.
3	xcom.glb	If the value is specified in the xcom.glb file, it overrides the value in the program. The values that are specified in xcom.glb are used by the system administrator to start XCOM Data Transport and for remotely initiated transfers.
4	Program Defaults	If a value is not specified anywhere, the program has its own defaults.

Parameter Format

The XCOM Data Transport parameters consist of assignment statements. The format for assignment statements is as follows:

- PARAMETER_NAME (always all uppercase, with underscore character (_) when indicated)
- An equal sign (=)
- A character string that is terminated by a newline

Syntax

The syntax for assignment statements is as follows:

```
PARAMETER_NAME=value
```

Example

In the following example, the parameter EXPIRATION_TIME is set to a value of 6000 seconds.

```
EXPIRATION_TIME=6000
```

This controls the maximum time in seconds that a transaction is held in the transfer queue.

Guidelines

The following guidelines apply when using XCOM Data Transport parameters:

- When you type trailing spaces and tabs from the command line as part of a parameter value, the command-line processor (the shell) strips them.
- When you type trailing spaces and tabs into a file or script with an editor, such as vi, they are treated as part of a parameter value. These trailing spaces and tabs is removed. If they are not removed they can cause confusion and unpredictable or undesirable results.
- Empty lines and lines beginning with a pound sign (#) are discarded.
- When the parameter value includes a special character, it is interpreted as a command by the command processor (the shell) then as a literal value. Use the appropriate escape sequence to specify the literal meaning.

Examples

In the following examples, the *spacespacespace* represents trailing spaces.

If you type the following command at the command line, the trailing spaces are ignored:

```
xcom62 -c1 LOCAL_FILE=xyzspacespacespace
```

If you type the following command into any configuration file using an editor, the *spacespacespace* would be treated as part of the name of the LOCAL_FILE:

```
LOCAL_FILE=xyzspacespacespace
```

If you type the following command at the command line, the “!” character is interpreted as history expansion command with the bash shell.

TRNENCRL_CIPHER=ALL:!AES

The xcomd Command

Use xcomd from the command line to start and stop the XCOM Data Transport daemon process, to kill the daemon process, to set trace levels, and to report the release level of xcomd.

The xcomd command controls the daemon. The daemon itself runs as a background process to control file transfers and manage XCOM Data Transport resources. The daemon does the following tasks:

- Schedules and synchronizes transfer requests.
- Controls shared memory for transfers.
- Establishes the default parameter values by reading the parameter file, xcom.glb and then running in the background.
- Controls the automatic restart of locally initiated transfers.
- Writes queue information out to disk periodically.
- Deletes aged entries from the queue.
- Notifies a local user by executing the xcomntfy script when LOCAL_NOTIFY is required.
- Communicates with active or pending transfers to terminate a transfer.

Control the Daemon

When XCOM Data Transport is installed, the superuser (root) has permission to control the daemon. Without authorized permission, other users cannot control the daemon. Users are typically given permission to perform transfers and check the status of the queue.

The daemon must be running before you can do any transfers. For performing transfers using SNA protocols, XCOM Data Transport also requires an active LU 6.2 session to transfer files. Once an LU 6.2 session is established, local users can begin using XCOM Data Transport to initiate transfers. For performing transfers using TCP/IP, your system must be configured for using XCOM Data Transport with TCP/IP.

Syntax

The syntax for using the xcomd command is as follows:

```
xcomd option
```

Options

The following table explains the options for xcomd:

-c[y]

Kill the daemon process and free shared memory. This option stops the daemon while transfers are running. **Note:** Use this option as a last resort because it brings the scheduler down immediately, whether there are running or scheduled transfers. If you use the **-c** option without the **y**, the system prompts you to confirm this action by typing **y** for yes.

> **WARNING**
> This option can cause corruption of the XCOM Data Transport queue. It can shut down XCOM Data Transportwhile it is updating the queue. To prevent problems, delete the queue by deleting all files in $XCOM_HOME/Q after issuing this command.

-d*tracelevel*

Set the trace level of the daemon process to *tracelevel*. Higher numbers (up to 10) give more trace information. The trace goes to stderr. For more information, see How to Trace Problems.

-r

Report the release level of xcomd and exit.

-s

Stop the XCOM Data Transport daemon by sending a request to the scheduler.
Note: This option does not stop the daemon immediately if there is an active transfer going on.

Start xcomd

To start xcomd, enter the following command at the system prompt:

```
$XCOM_HOME/sbin/xcomd
```

> **NOTE**
> Ensure the XCOM_HOME environment variable is set to the XCOM Data Transport install directory that is selected during installation.

The options for xcomd are described in the previous Options topic.

Global Parameters

The xcom.glb file lets you set defaults independently of the execution environment for transfers. This file is created as part of the installation process. Use vi or another editor to modify the parameters. For more information, see XCOM Parameters

Local Session Control Parameters

Learn how to use the XCOM.SES file to control the maximum number of locally initiated sessions that this partner supports.

The XCOMD XCOM Scheduler service uses the `XCOM.SES` file to control the maximum number of sessions that are available for locally initiated queued transfers for each remote system. By default, the `XCOM.SES` file is placed in the `$XCOM_HOME/config` directory. $XCOM_HOME is an environment variable.

You can set a maximum number of partners that can be described in this file by specifying a number in the MAX_SESSIONS_ENTRIES global parameter. For example, if you specify MAX_SESSIONS_ENTRIES=50, you can have up to 50 lines in this file, each one specifying the name of the remote system.

You limit the number of sessions available for XCOM Data Transport transfers by setting a parameter in the `XCOM.SES` file for each connection to the remote XCOM Data Transport partner. If a session is not available for immediate use, the transfer is queued for subsequent execution.

To specify a remote system, replace *connection_profile* with the destination address or name that is specified by the user in the remote system parameter (REMOTE_SYSTEM, REMOTE_SYSTEM_RF, REMOTE_SYSTEM_SJ, or REMOTE_SYSTEM_SR).

> **NOTE**
> XCOM Data Transport reserves memory for information about the number of sessions for each destination based on the *connection_profile* entries in this file.

Transfers Using SNA

For transfers using SNA, the *connection_profile* parameter represents the remote system name or partner LU name to which the session limit is applied.

Transfers Using TCP/IP

For transfers using TCP/IP, specify the remote system in the format that your site uses. If the IP address is used, specify *connection_profile* in the form of *###.###.##.##*. If host names or domain names are used in the remote system parameter, specify *connection_profile* in the appropriate form of the name.

Specify a *connection_profile* for each form of address or name that is used.

> **NOTE**
> If the `XCOM.SES` file does not have an entry for the host name or domain name, *connection_profile* defaults to one.

Example

The following sample `XCOM.SES` file indicates the following information:

- A maximum of eight sessions can be used at any one time for locally initiated transfers to a remote system identified as XCOM01
- A maximum of four sessions can be used at any one time for locally initiated transfers to a remote system identified as XCOM02.
- A maximum of four simultaneous transfers can go to ###.###.##.##.

```
# connection_profile=number_of_sessions_allowed
#
XCOM01=8
XCOM02=4
###.###.##.##=4
```

Remote Session Control Parameters

The maximum number of sessions available for remotely initiated transfers and metatransfers is controlled by the internet service daemon on your Linux or Unix system. On most systems the internet service daemon is usually either inetd or xinetd.

Session Control Parameters for inetd

During the installation of XCOM Data Transport, entries for the XCOM Data Transport txpi services were added to the system inetd.conf file. Up to four services could have been added depending on the Linux or Unix system:

- txpi - XCOM Data Transport service for IPv4
- txpi6 - XCOM Data Transport service for IPv6
- txpis - XCOM Data Transport for IPv4 Secure Socket (SSL)
- txpis6 - XCOM Data Transport for IPv6 Secure Socket (SSL)

On many Linux or Unix systems, inetd provides several parameters that can be used to control the maximum number of sessions available.

> **NOTE**
> For more information about these parameters, see the *inetd* documentation.

Session Control Parameters for xinetd

During the installation of XCOM Data Transport, entries for the XCOM Data Transport txpi services were created in the $XCOM_HOME/txpi directory. Links for the XCOM Data Transport services were also added to the /etc/xinetd.d directory. Up to four services could have been added depending on the Linux or Unix system:

- txpi - XCOM Data Transport service for IPv4
- txpi6 - XCOM Data Transport service for IPv6
- txpis - XCOM Data Transport for IPv4 Secure Socket (SSL)
- txpis6 - XCOM Data Transport for IPv6 Secure Socket (SSL)

On many Linux or Unix systems, xinetd provides several parameters that can be used to control the maximum number of sessions available.

> **NOTE**
> For more information about these parameters, see the *xinetd* documentation.

Manage Queue

Use xcomqm at the command line to control the activity of the XCOM Data Transport transfer queue, including stopping transfers and displaying the status of a transfer.

The XcomQAPI allows your site to write programs to access the queue. The input and options for the XcomQAPI are the same as those described in the following topics. For more information about the XcomQAPI, see The Application Programming Interface.

xcomqm Command

The xcomqm command lets you maintain XCOM Data Transport queues by using the command-line interface. This command lets you delete entries from the queue, suspend transfers, resume suspended transfers, display the list of entries in the queue, and display detailed information about the queue entries.

Syntax

The syntax for xcomqm is as follows:

```
xcomqm [option][option] . . .
```

You can use more than one option on a command line. You can also use the same option more than once. Each option must be separated by a space. These options can be listed in any order, but note that when the shell encounters options such as the **-r** option, it performs the command and exits xcomqm without reading the rest of the command line.

Options

The following list explains the options for xcomqm:

-r
Display the release level of xcomqm and exit.

-A*entryname*
Release a held transfer.

-C*entryname*
Get a trace of a transfer.

-D*entryname*
Display details about a queue entry.

-H*entryname*
Hold a scheduled transfer.

-L[a]
List your queued transfers. Use -La for all queued transfers.

-R*entryname*
Remove a queue entry.

-Rf*entryname*
Force the entry to be removed from the queue. Use this command when you want to remove a transfer that is still active.

-R*
Remove all completed or scheduled entries from the queue.
If you use this option on an active transfer, it interrupts the transfer.

-T*entryname*
Terminate an active transfer.

-T[f]*entryname*
Force termination of the transfer. Use this command when you want to remove a transfer that is still active.

> **NOTE**
> When using SNA/APPC protocols, the TP will ABEND and it may be necessary to restart the underlying SNA software. When using TCP/IP, xcomtcp will ABEND.

-S*entryname*
Suspend a transfer.

-E*entryname*
Resume a suspended transfer.

-d
Display debugging information.

no option
Display the help menu.

-z
Display logging text for transfer.

> **NOTE**
> *entryname* represents the Transaction ID, which is the six-digit name of the transfer entry in the queue.

Purge the Log File

CLEANLOG purges entries from the XCOM Data Transport transfer and error log. Run this program on a regular basis to ensure that the log file does not grow too large. Back up the XCOM.LOG before running this program as all items older than the specified age are deleted.

> **NOTE**
> XCOM Data Transport should be idle when you run this program.

Syntax

The syntax for CLEANLOG is as follows:

```
CLEANLOG number_of_days_old [log_file_name]
```

Where:

number_of_days_old
Entries in the log longer than the number of days specified are purged.

log_file_name
The name of the log file to clean (optional). The default is the XCOM.LOG file in the current directory. If not specified, the log file name defaults to $XCOM_HOME/xcom.log where $XCOM_HOME is an environment variable.

Example

```
CLEANLOG 5
```

This command removes entries in XCOM.LOG that are older than five days.

xcompre Pre-allocation Exit

Use the xcompre exit to validate all locally and remotely initiated transfers before they begin, and to customize parameters for that transfer.

The xcompre exit is invoked by specifying the XPRECMD parameter in xcom.glb. If XPRECMD is set when xcomd is invoked, the pre-allocation exit is active for remotely initiated transfers. Activating the exit gives the system administrator more access controls. For example, the system administrator can allow only a particular user to perform a transfer, or only allow transfers to or from a particular directory.

Customization is useful when the remote user omits the true file name and the local system uses a database to map the specified name to the true local name. Certain e-mail application systems work in this way. For example, an incoming transfer specifies a certain file name, but the administrator of the local system that is receiving the file wants to place the data in a different file, depending on the user ID of the sender. The script retrieves parameter values and sends the relevant information to standard output using the -g option and, if desired, changes the values using the -p option.

How to Use xcompre

An xcompre shell script is provided with XCOM Data Transport as a sample script, and is available online. The script runs an exit on the local system before the transfer starts. You can tailor the script to meet individual user requirements. The xcompre exit collects and passes transfer ID information (tid) to xcomqm, and also passes the parameter values that are specified by the -g and -p options. The -g and -p options in the sample script are intended for use only with this exit.

The pre-allocation exit is invoked after XCOM Data Transport receives the XCOM Data Transport header from the remote partner, but before the transfer commences.

For remotely initiated transfers, the pre-allocation exit is invoked after XCOM Data Transport receives the XCOM Data Transport header from the remote partner, but before the transfer commences. For locally initiated transfers, the pre-allocation exit is invoked based on the value of the global parameter XCOMPRE_LOCAL. If XCOMPRE_LOCAL is set, the pre-allocation exit is invoked before sending the XCOM Data Transport header to a remote partner.

xcompre Options

The following options are available for xcompre and their associated parameters:

-g

Passes the transfer ID to xcomqm. This option also retrieves the values for the following parameters from the local database and passes them to standard output:

- FILE_OPTION
- LOCAL_FILE
- REMOTE_FILE
- REMOTE_LU
- TRANSFER_TYPE
- USERID

-p

Changes the values for the following parameters based on the specified transfer ID (tid) and applies them to the transfer:

- FILE_OPTION
- LOCAL_FILE
- REMOTE_FILE

Sample Script

In the following example, only file transfers that have payroll or finance in the USERID are permitted. If the USERID does not meet one of these criteria, the transfer is rejected. If the transfer is permitted, then the -p option specifies the values to use for LOCAL_FILE, REMOTE_FILE, and FILE_OPTION.

```
#!/bin/sh# @(#)xcompre.sh#       This procedure is invoked by the CA XCOM Data Transport
#       transaction program for all locally and remotely initiated
#       transfers before they begin.

tid=$1

echo $1 > /tmp/$tid.tidlog

# required for HPUX-10
if [ `uname` = 'HP-UX' ]
then PATH=/opt/xcom/bin:$PATH
fi

xcomqm -g$tid |grep USERID > /tmp/$tid.userlog

'egrep' 'payroll|finance' /tmp/$tid.userlog

if [ $? -ne 0 ]
    then
    exit 100
else
    xcomqm -p$tid LOCAL_FILE=/u/jc/test/000040.local  \
    REMOTE_FILE=abcdefg
    FILE_OPTION=CREATE
    2>&1 > /tmp/$tid.errlog
    exit 0
fi
```

Post Processing Scripts

Use these scripts to handle processing after you have received a file. They are used for controlling print spooling, for managing notification facilities, and for other processing of incoming files after a successful file transfer is received.

The post processing shell scripts include xcompp, xcomend, xcomlp, and xcomntfy. These are provided with XCOM Data Transport as sample scripts and can be tailored to meet individual user requirements, if necessary.

Shells and Script Changes

XCOM Data Transport uses the shell that is specified by the SHELL_CMD parameter for processing scripts. If you customize a script using a shell other than the XCOM Data Transport default, you should change the default. However, all other XCOM Data Transport scripts will then use that default which may affect the behavior of these scripts.

Process Summary

Perform the following steps to modify a post processing script.

1. 1. Invoke an editor, such as vi, and open the script you want to modify.
2. 2. Go to the bottom of the script.
3. 3. Add the desired commands to the end of the script

   NOTE

   If the script ends with an exit statement, enter your commands above exit.
4. Save the file and exit from the editor.

Troubleshooting

For script problems, uncomment the debugging statements included in the sample script and rerun the script. The debugging section is clearly marked. Debugging information is written to the file specified in the first statement.

Modify Post Processing Parameters

Use the xcompp script after a successful file transfer is received. The XCOM Data Transport parameters that appear in the script can also be used for any modifications that are needed.

About xcompp

The xcompp script is a shell script that contains parameters for post processing of a file. After a successful file transfer is received, xcompp is always turned on (available) and will get invoked automatically. It is only invoked for incoming files and does not work for jobs or reports.

You can reference the parameters in xcompp by putting commands at the end of the script. If no modifications are made to the script, Xcompp does not take any visible actions.

Sample Script

```
#!/bin/sh
# @(#)xcompp.sh     1.1     3/27/92 20:19:33
#   This procedure is invoked by the CA XCOM Data Transport transaction
#   program after the transfer is finished
#   before the conversation is terminated.
#   Not every argument will be populated;
#   the values contained in the arguments depend on those
#   provided by the remote system.

###### START OF DEBUG SECTION ######
######
###### UNCOMMENT THE LINES BELOW TO DEBUG THIS SCRIPT ######
#exec >> /tmp/xcompp.out
#exec 2>&1
#set -vx
#PS4='[$0: $LINENO]+ '
#export PS4
###### END OF DEBUG SECTION ######

compression=$1          # Compression flag
shift
notify_flag=$1          # Notify flag
shift
notify_name=$1          # Notify Name
shift
remoteuser=$1           # Notify User
shift
filetype=$1             # File type (Whether it's file, job or report)
shift
fileaction=$1           # File action (Create, replace or append)
shift
datasettype=$1          # Dataset type
shift
carriage_flag=$1        # Carriage return flag
shift
code_flag=$1            # Code flag
shift
recfm=$1                # Record format
shift
lrecl=$1                # Logical record length
shift
remote_reqno=$1         # Remote request Number
shift
local_reqno=$1          # Local Request number
shift
group=$1                # Group name
shift
sysdata=$1              # System Dependent User Data
shift
xferdata=$1             # Transfer Dependent User Data
shift
ident=$1                # Ident
shift
truncation_flag=$1      # Truncation flag
```

```
shift
tmp_file=$1             # Local temporary file name
shift
file=$1                 # Local file name
shift
remote_file=$1          # Remote file name

#    POSTPROCESS HERE !
exit 0
```

Example

The following example shows how you change the permissions on a received file by adding the following snippet to the end of the xcompp script:

```
chmod 777 $tmp_file
```

This changes the permissions on tmp_file to read, write, and execute.

NOTE
Type a $ sign before the variable to reference the variable in the xcompp script.

Modify Post Transfer Script

Use the xcomend script for all types of transfers, after a transfer is finished and partner communications have ended, whether the transfer completed successfully or not.

About xcomend

The xcomend script is a script that is invoked by a CA XCOM Data Transport transfer program. This script can be used as provided, or it can be used by system administrators to modify printing commands and parameters to provide additional information about all types of transfers (remote and local; send and receive; successful or failed). Different arguments are passed depending on whether the transfer is a file, job, or report. This script can be used to define the context of the transfer, such as indicating that a remotely initiated send transfer failed with a particular message.

Sample Script

```
#!/bin/sh
# @(#)xcomend.sh     1.0     8/1/96 12:23:04

#   This procedure is invoked by the CA XCOM Data Transport transfer
#   program after the transfer is finished (whether successful or not).
#   Not every argument will be populated;
#   the values contained in the arguments depend on those provided by the
#   remote system.

###### START OF DEBUG SECTION ######
######
###### UNCOMMENT THE LINES BELOW TO DEBUG THIS SCRIPT ######
#exec > /tmp/xcomend.$1.out
#exec 2>&1
#set -vx
#PS4='[$0: $LINENO]+ '
```

```
#export PS4
###### END OF DEBUG SECTION ######

#
#  The first parameters are supplied for any type of transfer.

local_reqno=$1    # local request number/tid (000000 if unassigned)
shift
initiator=$1      # LOCAL or REMOTE
shift

transfer_type=$1  # FILE or JOB or REPORT
shift
direction=$1      # SEND or RECEIVE
shift
restarting=$1     # RESTARTING or FIRST_TRY
shift
start_time=$1     # Start time of transfer
shift
end_time=$1       # End time of transfer
shift
remote_system=$1  # Remote system name
shift
status=$1         # Status of transfer
shift
error=$1          # XCOM file transfer numeric error code
shift
msg=$1            # error code translated to message text
shift
status_msg=$1     # status message from partner when error on remote sys
shift

remoteuser=$1     # Remote user
shift

remote_reqno=$1   # Remote request Number
shift
group=$1          # Group name
shift
sysdata=$1        # System Dependent User Data
shift
xferdata=$1       # Transfer Dependent User Data
shift
transfer_name=$1  # Transfer Name
shift

tmp_file=$1       # Local temporary file name
shift
file=$1           # Local file name
shift
remote_file=$1    # Remote file name
shift
```

```
carriage_flag=$1  # Carriage return flag
shift
code_flag=$1      # Code flag
shift

compression=$1    # Compression flag
shift

file_recs=$1      # No. of records read/written
shift
file_bytes=$1     # No. of bytes   read/written
shift
blocks=$1         # No. of blocks  sent/received
shift
bytes=$1          # No. of bytes   sent/received
shift

#
#  The next parameters are supplied for file transfers only

if [ $transfer_type = "FILE" ]
then

fileaction=$1         # File action (C, R, or A for create, replace, or append)
shift
datasettype=$1        # Dataset type
shift
recfm=$1              # Record format
shift
lrecl=$1              # Logical record length
shift
truncation_flag=$1    # Truncation flag
shift

#
#  The next group of parameters are supplied for report transfers only

elif [ $transfer_type = "REPORT" ]

then

jobname=$1            # Job name field from JES
shift
jobnumber=$1          # Job number field from JES
shift
class=$1              # print class
shift
copies=$1             # Number of copies to print
shift
form=$1               # Type of form to print this job on.
shift
recfm=$1              # Record format of incoming print job.
shift
```

```
lrecl=$1              # Logical record length of incoming report.
shift
blksize=$1            # Block size of incoming report.
shift
ucs_name=$1           # Name of UCS to be used for this print job.
shift
fcb=$1                # Name of FCB (form control block) for this report.
shift
room_number=$1        # Room number field from JES.
shift
programmer_name=$1    # Programmer name field from JES.
shift
tso_notify=$1         # TSO notify field from JES.
shift
destination=$1        # Destination printer specification.
shift
carriagecontrol=$1    # Type of carriage control characters being used.
shift

#
# There are no extra parameters for remote jobs

fi

#
# PUT YOUR OWN XCOM ENDING CODE HERE !

exit 0
```

Modify Printing Processes

Use this script after a print job is received from a remote system. Use the script as provided or modify it, if necessary. The sample script assumes that the default print queue is used for printing XCOM Data Transport print jobs.

About xcomlp

The xcomlp script is a shell script that is invoked by the XCOM Data Transport transaction program that is processing a report transfer.

The contents of the parameters passed to xcomlp depend on what is supplied on the initiating side of the transfer. The only field that can be relied on is the file name. The sample procedure also assumes that the COPIES= field is valid, in addition to assigning each incoming argument to an appropriately named variable. For information about the processing of machine code characters, see Reports Containing Machine Code Characters.

Sample Script

```
#!/bin/sh
# @(#)xcomlp.sh  1.1     3/27/92 20:19:33

#   This procedure is invoked by the CA XCOM Data Transport transaction
#   program after an incoming print request is received. Not every argument will
#   be populated; the values contained in the arguments depend on those provided
#   by the remote system. The most general case is that of a host partner that
#   uses the user exit EXIT02  to extract and transmit the JES parameters
#   for the print job.

#   In all cases, the file and copies parameters will be valid.

###### START OF DEBUG SECTION ######
######
###### UNCOMMENT THE LINES BELOW TO DEBUG THIS SCRIPT ######
#exec > /tmp/xcomlp.out
#exec 2>&1
#set -vx
#PS4='[$0: $LINENO]+ '
#export PS4
###### END OF DEBUG SECTION ######

rluname=$1      # Remote lu name
shift
jobname=$1      # Job name field from JES
shift
jobnumber=$1    # Job number field from JES
shift
class=$1        # print class
shift
copies=$1       # Number of copies to print
shift
form=$1         # Type of form to print this job on.
shift
recfm=$1        # Record format of incoming print job.
shift
lrecl=$1        # Logical record length of incoming report.
shift
blksize=$1      # Block size of incoming report.
shift
ucs_name=$1     # Name of UCS to be used for this print job.
shift

fcb=$1          # Name of FCB (form control block) for this report.
shift
room_number=$           # Room number field from JES.
shift
programmer_name=$1      # Programmer name field from JES.
shift
tso_notify=$1           # TSO notify field from JES.
shift
file=$1   # Name of temporary file into which XCOM 6.2 has placed report
shift
destination=$1          # Destination printer specification.
shift
nodespec=$1             # connection profile name
shift
carriagecontrol=$1      # Type of carriage control characters being used.
```

```
lp  ${copies:+-n$copies} < $file
```

Customize Notification Process

The xcomntfy script is used, when notification is requested, to notify users that a transfer has completed successfully. System administrators can modify this script to customize notification procedures, if necessary. For any other post-processing modifications, we recommend that you use xcompp.

About xcomntfy

XCOMNTFY uses one of the following programs to read the queue entry to determine the user id and password to send an email.

- xcommail to use the MAPI (Messaging Application Programming Interface)
- xcomsmtp to use the SMTP interface

Notification is done using the notify script, which is defined to XCOM Data Transport with the parameter `XNOTIFYCMD $XCOM_HOME/cmd/xcomntfy` in the xcom.glb global file. $XCOM_HOME is an environment variable.

You need to set up the notify script to use xcommail or xcomsmtp. A sample notify script is distributed with XCOM Data Transport.

Sample Script

The supplied script uses mailx(1) to notify the named user logins.

```
#!/bin/sh
# @(#)xcomntfy.sh    1.2    3/28/92 08:42:41
#   This procedure (xcomntfy) is called by CA XCOM Data Transport
#   transaction program when the NOTIFY parameter is set to YES.
#
#   The arguments passed to the xcomntfy procedure are:
#
#   -q q_entryname
#
#   -h how
#
#   The following $how will be treated as 'M' (mail to user) by default:
#       T - TSO;
#       R - ROSCOE;
#       C - CICS;
#       L - another LU;
#       M - mail to user.
#   The following $how will be treated as 'W' (write to user) by default:
#       W - write to user.
#   The following $how will be treated as 'A' (write to all users) by
#   default:
#       A - all;
#
#   -u login
#
```

```
#   Note: 1.Other information about the transfer can be extracted
#       from the output of xcomqm -D$q_entryname command
#       for postprocessing
#         2.If specified in $how method fails, the mail message will be
#       sent to $LOGNAME or root.

#uncomment following to diagnose problems with script
# set -xv

if [ `uname` = 'SunOS' ]
then MAIL=Mail
else
     MAIL=mailx
fi

# required for HPUX-10


if [ `uname` = 'HP-UX' ]
then PATH=/opt/xcom/bin:$PATH
fi

while test $# -gt 0
do
    case "$1" in
        -h)
            how="$2"
            shift;;
        -q)
            q_entryname="$2"
            shift;;
        -u)
            login="$2"
            shift;;
    esac
    shift
done
case "$how" in
    [TRCLMV]* )
        xcomqm -D$q_entryname|
        $MAIL -s "xfer $q_entryname" ${login:-${LOGNAME:-root}}
>/dev/null 2>&1
        ;;
    [W]* )
        xcomqm -D$q_entryname|
        if [ "$login" != "" ]
        then
        (write $login || $MAIL -s "xfer $q_entryname"
${login:-${LOGNAME:-root}})  >/dev/null 2>&1
        fi
        ;;
    [A]* )
        xcomqm -D$q_entryname|
        (wall || $MAIL -s "xfer $q_entryname" ${login:-${LOGNAME:-root}})
>/dev/null 2>&1
        ;;
    * )
        ;;
esac

# put user defined notification
# script here

exit 0
```

Requirements for SMTP Notifies

Following the requirements for addresses and related parameters for SMTP notifies.

- From Addresses
- To Addresses

From Addresses

This section describes the requirements for to addresses and related parameters for SMTP notifies (local and remote).

Local Notifies

For an SMTP email *local* notify, XCOM Data Transport checks the following parameters to use as the from address:

- XCOM_USERID (in the global file)
  If neither of these parameters has been specified, then the results are as follows: No notify happens.
- XCOM Data Transport displays an error message.
- Set the following parameters:
- LCLNTFYL={ALL|WARN|ERROR}
- LOCAL_NOTIFY=(local user)
- MAIL_TYPE=SMTP (global file parameter)
- NOTIFYL=MAIL
- SMTP_SERVER={SMTP server address|name} (global file parameter)
- XNOTIFYCMD=$XCOM_HOME/cmd/xcomntfy (global file parameter)

Remote Notifies

For an SMTP email remote notifies, XCOM Data Transport checks the following parameters to use as the from address:

- First, the USERID of the remote user (in the configuration file)
- Second, XCOM_USERID (in the global file)

If parameters have of these been specified, then the results are as follows:

- No notify happens.
- XCOM Data Transport displays an error message.

NOTE

Note: The following parameters must also be set:

- MAIL_TYPE=SMTP (global file parameter)
- NOTIFY_NAME=(remote user)
- NOTIFYR=MAIL
- RMTNTFYL={ALL|WARN|ERROR}
- SMTP_SERVER={SMTP server address|name} (global file parameter)
- XNOTIFYCMD=$XCOM_HOME/cmd/xcomntfy (global file parameter)

To Addresses

This section describes the requirements for to addresses and related parameters for SMTP notifies (local and remote).

Local Notifies

For an SMTP email *local* notify, XCOM Data Transport uses the following parameter as the to address:

- LOCAL_NOTIFY (in the configuration file)
  If this parameter has not been specified, then the results are as follows:
- No notify happens.
- XCOM Data Transport displays an error message.

NOTE

The following parameters must also be set:

- LCLNTFYL={ALL/WARN/ERROR}
- MAIL_TYPE=SMTP (global file parameter)
- NOTIFYL=MAIL
- SMTP_SERVER={SMTP server address|name} (global file parameter)
- XNOTIFYCMD=$XCOM_HOME/cmd/xcomntfy (global file parameter)

Remote Notifies

For an SMTP email *remote* notify, XCOM Data Transport uses the following parameter as to the address:

- NOTIFY_NAME (in the configuration file)
  If this parameter has not been specified, then the results are as follows:
- No notify happens.
- XCOM Data Transport displays an error message.

NOTE

The following parameters must also be set:

- MAIL_TYPE=SMTP (global file parameter)
- NOTIFYR=MAIL
- RMTNTFYL={ALL/WARN/ERROR}
- SMTP_SERVER={SMTP server address|name} (global file parameter)
- XNOTIFYCMD=$XCOM_HOME/cmd/xcomntfy (global file parameter

Pluggable Authentication Modules (PAM) Based Authentication

Learn how administrators can select PAM-based authentication instead of the native mechanism available for the UNIX server.

PAM Overview

The Pluggable Authentication Modules (PAM) are an industry-standard framework providing authentication, account management, session management, and password services. PAM allows programs that rely on authentication to be written independently of the underlying authentication scheme. PAM uses the local password file to authenticate the user that is accessing the host. You can now take advantage of other authentication mechanisms such as LDAP.

Enable PAM Authentication

To enable PAM authentication, modify the following global parameters:

- Set the authentication type to PAM:
  ```
  AUTH_TYPE=PAM
  ```
- Specify the path for the PAM shared library:
  ```
  PAM_PATH=/usr/lib
  ```

Configure the PAM Service

Configure PAM through the `/etc/pam.conf` file or the `/etc/pam.d` directory. XCOM uses the service name of `xcomauth`.

- `/etc/pam.conf`
  Update the file with the `xcomauth` service name and the authentication settings that are detailed in the requirements.
  See the following example:
  ```
  xcomauth auth XXXX XXXX
  xcomauth account XXXX XXXX
  ```
- `/etc/pam.d` directory
  Each policy is contained in a separate file bearing the name of the service that it applies to. Create an `xcomauth` configuration file with the authentication settings that are detailed in the requirements.
  See the following example:
  ```
  auth required /opt/CA/XCOM/redistrib/pam_userpass/pam_userpass.so
  auth required pam_env.so
  auth sufficient pam_sss.so use_first_pass
  auth sufficient pam_unix.so nullok use_first_pass
  auth required pam_deny.so

  account required pam_access.so
  account required pam_unix.so broken_shadow
  account sufficient pam_localuser.so
  account sufficient pam_succeed_if.so uid < 500 quiet
  account [default=bad success=ok user_unknown=ignore] pam_sss.so
  account required pam_permit.so
  ```

  > **NOTE**
  > If you are using Linux PAM, the `pam_userpass` module must precede the actual authentication module. This module is distributed with XCOM and is installed in the `$XCOM_HOME/redistrib/pam_userpass` directory. For more details on PAM configuration, see the PAM documentation.

Symbolic Parameters in Configuration Files

Symbolic parameters let you store transfer parameters in configuration files with variable data that is resolved to other values at schedule time.

Standard symbolic parameters are supplied with XCOM Data Transport.

No setup is required to use the pre-defined variables that are integrated as part of XCOM Data Transport. If you place these variables in the parameter data set for a transfer, XCOM Data Transport invokes them when scheduling the transfer.

&DATE(format-code)

Causes the current date to be substituted dynamically in the current keyword value. The format of the date depends on the format code that is selected as a sub-parameter. If no format is specified, MMDDYYYY is used. Valid format-codes and examples of their output are:

- *MMDDYYYY* -- 12312012
- *DDMMYYYY* -- 31122012
- *YYYYMMDD* -- 20121231
- *YYMMDD* -- 121231
- *YYYY* -- 2012
- *MM* -- 12
- *YY* -- 12
- *DD* -- 31
- *YYDDD* -- 12366
- *YYYYDDD* -- 2012366
- *DDMONYYYY* -- 31DEC2012
- *MON* -- DEC
- *MONTH* -- DECEMBER

&ID

Causes the value that is entered for ID to be substituted dynamically in the current keyword value.

&IPNAME

Causes the value that is entered for IPNAME to be substituted dynamically in the current keyword value.

&LU

Causes the value that is entered for LU to be substituted dynamically in the current keyword value.

&LUSER

Causes the current local user ID (or the USERID of the current job) to be substituted dynamically in the current keyword value.

&TIME(format-code)

Causes the current time to be substituted dynamically in the current keyword value. The format of the time depends on the format code that is selected as a sub-parameter. Valid format-codes and examples of their output are:

- *HHMMSSTH* -- 15312811
- *HHMMSS* -- 153128
- *HHMM -- 1531*
- *MMSS* -- 3128
- *HH* -- 15
- *MM* -- 31
- *SS* -- 28
- *TH* -- 11
- *TH3* -- 110

&USERID

Causes the current remote user ID (or the USERID of the current job) to be substituted dynamically in the current keyword value.

Parameters Supporting Symbolic Variables

You can use symbolic parameters with the following parameters:

- DESTINATION
- FORM
- LOCAL_FILE, LOCAL_FILE_RF, LOCAL_FILE_SJ, LOCAL_FILE_SR, LOCAL_NOTIFY, LUSER
- NOTIFY_NAME
- OEDATE, OETIME, OFLMAX, OFLMIN, OID, OLMSG, OREQ, OSDATE, OSTIME, OTNAME, OUSER
- REMOTE_FILE, REMOTE_FILE_RF, REMOTE_SYSTEM, REMOTE_SYSTEM_RF, REMOTE_SYSTEM_SJ, REMOTE_SYSTEM_SR, REPORT_TITLE
- TRANSFER_ID, TRANSFER_USR_DATA
- UNIT, UNIT_RF, USER_DATA, USERID
- VOLUME, VOLUME_RF
- XCOM_CONFIG_SSL, XTCERRDECR, XTCERRINCR, XTCERRPURGE, XTCERRREL, XTCGOODDECR, XTCGOODINCR, XTCGOODPURGE, XTCGOODREL, XTCJOB, XTCNET

Command Examples for Metatransfers and Inquire on Status

The following sample script file performs a metatransfer and an inquire, and produces an Easytrieve report:

```
xcomtcp -c5 -f /tmp/asm2.cnf stcip=xcomwinca.ca.com
stcport=48316 inq_file=/tmp/xcom.inq.file
xcomtcp -c6 stcip=xcomwinca.ca.com stcport=48316
inq_file=/tmp/xcom.inq.file inq_wait=000005
hist_file=/tmp/xcomhist
echo Process Returned $?
```

NOTE

- The - c5 metatransfer parameters LUSERID and LPASSWORD are not shown in the previous example but are required and can be specified in the configuration file.
- Easytrieve reports cannot be generated using XCOM Data Transport for Linux PC.
  To generate an Easytrieve report, export the xcomhist file that is generated by the preceding script to a Windows XCOM Data Transport machine and follow the instructions in How to Generate Easytrieve Reports on History Records.

Suppose that this script file is named xmeta_script and it is located in /tmp.

To execute this script

1. Submit the following command to change to the directory where xcomtcp is located:

   ```
   cd $XCOM_HOME
   ```

   NOTE
   This step is not required when the XCOM Data Transport directory is in the system path.
2. Submit the following command to run the script file:

   ```
   ./tmp/xmeta_script
   ```

Application Programming Interface

The Application Programming Interface (API) is intended to give programmers the capability of developing their own XCOM Data Transport applications.

XCOM Data Transport APIs

The two XCOM Data Transport APIs, as follows:

XcomAPI

The XcomAPI routine uses C structures that are defined in the xcomapi.h file to pass information to XCOM Data Transport. The XCOM Data Transport XcomAPI function call takes a starting state and a parameter block structure as its arguments. For transfers using SNA, before invoking XcomAPI, an SNA session must be active with the desired remote system.
xcomapi.h can be at $XCOM_HOME/api/include

XcomQAPI

The XcomQAPI routine uses C structures that are defined in the xcomapi.h file to return information from the XCOM Data Transport Queue. The XCOM Data Transport XcomQAPI function call takes a parameter field as its argument, which allows you to delete entries from the queue, suspend a transfer, resume a suspended transfer, display the list of entries in the queue, and display detailed information about the queue entries.
For details about XcomQAPI option parameters and return codes, see XcomQAPI.

The XcomAPI provides the ability to submit transfers to XCOM Data Transport. The XcomQAPI provides the same functionality as the XCOMQM command line, but through an API.

Utility APIs

Also, the two utility APIs that work with these two APIs, as follows:

msgstring

The msgstring API is a utility API. It converts a return code to an appropriate message. The XCOM Data Transport msgstring API() function call takes a return code and sense as its arguments. The return code is the return value of XcomAPI or XcomQAPI.
Example: Suppose XcomAPI or XcomQAPI returns a code other than zero. This return code is passed to the msgstring API, which returns a message corresponding to the return code.

XsetSystemPriv

XsetSystemPriv is a utility API that allows xcomtcp and xcom62 to return to system privileges (user=root, group=xcomadm) after a transfer has been completed. The XCOM Data Transport XsetSystemPriv API() function call takes a type of privilege as its argument.
This function needs to be called at the end of transfer. Refer to API example to know usage of this function.

API Examples

The following API is an example of a file transfer. It uses the XcomAPI routine to pass information to XCOM Data Transport. This sample API program is the apitest.c file distributed with XCOM Data Transport. (For the location of this file, see the *XCOM Data Transport Installation Guide* for your platform.)

> **NOTE**
> The API example provided on your distribution media may have been updated for your XCOM Data Transport system and may be different from the example shown below.

C Language API File Transfer to Remote System File Example

The C Language API is distributed as file xcomapi.h in the directory $XCOM_HOME/api/include.

Queue Request Using XcomQAPI Routine

The Queue Request API is distributed as file qapitest.c in the directory $XCOM_HOME/api/src.

XcomQAPI

The XcomQAPI routine uses C structures that are defined in the xcomapi.h file to return information from the XCOM Data Transport Queue. The XCOM Data Transport XcomQAPI function call takes a parameter field as its argument, which allows you to delete entries from the queue, suspend a transfer, resume a suspended transfer, display the list of entries in the queue, and display detailed information about the queue entries.

Using XcomQAPI Option Parameters

The following XcomQAPI option parameters are for use by applications programmers to pass and retrieve information in a C structure to and from the XcomQAPI routine. These parameters need to be supplied by you and passed on to the XcomQAPI. The rest of parameters are a received buffer. They will receive data from XcomQAPI.

The option parameters in the following list are used to pass and retrieve information from the queue request.

funccode

Used to define a function parameter to pass to XcomQAPI().

r

Display the release level of xcomqm and exit.

A*entryname*

Release a held transfer.

C*entryname*

Get a trace of a transfer.

D*entryname*

Display details about a queue entry.

H*entryname*

Hold a scheduled transfer.

L[ ]

List your queued transfers.

La

List all queued transfers.

R*entryname*

Remove a queue entry.

Rf*entryname*

Force the entry to be removed from the queue. Use this command when you want to remove a transfer that is still active.

R*

Remove all completed or scheduled entries from the queue. If you use this option on an active transfer, it interrupts the transfer.

T*entryname*

Terminate an active transfer.

Tf*entryname*

Force termination of the transfer. Use this command when you want to remove a transfer that is still active.

> **WARNING**
> When using SNA/APPC protocols, the TP will ABEND and it may be necessary to restart the underlying SNA software. When using TCP/IP, xcomtcp will ABEND.

S*entryname*

Suspend a transfer.

E*entryname*

Resume a suspended transfer.

d

Create a trace file.

g*entryname*

Display queue entry information.

p*entryname*

Used to change the value of: FILE_OPTION LOCAL_FILE REMOTE_FILE.
Note: This option can be called only when xcbp->state is in PREALLOCATION_STATE.

> **NOTE**
> The *entryname* variable represents the Transaction ID, which is the six-digit name of the transfer entry in the queue.

nMax_Queue_Entries

Defines maximum queue entries. The value should be set to equal MAX_QUEUE_ENTRIES in the xcom.glb file.

queue_open_once

Open queue flag. Define this only once at first call of XcomQAPI. The value should be set to 0.

tid

A queue entry TID number.

trace_level

Defines the trace level. The value can be set to 0, 1, or 9.

trace_filename

It defines a trace file name and its location; for example:

$XCOM_HOME/trace/xcomqapi.tra.

NOTE
$XCOM_HOME is an environment variable.

XcomQAPI Return Codes

The return codes for XcomQAPI are as follows:

603
The TP state is invalid.

602
There was a general error in the queue function.

601
Permission denied.

600
Transfer is not active.

469
XCOM Scheduler service service is not running or is not compatible.

-1
The named file does not exist or is not accessible in the given mode.

-9
XCOMQM detected an unknown error.

-1100
XCOMQM is unable to find the entry in the queue.

-1801
XCOMQM is unable to open the index file.

-1802
XCOMQM is unable to close the index file.

-1803
XCOMQM detected an invalid command line option.

Using API Member Names

The API member names are used to create C structures that specify aspects about a transfer. The appropriate information is then passed to XCOM Data Transport by the XcomAPI routine.

For information about the XcomAPI member names, see the following links:

- ALLOCATION_TYPE
- AVGREC
- BLKSIZE
- CARRIAGE_CONTROL_CHARACTERS
- CARRIAGE_FLAG
- CHECKPOINT_COUNT
- CLASS
- CODE_FLAG
- CODETABL
- COMPRESS
- COMPRESS_PDS
- CONFIGSSL
- CONVERT_CLASSES
- COPIES
- CREATEDELETE
- DATACLAS
- DEBUG_FLAG
- DEN
- DESTINATION
- DISPOSITION
- DOMAIN
- DSNTYPE
- EAR_CIPHER
- EAR_DIGEST
- EAR_HASH
- EAR_KEY
- EATTR
- EOL_CLASSES
- EXPDT
- FCB
- FILE_OPTION
- FILE_TYPE
- FORM
- HOLD
- HOLD_TRANSFER
- LABEL
- LABELNUM
- LCLNTFYL
- LEAR_CIPHER
- LEAR_DIGEST
- LEAR_HASH
- LEAR_KEY
- LOCAL_CHARSET
- LOCAL_DELI
- LOCAL_FILE
- LOCAL_FILE_RF
- LOCAL_FILE_SJ
- LOCAL_FILE_SR
- LOCAL_NOTIFY
- LRECL
- MAXRECLEN
- MBCS_CONVERROR
- MBCS_INPUTERROR
- METACODE_CLASSES

Upgrading from Previous Releases

The XcomAPI and XcomQAPI distributed with previous releases of XCOM Data Transport have changed for this release. If you have versions of XcomAPI and XcomQAPI that were distributed before this release, recompile your XCOM Data Transport applications with the current version.

The XcomAPI distributed with previous versions of XCOM Data Transport is different from this API. If you have versions of the API that were distributed prior to this version, recompile your XCOM Data Transport applications with the current version.

The following changes have been made to the API:

- XCOM Data Transport now uses the following APIs:
  - XcomAPI
  - XcomQAPI
  - msgstring
  - XsetSystemPriv
- In previous releases, API users used XCOM Data Transport static/archive libraries (libxcom.a, libxcomtcp.a, libxcomtxpi.a, and libxcomcomp.a) to compile and link custom applications. Now, you must use a shared library (libxcomshared.so) to compile and link your custom applications. Therefore, API users can use libxcomshared.so to perform XCOM Data Transport transfers.

Changes to Access Permissions

Compile and link your programs, then perform the commands shown below, replacing *your_program* with the name of your program, as follows:

```
chown root your_program
chgrp xcomadm your_program
chmod 6755 your_program
```

These commands set the appropriate access permissions for using XCOM Data Transport with your program.

Link Libraries

When linking libraries, you must use the appropriate XCOM Data Transport library for transfers that use SNA/APPC or TCP/IP protocols.

Sample links for use with both SNA and TCP/IP transfers are provided in the makefile and qapimake files distributed with XCOM Data Transport. For the locations of the library files and the API source files, see the *Installation Guide* for your platform.

Starting States

Valid starting states are as follows:

LOCAL_SEND
Initiates the transmission of a local file to a remote system as a report, job, or file transfer.

LOCAL_RECEIVE
Initiates the transmission of a remote file to the local system.

Return Values
The value returned by XcomAPI is zero if the transfer was successful. If not, an error code as defined in the startst.h header file is generated.

Starting States in startst.h

The startst.h file contains the following:

- /* starting states to pass to XcomAPI */
- #define LOCAL_SEND 152
- #define LOCAL_RECEIVE 153

XCOM Parameters

Use parameter values to control the variables that govern the behavior of XCOM Data Transport

Using Parameters

Parameter values can be defined to XCOM Data Transport as indicated in the table below. The values take precedence depending on where they are specified. The order in which parameter values take precedence is as follows:

Order of Precedence	Parameter Value Specified in	Explanation
1	Command line	If the user specifies a value for a parameter on the command line, this overrides every other specification in the files or the program.
2	xcom.cnf or *filename.cnf* *filename.xml or* *GUI transfer record*	If the value is specified in the xcom.cnf file, or in a user-customized configuration file, *filename.cnf*, or in the transfer record created in the GUI or in an .xml configuration file saved through the GUI, it overrides the value in the xcom.glb file.
3	xcom.glb	If the value is specified in the xcom.glb file, it overrides the value in the program. The values specified in xcom.glb are generally used by the system administrator to start XCOM Data Transport.
4	Program defaults	If a value is not specified anywhere, the program has its own defaults.

Parameter Format

When using XCOM Data Transport from the command prompt or with a script, and when editing configuration files, the XCOM Data Transport parameters consist of assignment statements. The format for assignment statements is as follows:

- PARAMETER_NAME (always all uppercase, with underscore character (_) when indicated)
- An equal sign (=)
- A character string terminated by a new-line

> **NOTE**
> Parameter values are always in uppercase. However, when specifying directories, file names, user IDs and passwords, you may use uppercase and lowercase.

Syntax

The syntax for assignment statements is as follows:

```
PARAMETER_NAME=value
```

Example

In the following example, the parameter EXPIRATION_TIME is set to a value of 6000 seconds.

```
EXPIRATION_TIME=6000
```

This controls the maximum time in seconds that a transaction is held in the transfer queue after execution.

Guidelines

Note the following guidelines for using XCOM Data Transport parameters:

- When you type trailing spaces and tabs from the command line as part of a parameter value, they are stripped by the command line processor (the shell).
- When you type trailing spaces and tabs into a file or script with an editor, such as vi, they are treated as part of a parameter value. These trailing spaces and tabs should be removed. If they are not removed they can cause confusion and unpredictable or undesirable results.
- Empty lines and lines beginning with a pound sign (#) are discarded.

Examples:

In the following examples, the *spacespacespace* represents trailing spaces.

- If you type the following at the command line, the trailing spaces are ignored:

```
xcom62 -c1 LOCAL_FILE=xyzspacespacespace
```

- If you type the following into any configuration file using an editor, the *spacespacespace* would be treated as part of the name of the LOCAL_FILE.

```
LOCAL_FILE=xyzspacespacespace
```

List of Parameters

This section lists the parameters for XCOM Data Transport for UNIX and Linux. The default values are provided in the sample `xcom.glb` file that comes with the product. If this file does not specify a value for a parameter, the value in the sample `xcom.cnf` file is used. If this file also does not specify a value for a parameter, the product generates its own default value.

AGE

Specifies the number of days of history records that are retained when a purge procedure is executed.

1 to 999

Indicates the number of days of history records to be retained when executing a purge.

NOTE

- After XCOMUTIL has been run, the history file comprises the current date's records plus *nnn* days of records. For example, if the current Julian date is 12300 and AGE=1 is specified, then XCOMUTIL purges any history records written on or before 12298. Only the current and previous day's history records are saved.
- Use the AGE parameter if you intend to run XCOMUTIL on a periodic basis. It allows you to set up a procedure in which a specific date does not have to be continually modified.
- This parameter is mutually exclusive with the DATE parameter.
- This parameter applies only to metatransfers that are initiated on a z/OS system.

AGE_TIME

The number of seconds before waiting queue entries are removed from queue. If the value is 0, the waiting queue entries never age and are never removed from the queue.

Range: 0 to 86313600 (999 days)

Default: 432000 (5 days)

ALLOCATION_TYPE

This parameter indicates the unit of storage allocation for a data set that was created on an IBM mainframe.

CYL
Cylinders

TRK
Tracks

BLK
Blocks

REC
Records

Default: CYL

NOTE
This parameter is a Version 2 parameter.

ANALYTICS_API

The ANALYTICS_API parameter specifies the REST API endpoint for the HTTP Event Collector (HEC) in Splunk.

Specify this parameter only in the `xcomanalytics.cnf` file.

For the JSON data format, use the default value.

Range: 0 through 64 characters

Default: `/services/collector/event`

ANALYTICS_CNFFILE

The ANALYTICS_CNFFILE parameter specifies the full path to the XCOM analytics configuration file.

This file is used when you integrate XCOM Data Transport with the Splunk platform.

Specify this parameter only in the `XCOM.GLB` global parameters file.

Range: 0 through 256 characters

Default: `$XCOM_HOME/config/xcomanalytics.cnf`

> **NOTE**
> The `/usr/spool/xcom/config/` directory contains a soft link (also called a symbolic link) to the `xcomanalytics.cnf` file in the default directory.

ANALYTICS_FIELDS

The ANALYTICS_FIELDS parameter specifies the transfer fields to send to Splunk when the XCOM transfer completes.

Specify this parameter only in the `xcomanalytics.cnf` file.

By default, XCOM sends all fields. You can specify as many fields or as few fields as you want. Separate the field names with commas. A blank value is sent for any fields that you omit.

You can specify the following fields:

bytes
Indicates how many bytes were sent.

direction
Indicates whether the transfer was incoming or outgoing.

end_time
Indicates when the transfer ended.

err_msg
Indicates the transfer error messages.

initiator
Indicates whether the transfer was initiated locally or remotely.

local_file
Indicates the local file name.

local_requno
Indicates the transfer request number from the initiating side.

remote_file
Indicates the remote file name.

remote_reqno
Indicates the transfer request number from the remote side.

remote_system
Indicates the destination system where the transfer was sent.

remote_user
Indicates the user that was named in the sent file.

start_time
Indicates when the transfer started.

status
Indicates whether the transfer completed successfully or failed.

status_msg
Indicates the transfer success messages.

transfer_type
Indicates whether XCOM was transferring a file, job, or report.

Range: 0 through 256 characters

Default:
local_reqno,initiator,transfer_type,direction,start_time,end_time,remote_system,status,err_msg,status_msg,remote_user,remote_re

ANALYTICS_HOST

The ANALYTICS_HOST parameter specifies the host name or IP address of a Splunk server.

Specify this parameter only in the `xcomanalytics.cnf` file.

Range: 0 through 256 characters

Default: None

ANALYTICS_HTTP_AUTH_HEADER

The ANALYTICS_HTTP_AUTH_HEADER parameter specifies the authorization header for the HTTP Event Collector (HEC) in Splunk.

Specify this parameter only in the `xcomanalytics.cnf` file.

Use the 36-character hex HEC token that Splunk generates in the following format:

```
Authorization: Splunk 1a1a111a-1a1a-11aa-aaaa-1a1a1a11a111
```

Range: 0 through 256 characters

Default: None

ANALYTICS_HTTP_CUST_HEADER

The ANALYTICS_HTTP_CUST_HEADER parameter specifies a custom header for use with Splunk channels.

Specify this parameter only in the `xcomanalytics.cnf` file.

Use this parameter when you want to use Splunk channels, which require an X-Splunk-Request-Channel custom header. For more information about using Splunk channels, see the Splunk documentation.

Range: 0 through 256 characters

Default: None

ANALYTICS_PORT

The ANALYTICS_PORT parameter specifies the port number where the Splunk HTTP Event Collector (HEC) listens for requests.

Specify this parameter only in the `xcomanalytics.cnf` file.

Range: 1 through 65535

Default: 8088 (for HTTPS) or 8080 (for HTTP)

ANALYTICS_SCHEME

The ANALYTICS_SCHEME parameter specifies the network connection protocol to use when connecting to a Splunk server.

When SSL is enabled in the HTTP Event Collector (HEC) settings, the HEC uses the HTTPS protocol. When SSL is not enabled, the HEC uses HTTP.

Specify this parameter only in the `xcomanalytics.cnf` file.

Range: HTTP, HTTPS

Default: HTTPS

ATOE_FILENAME

The name of the file containing the ASCII-to-EBCDIC character conversion table.

This is a custom file used only for specifying custom translation tables from ASCII to EBCDIC, if needed.

Range: 0 to 256 characters

Default: $XCOM_HOME/convtab/atoe.tab

> **NOTE**
> $XCOM_HOME is an environment variable.

AUTH_TYPE

This parameter specifies the type of authentication to be used for transfers.

PAM

Enables Pluggable Authentication Modules authentication.

SYSTEM

Enables traditional UNIX authentication.

Default: SYSTEM

AVGREC

For a data set created on an IBM mainframe, Specifies the multiplier for Primary and Secondary allocation units when allocating based on the number of records. The record size is based on the value of the LRECL parameter.

U

Indicates that the PRIMARY_ALLOC and SECONDARY_ALLOC parameters specify the number of records to allocate for.

K

Indicates that PRIMARY_ALLOC and SECONDARY_ALLOC 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 3 K or 3072 records.

M

Indicates that PRIMARY_ALLOC and SECONDARY_ALLOC 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: None

> **NOTE**
> This parameter applies only when the SPACE parameter specifies a value of “REC”. The parameter indicates that a file is being allocated based on a specific number of records.

BLKSIZE

Specifies the block size of a data set created on an IBM mainframe.

> **NOTE**
> Used when FILE_OPTION=CREATE

Range: 0 to 32767

If the format is…	Then the block size must be…
Fixed or fixed block record	A multiple of the record length
Variable record	Four bytes larger than the record length
Undefined record	Larger than the largest record length

Default: 800

CACHE_READ_SZ

This parameter specifies the size (in KB) of the cache for reading files.

nnnn (KB)
The size of the cache for reading files.

Range: 0 - 9999

Default: 0

> **NOTE**
> In the XCOM.GLB file, a value of 0 to 1023 can be entered for this parameter without causing an error, but it is interpreted differently, as follows:

A value of 0 is interpreted to use the old method of reading files.

- A value of 1 to 1023 is automatically converted to the default value of 1024.

CACHE_WRITE_SZ

This parameter specifies the size (in KB) of the cache for writing files.

nnnn (KB)
The size of the cache for writing files.

Range: 0 - 9999

Default: 0

> **NOTE**
> In the XCOM.GLB file, a value of 0 to 1023 can be entered for this parameter without causing an error, but it is interpreted differently, as follows:

A value of 0 is interpreted to use the old method of reading files.

- A value of 1 to 1023 is automatically converted to the default value of 1024.

CAPKIHOME

This parameter specifies the CAPKI/ETPKI library path. Set the CAPKI/ETPKI library path before you use any of the following items:

- Password encryption transfers with cipher negotiation
- Encryption at rest transfers
- TLS/SSL transfers

The XCOM Data Transport installer installs the CAPKI/ETPKI library. Typically this library is installed to the common components directory. A default path is populated in XCOM.glb during installation. If CAPKI is installed at a different location, update CAPKIHOME with the correct path where libcapki.so resides.

Range: 0 to 256 characters

Default: The default depends on the platform, as follows:

- For AIX -- CAPKIHOME=/opt/CA/SharedComponents/CAPKI/AIX/powerpc/32/lib
- For AIX 64 -- CAPKIHOME=/opt/CA/SharedComponents/CAPKI/CAPKI5/AIX/powerpc/xlc64/lib
- For HP-UX IA64 -- CAPKIHOME= /opt/CA/SharedComponents/CAPKI/CAPKI5/HP-UX/Itanium/64/lib
- For Linux x64 -- CAPKIHOME=/opt/CA/SharedComponents/CAPKI/CAPKI5/Linux/amd64/64/lib
- For Solaris 64 -- CAPKIHOME=/opt/CA/SharedComponents/CAPKI/CAPKI5/SunOS/sparc/64/lib
- For Solaris Sparc 64 -- CAPKIHOME= /opt/CA/SharedComponents/CAPKI/CAPKI5/SunOS/amd64/64/lib
- For Linux s390x -- CAPKIHOME= /opt/CA/SharedComponents/CAPKI/CAPKI5/Linux/s390x/64/lib

CARRIAGE_CONTROL_CHARACTERS

Indicates the type of printer carriage-control codes, if any, that are included in the report file.

> **NOTE**
> For report transfers only.

ASA

ASA control codes in column 1.

IBM

IBM Machine Characters (valid only for IBM mainframes).

BYPASSASA

If data is already in ASA format, bypass conversion.

OTHER

No carriage-control codes are used.

Default: OTHER

CARRIAGE_FLAG

Specifies the type of file being transferred and some special characteristics of the conversion done during the transfer.

YES

Indicates that the transferred file is a text file and a newline character should be added to the end of incoming records. Also, newline characters are removed from the ends of lines before an outgoing record is sent.

NO

Indicates no special processing.

MPACK

Indicates a text file with record packing. Uses 2K pack buffer.

VLR

Indicates a binary file of variable-length records with a field of four bytes preceding each record. Applies to locally initiated transfers only.

XPACK

Indicates a text file with record packing. Uses 31K pack buffer.

> **NOTE**
> MPACK does not support a MAXRECLEN (actual record length) over 2K. XPACK does not support a MAXRECLEN (actual record length) over 31K.

Default: YES

CHARS

This parameter indicates the character set JES uses when the report is sent to a remote system.

xxxx

Specifies up to four alphanumeric characters representing the character set JES uses when a report is sent to a remote system.

Default: None

CHECKPOINT_COUNT

Defines how often (based on record count) the sending system requests a checkpoint to be taken. The value 0000 indicates no checkpointing.

Range: 0 to 9999

Default: 1000

- XCOM Data Transport automatically sets CHECKPOINT_COUNT to 0 if any of the following parameter values are set:
  - (L)EAR_CIPHER=XXX-OFB
  - (L)EAR_CIPHER=RC4
  - (L)EAR_HASH=XXX
- This is a Version 2 parameter.

CKPT

Defines how often (based on record count) the sending system requests a checkpoint to be taken. The value 0 indicates no checkpointing.

> **NOTE**
> Setting CKPT affects performance. Each time a checkpoint is taken, the output buffers on the receiving system are written to the disk. On Token Ring, Ethernet, and other high-speed networks, CKPT should be set to 0 or to the highest allowable value. CKPT should not be set for small files.

Range: 0 to 9999

Default: 1000

Notes:

- Alias of CHECKPOINT_COUNT
- XCOM Data Transport automatically sets CHECKPOINT_COUNT to 0 if any of the following parameter values are set:
  - (L)EAR_CIPHER=XXX-OFB
  - (L)EAR_CIPHER=RC4
  - (L)EAR_HASH=XXX
- This is a Version 2 parameter.

CLASS

The print class assigned to a report transferred to a remote system.

If the remote system is an IBM mainframe, this field designates the JES SYSOUT class.

> **NOTE**
> For report transfers only.

Example:

Enter **B** to print the report through SYSOUT=B.

Range: 1 character

Default: None

CODE

Used to identify the type of data being transferred.

ASCII

An ASCII file is being transferred. This indicates that the incoming file is assumed to be ASCII format, and is not translated. Therefore the file on the remote system should be in ASCII format before it is transferred.

BINARY

A binary file, such as an executable file, is being transferred. This indicates to a remote system that it is not to translate the data it is exchanging with your system.

EBCDIC

An EBCDIC file is being transferred. The transferred data is translated from EBCDIC to ASCII format when the local system receives the data, and from ASCII to EBCDIC format when the local system sends the data.

UTF8

A Unicode file that is based on the UTF8 encoding system is being transferred. When this option is set, the LOCAL_CHARSET/ REMOTE_CHARSET parameters are used to identify the character encoding of the input file and the required encoding for the output file.

UTF16

A Unicode file that is based on the UTF16 encoding system is being transferred. When this option is set, the LOCAL_CHARSET/ REMOTE_CHARSET parameters are used to identify the character encoding of the input file and the required encoding for the output file.

Default: ASCII

Note:

Alias of CODE_FLAG

- XPACK packing is selected automatically when UT8/UTF16 options are selected.
- Block level I/O is disabled automatically when UTF8/UTF16 options are selected.

CODE_FLAG

Used to identify the type of data being transferred.

ASCII

An ASCII file is being transferred. This indicates that the incoming file is assumed to be ASCII format, and is not translated. Therefore the file on the remote system should be in ASCII format before it is transferred.

BINARY

A binary file, such as an executable file, is being transferred. This indicates to a remote system that it is not to translate the data it is exchanging with your system.

EBCDIC

An EBCDIC file is being transferred. The transferred data is translated from EBCDIC to ASCII format when the local system receives the data, and from ASCII to EBCDIC format when the local system sends the data.

UTF8

A Unicode file that is based on the UTF8 encoding system is being transferred. When this option is set, the LOCAL_CHARSET/ REMOTE_CHARSET parameters are used to identify the character encoding of the input file and the required encoding for the output file.

UTF16

A Unicode file that is based on the UTF16 encoding system is being transferred. When this option is set, the LOCAL_CHARSET/ REMOTE_CHARSET parameters are used to identify the character encoding of the input file and the required encoding for the output file.

Default: ASCII

Note:

XPACK packing is selected automatically when UT8/UTF16 options are selected.

- Block level I/O is disabled automatically when UTF8/UTF16 options are selected.

CODETABL

Specifies the 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. These custom character conversion tables determine which external translation tables will be used by the transfer.

This parameter is valid only if INTERNAL_CONVERSION_TABLES=NO.

Range: 0 to 3 alphanumeric characters

Default: None

COMPRESS

Indicates the compression type. When communicating with an IBM mainframe, if the data file contains any empty lines, COMPRESS can only be set to YES, LZSMALL, LZMEDIUM, or LZLARGE.

NOTE

- Not all compression types are supported on all platforms. For supported compression types, see the partner platform documentation.
- The LZ values enable LZ (Lempel-Ziv) compression to replace sequences of data bytes that occur more than once in a data stream with a code value.
- **COMPACT**
  RJE compaction algorithm optimized for the uppercase English text.
- **LCOMPACT**
  RJE compaction algorithm optimized for the lowercase English text.
- **LZLARGE**
  Activates LZ compression to search back 32K in the datastream for a matching string.
- **LZMEDIUM**
  Activates LZ compression to search back 16K in the datastream for a matching string.
- **LZRW3**

General-purpose algorithm that runs fast and gives a reasonable compression.
- **LZSMALL**
  Activates LZ compression to search back 4K in the datastream for a matching string.
- **NO**
  Indicates no compression.
- **RLE**
  Run length encoding.
- **YES**
  Indicates run length encoding of binary zeros and blanks only.
- **ZLIB*n***
  Greater compression than LZRW3 but less than LZSMALL, LZMEDIUM, and LZLARGE. The *n* value can be 1 through 9.

Default: YES

COMPRESS_PDS

The COMPRESS_PDS parameter controls when a partitioned data set (PDS) is compressed for XCOM Data Transport for z/OS partners that support this functionality.

The value of the CMPRS_PDS_ALLOW parameter in the z/OS default table (XCOMDFLT) or the destination member (XCOMDFLT) determines whether PDS compression is allowed. When this parameter is set to YES, then PDS compression can occur on the XCOM Data Transport z/OS partner.

COMPRESS_PDS applies only to partitioned data sets that are opened for output as the target of a XCOM Data Transport transfer.

NONE
Suppresses the compression of an output PDS as part of an XCOM Data Transport transfer.

BEFORE
Compresses the output PDS before the transfer of user data begins.

AFTER
Compresses the output PDS is compressed after the transfer of user data has completed.

BOTH
Compresses the output PDS is compressed both before and after the transfer of user data.

Default: NONE

CONFIGSSL

This *API only* parameter specifies the configssl.cnf file path and file name.

Range: 1 to 256 characters

Default: $XCOM_HOME\config\configssl.cnf

NOTE
$XCOM_HOME is an environment variable.

CONTROL

For multiple transfers. Use the syntax for performing a single transfer and then separate parameters for different transfers in the same configuration file by using this parameter.

Range: NEWXFER or NONE

Default: NEWXFER

CONVERT_CLASSES

A character string containing print classes for EBCDIC-to-ASCII conversions to be performed. For incoming report transfers only.

Range: 1 to 64 characters

Default: None

COPIES

The number of copies that are to be sent. If this parameter is not specified, the remote system queues one copy of the report to the system's default printer. For report transfers only.

Range: 1 to 999

Default: 1

CREATE_DIRECTORIES

Indicates whether to create the specified directory if it does not exist.

YES

Create the directory if it does not exist. For the connection messages to be included, see LOG_CONNECT_MSG.

NO

Do not create the directory if it does not exist.

Default: YES

CREATEDELETE

This parameter specifies whether an existing data set can be deleted and a new data set allocated at the start of a FILEOPT=CREATE transfer. This functionality is for XCOM Data Transport z/OS partners with support for CREATEDELETE. The CREATEDELETE parameter in the z/OS default table (XCOMDFLT) or destination member (XCOMCNTL) specified by the z/OS XCOM Data Transport Administrator affects the functionality of this parameter.

- When z/OS has CREATEDELETE=ALLOW:

YES

If FILEOPT=CREATE and the data set exists, then the data set is deleted and a new data set is allocated at the start of the transfer.

NO

If FILEOPT=CREATE and the data set exists, then the transfer fails with a catalog/file error.

- When z/OS has CREATEDELETE=YES:

YES or NO

If FILEOPT=CREATE and the data set exists, then the data set is deleted and a new data set is allocated at the start of the transfer.

- When z/OS has CREATEDELETE=NO:

YES or NO

If FILEOPT=CREATE and the data set exists, then the transfer fails with a catalog/file error.

Default: NO

NOTE

The attributes of the existing dataset are deleted and the new data set is allocated with the attributes specified in the transfer when CREATEDELETE=YES.

- CREATEDELETE applies only if the target data set is a sequential data set or an entire PDS/PDSE. CREATEDELETE is ignored for other types of data sets (such as PDS members, PDSE members, VSAM, and USS files).
- CREATEDELETE applies to relative GDGs when the dataset is specified using the fully qualified GxxxxVxx name.

DATACLAS

Specifies the name of the data class to use when allocating a new SMS-managed data set.

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

Range: One to eight characters

Default: None

DEBUG_FLAG

This parameter specifies whether a trace will be output to the file /tmp/<tid>.TRA instead of to the default path $XCOM_HOME/trace/<tid>.TRA. In addition to the transfer trace information, the initialization will also be traced. The trace level is still taken from the variable XTRACE.

NOTE
The <tid> variable refers to the six-digit transfer ID number.

- $XCOM_HOME is an environment variable

YES

The trace is output to /tmp/<tid>

NO

The trace is output to $XCOM_HOME/trace/<tid>.TRA.

Default: NO

DEFAULT_CHARSET

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

CCSID#nnnnn
nnnnnn - specifies the CCSID number that corresponds to the character set. Valid values are 1 - 65535. Alternatively, the value can be specified as an IANA character set name, or (ICU) acceptable alias name.

Range: 0 to 60 characters

Default: ISO-8859-1

DEFAULT_CONVERROR

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

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

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).

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.

FAIL

The transfer terminates with an error condition.

Default: FAIL

DEFAULT_DELIM

DEFAULT_DELIM specifies an optional encoding for which the specified DEFAULT_CHARSET is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list.

Additionally, specifies a colon separated list of record delimiters that are used to mark and detect the end of a record can be specified.

This parameter is used only for UNICODE transfers (CODE_FLAG=UTF8 or CODE_FLAG=UTF16).

- **EBCDIC Rules**

Record Delimiter	Mutually exclusive with
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
FF	

- **ASCII Rules**

Record Delimiter	Mutually exclusive with
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
VT	
FF	

- **UTF Rules**

Record Delimiter	Mutually exclusive with
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
VT	
FF	
LS	
PS	

Range: 0 to 60 characters

Default: ASCII:LF:CRLF

Tables indicate valid record delimiters for various encodings. If a record delimiter is not applicable for encoding on which XX_CHARSET (LOCAL_CHARSET or REMOTE_CHARSET or DEFAULT CHARSET, whichever of these is in effect at the time of transfer) is based, it should not be used.

Example: If XX_CHARSET is not UTF based, PS and LS delimiters should not be used. Similarly if XX_CHARSET is EBCDIC based, VT, PS, and LS delimiters should not be used.

In these cases, these delimiters will be treated as invalid and disregarded if used. If no valid delimiter is found in DEFAULT_DELIM, delimiters will be derived as explained below.

If DEFAULT_DELIM is not specified or doesn’t contain at least one valid delimiter, platform default values are used as shown below,

Default for ASCII: ‘LF,CRLF’

- Default for EBCDIC: ‘NL’

DEFAULT_INPUTERROR

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

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

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.

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.

FAIL

The transfer terminates with an error condition.

Default: FAIL

DEN

Specifies the density to be used in creating a tape on the remote system. Valid values are the same as those for the DEN parameter in JCL.

Range: 1 to 4

Default: None

DESTINATION

Identifies the printer or other device on the remote system where the report is to be sent. If this parameter is not specified, the remote system sends the report to the system's default printer. For report transfers only.

0 to 16 characters

For indirect transfers and for Version 1.

0 to 21 characters

For transfers that are not indirect and for Version 2.

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

DESTINATION_TYPE

The type of target system.

Range: HOST, MIDRANGE, or OTHER

Default: OTHER

DISPOSITION

Indicates what the remote system does with the report file after the report has been printed. For report transfers only.

NOTE
This field is not used when the remote system is an IBM mainframe.

DELETE

After printing the report is deleted.

KEEP

After printing the report is kept.

HOLD

After printing the report is held.

Default: DELETE

DOMAIN

The Windows domain name for use in authenticating the user ID and password when accessing a Windows-based machine that has sharable disks and drives that belong to that domain. This allows users to access these sharable drives without having to have a local user ID or password defined to the machine.

Range: 1 to 15 characters

Default: None

DROPSESS

Indicates whether XCOM Data Transport drops an LU-to-LU session at the conclusion of a scheduled file transfer.

YES

Indicates that XCOM Data Transport drops the session.

NO

Indicates that XCOM Data Transport does not drop the session.

QEMPTY

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

Default: NO

DSNTYPE

This parameter 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 is used.

<blank>

Defines a partitioned or sequential data set based on the data set characteristics entered.

NOTE
These values are IBM standards for SMS processing.

Range: One to eight characters

Default: None

EAR_CIPHER

Indicates which encryption algorithm is to be used.

Default: None

Value	FIPS MODE	Comments
DES-CBC DES-ECB DES-CFB DES-OFB DES	YES/NO	Algorithm uses fixed key size(8 bytes)
3DES-CBC 3DES-ECB 3DES-CFB 3DES-OFB 3DES	YES/NO	Algorithm uses fixed key size(24 bytes)
AES128-CBC AES128-ECB AES128-CFB AES128-OFB AES128	YES/NO	Algorithm uses fixed key size(16 bytes)
AES192-CBC AES192-ECB AES192-CFB AES192-OFB AES192	YES/NO	Algorithm uses fixed key size(24 bytes)
AES256-CBC AES256-ECB AES256-CFB AES256-OFB AES256	YES/NO	Algorithm uses fixed key size(32 bytes)
RC2-CBC RC2-ECB RC2-CFB RC2-OFB RC2	NO	Product can choose from supported range
RC4-STREAM RC4	NO	Product can choose from supported range

Example:

```
EAR_CIPHER=3DES-CBC
```

- This is one of a pair of parameters, as follows:
  - LEAR_CIPHER (local)
  - EAR_CIPHER (remote)
- XCOM Data Transport automatically sets CHECKPOINT_COUNT to 0 if any of the following parameter values are set:

- (L)EAR_CIPHER=XXX-OFB
- (L)EAR_CIPHER=RC4
- (L)EAR_HASH=XXX

EAR_DIGEST

The EAR_DIGEST parameter specifies the digest value to match with the generated digest when the transfer type is a receive file.

You must have already generated a digest value on the input file.

For a send file transfer, XCOM Data Transport generates the digest on the remote file.

XCOM Data Transport accepts the hexadecimal value of the actual digest. The specified value must be upper case.

This parameter is one of a pair of parameters, as follows:

- lear_digest (local)
- ear_digest (remote)

Range: 0 to 128 characters

Default: None

EAR_HASH

The EAR_HASH parameter specifies the hash algorithm to use.

This parameter is one of a pair of parameters, as follows:

- lear_hash (local)
- ear_hash (remote)

If any of the following parameter values are set, XCOM Data Transport automatically sets checkpoint_count to 0:

- (l)ear_cipher=XXX-OFB
- (l)ear_cipher=RC4
- (l)ear_hash=XXX

Value	FIPS MODE	Comments
SHA1	YES	Secure Hash Algorithm 1
MD5	NO	Message Digest-5 hash algorithm
MD4	NO	Message Digest-4 hash algorithm

Default: None

EAR_KEY

The EAR_KEY parameter specifies the symmetric encryption key to use for encryption and decryption.

XCOM Data Transport accepts the hexadecimal value of the actual key. The specified value must be upper case.

Default: None

EATTR

This parameter identifies if the dataset can have extended attributes when the dataset is 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: None

NOTE
This parameter is applicable only for data set creation on an IBM mainframe.

ENABLE_ANALYTICS

The ENABLE_ANALYTICS parameter specifies whether to send XCOM events to Splunk.

Specify this parameter only in the `XCOM.GLB` global parameters file.

Range: YES, NO, Y, N

Default: NO

EOL_CLASSES

A character string containing print classes for an ASCII newline that is appended to each record. For incoming report transfers only.

Range: 1 to 64 characters

Default: None

EPRTY

Indicates the execution priority for the request. The lowest priority is 1.

1 to 255

Specifies an execution priority from 1 to 255, where 1 is the lowest priority.

- In an environment with multiple concurrent XCOM Data Transport transmissions, transfers with higher priorities receive preferential servicing.
- All other considerations being equal, give short file transfers a higher EPRTY than very long-running transmissions.

Default: 16

ETOA_FILENAME

The name of the file containing the EBCDIC-to-ASCII character conversion table.

This is a custom file used only for creating custom translation tables from EBCDIC to ASCII, if needed.

Range: 1 to 256 characters

Default: $XCOM_HOME/convtab/etoa.tab

NOTE
$XCOM_HOME is an environment variable.

EXPDT

Specifies the expiration date to be placed on the tape data set being created.

yyddd

Specifies an expiration date for the tape data set as a two-digit designation for the year and a three-digit designation for the day of the year. For example, in the expiration date 13021, 13 is the year (namely, 2013) and 021 is the 21st day of that year. The tape data set will expire on January 21, 2013.

yyyy/ddd

Specifies an expiration date for the tape data set in terms of a four-digit designation for the year and a three-digit designation for the day of the year. For example, in the expiration date 2013/021, 2013 is the year and 021 is the 21st day of that year. The tape data set will expire on January 21, 2013.

NOTE
EXPDT and RETPD are mutually exclusive; specify one or the other.

EXPIRATION_TIME

The maximum time, in seconds, that a transaction is held in the transfer queue after completion. When the maximum time is reached, all references to the transaction are removed from the queue, including trace files and temporary files.

NOTE
If EXPIRATION_TIME is set to no value in xcom.glb, the program default of 6000 is used.

Range: 0 to 32767

Default: 6000

FCB

Identifies the FCB JCL parameter when sending the report file to an IBM mainframe, defining print density, lines per page, and so on. For report transfers only.

Range: Zero to four characters

Default: None

FILE_OPTION

Indicates how the transferred data is to be processed by the receiving system. For file transfers only.

For most file transfers, the parameter values are as follows:

CREATE

Create a new file on the receiving system.

APPEND

Append the transferred data to an existing file on the receiving system.

REPLACE

Replace an existing file on the receiving system.

For wildcard transfers, the parameter values are as follows:

CREATE

Create the PDS/Directory and add the transferred members. If the PDS/Directory already exists, the transfer fails with an error.

Note: The transfer will not fail if the PDS already exists if CREATEDELETE=YES is specified and the remote z/OS system has been set up to allow the PDS to be recreated through the default table (XCOMDFLT) or destination member (XCOMCNTL).

APPEND

Add transferred members/files. If the PDS/Directory does not exist or the member/file already exists, the transfer fails with an error.

REPLACE

Add or replace transferred members/files. If the PDS/Directory does not exist, the transfer fails with error XCOMN0403E Cannot open output file -- No such file or directory.

NOTE

When creating a file on an IBM mainframe system, some additional information may be necessary. For more information, see the explanations for RECORD_FORMAT, BLKSIZE, VOLUME, and UNIT parameters.

Default: CREATE

FILE_OPTION_RF

Indicates how the transferred data is to be processed by the receiving system (that is, the local system). Used when the transfer type is Retrieve File. If a value is not specified, then the value of FILE_OPTION is used. If no default is specified in FILE_OPTION or FILE_OPTION_RF, then the value defaults to CREATE.

For most file transfers, the parameter values are as follows:

CREATE

Create a new file on the receiving system.

APPEND

Append the transferred data to an existing file on the receiving system.

REPLACE

Replace an existing file on the receiving system.

For wildcard transfers, the parameter values are as follows:

CREATE

Create the PDS/Directory and add the transferred members. If the PDS/Directory already exists, the transfer fails with an error.

Note: The transfer will not fail if the PDS already exists if CREATEDELETE=YES is specified and the remote z/OS system has been set up to allow the PDS to be recreated through the default table (XCOMDFLT) or destination member (XCOMCNTL).

APPEND

Add transferred members/files. If the PDS/Directory does not exist or the member/file already exists, the transfer fails with an error.

REPLACE

Add or replace transferred members/files. If the PDS/Directory does not exist, the transfer fails with error XCOMN0403E Cannot open output file-No such file or directory.

Default: CREATE

FILE_TYPE

Indicates the type of transmission.

SEND_FILE

File transfer

SEND_REPORT
Report to be printed

SEND_JOB
Job

RECEIVE_FILE
Retrieval of a file

FILEDATA

Indicates how a remote USS file is to be allocated.

B
Binary

T
Text

If you do not specify a value for FILEDATA, then the allocation is determined based on the CODE= specification, as determined by the value of the EBCDIC/Binary/ASCII/VLR(E/B/A/V) field, as follows:

- If EBCDIC/Binary/ASCII/VLR(E/B/A/V)=B, then the file is allocated and processed as binary data.
- If EBCDIC/Binary/ASCII/VLR(E/B/A/V)=E, the type of allocation and processing depends on the value of FILEDATA, as follows:
  - If you do not specify FILEDATA=B, then the file is allocated as a text file and processed as an EBCDIC text file.
  - If you do specify FILEDATA=B, then the file is allocated as a binary file, but processed as an EBCDIC file.

Important! If you do specify FILEDATA=B, then you need to specify a value for USSLRECL, to tell XCOM Data Transport how many bytes there are in each logical record.

FIPS_MODE

This parameter specifies the mode to enable or disable the FIPS-compliant cipher and digest algorithms to be used for TLS/SSL transfers or for encryption at rest.

YES
Indicates that encryption/decryption will be running in FIPS mode.

NO
Indicates that encryption/decryption will not be running in FIPS mode.

Default: NO

FORM

The type of form that should be used to print the report. Because XCOM Data Transport places the print job in the remote system's print queue, the print control functions depend on the remote system. The user must verify beforehand that the requested form is available at the remote site. For report transfers only.

> **NOTE**
> When sending a report to a VAX computer, leave this parameter blank unless you are certain that this is a valid form type. VMS interprets this to mean that no special form is being requested.

Range: 0 to 10 characters

Default: None

HIST_FILE

This parameter specifies the complete path information of the file to contain the history records returned by an inquire metatransfer (-c6) or a get history record retrieval metatransfer (-c7).

Range: 0 to 256 characters

Default: None

HOLD_SESS

Specifies whether a session is to be held.

Y

The session is held.

N

The session is not held.

HOLD_TRANSFER

The HOLD_TRANSFER parameter specifies whether to hold a queued transfer.

This parameter applies only when QUEUE=YES is also specified.

Y

Holds the queued transfer.

N

Does not hold the queued transfer.

Default: N

HOLD

Prevents a TYPE=SCHEDULE transfer from starting until explicitly released.

YES

This transfer is not initiated until it is released in one of the following ways:

- Through the Menu Interface
- By an operator with a RELEASE command
- Through the XTC facility

NO

This transfer is not held.

Default: NO

HOLDCOUNT

Associates a number with a transfer request that is incremented or decremented by the successful or unsuccessful completion of other transfer requests. As long as the number is greater than 0, the transfer is not released.

0 to 255

Specifies a value that controls the holding/releasing of a transfer request. The transfer is released when the value of the parameter reaches 0.

NOTE
See the description of the parameters XTCERRDECR, XTCERRINCR, XTCGOODDECR, and XTCGOODINCR, which can decrement and increment the value of the HOLDCOUNT parameter.

Default: 0

HOLDFLAG

The HOLDFLAG parameter specifies whether to place a transferred report file on hold on the remote system.

This parameter applies to report transfers only.

The alias for HOLDFLAG is HOLD.

Specifying NO prints the report immediately.

Range: YES or NO

Default: NO

IDENT

Allows the user to enter information to identify the file transfer request.

Note: This is equivalent to the TRANSFER_ID field on Windows.

Range: 0 to 10 characters

Default: None

Note: You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

INQ_FILE

Specifies the complete path information of the file to contain the information required to inquire about the transfer.

Range: 1 to 256 characters

Default: None

INQ_WAIT

Specifies how long XCOM Data Transport should wait - in hours (*hh*), minutes (*mm*), and seconds (*ss*) - for transfers to complete.

hhmmss

Specifies in hours (*hh*), minutes (*mm*) and seconds (*ss*) the length of the time that XCOM Data Transportshould wait for a transfer request to complete.

Default: 001000 (10 minutes)

Note: This parameter's value is expressed as a number of up to six digits (for example, 010000 for 1 hour).

INTERNAL_CONVERSION_TABLES

Indicates whether internal or external conversion tables should be used for ASCII-to-EBCDIC conversion and EBCDIC-to-ASCII conversion. The external conversion files may be customized to meet your needs. For the appropriate directory and file name, see ATOE_FILENAME and ETOA_FILENAME.

YES

Use internal conversion tables.

NO

Use the external tables contained in ATOE_FILENAME and ETOA_FILENAME.

Default: YES

IPNAME

For TCP/IP protocols, the name of the remote system that receives a file, job, or report.

The value can be the remote system's IP address, hostname, or domain name.

Range: 1 to 256 characters

Default: None

IPPORT

The number of the TCP/IP port on the remote XCOM Data Transport server. Used for TCP/IP transfers only.

Range: 1 to 65535

Default: 8044

LABEL

The LABEL parameter specifies the type of processing to apply to a tape data set.

The valid processing types are AL, AUL, BLP, LTM, NL, NSL, SL, and SUL.

> **NOTE**
> XCOM Data Transport for z/OS supports only standard label tapes.

Default: SL

LABELNUM

Indicates the sequence number of the data set on the tape.

Sequence number (0001 to 9999)

This value identifies the sequence number of a data set on tape.

Example:

LABELNUM=2

This specification refers to the second data set on the tape.

Default: 0001

LCLNTFYL

Specifies the local user notification level.

A or ALL

Notify on transfer completion.

W or WARN

Notify only if the transfer received a warning or error.

E or ERROR

Notify only if the transfer received an error.

Default: ALL

LDOMAIN

The domain associated with LUSERID and LPASSWORD, if the target system is UNIX or Linux.

Range: 1 to 15 characters

Default: None

LEAR_CIPHER

Indicates which encryption algorithm is to be used.

Default: None

Value	FIPS MODE	Comments
DES-CBC DES-ECB DES-CFB DES-OFB DES	YES/NO	Algorithm uses fixed key size(8 bytes)
3DES-CBC 3DES-ECB 3DES-CFB 3DES-OFB 3DES	YES/NO	Algorithm uses fixed key size(24 bytes)
AES128-CBC AES128-ECB AES128-CFB AES128-OFB AES128	YES/NO	Algorithm uses fixed key size(16 bytes)
AES192-CBC AES192-ECB AES192-CFB AES192-OFB AES192	YES/NO	Algorithm uses fixed key size(24 bytes)
AES256-CBC AES256-ECB AES256-CFB AES256-OFB AES256	YES/NO	Algorithm uses fixed key size(32 bytes)

RC2-CBC RC2-ECB RC2-CFB RC2-OFB RC2	NO	Product can choose from supported range
RC4-STREAM RC4	NO	Product can choose from supported range

Example:

```
LEAR_CIPHER=3DES-CBC
```

- This is one of a pair of parameters, as follows:
  - LEAR_CIPHER (local)
  - EAR_CIPHER (remote)
- XCOM Data Transport automatically sets CHECKPOINT_COUNT to 0 if any of the following parameter values are set:
  - (L)EAR_CIPHER=XXX-OFB
  - (L)EAR_CIPHER=RC4
  - (L)EAR_HASH=XXX

LEAR_DIGEST

The LEAR_DIGEST parameter specifies the digest value to match with the generated digest when the transfer type is a send file, job, or report.

You must have already generated a digest value on the input file.

For a receive file transfer, XCOM Data Transport generates the digest on the local file.

XCOM Data Transport accepts the hexadecimal value of the actual digest. The specified value must be upper case.

This parameter is one of a pair of parameters, as follows:

- lear_digest (local)
- ear_digest (remote)

Range: 0 to 128 characters

Default: None

LEAR_HASH

The LEAR_HASH parameter specifies the hash algorithm to use.

This parameter is one of a pair of parameters, as follows:

- LEAR_HASH (local)
- EAR_HASH (remote)

If any of the following parameter values are set, XCOM Data Transport automatically sets CHECKPOINT_COUNT to 0:

- (L)EAR_CIPHER=XXX-OFB
- (L)EAR_CIPHER=RC4
- (L)EAR_HASH=XXX

Value	FIPS MODE	Comments
SHA1	YES	Secure Hash Algorithm 1
MD5	NO	Message Digest-5 hash algorithm
MD4	NO	Message Digest-4 hash algorithm

Default: None

LEAR_KEY

The LEAR_KEY parameter specifies the symmetric encryption key to use for encryption and decryption.

XCOM Data Transport accepts the hexadecimal value of the actual key. The specified value must be upper case.

Range: 0 to 128 characters

Default: None

LFILE

The name of the file on the local system that is being transferred. At the command prompt or in a script, if this variable is null or unset, standard input is read. In this manner, XCOM Data Transport commands can be used in a pipeline or with redirection. All Windows and UNC file naming conventions apply. If more than one file is included in the transfer, L_FILE must include wildcard characters (*).

> **NOTE**
> If QUEUE=YES, user must specify full path name.

Range: 0 to 256 characters

Default: None

LOCAL_CHARSET

LOCAL_CHARSET specifies the local character set XCOM Data Transport uses for Unicode transfers (CODE_FLAG=UTF8 or CODE_FLAG=UTF16).

CCSID#nnnnn

nnnnnn - specifies the CCSID number that corresponds to the character set. Valid values are 1 - 65535. Alternatively, can be specified as an IANA character set name, or (ICU) acceptable alias name.

Range: 0 to 60 characters

Default: The DEFAULT_CHARSET parameter in the XCOM Data Transport Global Parameters specifies the default.

LOCAL_DELIM

LOCAL_DELIM specifies an optional encoding for which the specified LOCAL_CHARSET is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list. If LOCAL_CHARSET is UTF based charset, UTF rules will be applied disregarding encoding specified in LOCAL_DELIM.

Additionally, a colon-separated list of record delimiters can be specified that are used to mark and detect the end of a record.

This parameter is used only for UNICODE transfers (CODE_FLAG=UTF8 or CODE_FLAG=UTF16).

- **EBCDIC Rules**

Record Delimiter Option	Mutually exclusive with
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
FF	
NA	

- **ASCII Rules**

Record Delimiter Option	Mutually exclusive with
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
VT	
FF	

- **UTF Rules**

Record Delimiter Option	Mutually exclusive with
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
VT	
FF	
LS	
PS	

Range: 0 to 60 characters

Default: The DEFAULT_DELIM parameter in the XCOM Data Transport Global Parameters specifies the default.

The tables indicate valid record delimiters for various encodings. If a record delimiter is not applicable for encoding on which LOCAL_CHARSET is based, it should not be used.

Example: If LOCAL_CHARSET is not UTF based, PS and LS delimiters should not be used. Similarly if LOCAL_CHARSET is EBCDIC based, VT, PS, and LS delimiters should not be used.

In these cases, these delimiters will be treated as invalid and disregarded if used. If no valid delimiter is found in LOCAL_DELIM, delimiters will be derived as explained below.

If LOCAL_DELIM is not specified or doesn't contain at least one valid delimiter, DEFAULT_DELIM parameter in the XCOM Data Transport Global Parameters is considered. If DEFAULT_DELIM is also not specified or doesn't contain at least one valid delimiter, platform default values are used as shown below,

Default for ASCII: 'LF,CRLF'

- Default for EBCDIC: 'NL'

NOTE
During Unicode conversion, some of the delimiters may fail to convert to some code pages.

For example, In case of Receive File transfer, when LOCAL_CHARSET set to ASCII based charset and LOCAL_DELIM set to NL/CRNL, then transfer may fail. The Unicode conversion fails as ASCII based code pages may not have mapping character for NL/CRNL.

LOCAL_FILE

The name of the file on the local system that is being transferred. At the command prompt or in a script, if this variable is null or unset, standard input is read. In this manner, XCOM Data Transport commands can be used in a pipeline or with redirection. All UNIX or Linux file naming conventions apply.

For wildcard transfers, use an asterisk (*) as a file name to indicate that all files within the specified directory are to be transferred. For example, the statement LOCAL_FILE=/NAMES/* indicates that all files under the NAMES directory are to be transferred.

When a prefix is followed by an asterisk, all members beginning with a specific prefix are to be transferred. For example, LOCAL_FILE=/NAMES/AL* requests that files AL, ALEX, and ALICE are all to be transferred. The same rules apply if an asterisk is followed by a suffix.

The actual file name range (not including its path) for wildcard transfers can be between 0 and 71 characters. This also includes the file extension where applicable. File names over 71 characters are truncated. However, when sending files to a mainframe PDS, any file name over 8 characters in length is truncated. These systems do not recognize file extensions. For example, a file called *longfilename.txt* will be truncated to *longfile* and a file called *file.txt* will be truncated to *file*.

NOTE
If QUEUE=YES, the full pathname must be specified.

Range: 0 to 256 characters

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

LOCAL_FILE_RF

The file name that is created, appended, or replaced on the local system when it receives a file. At the command prompt or in a script, if this value is null or unset, then XCOM Data Transport writes to the stdout. All UNIX or Linux file naming conventions apply.

For wildcard transfers, use an asterisk (*) as a file name to indicate that multiple files will be received.

Example

```
LOCAL_FILE_RF=/PAYROLL/*.
```

If multiple files are received and the user specifies a file name, all files received by the partner are written to that specified file as one single file.

For platforms that support it, you can specify a common file extension to be appended to each file name.

Example

```
LOCAL_FILE_RF=/PAYROLL/*.TXT.
```

NOTE
If QUEUE=YES, the full pathname must be specified.

Range: 0 to 256 characters

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

LOCAL_FILE_SJ

Indicates the name of the file on the local system to be sent as a job. All the UNIX or Linux file naming conventions apply. If this value is null or unset, then XCOM Data Transport reads the standard input file.

NOTE
If QUEUE=YES, user must specify full path name.

Range: 0 to 256 characters

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

LOCAL_FILE_SR

Local file name to be sent as a report to the remote system. If this value is null or unset, then XCOM Data Transport reads the standard input file. For report transfers only.

NOTE
If QUEUE=YES, user must specify full path name.

Range: 0 to 256 characters

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

LOCAL_NOTIFY

Specifies the user on the local system who is to be notified that XCOM Data Transport has completed a transfer. XCOM Data Transport uses the NOTIFYL parameter to determine the type of notification to use.

Range: 0 to 64 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

LOG_CONNECT_MSG

This parameter specifies whether the XCOM Data Transport log file should include the following connection messages:

- XCOMN0785I
- XCOMN0786I
- XCOMN0793I
- XCOMN0805I
- XCOMN0811I
- XCOMN0812I
- XCOMN0813I
- XCOMN0814I
- XCOMN0818I

Options: YES or NO

- YES: Include the connection messages in XCOM Data Transport log.
- NO: Do not include the connection messages in XCOM Data Transport log.

Default: YES

LPASS

The password of the local user to be validated on the XCOM Data Transport server handling a meta-transfer.

LPASSWORD

LPASSWORD specifies the password of the local user to validate on the XCOM Data Transport server that is handling a metatransfer.

Range: 1 to 31 characters

> **NOTE**
>
> Service pack 11.6.01 with corresponding PTFs supports passphrase; hence, the accepted range is from 1 to 100 characters. To implement the passphrase support, apply the following PTFs to the corresponding platforms: SO05627(Linux), SO05628(AIX), SO05629(Solaris Sparc), and SO05630(Solaris x86).

Default: None

LRECL

Specifies the actual or maximum length, in bytes, of a logical record. Corresponds to the JCL LRECL subparameter.

Range: 0 to 32767

If the format is…	**Then the maximum length of a logical record must be equal to the…**
Variable blocked record	Maximum record length plus four
Fixed or fixed block record	Constant record length

Default: 160

LUSER

The user ID to use on the XCOM Data Transport system receiving a meta-transfer.

Range: 1 to 12 characters

Default: None

> **NOTE**
> LUSER is an alias of LUSERID.

LUSERID

The user ID to use on the XCOM Data Transport system receiving a metatransfer.

Range: 1 to 12 characters

Default: None

MAIL_TYPE

This parameter specifies the type of MAIL server used for sending mail notifications.

MAPI
Windows MAPI server

SMTP
SMTP server

Default: None

MAX_QUEUE_ENTRIES

The maximum number of entries allowed in the transfer queue. Once the maximum number of queue entries is reached, subsequent transfer attempts are rejected by XCOM Data Transport.

> **NOTE**
> This value depends on the memory available when XCOM Data Transport is started.

Range: 0 to 32767

The value cannot be set larger than the maximum allowed shared memory segment divided by 512.

Default: 50

MAX_REMOTE_TCP

The maximum number of simultaneous remote TCP/IP transfers accepted by XCOM Data Transport. Transfers received after this limit has been reached are rejected.

> **NOTE**
> Setting this parameter to 0 prohibits any remote transfers from taking place.

Range: 0 to 999

Default: 32

MAX_SESSIONS_ENTRIES

The maximum number of partners that can be described in the XCOM.SES file.

Range: 1 to 999

Default: 15

MAXRECLEN

For Windows, UNIX, and Linux systems, the locally initiating XCOM Data Transport system determines the values for MAXRECLEN, TRUNCATION, and LRECL, for send and receive operations. When the local XCOM Data Transport system initiates a transfer of a text file, this parameter designates the length, in bytes, of the largest record that can be transferred. If a record length is longer than this value, XCOM Data Transport uses the value in the TRUNCATION parameter on the initiating side to determine whether to terminate the transfer or to truncate the record and continue the transfer. When XCOM Data Transport transfers binary files, this value indicates the length of the records that are transferred. On a receive operation, MAXRECLEN is set to whatever the LRECL value is on the initiating side.

Range: 1 to 32767

> **NOTE**
>
> Note: It is recommended to use values >=24 for MAXRECLEN parameter when used in combination with Encryption At Rest feature to transfer encrypted files.

Default: 1024

MBCS_CONVERROR

MBCS_CONVERROR identifies the appropriate action when the input file contains characters that cannot be converted. 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 and is treated as REPLACE.

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 XCOM Data Transport Global Parameters specifies the default.

MBCS_INPUTERROR

MBCS_INPUTERROR 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 and is treated as REPLACE.

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 XCOM Data Transport Global Parameters specifies the default.

METACODE_CLASSES

Classes of print jobs saved in metacode format, a variable length record format. For incoming report transfers only.

Range: 0 to 64 characters

Default: None

MGMTCLAS

Specifies the name of the management class to use when allocating a new SMS-managed data set.

> **NOTE**
> This parameter applies only to mainframe SMS data sets.

Range: One to eight characters

Default: None

NOTIFY_NAME

The user on the remote system who is to be notified when XCOM Data Transport completes a transfer.

> **NOTE**
> If the remote system is an IBM mainframe, XCOM Data Transport uses the value of NOTIFYR to determine the type of notification to deliver.

If the remote system is a UNIX or Linux system, the user receives a mail message.

Range: 0 to 12 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

NOTIFY_TERM

Specifies which terminals to write to if NOTIFYL=WRITE. If NOTIFY_TERM is not set, all users specified in LOCAL_NOTIFY are notified at the first terminal where they are logged in, as found in the system table.

Range: 0 to 256 characters

Default: None

NOTIFYL

The local user notification flag.

WRITE
A message is displayed on the workstation where the user is logged in.

MAIL
A mail message will be sent to the user.

NONE
No notification is sent.

ALL
A message is displayed on all workstations attached to the server.

Default: None

> **NOTE**
> The L in NOTIFYL indicates that the local system governs the processing of the resulting notification on that system.

NOTIFYR

Specifies the remote user notification type when sending data to a remote system.

WRITE
A message is displayed on the screen.

MAIL
A mail message is sent to the user.

TSO
The specified TSO user is notified.

WTO
XCOM Data Transport writes to the log only (WTO).

CICS
The specified CICS user is notified.

LU
The specified Logical Unit is notified.

ROSCOE
Notify Roscoe user.

NONE

No notification is sent.

ALL

Write to all users.

Default: None

> **NOTE**
> The R in NOTIFYR indicates that the remote system governs the processing of the resulting notification on that system.

NUM_OF_DIR_BLOCKS

This parameter specifies the number of directory blocks to allocate for a data set that was created on an IBM mainframe.

Range: 0 to 16,777,215

Default: 0

> **NOTE**
> This parameter is a Version 2 parameter.

NUMBER_OF_RETRIES

Maximum number of retries before a transfer is logged as failed and taken out of the transfer queue. If the value is 0, no retries are attempted.

Range: 0 to 255

Default: 1

> **NOTE**
> This is a Version 2 parameter.

OEDATE

Limits the history request to only those file transfers that were scheduled or completed on or before the end date and time.

YYYYMMDD

The end date used to limit the history request to only those file transfers that were scheduled or completed on or before the end date and time specified.

YYYY

The four-digit year

MM

The two-digit number of the month, as follows:
01 = January 02 = February 03 = March
04 = April 05 = May 06 = June
07 = July 08 = August 09 = September
10 = October 11 = November 12 = December

DD

The two-digit day of the month (01 to 31)

Default: None

- OEDATE and OETIME form an end date and time used to limit the history request to only those file transfers that were scheduled or completed on or before the date and time specified.
- See OSDATE and OSTIME for the start date and time.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

OETIME

Limits the history request to only those file transfers that were scheduled or completed on or before the end date and time.

HHMMSS

The end time used to limit the history request to only those file transfers that were scheduled or completed on or before the end date and time specified.

HH

The two-digit hour (00 through 23)

MM

The two-digit minute (00 through 59)

SS

The two-digit second (00 through 59)

Default: 235959

- OEDATE and OETIME form an end date and time used to limit the history request to only those file transfers that were scheduled or completed on or before the date and time specified.
- See OSDATE and OSTIME for the start date and time.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

OFILE

Specifies the file name, local, or remote, to match for a history request.

You can use the following wildcard characters when you specify the file name:

*** or %**

Represents a string of zero or more characters.

_

Represents any single character.

Range: 1 to 256

Default: None

Example:

An OFILE value of %MASTER.FIL_.G* tells XCOM Data Transport to locate a file with following attributes:

- Starting with anything
  - Ending with anything
  - With the characters MASTER.FIL found in the name, followed by any single character and .G.

Notes:

- Supported in - c7 CNF meta transfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OFILECASE

Specifies whether the specified file name (OFILE parameter) search is case-sensitive.

YES

The value that is specified for the OFILE parameter is case-sensitive.

NO

The value that is specified for the OFILE parameter is not case-sensitive.

Default: NO

NOTE

- For a case-sensitive search, case-sensitive collation is set to the history database table. Such collation can also be set to ‘file’ and ‘lfile’ columns in the history database table.
- Supported in - c7 CNF meta transfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.

OFILETYPE

Limits the history request to those transfers with the specified FILETYPE.

JOB

Restrict the search for file transfers to only FILETYPE JOB transfers.

REPORT

Restrict the search for file transfers to only FILETYPE REPORT transfers.

FILE

Restrict the search for file transfers to only FILETYPE FILE transfers.

Default: None

OFLMAX

Limits the history request to only those file transfers where the number of bytes transferred is equal to or less than the value specified.

NNNNNNNNN(*N*|*X*)

A 1- to 10-digit number, where the last digit can be either another numeric digit or a 1-character qualifier. This parameter is used to restrict the search for file transfers to only those file transfers where the number of bytes transferred is equal to or less than the value specified.

X

One of the following qualifiers (default B):

- B = Bytes
- K = Kilobytes
- M = Megabytes
- G = Gigabytes

Default: None

- Use OFLMIN and OFLMAX to specify a range that can be used to limit the history request by number of bytes transferred.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

OFLMIN

Limits the history request to only those file transfers where the number of bytes transferred is equal to or greater than the value specified.

NNNNNNNNN(*N*|*X*)

A 1- to 10-digit number, where the last digit can be either another numeric digit or a 1-character qualifier. This parameter is used to restrict the search for file transfers to only those file transfers where the number of bytes transferred is equal to or less than the value specified.

X

One of the following qualifiers (default B):

- B = Bytes
- K = Kilobytes
- M = Megabytes
- G = Gigabytes

Default: None

- Use OFLMIN and OFLMAX to specify a range that can be used to limit the history request by number of bytes transferred.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

OID

Limits the history request to only those file transfers with a specific transfer ID. The transfer ID is a user-defined identifier for file transfer requests.

xxxxxxxxxx

A 1- to 10-character transfer ID used to limit the history request to only those file transfers that contain the specified transfer ID.

Default: None

- The wildcard character, *, can be used for this parameter only when specified as the last character.
- This parameter is not case-sensitive. So filtering history records based on this parameter returns case-insensitive results. For example, specifying **TRANSFER01** returns the same results as specifying **transfer01**.
- Supported in - c7 CNF metatransfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer will be authorized to access.

OINIT

Limits the history request to only locally initiated transfers or only remotely initiated transfers.

LOCAL or L

Restrict the search for file transfers to only locally initiated transfers.

REMOTE or R

Restrict the search for file transfers to only remotely initiated transfers.

Default: None

NOTE

You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

OJOB

Specifies the invoking job name to match for a history request.

Range: 1 to 8 characters

Default: None

NOTE

- Supported in - c7 CNF meta transfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.
- You can use the * wildcard character when you specify the job name.

OLMSG

Limits the history request by the transfer's last message. The format to use for XCOM Data Transport messages is as follows:

XCOM*XNNNNS*

A 1- to 10-character name used to restrict the search for file transfers to those where the last message matches the value specified.

XCOM

Indicates that the message is from XCOM Data Transport.

X

Identifies the system.

NNNN

Is the message number.

S

Is the message severity:

- I = Informational
- W = Warning
- E = Error

Default: None

NOTE

- The wildcard character, *, can be used for this parameter only when specified as the last character.
- This parameter is case sensitive.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

OLOCATN

Limits the history request to transfers in one of the following locations:

- Transfers that have been completed and have been stored in a database
- Transfers still in the queue
- **DATABASE**
  Restrict the search for file transfers to those stored in the database. This applies only if a history database has been set up at your site.
- **QUEUE**
  Restrict the search for file transfers to those still in the queue.

Default: Transfers located in both the QUEUE and DATABASE.

NOTE

- Supported in - c7 CNF metatransfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer will be authorized to access.

OLU

Limits the history request to only those file transfers with a specific remote LU name.

xxxxxxxx

A one- to eight-character LU name used to limit the history request to only those file transfers for the specified remote LU.

Default: None

- The wildcard character, *, can be used for this parameter only when specified as the last character.
- This parameter is not case-sensitive. So filtering history records based on this parameter returns case-insensitive results. For example, specifying **LU01** returns the same results as specifying **lu01**.
- Supported in - c7 CNF metatransfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer will be authorized to access.

OREQ

Limits the history request to only those file transfers that contain this specific request number.

NNNNNN

A one- to six-character request number used to limit the history request to a specific request number.

Default: All request numbers

Note: You can use symbolic variables with this parameter in batch SYSIN01. For more information, see Symbolic Parameters in Configuration Files.

OSDATE

Limits the history request to only those file transfers that were scheduled or completed on or after the start date and time.

YYYYMMDD

The start date used to limit the history request to only those file transfers that were scheduled or completed on or after the start date and time specified.

YYYY

The four-digit year

MM

The two-digit number of the month, as follows:
01 = January 02 = February 03 = March
04 = April 05 = May 06 = June
07 = July 08 = August 09 = September
10 = October 11 = November 12 = December

DD

The two-digit day of the month (01 to 31)

Default: None

- OSDATE and OSTIME form a start date and time used to limit the history request to only those file transfers that were scheduled or completed on or after the date and time specified.
- See OEDATE and OETIME for the end date and time.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

OSTIME

Limits the history request to only those file transfers that were scheduled or completed on or after the start date and time.

HHMMSS

The start time used to limit the history request to only those file transfers that were scheduled or completed on or after the start date and time specified.

HH

The two-digit hour (00 through 23)

MM

The two-digit minute (00 through 59)

SS

The two-digit second (00 through 59)

Default: 235959

NOTE

- OSDATE and OSTIME form a start date and time used to limit the history request to only those file transfers that were scheduled or completed on or after the date and time specified.
- See OEDATE and OETIME for the end date and time.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

OSYSID

The OSYSID parameter limits the history request to file transfers that XCOM Data Transfer executes on the specified system ID.

OSYSNAME and OSYSID are used together to uniquely identify an XCOM Data Transport server that is r11.5 or higher. OSYSID is supported only in - c7 CNF metatransfers. When a user requests history records, the defined users in groups XCOMADM and XCOMSADM determine what records the user is authorized to access.

Range: 1 to 4 characters

Default: None

OSYSNAME

The OSYSNAME parameter limits the history request to file transfers that XCOM Data Transfer executes on the specified system name.

OSYSNAME and OSYSID are used together to uniquely identify an XCOM Data Transport server that is r11.5 or higher. OSYSNAME is supported only in - c7 CNF metatransfers. When a user requests history records, the defined users in groups XCOMADM and XCOMSADM determine what records the user is authorized to access.

Range: 1 to 8 characters

Default: None

OTNAME

Limits the history request to only those file transfers with a specific remote TCP/IP name or TCP/IP address.

xxxxxxxx...x

A 1- to 64-character TCP/IP name or address used to limit the history request to only those file transfers for the specified TCP/IP name or address.

Default: None

- The wildcard character, *, can be used for this parameter only when specified as the last character.
- This parameter is not case-sensitive. So filtering history records based on this parameter returns case-insensitive results. For example, specifying **TCPIP01** returns the same results as specifying **tcpip01**.
- Supported in - c7 CNF metatransfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer will be authorized to access.

OTYPE

Specifies if the history request should include inactive transfer requests, active transfer requests, or completed transfers.

I

Selecting this option retrieves history records for file transfers whose execution is still pending.

A

Selecting this option retrieves history records for file transfers that are currently in progress.

C

Selecting this option retrieves history records for file transfers that have been successfully or unsuccessfully completed.

ALL|AIC|*

Selecting this option retrieves history records for all file transfers, independent of their status.

Default: AIC

NOTE
You can also specify values in combination; for example, specify AI to request history records for file transfers whose execution status is inactive and active.

OTYPEREQ

Limits the history request to only send transfers or only receive transfers.

SEND or S
Restrict the search for file transfers to only send transfers.

RECEIVE or R
Restrict the search for file transfers to only receive transfers.

Default: None

NOTE
This parameter is case sensitive.

OUSER

Limits the history request to only those file transfers submitted by a specific user.

xxxxxxxxxxxx
A 1- to 12-character user name used to limit the history request to only those file transfers submitted by the specified user.

Default: None

NOTE

- The wildcard character, *, can be used for this parameter only when specified as the last character.
- This parameter is not case-sensitive. So filtering history records based on this parameter returns case-insensitive results. For example, specifying **USER01** returns the same results as specifying **user01**.
- Supported in - c7 CNF metatransfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer will be authorized to access.

OVOL

Specifies the volser (local or remote) to match for a history request.

Range: 1 to 6 characters

Default: None

NOTE

- Supported in - c7 CNF meta transfers only.
- The users defined in the groups XCOMADM and/or XCOMSADM determine what records the user running the get history record retrieval transfer is authorized to access.
- You can use the * wildcard character when you specify the volser.

PAM_PATH

The PAM_PATH parameter specifies the path to your Pluggable Authentication Modules (PAM) library for your current UNIX or Linux platform.

This parameter is valid only when AUTH_TYPE=PAM.

Do not specify the library name in the path. For example, if the library is available at `/usr/lib64`, specify `PAM_PATH=/usr/lib64`. XCOM appends the library name to the path internally.

Range: 0 to 256 characters

Default: None

PASSWORD

PASSWORD specifies the password that is associated with the user ID on a remote system.

Range: 0 to 31 characters

> **NOTE**
>
> Service pack 11.6.01 with corresponding PTFs supports passphrase; hence, the accepted range is from 1 to 100 characters. To implement the passphrase support, apply the following PTFs to the corresponding platforms: SO05627(Linux), SO05628(AIX), SO05629(Solaris Sparc), and SO05630(Solaris x86).

Default: None

PORT

The number of the TCP/IP port on the remote XCOM Data Transport server. Used for TCP/IP transfers only.

Range: 1 to 65535

Default: 8044

PRIMARY_ALLOC

This parameter identifies primary storage allocation for a data set that was created on an IBM mainframe.

Range: 0 to 16,777,215

Default: 1

> **NOTE**
> This parameter is a Version 2 parameter.

PRINT_CLASS

The print class assigned to a report transferred to a remote system. If the remote system is an IBM mainframe, this field designates the JES SYSOUT class.

> **NOTE**
> For report transfers only.

Range: 1 character

Default: None

Example:

Enter **B** to print the report through SYSOUT=B.

PRIORITY

Indicates the priority that XCOM Data Transport uses for scheduling a transfer. If two transfers are scheduled for the same time, the one with the high priority is processed before one with a normal or low priority.

HIGH
Set high priority.

NORMAL
Set medium priority.

LOW
Set low priority.

Default: NORMAL

PROGLIB

Specifies whether the transfer is a z/OS transfer of a PDSE program library or not.

YES
Indicates a z/OS transfer of a PDSE program library.

NO
Indicates that the transfer is not of a PDSE program library.

Default: None

PROTOCOL

The type of communication protocols to use.

SNA
For transfers using SNA/APPC communication protocols

TCPIP
For transfers using TCP/IP communication protocols

Default: SNA

NOTE
If only the TCP/IP component is installed with the base components of XCOM Data Transport, the default value is TCPIP.

QUEUE

Indicates whether to execute the transmission request immediately or to allow the request to be queued. If the user does not specify a .cnf file, and has not changed a .cnf file, the default value is YES.

NOTE
If NO is specified and the remote system is unavailable the request aborts. If YES is specified, START_TIME and START_DATE are read.

YES
The transfer request goes into a queue and executes depending on the traffic in the queue and START_DATE and START_TIME.

NO

The transfer starts immediately.

Default: YES

WARNING
If you load a .cnf file into the XCOM Data Transport GUI, the value (QUEUE=YES or QUEUE=NO) specified in the .cnf file is shown correctly in the generated xml file. However, when you submit the transfer from the XCOM Data Transport GUI, it is always treated as though QUEUE=YES.

QUEUE_PATH

Directory containing the transfer queue data files.

Range: 0 to 256 characters

Default: $XCOM_HOME/Q

NOTE
$XCOM_HOME is an environment variable.

RECORD_FORMAT

Specifies the record format of a data set created on an IBM mainframe. This corresponds to the JCL RECFM subparameter.

Range: The range of values is listed in the following table:

Value	Description	Record Length	Comment
F	Fixed unblocked	The same length as the data set	
FA	Fixed unblocked ANSI	The same length as the data set	Contains ISO/ANSI/FIPS control characters
FB	Fixed blocked	Fixed	Fixed record length with multiple records per block
FBA	Fixed blocked ANSI	Fixed	Multiple records per block where these records contain ISO/ANSI/FIPS control characters
FBS	Fixed blocked spanned	Fixed	Multiple records per block written as standard blocks
FM	Fixed unblocked machine	The same length as the data set	Contains machine code control characters
FS	Fixed unblocked spanned	The same length as the data set	Written as standard blocks where these records do not contain any truncated blocks or unfilled tracks
U	Undefined	Undefined	
V	Variable unblocked	Variable	
VA	Variable unblocked ANSI	Variable	Contains ISO/ANSI/FIPS control characters
VB	Variable blocked	Variable	Multiple records per block

VBA	Variable blocked ANSI	Variable	Multiple records per block where these records contain ISO/ANSI/ FIPS control characters
VBM	Variable blocked machine	Variable	Multiple records per block where these records contain machine code control characters
VBS	Variable blocked spanned	Variable	May have multiple records per block where these records can span more than one block
VM	Variable unblocked machine	Variable	Contains machine code control characters
VS	Variable unblocked spanned	Variable	A record can span more than one block

Default: VB

RELEASE

Specifies whether the remote z/OS partner is to release unused DASD space when creating a new file.

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.

Default: No

REMOTE_CHARSET

REMOTE_CHARSET specifies the remote character set XCOM Data Transport uses for Unicode transfers (CODE_FLAG=UTF8 or CODE_FLAG=UTF16).

When the remote system is XCOM Data Transport for z/OS:

CCSID#nnnnn/tttttttt

nnnnnn - specifies the CCSID number that corresponds to the character set of the local file. 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

NOTE
If the technique search order is not specified, Unicode Services defaults to ‘RECLM’.

When the remote system is XCOM Data Transport for Windows or XCOM Data Transport for Linux/Unix:

CCSID#nnnnn

nnnnnn - specifies the CCSID number that corresponds to the character set. Valid values are 1 - 65535. Alternatively, can be specified as an IANA character set name, or (ICU) acceptable alias name.

Range: 0 to 60 characters

Default: The DEFAULT_CHARSET parameter specifies the default in the XCOM Data Transport Default Options Table/ Global Parameters on the remote system.

REMOTE_DELIM

REMOTE_DELIM specifies an optional encoding for which the specified REMOTE_CHARSET is based. The encoding can be either ASCII or EBCDIC. If specified, the encoding must be the first option in the list. If REMOTE_CHARSET is UTF based charset, UTF rules will be applied disregarding encoding specified in REMOTE_DELIM.

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

When the remote system is XCOM Data Transport for z/OS:

Additionally it specifies the delimiter to use for USS-based output files when FILEDATA=TEXT.

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

CRLFL - Carriage return/Line feed

LFCR - Line feed/Carriage return

CRNL - Carriage return/New line

Note: The only valid encoding for XCOM Data Transport for z/OS is EBCDIC.

When the remote system is XCOM Data Transport for Windows or XCOM Data Transport for Linux/Unix:

Additionally, specifies a colon separated list of record delimiters that are used to mark and detect the end of a record.

- **EBCDIC Rules**

Record Delimiter Option	Mutually exclusive with
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
FF	
NA	

- **ASCII Rules**

Record Delimiter Option	**Mutually exclusive with**
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
VT	
FF	

- **UTF Rules**

Record Delimiter Option	**Mutually exclusive with**
CR	CRLF, CRNL
LF	LFCR
NL	
CRLF	CR
LFCR	LF
CRNL	CR
VT	
FF	
LS	
PS	

Range: 0 to 60 characters

Default: The DEFAULT_DELIM parameter specifies the default in the XCOM Data Transport Default Options Table/Global Parameters on the remote system.

The tables indicate valid record delimiters for various encodings. If a record delimiter is not applicable for encoding on which REMOTE_CHARSET is based, it should not be used.

Example: If REMOTE_CHARSET is not UTF based, PS and LS delimiters should not be used. Similarly if REMOTE_CHARSET is EBCDIC based, VT, PS, and LS delimiters should not be used.

In these cases, these delimiters will be treated as invalid and disregarded if used. If no valid delimiter is found in REMOTE_DELIM, delimiters will be derived as explained below.

If REMOTE_DELIM is not specified or doesn't contain at least one valid delimiter, DEFAULT_DELIM parameter in the XCOM Data Transport Default Options Table/Global Parameters on the remote system is considered. If DEFAULT_DELIM is also not specified or doesn't contain at least one valid delimiter, platform default values are used as shown below,

Default for ASCII: 'LF,CRLF'

- Default for EBCDIC: 'NL'

Note: During Unicode conversion some of the delimiters may fail to convert to some code pages.

For example, In case of Send File transfer, when REMOTE_CHARSET set to ASCII based charset and REMOTE_DELIM set to NL/CRNL, then transfer may fail. The Unicode conversion fails as ASCII based code pages may not have mapping character for NL/CRNL.

REMOTE_FILE

Indicates the file on the remote computer to which the transferred data is being written. If you are creating the file (FILE_OPTION=CREATE), the file name must be consistent with the file naming conventions of the remote system. The local XCOM Data Transport system does not validate this name. The remote I/O system determines whether the file name is valid.

For wildcard transfers, use an asterisk (*) as a file name to indicate and to inform the receiving partner that multiple files will be sent. For example, REMOTE_FILE=/PAYROLL/*.

If multiple files are sent and the user specifies a file name, all files received by the partner are written to that specified file as one single file.

An asterisk used to send files to an IBM mainframe system shows that all files are to be transferred to a partitioned data set (PDS).

For platforms that support it, you can specify a common file extension to append to each file name. For example, REMOTE_FILE=/PAYROLL/*.TXT.

NOTE
For send file transfers only.

Range: 1 to 256 characters

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_FILE_RF

Indicates the name of the file to be retrieved from the remote system.

For wildcard transfers, use an asterisk (*) as a file name to indicate that all files within the specified PDS/Directory should be transferred. For example, the statement REMOTE_FILE_RF=/NAMES/* indicates that all files under the NAMES directory should be transferred.

When a prefix is followed by an asterisk, all members beginning with a specific prefix are transferred. For example, REMOTE_FILE_RF=/NAMES/AL* requests that files AL, ALEX, and ALICE should be transferred. The same rules apply if an asterisk is followed by a suffix.

The actual file name range (not including its path) for wildcard transfers can be between 0 and 71 characters. This also includes the file extension where applicable. File names over 71 characters are truncated.

NOTE
For retrieve file transfers only.

Range: 1 to 256 characters

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_SYSTEM

The name of the remote system that receives a file, job, report, or ping request.

For SNA/APPC protocols, the name is as specified in the SNA/APPC configuration of the local system. For RS/6000, the name is the LU6.2 Side Information record.

For TCP/IP protocols, the value can be the remote systems IP address, host name, or domain name.

Range: 1 to 256 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_SYSTEM_RF

The name of the remote system that sends a file on a receive file operation. If no value is specified here, the value in REMOTE_SYSTEM is used.

For SNA/APPC protocols, the name is as specified in the SNA/APPC configuration of the local system. For RS/6000, this is the LU6.2 Side Information record.

For TCP/IP protocols, the value can be the remote system's IP address, host name, or domain name.

Range: 1 to 256 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_SYSTEM_SJ

The name of the remote system to which a job is sent. If no value is specified here, the value in REMOTE_SYSTEM is used.

For SNA/APPC protocols, the name is as specified in the SNA/APPC configuration of the local system. For RS/6000, this is the LU6.2 Side Information record.

For TCP/IP protocols, the value can be the remote system's IP address, host name, or domain name.

Range: 1 to 256 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOTE_SYSTEM_SR

The name of the remote system to which a report is sent. If no value is specified here, the value in REMOTE_SYSTEM is used.

For SNA/APPC protocols, the name is as specified in the SNA/APPC configuration of the local system. For RS/6000, this is the LU6.2 Side Information record.

For TCP/IP protocols, the value can be the remote system's IP address, host name, or domain name.

Range: 1 to 256 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

REMOVE_TRAIL_BLANKS

Indicates whether or not to remove the blanks at the end of each record when receiving a text file from a remote partner.

Range: Yes or No

Default: No

REPORT_TITLE

This field contains the report title and job number that will be printed on the report. The field has the following format:

8 characters	2 characters	8 characters
Job Name	Blanks	Job Number

The job is optional and can be skipped. The job name can also be skipped, but if you skip the job name and want to use the job number, you must pad the number with 10 blanks.

> **NOTE**
> For report transfers only.

Examples

```
REPORT_TITLE="Salary94  Job12345"
REPORT_TITLE="          Job23456"
```

Non-example

```
REPORT_TITLE="     Job34567"
```

This is not a valid REPORT_TITLE because the job number spans both subfields.

This parameter is used by XCOM Data Transport on remote systems in the following ways:

System	Uses the REPORT_TITLE…
z/OS	To interpret a non-blank value in this field as specifying the generation of a separator (banner) page for this value.
VAX/VMS	To print with the report.
UNIX/Linux	To allow XCOM Data Transport to pass this field to the LP spooler as a title field.
Other systems	As a descriptive comment only and does not print it as part of the report.

Range: 0 to 21 alphanumeric or blank characters

Default: None

NOTE
You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

RESTART_SUPPORTED

Specifies whether automatic restart is to be supported on a transfer.

Range: YES or NO

Default: YES

NOTE
This is a Version 2 parameter.

RETPD

Specifies the number of days (1 to 9999) that the tape data set being created is to be retained.

Range: 1 to 9999

Default: None

NOTE
RETPD and EXPDT are mutually exclusive; specify one or the other.

RETRY_TIME

The number of seconds between retries of unsuccessful transfers.

Range: 0 to 99999

Default: 1

NOTE
This is a Version 2 parameter.

RMTNTFYL

Specifies the remote user notification level when sending data to a remote system.

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: ALL

RNOTIFY

Specifies the remote user notification type when sending data to a remote system.

MAIL
Mail is sent to the user

WRITE
A message is displayed on the local Windows system console.

ALL
A message is displayed on the local Windows system console.

TSO
Notify specified TSO user.

WTO
Write to log only (WTO).

ROSCOE
Notify the specified Roscoe user.

CICS
Notify CICS user (not used in Release 1).

LU
Notify Logical Unit (not used in Release 1).

NONE
No notification is sent.

Default: NONE

- The R in NOTIFYR indicates that the remote system governs the processing of the resulting notification on that system.
- A MAIL message is sent to the specified user only if a MAPI-compliant provider is configured as a non-domain account and/or allows local accounts (such as the xcomd service and the local user ID) access to the Windows system (for example, Microsoft Mail boxes and address books on the local system).
- Some Microsoft Outlook 2003 security updates may prevent mail notification from working.

RNOTIFYNAME

Specifies the remote user notification type when sending data to a remote system.

MAIL
Mail is sent to the user

WRITE
A message is displayed on the local Windows system console.

ALL
A message is displayed on the local Windows system console.

TSO
Notify specified TSO user.

WTO
Write to log only (WTO).

ROSCOE
Notify the specified Roscoe user.

CICS
Notify CICS user (not used in Release 1).

LU
Notify Logical Unit (not used in Release 1).

NONE

No notification is sent.

Default: NONE

- The R in NOTIFYR indicates that the remote system governs the processing of the resulting notification on that system.
- A MAIL message is sent to the specified user only if a MAPI-compliant provider is configured as a non-domain account and/or allows local accounts (such as the xcomd service and the local user ID) access to the Windows system (for example, Microsoft Mail boxes and address books on the local system).
- Some Microsoft Outlook 2003 security updates may prevent mail notification from working.

SAVE_PASSWORD_IN_CNF

The SAVE_PASSWORD_IN_CNF parameter specifies whether the xcomtool GUI saves the password in the .CNF file.

Range:

YES

Saves the password in the .CNF file in an encrypted format.

NO

Does not save the password in the .CNF file and blanks the password field out after each operation.

Default: NO

NOTE
XCOM Data Transport for Linux PC x64, XCOM Data Transport for AIX 64, XCOM Data Transport for Solaris Sparc 64, XCOM Data Transport for Solaris x86 64, and XCOM Data Transport for Linux s390x no longer support xcomtool. The xcomtool utility is not shipped with the product.

SECONDARY_ALLOC

This parameter identifies secondary storage allocation for a data set that was created on an IBM mainframe.

Range: 0 to 16,777,215

Default: 0

NOTE
This parameter is a Version 2 parameter.

SECURE_SCHEDULE

Specifies whether the meta-transfer uses SSL.

Y

The meta-transfer uses SSL.

N

The meta-transfer does not use SSL.

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 to connect to a TLS or an SSL listener on the remote partner.

NO

Performs a non-secure transfer. The transfer uses a non-OpenSSL socket.

Default: NO

SESSIONS_FILE

The path name of the xcom.ses file, which tells XCOM Data Transport how many sessions can be used by a single LU.

Range: 0 to 256 characters

Default: $XCOM_HOME/xcom.ses

> **NOTE**
> $XCOM_HOME is an environment variable.

SHELL_CMD

Name of the command that runs jobs, reports, notification scripts, and post-processing scripts on the local system.

Range: 1 to 256 characters

Default: /bin/sh

SMTP_SERVER

This parameter specifies the name of the SMTP server.

> **NOTE**
> This is required only when MAIL_TYPE=SMTP.

Range: 1 to 64 characters

Default: None

SOCK_DELAY

TCP/IP socket option TCP_NODELAY. Refers to the Nagle algorithm for send coalescing. By default, small sends may be delayed. Should have no impact for normal XCOM Data Transport record sizes. Used for TCP/IP transfers only.

> **NOTE**
> Socket options affect the way XCOM Data Transport uses the TCP/IP stack implementation.

Range: The range of values is listed in the following table:

YES

Small sends may be delayed. (Does not disable the Nagle algorithm.)

NO

All sends are immediate. (Disables the Nagle algorithm.)

Default: YES

SOCK_RCV_BUF_SIZE

TCP/IP Socket option SO_RCVBUF. The buffer size for receives. Use zero for the default size provided by the socket implementation. The value for SOCK_RCV_BUF_SIZE can be smaller than the value for TXPI_BUF_SIZE. Used for TCP/IP transfers only.

> **NOTE**
> Socket options affect the way XCOM Data Transport uses the TCP/IP stack implementation.

Range: 0 to 65536

Default: 0

SOCK_SEND_BUF_SIZE

TCP/IP Socket option SO_SNDBUF. The buffer size for sends. Use zero for the default size provided by the socket implementation. The value for SOCK_SEND_BUF_SIZE can be smaller than the value for TXPI_BUF_SIZE. Used for TCP/IP transfers only.

> **NOTE**
> Socket options affect the way XCOM Data Transport uses the TCP/IP stack implementation.

Range: 0 to 65536

Default: 0

SPACE

The unit of storage allocation for the remote file.

CYL

Cylinders

TRK

Tracks

BLK

Blocks

REC

Records

Specify by:

Primary allocation space for the remote file

- Secondary allocation for the remote file
- Directory blocks for the remote file

Default: CYL

SPOOL_FLAG

Indicates whether the report is to be spooled to disk or printed immediately. For report transfers only.

> **NOTE**
> If the remote system is an IBM mainframe, this parameter has no effect on the transfer.

Range: YES or NO

Default: YES

SPRTY

Indicates the scheduling priority of this meta-transfer request. When multiple file transfer requests are eligible for initiation (that is, they are past their start date/time) to the same LU or GROUP, those with higher SPRTY values are scheduled first. One is the lowest priority.

1 to 255

Specifies a scheduling priority for this file transfer request.

> **NOTE**
> This feature has no effect once the transfer begins executing. Do not confuse it with EPRTY, the execution priority.

Default: 16

START_DATE

Indicates the date on which the transfer becomes eligible for execution. The format is mm/dd/yy. If this field is blank, the current date is used.

Example:

A value of 02/28/13 indicates February 28, 2013 as the start date.

Format: *mm/dd/yy*

Default: None

Note: You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

START_TIME

Indicates the time when the transfer becomes eligible for execution. The military format of *hh*:*mm*:*ss* is used. If this field is blank, then the current time is used.

Example:

A value of 14:00:00 indicates 2 p.m. as the start time.

Format: *hh*:*mm*:*ss*

Default: None

Note: You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

STARTDATE

Specifies the date when the transfer is to begin.

The transfer start date can be provided in one of three formats:

1. *yyyymmdd*

yyyy

Is a four-digit designation for a year (for example, 2013).

mm

Is a two-digit designation for one of the twelve months of the year as shown in the following chart:

```
01 = January      02 = February      03 = March
```

```
04 = April        05 = May          06 = June
07 = July         08 = August       09 = September
10 = October      11 = November     12 = December
```

dd

Is a two-digit number in the range 01 to 31 designating a day of the month.

Example:

```
STARTDATE=20130201 schedules a transfer to begin on February 1, 2013.
```

- *yyddd* (Julian date)

yy

Is a two-digit designation for a year (for example, 13).

ddd

Is a three-digit number in the range 001 to 366 designating a day of the year.

Example:

```
STARTDATE=13032 schedules a transfer to start on the 32nd day of 2013, which is the same as February 1, 2013.
```

- +*nnn* (*nnn* days from today)

nnn

Is a number in the range 1 to 999.

Example:

Specifying STARTDATE=+31 on January 1, 2013 schedules a transfer to begin on February 1, 2013.

Default: Current date

STARTTIME

Specifies the time (*hhmm*) this transfer becomes eligible for execution.

0000 to 2400

Specifies the time (*hhmm*) this transfer becomes eligible for execution.

- You can also set the time for the future in terms of its separation in hours and minutes (+*hhmm*) from the current time.
- For example, if a transmission is to start no earlier than 2 p.m.:
  STARTTIME=1400
- Or if the start time is two hours from now:
  STARTTIME=+0200.

Default: Current time

STAT_FREQUENCY

Indicates the frequency with which transfer statistics are made available to xcomqm. Intended for tuning high-speed links. Longer values help performance, but byte/record counts in xcomqm -D may be slightly behind the actual counts.

Range: 1 to 9999 records

Default: 10

STCIP

Specifies the IP address of the XCOM Data Transport server to handle the metatransfer.

Range: 1 to 64 characters

Default: None

STCPORT

Specifies the TCP/IP port number of the XCOM Data Transport server to handle the metatransfer request.

Range: 1 to 65536

Default: 8044

STCTRNENCRL_CIPHER

STCTRNENCRL_CIPHER specifies the ciphers that XCOM Data Transport uses to encrypt the password fields for locally initiated -c5 meta-transfer requests.

The cipher list consists of one or more ciphers from the table. The ciphers are separated by colons. An exclamation point (!) or hyphen (-) can precede each cipher in the cipher list. If an exclamation point is used, the ciphers are permanently deleted from the list. The ciphers that are deleted can never reappear in the list even if they are explicitly stated. If a hyphen is used, the ciphers are deleted from the list, but some or all ciphers are added again using later options.

Default: COMPAT

Value	Comments
ALL	ALL ciphers: `AES:3DES:RC4:RC2:DES:XCOM` ALL does NOT include the COMPAT value.
DES	All DES ciphers: `DES-CBC:DES-ECB:DES-CFB:DES-OFB`
DES-CBC	DES cipher with cipher-block chaining
DES-ECB	DES cipher with electronic codebook
DES-CFB	DES cipher with cipher feedback
DEC-OFB	DES cipher with output feedback
3DES	All 3DES ciphers: `3DES-CBC:3DES-ECB:3DES-CFB:3DES-OFB`
3DES-CBC	3DES cipher with cipher-block chaining
3DES-ECB	3DES cipher with electronic codebook
3DES-CFB	3DES cipher with cipher feedback
3DES-OFB	3DES cipher with output feedback
AES	All AES ciphers: `AES256-CBC:AES256-ECB:AES256-CFB:AES256-OFB:AES192-CBC:AES192-ECB:AES192-CFB:AES192-OFB:AES128-CBC:AES128-ECB:AES128-CFB:AES128-OFB`
AES128	All AES 128-bit ciphers: `AES128-CBC:AES128-ECB:AES128-CFB:AES128-OFB`
AES128-CBC	AES 128-bit cipher with cipher-block chaining
AES128-ECB	AES 128-bit cipher with electronic codebook
AES128-CFB	AES 128-bit cipher with cipher feedback
AES128-OFB	AES 128-bit cipher with output feedback

AES192	All AES 192-bit ciphers: `AES192-CBC:AES192-ECB:AES192-CFB:AES192-OFB`
AES192-CBC	AES 192-bit cipher with cipher-block chaining
AES192-ECB	AES 192-bit cipher with electronic codebook
AES192-CFB	AES 192-bit cipher with cipher feedback
AES192-OFB	AES 192-bit cipher with output feedback
AES256	All AES 256-bit ciphers: `AES256-CBC:AES256-ECB:AES256-CFB:AES256-OFB`
AES256-CBC	AES 256-bit cipher with cipher-block chaining
AES256-ECB	AES 256-bit cipher with electronic codebook
AES256-CFB	AES 256-bit cipher with cipher feedback
AES256-OFB	AES 256-bit cipher with output feedback
RC2	All RC2 ciphers
RC2-CBC	RC2 cipher with cipher-block chaining
RC2-ECB	RC2 cipher with electronic codebook
RC2-CFB	RC2 cipher with cipher feedback
RC2-OFB	RC2 cipher with output feedback
RC4	RC4 cipher
XCOM	XCOM Data Transport proprietary cipher
COMPAT	Permits the XCOM Data Transport proprietary cipher without the cipher negotiation that is required for backward password compatibility with XCOM Data Transport releases before 11.6. This cipher is also required for transfers that are sent to earlier XCOM Data Transport for z/OS releases.

Examples:

To request all ciphers except for any of the DES ciphers, use the following command:

```
STCTRNENCRL_CIPHER=ALL:!DES
```

To request only a 3DES or AES cipher, use the following command:

```
STCTRNENCRL_CIPHER=3DES:AES
```

To disable the cipher negotiation or remain backward compatible with the earlier releases of XCOM Data Transport, use the following command:

```
STCTRNENCRL_CIPHER=COMPAT
```

STORCLAS

Specifies the name of the storage class for a new SMS-managed data set.

> **NOTE**
> This parameter applies only to mainframe SMS data sets.

Range: One to eight characters

Default: None

SYSID

This parameter specifies the system ID (one to four characters).

This value is set during the installation process and is used for Trusted Transfers and for getting history records.

SYSID and SYSNAME together provide a unique system identifier.

Range: 1 to 4 characters

Default: None

SYSNAME

This parameter specifies the system name.

This value is set during the installation process and is used for Trusted Transfers and for getting history records.

SYSID and SYSNAME together provide a unique system identifier.

Range: 1 to 8 characters

Default: None

SYSTEM_USER_DATA

A user-defined text data field that can be used by an application as a customized system identifier. When specified in the configuration file of a locally initiated transfer, this identifying information is sent to the remote system and can be used to identify the system to certain applications. When received, the identifying information is displayed in the xcomqm detail display if the -D option is used, and is labeled System user data.

Range: 0 to 10 characters

Default: None

SYSUDATA

An open field where a user can specify any text associated with the transfer.

Note: This is equivalent to the USER_DATA field on Windows.

Range: 0 to 10 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

TAPE

Indicates to the remote system whether the transfer is to a tape volume or a disk file.

YES

The transfer is to a tape volume. Mounts are allowed when performing dynamic allocation.

NO

The transfer is to a disk file.

Default: NO

TCP_CMD

Path and name of the XCOM Data Transport program started by the XCOMD service that is used for queued locally initiated transfers, and for all remotely initiated transfers for TCP/IP protocols.

Range: 0 to 256

Default: $XCOM_HOME/bin/xcomtcp

> **NOTE**
> $XCOM_HOME is an environment variable.

TDUDATA

An open field where a user can specify any text associated with the transfer.

Note: This is equivalent to the XFERDATA field on UNIX and Linux.

Range: 0 to 10 characters

Default: None

TEMPDIR

This parameter indicates the directory in which temporary files for jobs and reports can be created.

Range: 1 to 256 characters

Default: $XCOM_HOME/tmp

> **NOTE**
> $XCOM_HOME is an environment variable.

TP_CMD

Command to start a transaction program using SNA protocols.

Range: 0 to 256 characters

Default: $XCOM_HOME/bin/xcomtp

TRACE_PATH

The directory containing the trace data files.

Range: 0 to 256 characters

Default: $XCOM_HOME/trace

> **NOTE**
> $XCOM_HOME is an environment variable.

TRANSFER_NAME

Allows a user to assign a name to a transfer. This name is also available to the remote XCOM Data Transport system. In addition, if the remote XCOM Data Transport system has defined a transfer identifier, that name is available to the local XCOM Data Transport for UNIX or Linux system. When received, the transfer identifier is displayed in the xcomqm display if the -D option is used, and is labeled Transfer name. The equivalent to this parameter on z/OS is the transfer identifier (XFERID).

Range: 0 to 10 characters

Default: None

TRANSFER_TYPE

Generated by the graphical user interface to specify the type of transfer to initiate.

Can also be used in a configuration file that has multiple transfers in it, to specify a different transfer type from the default value.

Range: 1 to 4

1
Send job

2
Send report

3
Send file

4
Receive file

Default: 3

> **NOTE**
> The functionality of this parameter is similar to using the **-c** *number* command with xcom62 or xcomtcp at the command line. However, the values used to indicate the type of transfer are different.

TRANSFER_USER_DATA

A user-defined text data field that can be used by an application as a customized transfer identifier. When specified in the configuration file of a locally initiated transfer, this identifying information is sent to the remote system and can be used to identify the transfer to certain applications. When received, the identifying information is displayed in the xcomqm detail display if the -D option is used, and is labeled Transfer user data.

Range: 0 to 10 characters

Default: None

> **NOTE**
> You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

TRNENCRL_CIPHER

TRNENCRL_CIPHER specifies the ciphers that XCOM Data Transport use to encrypt the password fields for locally initiated transfers.

The cipher list consists of one or more ciphers from the table. The ciphers are separated by colons. An exclamation point (!) or hyphen (-) can precede each cipher in the cipher list. If an exclamation point is used, the ciphers are permanently deleted from the list. The ciphers that are deleted can never reappear in the list even if they are explicitly stated. If a hyphen is used, the ciphers are deleted from the list, but some or all ciphers are added again using later options.

Default: COMPAT

Value	Comments
ALL	ALL ciphers: `AES:3DES:RC4:RC2:DES:XCOM` ALL does NOT include the COMPAT value.
DES	All DES ciphers: `DES-CBC:DES-ECB:DES-CFB:DES-OFB`
DES-CBC	DES cipher with cipher-block chaining
DES-ECB	DES cipher with electronic codebook
DES-CFB	DES cipher with cipher feedback
DEC-OFB	DES cipher with output feedback
3DES	All 3DES ciphers: `3DES-CBC:3DES-ECB:3DES-CFB:3DES-OFB`
3DES-CBC	3DES cipher with cipher-block chaining
3DES-ECB	3DES cipher with electronic codebook
3DES-CFB	3DES cipher with cipher feedback
3DES-OFB	3DES cipher with output feedback
AES	All AES ciphers: `AES256-CBC:AES256-ECB:AES256-CFB:AES256-OFB:AES192-CBC:AES192-ECB:AES192-CFB:AES192-OFB:AES128-CBC:AES128-ECB:AES128-CFB:AES128-OFB`
AES128	All AES 128-bit ciphers: `AES128-CBC:AES128-ECB:AES128-CFB:AES128-OFB`
AES128-CBC	AES 128-bit cipher with cipher-block chaining
AES128-ECB	AES 128-bit cipher with electronic codebook
AES128-CFB	AES 128-bit cipher with cipher feedback
AES128-OFB	AES 128-bit cipher with output feedback
AES192	All AES 192-bit ciphers: `AES192-CBC:AES192-ECB:AES192-CFB:AES192-OFB`
AES192-CBC	AES 192-bit cipher with cipher-block chaining
AES192-ECB	AES 192-bit cipher with electronic codebook
AES192-CFB	AES 192-bit cipher with cipher feedback
AES192-OFB	AES 192-bit cipher with output feedback
AES256	All AES 256-bit ciphers: `AES256-CBC:AES256-ECB:AES256-CFB:AES256-OFB`
AES256-CBC	AES 256-bit cipher with cipher-block chaining
AES256-ECB	AES 256-bit cipher with electronic codebook
AES256-CFB	AES 256-bit cipher with cipher feedback
AES256-OFB	AES 256-bit cipher with output feedback
RC2	All RC2 ciphers
RC2-CBC	RC2 cipher with cipher-block chaining
RC2-ECB	RC2 cipher with electronic codebook
RC2-CFB	RC2 cipher with cipher feedback
RC2-OFB	RC2 cipher with output feedback

RC4	RC4 cipher
XCOM	XCOM Data Transport proprietary cipher
COMPAT	Permits the XCOM Data Transport proprietary cipher without the cipher negotiation that is required for backward password compatibility with XCOM Data Transport releases before 11.6. This cipher is also required for transfers that are sent to earlier XCOM Data Transport for z/OS releases.

Examples:

To request all ciphers except for any of the DES ciphers, use the following command:

```
TRNENCRL_CIPHER=ALL:!DES
```

To request only a 3DES or AES cipher, use the following command:

```
TRNENCRL_CIPHER=3DES:AES
```

To disable the cipher negotiation or remain backward compatible with the earlier releases of XCOM Data Transport, use the following command:

```
TRNENCRL_CIPHER=COMPAT
```

TRNENCRR_CIPHER

The TRNENCRR_CIPHER parameter specifies the ciphers that XCOM Data Transport uses to encrypt the password fields for transfers that are initiated remotely.

The permitted list of ciphers is matched against the requested list of ciphers that is provided by the local system using the TRNENCRL_CIPHER parameter. The common cipher with the highest ranking is selected to encrypt the password fields.

The cipher list consists of one or more ciphers. Ciphers are separated by colons. The highest ranked cipher is listed first, and the lowest ranked cipher is listed last.

Each cipher in the list can be preceded by an exclamation point (!), hyphen (-), or plus sign (+).

- If an exclamation point is used, the ciphers are permanently deleted from the list. The ciphers that are deleted can never reappear in the list even when they are explicitly stated.
- If a hyphen is used, the ciphers are deleted from the list. Some or all ciphers are added again using later options.
- If a plus sign is used, the ciphers are moved to the end of the list. This option does not add any new ciphers. The option just looks for existing ones to move to the end of the list.

Default: XCOM:ALL:COMPAT

The following table shows the permitted ciphers:

Value	**Comments**
ALL	ALL ciphers: `AES:3DES:RC4:RC2:DES:XCOM` ALL does NOT include the COMPAT value.
DES	All DES ciphers: `DES-CBC:DES-ECB:DES-CFB:DES-OFB`
DES-CBC	DES cipher with cipher-block chaining
DES-ECB	DES cipher with electronic codebook
DES-CFB	DES cipher with cipher feedback
DEC-OFB	DES cipher with output feedback

3DES	All 3DES ciphers: `3DES-CBC:3DES-ECB:3DES-CFB:3DES-OFB`
3DES-CBC	3DES cipher with cipher-block chaining
3DES-ECB	3DES cipher with electronic codebook
3DES-CFB	3DES cipher with cipher feedback
3DES-OFB	3DES cipher with output feedback
AES	All AES ciphers: `AES256-CBC:AES256-ECB:AES256-CFB:AES256-OFB:AES192-CBC:AES192-ECB:AES192-CFB:AES192-OFB:AES128-CBC:AES128-ECB:AES128-CFB:AES128-OFB`
AES128	All AES 128-bit ciphers: `AES128-CBC:AES128-ECB:AES128-CFB:AES128-OFB`
AES128-CBC	AES 128-bit cipher with cipher-block chaining
AES128-ECB	AES 128-bit cipher with electronic codebook
AES128-CFB	AES 128-bit cipher with cipher feedback
AES128-OFB	AES 128-bit cipher with output feedback
AES192	All AES 192-bit ciphers: `AES192-CBC:AES192-ECB:AES192-CFB:AES192-OFB`
AES192-CBC	AES 192-bit cipher with cipher-block chaining
AES192-ECB	AES 192-bit cipher with electronic codebook
AES192-CFB	AES 192-bit cipher with cipher feedback
AES192-OFB	AES 192-bit cipher with output feedback
AES256	All AES 256-bit ciphers: `AES256-CBC:AES256-ECB:AES256-CFB:AES256-OFB`
AES256-CBC	AES 256-bit cipher with cipher-block chaining
AES256-ECB	AES 256-bit cipher with electronic codebook
AES256-CFB	AES 256-bit cipher with cipher feedback
AES256-OFB	AES 256-bit cipher with output feedback
RC2	All RC2 ciphers
RC2-CBC	RC2 cipher with cipher-block chaining
RC2-ECB	RC2 cipher with electronic codebook
RC2-CFB	RC2 cipher with cipher feedback
RC2-OFB	RC2 cipher with output feedback
RC4	RC4 cipher
XCOM	XCOM Data Transport proprietary cipher
COMPAT	Permits the XCOM Data Transport proprietary cipher without cipher negotiation. This parameter is required for backward compatibility with XCOM Data Transport versions before 11.6. This cipher is also required for transfers that are sent to XCOM Data Transport for z/OS r11.6 and earlier versions.

Examples:

To permit all ciphers except for any DES ciphers and remain compatible with XCOM Data Transport versions before r11.6, use the following example:

```
TRNENCRR_CIPHER=ALL:!DES:COMPAT
```

To permit only a 3DES or AES cipher, use the following example:

```
TRNENCRR_CIPHER=3DES:AES
```

To permit all ciphers for transfers from 11.6 or higher but remain backward compatible with earlier versions of XCOM Data Transport, use the following example:

```
TRNENCRR_CIPHER=ALL:!XCOM:COMPAT
```

TRNENCRR_DHBITS

Specify the size (in bits) of the prime number that is used during DH (Diffie-Hellman) exchange for remotely initiated transfers or meta-transfers. The exchanged value generates the key that is used to encrypt the password fields transmitted in the XCOM Data Transport header.

Default: 1024

Value	Comments
256	DH exchange using a 256-bit prime number
512	DH exchange using a 512-bit prime number
1024	DH exchange using a 1024-bit prime number
2048	DH exchange using a 2048-bit prime number
4096	DH exchange using a 4096-bit prime number

NOTE

- The more bits used for the prime number the more secure the key exchange.
- The more bits used for the prime number the more CPU overhead is required to negotiate a secret value that is based on the prime number.
- Each XCOM Data Transport connection generates a unique secret value.
- No DH exchange is performed when the cipher the TRNENCRL_CIPHER/TRNENCRR_CIPHER negotiates is XCOM.

TRUNCATION

Indicates whether XCOM Data Transport truncates excess characters in the source file if the record exceeds the maximum record length as indicated by the MAXRECLEN parameter. If NO is selected, and the maximum record length is exceeded, XCOM Data Transport aborts the transfer. This parameter is ignored if CARRIAGE_FLAG=NO.

NOTE
Set **TRUNCATE** to **NO** for Binary transfers. Otherwise, it can result in truncated records on receiving platform based on LRECL and MAXRECLEN values.

Range: YES or NO

Default: NO

TRUST_DATABASE_NAME

This parameter specifies the name of the database where the trusted tables were created.

Range: 1 to 256 characters

Default: None

TRUST_DATABASE_TYPE

The TRUST_DATABASE_TYPE parameter specifies the database type that resides on the database server.

Range: Db2, MySQL, or Oracle

Default: Db2

TRUST_ODBC

This parameter specifies the ODBC Data Source Name for the Trusted Tables.

Range: 1 to 32 characters

Default: None

TRUST_OWNER

This parameter identifies the owner of the database tables. May be omitted if using MySQL or if it is the same as TRUST_USER when using DB2.

Range: 1 to 32 characters

Default: None

TRUST_PASS

This parameter specifies the password of the user (TRUSTED_USER) for the Trusted Tables.

> **NOTE**
> When any changes are made via the Global Parameters tab, the updates encrypt this password and it is saved as TRUST_PASS.ENCRYPTED.

Range: 1 to 128 characters

Default: None

TRUST_PORT

This parameter specifies the database server port where the trusted database resides.

Range: 1 to 65535

Default: 50000

TRUST_USER

This is a generic user ID that has been defined to the RDMS for the Trusted Tables.

Range: 1 to 64 characters

Default: None

TRUST_SERVER

This parameter specifies the IP address or name of the server where the trusted database resides.

Range: 1 to 256 characters

Default: None

TRUST_TABLE_PFX

This parameter specifies the prefix to use for the names of the Trusted Tables.

Range: 1 to 16 characters

Default: XCOM_TRUSTED

TRUSTED

Allows the user to request a trusted transfer and the partner's XCOM Data Transport TRUSTED database to be searched to verify the user's credentials. This eliminates the need for the user to specify a USERID and PASSWORD. If XCOM_TRUSTED_OVR is set to NO or no USERID is specified, the USERID of the process that initiated the transfer is used.

> **NOTE**
> TRUSTED=YES cannot be specified with indirect transfers, because this is not supported.

Range: YES, NO, Y, N

Default: NO

TXPI_BUF_SIZE

For TCP/IP transfers, the internal buffer size for sends and receives. The default size allows multiple XCOM Data Transport records to be received in a single socket call. With this default, if your XCOM Data Transport record size is less than 32K, XCOM Data Transport attempts to receive multiple records in a single socket call. Used for TCP/IP transfers only.

Range: 0 to 65536

Default: 32768

TXPI_RECEIVE_TIMEOUT

Maximum wait time, in seconds, that this XCOM Data Transport waits to receive from the partner system. If a value of 0 is specified, it waits indefinitely. Use for TCP/IP transfers only.

Range: 0 to 999 seconds

Default: 0

TXPI_SEND_CHECK_FREQ

Indicates the frequency with which 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.

Range: 0 to 9999

Default: 10

TXPI_TERM_TIMEOUT

Maximum wait time, in seconds, for 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 occurs only during an error in the termination of the connection. Used for TCP/IP transfers only.

Range: 0 to 999 seconds

Default: 20 seconds

UMASK

Used to set the permissions assigned to a file when the file is being created and received on the 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 umask command.

Range: 000 to 777

Default: 022

Note:

For directories -XCOM Data Transport sets permissions for a created directory to 7xx, no matter what owner UMASK value 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, XCOM Data Transport sets permissions for a created file to 6xx, where xx represents the permissions with the specified UMASK removed. After the transfer has been completed, XCOM Data Transport sets the owner permission with the specified UMASK removed.

UNIT

Specifies the unit on which a data set is to be created on an IBM mainframe.

> **NOTE**
> Used when FILE_OPTION=CREATE.

Range: Zero to six characters

Default: None

UNIT_RF

Specifies the unit from which a data set is to be retrieved from an IBM mainframe.

Range: Zero to six characters

Default: None

UNITCT

Specifies the number of units to be allocated on the remote system.

This is a tape parameter and is used when the partner is an IBM mainframe.

Range: 1 to 20

Default: None

USE_TP_SECURITY

Only for NCR systems using SNA. When set to YES, XCOM Data Transport instructs the APPC to send out the user ID and password in the APPC attach request.

> **NOTE**
> This parameter applies to outgoing security only.

Range: YES or NO

Default: NO

USEROVR

This parameter specifies whether the local user ID parameter LUSERID can be used when scheduling a transfer to a remote machine, using the xcomtcp -c5 option.

YES

Specifies that LUSERID can be used.

NO

Indicates that the local user ID (LUSERID) cannot be changed.

Default: YES

USERID

The user ID that the security system on the remote system checks before granting access for the file transfer.

Range: 0 to 12 characters

Default: None

VERSION

Indicates the version of the XCOM Data Transport protocol to be used for this transfer. For TCP/IP, only a value of 2 is valid.

Range: 1 or 2

Default: 2

> **NOTE**
> This is a Version 2 parameter.

VOLCT

Specifies the maximum number of volumes to be used in processing a multi-volume output tape data set on the remote system.

Range: 1 to 255

Default: None

VOLSQ

Specifies the sequence number of the first volume of a multi-volume remote data set to be used.

Range: 1 to 255

Default: None

VOLUME

Specifies the volume on which a data set is to be created on an IBM mainframe.

Range: Zero to six characters

Default: None

VOLUME_RF

Specifies the volume from which a data set is to be retrieved from an IBM mainframe.

Range: Zero to six characters

Default: None

WRITER

Specifies the name of the external writer that is to process the report on the remote system.

xxxxxxxx

Specifies up to eight alphanumeric characters identifying the external writer that is to process this report on the remote system.

NOTE

This parameter cannot be specified with Version 1 transfers.

XBUFFSIZE

The size of XCOM Data Transport's internal buffer.

WARNING
Do not change this parameter unless you are instructed to do so by Technical Support.

Range: 1 to 32767

Default: 32767

XCOM_CONFIG_SSL

This parameter specifies the configssl.cnf file path and file name.

Range: 1 to 256 characters

Default: $XCOM_HOME/config/configssl.cnf

NOTE

$XCOM_HOME is an environment variable

- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XCOM_ICUPATH

This parameter specifies the path to ICU shared libraries icudata and icuuc.

Range: 0 to 256 characters.

Default: $XCOM_HOME/bin

XCOM_JVM

The XCOM_JVM parameter specifies the full path and file name for the Java JVM shared library (`libjvm.so`).

Range: 1 to 256 characters

Default: The default value depends on the platform:

- AIX 64 -- `XCOM_JVM=/opt/CA/XCOM/JRE/1.8.0_SR3_64Bit/lib/ppc64/j9vm/libjvm.so`
- HP-UX IA64 -- `XCOM_JVM=/opt/CA/SharedComponents/JRE/1.6.0_02_ALL/lib/IA64W/server/libjvm.so`
- Linux x64 -- `XCOM_JVM=/opt/CA/XCOM/JRE/1.8.0_77/lib/amd64/server/libjvm.so`
- Solaris Sparc 64 -- `XCOM_JVM=/opt/CA/XCOM/JRE/1.8.0_131_64Bit/lib/sparcv9/server/libjvm.so`
- Solaris x86 64 -- `XCOM_JVM=/opt/CA/XCOM/JRE/1.8.0_131_64Bit/lib/amd64/server/libjvm.so`

XCOM_MYSQLODBC

This parameter specifies the MySQL ODBC driver. It is used for HP only.

Default: /usr/lib/libmyodbc5.sl

XCOM_ODBC

This parameter specifies the full path to the ODBC library (libodbc.so) on Unix.

Range: 0 to 256 characters

Default: None

XCOM_PASSWORD

This parameter specifies the default password for remotely initiated transfers.

Range: 0 to 31 characters

Default: None

XCOM_SHLIBPATH

This parameter specifies the path to ODBC driver to be added to SHLIB_PATH. It is used for HP only.

Default: None

XCOM_SHOW_CIPHER

Specifies whether to display encryption algorithms in the XCOM Data Transport queue detailed information, which is used for transfers.

NO

Do not display encryption algorithms in the queue detail information.

YES

Display encryption algorithms in the queue detail information.

Default: NO

XCOM_TRUSTED_OVR

Specifies if the user is permitted to override the user ID by using the USERID parameter for locally initiated trusted transfers. If XCOM_TRUSTED_OVR is set to YES, the user is permitted to override the user ID by using the USERID parameter. If it is set to NO, the user ID of the process that initiated the transfer is used.

Range: YES, NO, Y, N

Default: NO

XCOM_USERID

This is the default user ID for remotely initiated transfers.

Range: 0 to 12 characters

Default: xcom

XCOMDB_SQLCONNECT_TIMEOUT

This parameter is used to set the database connection time-out; and is useful when the database server is slow. Time is provided in seconds.

Range: 0 to 300 seconds

Default: 10 seconds

XCOMFULLSSL

This *API only* parameter 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 a TLS or an SSL listener on the remote partner.

NO

Performs a non-secure transfer. The transfer uses a non-OpenSSL socket.

Default: NO

XCOMHIST

The ODBC Data Source Name that is used to connect to the History Database.

Range: 1 to 32 characters

Default: None

XCOMHIST_BACKSLASH

Treat a backslash in a file name as a single backslash. This is dependent on the target ODBC system.

Y

Treat a backslash in a file name as a single backslash (\).

N

Treat a backslash in a file name as a double backslash (\\).

Examples

If the ODBC is z/OS or DB2, a single \ is needed in order to display the data correctly.

If the ODBC is PC based, like Mysql, for example, then \\ is treated as a single \.

Default: None

XCOMHIST_FILE

If the database machine is not connected and available, the name of the flat file to contain insert records. Records are also written into this file as a result of an SQL failure.

Range: 0 to 256 characters

Default: $XCOM_HOME/config/xcomhist.inserts

NOTE
$XCOM_HOME is an environment variable.

XCOMPRE_LOCAL

Indicates whether the xcompre script file for a locally initiated transfer should be run.

Range: YES, NO

Default: NO

XCOMHIST_OWNER

Specifies the ID of the creator of the History Table. May be omitted if it is the same as XCOMHIST_USER.

Range: 0 to 32 characters

Default: None

XCOMHIST_PASSWORD

The password of XCOMHIST_USER.

Range: 1 to 32 characters

Default: None

XCOMHIST_SPLIT_FILE

If specified as Y, then if an insert fails, the query will be written out as 72-byte records for z/OS SPUFI compatibility.

Y

Split the file into 72-byte records.

N

Do not split the file.

Default: N

XCOMHIST_TBL

The name of the table created for the XCOM Data Transport history records.

Range: 1 to 18 characters

Default: xcom_history_tbl

XCOMHIST_USER

Specifies the user ID that has complete access to the XCOMHIST database.

Range: 1 to 32 characters

Default: None

XENDCMD

The name of the post-processing command optionally invoked by the XCOM Data Transport transfer program after any type of transfer is finished, whether successful or not. Invoked after partner communications have ended.

Range: 0 to 256 characters

Default: $XCOM_HOME/cmd/xcomend

NOTE
$XCOM_HOME is an environment variable.

XIDEST

Specifies the name of the remote system on the intermediate destination that is designated for store-and-forward transfers. If this variable is null or unset, then a direct connection to a remote system is attempted.

NOTE
For store-and-forward transfers only.

Range: 0 to 14 characters

Default: None

XLOGFILE

The name of the file where XCOM Data Transport logs activity. If you do not specify this parameter, the system-wide log file $XCOM_HOME/xcom.log is used. If you specify this parameter with a different file name, the logging information is only sent to the specified file.

NOTE

If QUEUE=YES, specify the full path n*a*me.

- $XCOM_HOME is an environment variable.

Range: 0 to 256 characters

Default: $XCOM_HOME/xcom.log

XLPCMD

The name of the post-processing command file used to send print jobs to the spooler. For incoming reports only.

Range: 0 to 256 characters

Default: $XCOM_HOME/cmd/xcomlp

> **NOTE**
>
> $XCOM_HOME is an environment variable.

XLUNAME

Indicates the name of the local LU. This name must match the remote LU name on the remote system.

Range: 1 to 17 characters

Default: None

> **NOTE**
> Not used in XCOM Data Transport for RS/6000 SNA Services or RS/6000 SNA Server.

XMODE

Indicates the mode name associated with the SNA/APPC configuration for the XLUNAME-Remote LU name pair. This name must match the mode name defined on the remote system.

Range: One to eight characters

Default: XCOMMODE

> **NOTE**
> Not used in XCOM Data Transport for RS/6000 SNA Services or RS/6000 SNA Server.

XNODESPEC

Required for Brixton and SunLink APPCs. Indicates the name of the node that specifies the gateway to use for the transfer. Does not apply to other APPCs.

Range: 1 to 64 characters

Default: None

XNOTIFYCMD

The name of the command file that XCOM Data Transport will use to notify users on the local system of the completion of a transfer. This is normally a shell script that composes a message and invokes mail or write as appropriate.

Range: 0 to 256 characters

Default: $XCOM_HOME/cmd/xcomntfy

XPPCMD

The name of the command file used for user-defined post processing, for file transfers only. Only used when the local system is receiving the file.

Range: 0 to 64 characters

Default: $XCOM_HOME/cmd/xcompp

NOTE
$XCOM_HOME is an environment variable.

XPRECMD

The name of the command file used for user-defined pre-allocation processing for locally and remotely initiated transfers. Specify this parameter in xcom.glb to invoke the pre-allocation exit xcompre. A sample command file is provided in $XCOM_HOME/cmd/xcompre.bat.

Range: 0 to 64 characters

Default: $XCOM_HOME/cmd/xcompre.bat

NOTE
$XCOM_HOME is an environment variable.

XTCERRDECR

Specifies the transfer requests for which the HOLDCOUNT parameter value is decremented when the current transfer completes unsuccessfully.

Up to eight transfer request names
Specifies up to eight transfers whose hold counts are to be decremented if a transfer completes unsuccessfully.

VTAM errors are not considered failed file transfers, because they are retried later.

- If the current file transfer fails, the value of the HOLDCOUNT parameter of the transfers assigned to this parameter is reduced by one, unless the next transfer has already started or the HOLDCOUNT value has already reached zero.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCERRINCR

Specifies the transfer requests for which the HOLDCOUNT parameter value is incremented if the current file transfer fails.

Up to eight transfer request names
Specifies up to eight transfers whose hold counts are to be incremented if the current file transfer fails.

NOTE

- A VTAM communications error does not count as a failure.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCERRPURGE

Specifies the transfer requests to be purged if the transfer concludes unsuccessfully.

Up to eight transfer request names

Specifies up to eight transfers to be purged if the current file transfer concludes unsuccessfully.

NOTE

You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCERRREL

Specifies the transfers to be released if the current transfer completes unsuccessfully.

Up to eight transfer request names

Specifies up to eight transfers to be released if the current transfer completes unsuccessfully.

NOTE

You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCGOODDECR

Indicates an XTCNET job whose hold count decrements if the file transfer completes successfully.

Up to eight transfer request names

Specifies up to eight transfers whose hold counts are to be decremented if the current transfer completes successfully.

NOTE

- If the current file transfer completes successfully, the value of the HOLDCOUNT parameter of the transfers assigned to this parameter is reduced by one, unless the next transfer has already started or the HOLDCOUNT value has already reached zero.
- You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCGOODINCR

Indicates the transfer requests whose HOLDCOUNT parameter is incremented when the current file transfer completes successfully.

Up to eight transfer request names

Specifies up to eight transfers whose hold counts are to be incremented if a transfer completes successfully.

NOTE

You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCGOODPURGE

Specifies the transfer requests to be purged when the current file transfer completes successfully.

Up to eight transfer request names

Specifies up to eight jobs to be purged if a transfer completes successfully.

NOTE

You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCGOODREL

Specifies the transfer requests to be released if the current file transfer concludes successfully.

Up to eight transfer request names

Specifies up to eight jobs to be released if a transfer completes successfully.

NOTE

You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCHOLD_COUNT

This *API only* parameter associates a number with a transfer request that is incremented or decremented by the successful or unsuccessful completion of other transfer requests. As long as the number is greater than 0, the transfer is not released.

0 to 255

Specifies a value that controls the holding/releasing of a transfer request. The transfer is released when the value of the parameter reaches 0.

NOTE

See the description of the parameters xtcerrdecr, xtcerrincr, xtcgooddecr, and xtcgoodincr, which can decrement and increment the value of the xtchold_count parameter.

- hold_transfer=YES overrides xtchold_count.

XTCJOB

Defines the name of a transfer request belonging to the group of interdependent transfer requests named through the XTCNET parameter.

xxxxxxxx

Specifies the name of a transfer request in a group of interrelated transfer requests.

NOTE

You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTCNET

Defines the name of a group of interdependent transfer requests.

xxxxxxxx

Specifies up to eight alphanumeric characters representing the name of the XTC network running this transfer.

NOTE

You can use symbolic variables with this parameter. For more information, see Symbolic Parameters in Configuration Files.

XTRACE

Indicates the level of desired execution tracing.

0

No tracing.

1 to 8

Levels of tracing from 1 (lowest) to 8 (highest).

9

Show the contents of the data buffers.

10

Includes levels 1 to 9 with additional detailed technical information.

Default: 0

NOTE
Turning on the trace can seriously degrade performance.

Character Sets for Unicode Transfer

This section lists the character sets supported by XCOM Data Transport for Unicode Transfers.

IANA is the main source of converter aliases on the Internet. Since IANA does not specify the Unicode mappings for every codepage and alias, and every platform supports other aliases besides the IANA aliases, ICU (International Components for Unicode) is used by XCOM Data Transport to provide a way to target the codepage conversion based upon the standard or platform. This allows you to use the right converter name and implementation based upon which standard you are targeting.

The column marked as ‘Canonical Name’ is a Unique ICU converter name. The column marked as ‘IANA’ specifies the code page name as per IANA registry. The column marked as ‘All Aliases’ has all the aliases by which specified code page is known regardless of which standards support the converter alias name.

Character sets are categorized according to converter type. Character sets are separated with the space.

Unicode

Canonical Name	IANA	All Aliases
UTF-8	UTF-8	UTF-8 ibm-1208 ibm-1209 ibm-5304 ibm-5305 ibm-13496 ibm-13497 ibm-17592 ibm-17593 windows-65001 cp1208 x-UTF_8J
UTF-16	UTF-16 ISO-10646-UCS-2	UTF-16 ISO-10646-UCS-2 ibm-1204 ibm-1205 Unicode csUnicode ucs-2
UTF-16BE	UTF-16BE	UTF-16BE x-utf-16be Unicode Big Unmarked ibm-1200 ibm-1201 ibm-13488 ibm-13489 ibm-17584 ibm-17585 ibm-21680 ibm-21681 ibm-25776 ibm-25777 ibm-29872 ibm-29873 ibm-61955 ibm-61956 windows-1201 cp1200 cp1201 UTF16_BigEndian

UTF-16LE	UTF-16LE	UTF-16LE x-utf-16le Unicode Little Unmarked ibm-1202 ibm-1203 ibm-13490 ibm-13491 ibm-17586 ibm-17587 ibm-21682 ibm-21683 ibm-25778 ibm-25779 ibm-29874 ibm-29875 UTF16_LittleEndian windows-1200

Unicode (64-bit)

The following table lists the Unicode charsets that are supported by XCOM Data Transport for Unicode transfers on Linux, AIX, Solaris Sparc, Solaris x86, and Linux s390x 64-bit platforms.

Canonical Name	IANA	All Aliases
UTF-8	UTF-8	UTF-8 ibm-1208 ibm-1209 ibm-5304 ibm-5305 ibm-13496 ibm-13497 ibm-17592 ibm-17593 windows-65001 cp1208 x-UTF_8J unicode-1-1-utf-8 unicode-2-0-utf-8
UTF-16	UTF-16 ISO-10646-UCS-2	UTF-16 ISO-10646-UCS-2 ibm-1204 ibm-1205 unicode csUnicode ucs-2

UTF-16BE	UTF-16BE	UTF-16BE x-utf-16be UnicodeBigUnmarked ibm-1200 ibm-1201 ibm-13488 ibm-13489 ibm-17584 ibm-17585 ibm-21680 ibm-21681 ibm-25776 ibm-25777 ibm-29872 ibm-29873 ibm-61955 ibm-61956 windows-1201 cp1200 cp1201 UTF16_BigEndian
UTF-16LE	UTF-16LE	UTF-16LE x-utf-16le UnicodeLittleUnmarked ibm-1202 ibm-1203 ibm-13490 ibm-13491 ibm-17586 ibm-17587 ibm-21682 ibm-21683 ibm-25778 ibm-25779 ibm-29874 ibm-29875 UTF16_LittleEndian windows-1200

LATIN_1

Canonical Name	IANA	All Aliases
ISO-8859-1	ISO_8859-1:1987 ISO-8859-1 IBM819 cp819 latin1 csISOLatin1 iso-ir-100 l1	ISO-8859-1 ibm-819 IBM819 cp819 latin1 8859_1 csISOLatin1 iso-ir-100 ISO_8859-1:1987 l1 819

US_ASCII

Canonical Name	IANA	All Aliases
US-ASCII	ANSI_X3.4-1968 US-ASCII ASCII ANSI_X3.4-1986 ISO_646.irv:1991 ISO646-US us csASCII iso-ir-6 cp367 IBM367	US-ASCII ASCII ANSI_X3.4-1968 ANSI_X3.4-1986 ISO_646.irv:1991 iso_646.irv:1983 ISO646-US us csASCII iso-ir-6 cp367 ascii7 646 windows-20127 ibm-367 IBM367

Multibyte character sets (MBCS)

Canonical Name	IANA	All Aliases
gb18030	gb18030	gb18030 ibm-1392 windows-54936 GB18030
ibm-942_P12A-1999		ibm-942_P12A-1999 ibm-942 ibm-932 cp932 shift_jis78 sjis78 ibm-942_VSUB_VPUA ibm-932_VSUB_VPUA x-IBM942 x-IBM942C
ibm-943_P15A-2003	Shift_JIS MS_Kanji csShiftJIS windows-31j csWindows31J	ibm-943_P15A-2003 ibm-943 Shift_JIS MS_Kanji csShiftJIS windows-31j csWindows31J x-sjis x-ms-cp932 cp932 windows-932 cp943c IBM-943C ms932 pck sjis ibm-943_VSUB_VPUA x-MS932_0213 x-JISAutoDetect
ibm-943_P130-1999		ibm-943_P130-1999 ibm-943 Shift_JIS cp943 943 ibm-943_VASCII_VSUB_VPUA x-IBM943

ibm-33722_P12A_P12A-2004_U2	Extended_UNIX_Code_Packed_Format_for_Japanese EUC-JP csEUCPkdFmtJapanese	ibm-33722_P12A_P12A-2004_U2 ibm-33722 ibm-5050 EUC-JP Extended_UNIX_Code_Packed_Format_for_Japanes csEUCPkdFmtJapanese X-EUC-JP windows-51932 ibm-33722_VPUA IBM-eucJP
ibm-33722_P120-1999		ibm-33722_P120-1999 ibm-33722 ibm-5050 cp33722 33722 ibm-33722_VASCII_VPUA x-IBM33722 x-IBM33722A x-IBM33722C
ibm-954_P101-2007		ibm-954_P101-2007 ibm-954 EUC-JP Extended_UNIX_Code_Packed_Format_for_Japanes csEUCPkdFmtJapanese X-EUC-JP eucjis ujis x-IBM954 x-IBM954C
ibm-1373_P100-2002		ibm-1373_P100-2002 ibm-1373 windows-950
windows-950-2000	Big5 csBig5	windows-950-2000 Big5 csBig5 windows-950 x-windows-950 x-big5
ibm-950_P110-1999		ibm-950_P110-1999 ibm-950 cp950 950 x-IBM950
ibm-1375_P100-2007	Big5-HKSCS	ibm-1375_P100-2007 ibm-1375 Big5-HKSCS big5hk HKSCS-BIG5
ibm-5471_P100-2006		ibm-5471_P100-2006 ibm-5471 Big5-HKSCS MS950_HKSCS hkbig5 big5-hkscs:unicode3.0 x-MS950-HKSCS
ibm-1386_P100-2001		ibm-1386_P100-2001 ibm-1386 cp1386 windows-936 ibm-1386_VSUB_VPUA
windows-936-2000	GBK CP936 MS936 windows-936	windows-936-2000 GBK CP936 MS936 windows-936
ibm-1383_P110-1999	GB2312 csGB2312	ibm-1383_P110-1999 ibm-1383 GB2312 csGB2312 cp1383 1383 EUC-CN ibm-eucCN hp15CN ibm-1383_VPUA
ibm-964_P110-1999		ibm-964_P110-1999 ibm-964 EUC-TW ibm-eucTW cns11643 cp964 964 ibm-964_VPUA x-IBM964
ibm-949_P110-1999		ibm-949_P110-1999 ibm-949 cp949 949 ibm-949_VASCII_VSUB_VPUA x-IBM949
ibm-949_P11A-1999		ibm-949_P11A-1999 ibm-949 cp949c ibm-949_VSUB_VPUA x-IBM949C IBM-949C
ibm-970_P110_P110-2006_U2	EUC-KR csEUCKR	ibm-970_P110_P110-2006_U2 ibm-970 EUC-KR KS_C_5601-1987 windows-51949 csEUCKR ibm-eucKR KSC_5601 5601 cp970 970 ibm-970_VPUA x-IBM970

ibm-1363_P11B-1998	KS_C_5601-1987 KS_C_5601-1989 KSC_5601 csKSC56011987 korean iso-ir-149	ibm-1363_P11B-1998 ibm-1363 KS_C_5601-1987 KS_C_5601-1989 KSC_5601 csKSC56011987 korean iso-ir-149 cp1363 5601 ksc windows-949 ibm-1363_VSUB_VPUA x-IBM1363C
ibm-1363_P110-1997		ibm-1363_P110-1997 ibm-1363 ibm-1363_VASCII_VSUB_VPUA x-IBM1363
windows-949-2000		windows-949-2000 windows-949 KS_C_5601-1987 KS_C_5601-1989 KSC_5601 csKSC56011987 korean iso-ir-149 ms949 x-KSC5601

Multibyte character sets (MBCS) 64-bit

The following table lists the MBCS charsets that are supported by XCOM Data Transport for Unicode transfers on Linux, AIX, Solaris Sparc, Solaris x86, and Linux s390x 64-bit platforms.

Canonical Name	IANA	All Aliases
gb18030	gb18030	gb18030 ibm-1392 windows-54936 GB18030
ibm-942_P12A-1999		ibm-942_P12A-1999 ibm-942 ibm-932 cp932 shift_jis78 sjis78 ibm-942_VSUB_VPUA ibm-932_VSUB_VPUA x-IBM942 x-IBM942C

ibm-943_P15A-2003	Shift_JIS MS_Kanji csShiftJIS windows-31j csWindows31J	ibm-943_P15A-2003 ibm-943 Shift_JIS MS_Kanji csShiftJIS windows-31j csWindows31J x-sjis x-ms-cp932 cp932 windows-932 cp943c IBM-943C ms932 pck sjis ibm-943_VSUB_VPUA x-MS932_0213 x-JISAutoDetect
ibm-943_P130-1999		ibm-943_P130-1999 ibm-943| Shift_JIS cp943 943 ibm-943_VASCII_VSUB_VPUA x-IBM943
ibm-33722_P12A_P12A-2009_U2		ibm-33722_P12A_P12A-2009_U2 ibm-33722 ibm-5050 ibm-33722_VPUA IBM-eucJP
ibm-33722_P120-1999		ibm-33722_P120-1999 ibm-33722 ibm-5050 cp33722 33722 ibm-33722_VASCII_VPUA x-IBM33722 x-IBM33722A x-IBM33722C
ibm-954_P101-2007		ibm-954_P101-2007 ibm-954 x-IBM954 x-IBM954C

euc-jp-2007	*Extended* *_UNIX* *_Code* *_Packed* *_Format* *_for* *_Japanese* EUC-JP csEUCPkdFmtJapanese	euc-jp-2007 EUC-JP *Extended* *_UNIX* *_Code* *_Packed* *_Format* *_for* *_Japanese* csEUCPkdFmtJapanese X-EUC-JP eucjis ujis
ibm-1373_P100-2002		ibm-1373_P100-2002 ibm-1373 windows-950
windows-950-2000	Big5 csBig5	windows-950-2000 Big5 csBig5 windows-950 x-windows-950 x-big5
ibm-950_P110-1999		ibm-950_P110-1999 ibm-950 cp950 950 x-IBM950
ibm-1375_P100-2007	Big5-HKSCS	ibm-1375_P100-2007 ibm-1375 Big5-HKSCS big5hk HKSCS-BIG5
ibm-5471_P100-2006		ibm-5471_P100-2006 ibm-5471 Big5-HKSCS MS950_HKSCS hkbig5 big5-hkscs:unicode3.0 x-MS950-HKSCS
ibm-1386_P100-2001		ibm-1386_P100-2001 ibm-1386 cp1386 windows-936 ibm-1386_VSUB_VPUA

windows-936-2000	GBK CP936 MS936 windows-936	windows-936-2000 GBK CP936 MS936 windows-936
ibm-1383_P110-1999	GB2312 csGB2312	ibm-1383_P110-1999 ibm-1383 GB2312 csGB2312 cp1383 1383 EUC-CN ibm-eucCN hp15CN ibm-1383_VPUA
ibm-964_P110-1999		ibm-964_P110-1999 ibm-964 EUC-TW ibm-eucTW cns11643 cp964 964 ibm-964_VPUA x-IBM964
ibm-949_P110-1999		ibm-949_P110-1999 ibm-949 cp949 949 ibm-949_VASCII_VSUB_VPUA x-IBM949
ibm-949_P11A-1999		ibm-949_P11A-1999 ibm-949 cp949c ibm-949_VSUB_VPUA x-IBM949C IBM-949C

ibm-970_P110_P110-2006_U2	EUC-KR csEUCKR	ibm-970_P110_P110-2006_U2 ibm-970 EUC-KR KS_C_5601-1987 windows-51949 csEUCKR ibm-eucKR KSC_5601 5601 cp970 970 ibm-970_VPUA x-IBM970
ibm-1363_P11B-1998	KS_C_5601-1987 KS_C_5601-1989 KSC_5601 csKSC56011987 korean iso-ir-149	ibm-1363_P11B-1998 ibm-1363 KS_C_5601-1987 KS_C_5601-1989 KSC_5601 csKSC56011987 korean iso-ir-149 cp1363 5601 ksc windows-949 ibm-1363_VSUB_VPUA x-IBM1363C
ibm-1363_P110-1997		ibm-1363_P110-1997 ibm-1363 ibm-1363_VASCII_VSUB_VPUA x-IBM1363
windows-949-2000		windows-949-2000 windows-949 KS_C_5601-1987 KS_C_5601-1989 KSC_5601 csKSC56011987 korean iso-ir-149 ms949 x-KSC5601

Single Byte Character Sets (SBCS)

Canonical Name	IANA	All Aliases
ibm-912_P100-1995	ISO_8859-2:1987 ISO-8859-2 latin2 csISOLatin2 iso-ir-101 l2	ibm-912_P100-1995 ibm-912 ISO-8859-2 ISO_8859-2:1987 latin2 csISOLatin2 iso-ir-101 l2 8859_2 cp912 912 windows-28592
ibm-913_P100-2000	ISO_8859-3:1988 ISO-8859-3 latin3 csISOLatin3 iso-ir-109 l3	ibm-913_P100-2000 ibm-913 ISO-8859-3 ISO_8859-3:1988 latin3 csISOLatin3 iso-ir-109 l3 8859_3 cp913 913 windows-28593
ibm-914_P100-1995	ISO_8859-4:1988 ISO-8859-4 latin4 csISOLatin4 iso-ir-110 l4	ibm-914_P100-1995 ibm-914 ISO-8859-4 latin4 csISOLatin4 iso-ir-110 ISO_8859-4:1988 l4 8859_4 cp914 914 windows-28594
ibm-915_P100-1995	ISO_8859-5:1988 ISO-8859-5 cyrillic csISOLatinCyrillic iso-ir-144	ibm-915_P100-1995 ibm-915 ISO-8859-5 cyrillic csISOLatinCyrillic iso-ir-144 ISO_8859-5:1988 8859_5 cp915 915 windows-28595
ibm-1089_P100-1995	ISO_8859-6:1987 ISO-8859-6 arabic csISOLatinArabic iso-ir-127 ECMA-114 ASMO-708 ISO-8859-6-I ISO-8859-6-E	ibm-1089_P100-1995 ibm-1089 ISO-8859-6 arabic csISOLatinArabic iso-ir-127 ISO_8859-6:1987 ECMA-114 ASMO-708 8859_6 cp1089 1089 windows-28596 ISO-8859-6-I ISO-8859-6-E x-ISO-8859-6S
ibm-9005_X110-2007	ISO_8859-7:1987 ISO-8859-7 greek greek8 ELOT_928 ECMA-118 csISOLatinGreek iso-ir-126	ibm-9005_X110-2007 ibm-9005 ISO-8859-7 greek greek8 ELOT_928 ECMA-118 csISOLatinGreek iso-ir-126 ISO_8859-7:1987 windows-28597 sun_eu_greek
ibm-813_P100-1995		ibm-813_P100-1995 ibm-813 ISO-8859-7 greek greek8 ELOT_928 ECMA-118 csISOLatinGreek iso-ir-126 ISO_8859-7:1987 8859_7 cp813 813

ibm-5012_P100-1999	ISO_8859-8:1988 ISO-8859-8 hebrew csISOLatinHebrew iso-ir-138 ISO-8859-8-I ISO-8859-8-E	ibm-5012_P100-1999 ibm-5012 ISO-8859-8 hebrew csISOLatinHebrew iso-ir-138 ISO_8859-8:1988 ISO-8859-8-I ISO-8859-8-E 8859_8 windows-28598 hebrew8
ibm-916_P100-1995		ibm-916_P100-1995 ibm-916 cp916 916
ibm-920_P100-1995	ISO_8859-9:1989 ISO-8859-9 latin5 csISOLatin5 iso-ir-148 l5	ibm-920_P100-1995 ibm-920 ISO-8859-9 latin5 csISOLatin5 iso-ir-148 ISO_8859-9:1989 l5 8859_9 cp920 920 windows-28599 ECMA-128 turkish8 turkish
iso-8859_10-1998	ISO-8859-10 iso-ir-157 l6 ISO_8859-10:1992 csISOLatin6 latin6	iso-8859_10-1998 ISO-8859-10 iso-ir-157 l6 ISO_8859-10:1992 csISOLatin6 latin6
iso-8859_11-2001		iso-8859_11-2001 ISO-8859-11 thai8 x-iso-8859-11
ibm-921_P100-1995	ISO-8859-13	ibm-921_P100-1995 ibm-921 ISO-8859-13 8859_13 windows-28603 cp921 921 x-IBM921
iso-8859_14-1998	ISO-8859-14 iso-ir-199 ISO_8859-14:1998 latin8 iso-celtic l8	iso-8859_14-1998 ISO-8859-14 iso-ir-199 ISO_8859-14:1998 latin8 iso-celtic l8
ibm-923_P100-1998	ISO-8859-15 Latin-9	ibm-923_P100-1998 ibm-923 ISO-8859-15 Latin-9 l9 8859_15 latin0 csisolatin0 csisolatin9 iso8859_15_fdis cp923 923 windows-28605
windows-874-2000		windows-874-2000 TIS-620 windows-874 MS874 x-windows-874
ibm-874_P100-1995	TIS-620	ibm-874_P100-1995 ibm-874 ibm-9066 cp874 TIS-620 tis620.2533 eucTH x-IBM874
ibm-1162_P100-1999		ibm-1162_P100-1999 ibm-1162
ibm-437_P100-1995	IBM437 cp437 437 csPC8CodePage437	ibm-437_P100-1995 ibm-437 IBM437 cp437 437 csPC8CodePage437 windows-437
ibm-720_P100-1997		ibm-720_P100-1997 ibm-720 windows-720 DOS-720 x-IBM720

ibm-737_P100-1997		ibm-737_P100-1997 ibm-737 IBM737 cp737 windows-737 737 x-IBM737
ibm-775_P100-1996	IBM775 cp775 csPC775Baltic	ibm-775_P100-1996 ibm-775 IBM775 cp775 csPC775Baltic windows-775 775
ibm-850_P100-1995	IBM850 cp850 850 csPC850Multilingual	ibm-850_P100-1995 ibm-850 IBM850 cp850 850 csPC850Multilingual windows-850
ibm-851_P100-1995	IBM851 cp851 851 csPC851	ibm-851_P100-1995 ibm-851 IBM851 cp851 851 csPC851
ibm-852_P100-1995	IBM852 cp852 852 csPCp852	ibm-852_P100-1995 ibm-852 IBM852 cp852 852 csPCp852 windows-852
ibm-855_P100-1995	IBM855 cp855 855 csIBM855	ibm-855_P100-1995 ibm-855 IBM855 cp855 855 csIBM855 csPCp855 windows-855
ibm-856_P100-1995		ibm-856_P100-1995 ibm-856 IBM856 cp856 856 x-IBM856
ibm-857_P100-1995	IBM857 cp857 857 csIBM857	ibm-857_P100-1995 ibm-857 IBM857 cp857 857 csIBM857 windows-857
ibm-858_P100-1997	IBM00858 CCSID00858 CP00858 PC-Multilingual-850+euro	ibm-858_P100-1997 ibm-858 IBM00858 CCSID00858 CP00858 PC-Multilingual-850+euro cp858 windows-858
ibm-860_P100-1995	IBM860 cp860 860 csIBM860	ibm-860_P100-1995 ibm-860 IBM860 cp860 860 csIBM860
ibm-861_P100-1995	IBM861 cp861 861 cp-is csIBM861	ibm-861_P100-1995 ibm-861 IBM861 cp861 861 cp-is csIBM861 windows-861
ibm-862_P100-1995	IBM862 cp862 862 csPC862LatinHebrew	ibm-862_P100-1995 ibm-862 IBM862 cp862 862 csPC862LatinHebrew DOS-862 windows-862

ibm-863_P100-1995	IBM863 cp863 863 csIBM863	ibm-863_P100-1995 ibm-863 IBM863 cp863 863 csIBM863
ibm-864_X110-1999	IBM864 cp864 csIBM864	ibm-864_X110-1999 ibm-864 IBM864 cp864 csIBM864
ibm-865_P100-1995	IBM865 cp865 865 csIBM865	ibm-865_P100-1995 ibm-865 IBM865 cp865 865 csIBM865
ibm-866_P100-1995	IBM866 cp866 866 csIBM866	ibm-866_P100-1995 ibm-866 IBM866 cp866 866 csIBM866 windows-866
ibm-867_P100-1998		ibm-867_P100-1998 ibm-867 x-IBM867
ibm-868_P100-1995	IBM868 CP868 csIBM868 cp-ar	ibm-868_P100-1995 ibm-868 IBM868 CP868 868 csIBM868 cp-ar
ibm-869_P100-1995	IBM869 cp869 869 cp-gr csIBM869	ibm-869_P100-1995 ibm-869 IBM869 cp869 869 cp-gr csIBM869 windows-869
ibm-878_P100-1996	KOI8-R csKOI8R	ibm-878_P100-1996 ibm-878 KOI8-R koi8 csKOI8R windows-20866 cp878
ibm-901_P100-1999		ibm-901_P100-1999 ibm-901
ibm-902_P100-1999		ibm-902_P100-1999 ibm-902
ibm-922_P100-1999		ibm-922_P100-1999 ibm-922 IBM922 cp922 922 x-IBM922
ibm-1168_P100-2002	KOI8-U	ibm-1168_P100-2002 ibm-1168 KOI8-U windows-21866
ibm-4909_P100-1999		ibm-4909_P100-1999 ibm-4909
ibm-5346_P100-1998	windows-1250	ibm-5346_P100-1998 ibm-5346 windows-1250 cp1250
ibm-5347_P100-1998	windows-1251	ibm-5347_P100-1998 ibm-5347 windows-1251 cp1251 ANSI1251
ibm-5348_P100-1997	windows-1252	ibm-5348_P100-1997 ibm-5348 windows-1252 cp1252
ibm-5349_P100-1998	windows-1253	ibm-5349_P100-1998 ibm-5349 windows-1253 cp1253
ibm-5350_P100-1998	windows-1254	ibm-5350_P100-1998 ibm-5350 windows-1254 cp1254
ibm-9447_P100-2002	windows-1255	ibm-9447_P100-2002 ibm-9447 windows-1255 cp1255

ibm-9448_X100-2005	windows-1256	ibm-9448_X100-2005 ibm-9448 windows-1256 cp1256 x-windows-1256S
ibm-9449_P100-2002	windows-1257	ibm-9449_P100-2002 ibm-9449 windows-1257 cp1257
ibm-5354_P100-1998	windows-1258	ibm-5354_P100-1998 ibm-5354 windows-1258 cp1258
ibm-1250_P100-1995		ibm-1250_P100-1995 ibm-1250 windows-1250
ibm-1251_P100-1995		ibm-1251_P100-1995 ibm-1251 windows-1251
ibm-1252_P100-2000		ibm-1252_P100-2000 ibm-1252 windows-1252
ibm-1253_P100-1995		ibm-1253_P100-1995 ibm-1253 windows-1253
ibm-1254_P100-1995		ibm-1254_P100-1995 ibm-1254 windows-1254
ibm-1255_P100-1995		ibm-1255_P100-1995 ibm-1255
ibm-5351_P100-1998		ibm-5351_P100-1998 ibm-5351 windows-1255
ibm-1256_P110-1997		ibm-1256_P110-1997 ibm-1256
ibm-5352_P100-1998		ibm-5352_P100-1998 ibm-5352 windows-1256
ibm-1257_P100-1995		ibm-1257_P100-1995 ibm-1257
ibm-5353_P100-1998		ibm-5353_P100-1998 ibm-5353 windows-1257
ibm-1258_P100-1997		ibm-1258_P100-1997 ibm-1258 windows-1258
macos-0_2-10.2	macintosh mac csMacintosh	macos-0_2-10.2 macintosh mac csMacintosh windows-10000 macroman x-macroman
macos-6_2-10.4		macos-6_2-10.4 x-mac-greek windows-10006 macgr x-MacGreek
macos-7_3-10.2		macos-7_3-10.2 x-mac-cyrillic windows-10007 mac-cyrillic maccy x-MacCyrillic x-MacUkraine
macos-29-10.2		macos-29-10.2 x-mac-centraleurroman windows-10029 x-mac-ce macce maccentraleurope x-MacCentralEurope
macos-35-10.2		macos-35-10.2 x-mac-turkish windows-10081 mactr x-MacTurkish
ibm-1051_P100-1995	hp-roman8 roman8 r8 csHPRoman8	ibm-1051_P100-1995 ibm-1051 hp-roman8 roman8 r8 csHPRoman8
ibm-1276_P100-1995	Adobe-Standard-Encoding csAdobeStandardEncoding	ibm-1276_P100-1995 ibm-1276 Adobe-Standard-Encoding csAdobeStandardEncoding

ibm-1006_P100-1995		ibm-1006_P100-1995 ibm-1006 IBM1006 cp1006 1006 x-IBM1006
ibm-1098_P100-1995		ibm-1098_P100-1995 ibm-1098 IBM1098 cp1098 1098 x-IBM1098
ibm-1124_P100-1996		ibm-1124_P100-1996 ibm-1124 cp1124 1124 x-IBM1124
ibm-1125_P100-1997		ibm-1125_P100-1997 ibm-1125 cp1125
ibm-1129_P100-1997		ibm-1129_P100-1997 ibm-1129
ibm-1131_P100-1997		ibm-1131_P100-1997 ibm-1131 cp1131
ibm-1133_P100-1997		ibm-1133_P100-1997 ibm-1133
ibm-37_P100-1995	IBM037 ebcdic-cp-us ebcdic-cp-ca ebcdic-cp-wt ebcdic-cp-nl csIBM037	ibm-37_P100-1995 ibm-37 IBM037 ibm-037 ebcdic-cp-us ebcdic-cp-ca ebcdic-cp-wt ebcdic-cp-nl csIBM037 cp037 037 cpibm37 cp37
ibm-273_P100-1995	IBM273 CP273 csIBM273	ibm-273_P100-1995 ibm-273 IBM273 CP273 csIBM273 ebcdic-de 273
ibm-277_P100-1995	IBM277 EBCDIC-CP-DK EBCDIC-CP-NO csIBM277	ibm-277_P100-1995 ibm-277 IBM277 cp277 EBCDIC-CP-DK EBCDIC-CP-NO csIBM277 ebcdic-dk 277
ibm-278_P100-1995	IBM278 ebcdic-cp-fi ebcdic-cp-se csIBM278	ibm-278_P100-1995 ibm-278 IBM278 cp278 ebcdic-cp-fi ebcdic-cp-se csIBM278 ebcdic-sv 278
ibm-280_P100-1995	IBM280 CP280 ebcdic-cp-it csIBM280	ibm-280_P100-1995 ibm-280 IBM280 CP280 ebcdic-cp-it csIBM280 280
ibm-284_P100-1995	IBM284 CP284 ebcdic-cp-es csIBM284	ibm-284_P100-1995 ibm-284 IBM284 CP284 ebcdic-cp-es csIBM284 cpibm284 284
ibm-285_P100-1995	IBM285 CP285 ebcdic-cp-gb csIBM285	ibm-285_P100-1995 ibm-285 IBM285 CP285 ebcdic-cp-gb csIBM285 cpibm285 ebcdic-gb 285
ibm-290_P100-1995	IBM290 cp290 EBCDIC-JP-kana csIBM290	ibm-290_P100-1995 ibm-290 IBM290 cp290 EBCDIC-JP-kana csIBM290
ibm-297_P100-1995	IBM297 cp297 ebcdic-cp-fr csIBM297	ibm-297_P100-1995 ibm-297 IBM297 cp297 ebcdic-cp-fr csIBM297 cpibm297 297

ibm-420_X120-1999	IBM420 cp420 ebcdic-cp-ar1 csIBM420	ibm-420_X120-1999 ibm-420 IBM420 cp420 ebcdic-cp-ar1 csIBM420 420
ibm-424_P100-1995	IBM424 cp424 ebcdic-cp-he csIBM424	ibm-424_P100-1995 ibm-424 IBM424 cp424 ebcdic-cp-he csIBM424 424
ibm-500_P100-1995	IBM500 CP500 ebcdic-cp-be csIBM500 ebcdic-cp-ch	ibm-500_P100-1995 ibm-500 IBM500 CP500 ebcdic-cp-be csIBM500 ebcdic-cp-ch 500
ibm-803_P100-1999		ibm-803_P100-1999 ibm-803 cp803
ibm-838_P100-1995	IBM-Thai csIBMThai	ibm-838_P100-1995 ibm-838 IBM838 IBM-Thai csIBMThai cp838 838 ibm-9030
ibm-870_P100-1995	IBM870 CP870 ebcdic-cp-roece ebcdic-cp-yu csIBM870	ibm-870_P100-1995 ibm-870 IBM870 CP870 ebcdic-cp-roece ebcdic-cp-yu csIBM870
ibm-871_P100-1995	IBM871 ebcdic-cp-is csIBM871 CP871	ibm-871_P100-1995 ibm-871 IBM871 ebcdic-cp-is csIBM871 CP871 ebcdic-is 871
ibm-875_P100-1995		ibm-875_P100-1995 ibm-875 IBM875 cp875 875 x-IBM875
ibm-918_P100-1995	IBM918 CP918 ebcdic-cp-ar2 csIBM918	ibm-918_P100-1995 ibm-918 IBM918 CP918 ebcdic-cp-ar2 csIBM918
ibm-1025_P100-1995		ibm-1025_P100-1995 ibm-1025 cp1025 1025 x-IBM1025
ibm-1026_P100-1995	IBM1026 CP1026 csIBM1026	ibm-1026_P100-1995 ibm-1026 IBM1026 CP1026 csIBM1026 1026
ibm-1047_P100-1995	IBM1047	ibm-1047_P100-1995 ibm-1047 IBM1047 cp1047 1047
ibm-1097_P100-1995		ibm-1097_P100-1995 ibm-1097 cp1097 1097 x-IBM1097
ibm-1112_P100-1995		ibm-1112_P100-1995 ibm-1112 cp1112 1112 x-IBM1112
ibm-1122_P100-1999		ibm-1122_P100-1999 ibm-1122 cp1122 1122 x-IBM1122
ibm-1123_P100-1995		ibm-1123_P100-1995 ibm-1123 cp1123 1123 x-IBM1123
ibm-1130_P100-1997		ibm-1130_P100-1997 ibm-1130

ibm-1132_P100-1998		ibm-1132_P100-1998 ibm-1132
ibm-1137_P100-1999		ibm-1137_P100-1999 ibm-1137
ibm-4517_P100-2005		ibm-4517_P100-2005 ibm-4517
ibm-1140_P100-1997	IBM01140 CCSID01140 CP01140 ebcdic-us-37+euro	ibm-1140_P100-1997 ibm-1140 IBM01140 CCSID01140 CP01140 cp1140 ebcdic-us-37+euro
ibm-1141_P100-1997	IBM01141 CCSID01141 CP01141 ebcdic-de-273+euro	ibm-1141_P100-1997 ibm-1141 IBM01141 CCSID01141 CP01141 cp1141 ebcdic-de-273+euro
ibm-1142_P100-1997	IBM01142 CCSID01142 CP01142 ebcdic-dk-277+euro ebcdic-no-277+euro	ibm-1142_P100-1997 ibm-1142 IBM01142 CCSID01142 CP01142 cp1142 ebcdic-dk-277+euro ebcdic-no-277+euro
ibm-1143_P100-1997	IBM01143 CCSID01143 CP01143 ebcdic-fi-278+euro ebcdic-se-278+euro	ibm-1143_P100-1997 ibm-1143 IBM01143 CCSID01143 CP01143 cp1143 ebcdic-fi-278+euro ebcdic-se-278+euro
ibm-1144_P100-1997	IBM01144 CCSID01144 CP01144 ebcdic-it-280+euro	ibm-1144_P100-1997 ibm-1144 IBM01144 CCSID01144 CP01144 cp1144 ebcdic-it-280+euro
ibm-1145_P100-1997	IBM01145 CCSID01145 CP01145 ebcdic-es-284+euro	ibm-1145_P100-1997 ibm-1145 IBM01145 CCSID01145 CP01145 cp1145 ebcdic-es-284+euro
ibm-1146_P100-1997	IBM01146 CCSID01146 CP01146 ebcdic-gb-285+euro	ibm-1146_P100-1997 ibm-1146 IBM01146 CCSID01146 CP01146 cp1146 ebcdic-gb-285+euro
ibm-1147_P100-1997	IBM01147 CCSID01147 CP01147 ebcdic-fr-297+euro	ibm-1147_P100-1997 ibm-1147 IBM01147 CCSID01147 CP01147 cp1147 ebcdic-fr-297+euro
ibm-1148_P100-1997	IBM01148 CCSID01148 CP01148 ebcdic-international-500+euro	ibm-1148_P100-1997 ibm-1148 IBM01148 CCSID01148 CP01148 cp1148 ebcdic-international-500+euro
ibm-1149_P100-1997	IBM01149 CCSID01149 CP01149 ebcdic-is-871+euro	ibm-1149_P100-1997 ibm-1149 IBM01149 CCSID01149 CP01149 cp1149 ebcdic-is-871+euro
ibm-1153_P100-1999		ibm-1153_P100-1999 ibm-1153 IBM1153 x-IBM1153

ibm-1154_P100-1999		ibm-1154_P100-1999 ibm-1154
ibm-1155_P100-1999		ibm-1155_P100-1999 ibm-1155
ibm-1156_P100-1999		ibm-1156_P100-1999 ibm-1156
ibm-1157_P100-1999		ibm-1157_P100-1999 ibm-1157
ibm-1158_P100-1999		ibm-1158_P100-1999 ibm-1158
ibm-1160_P100-1999		ibm-1160_P100-1999 ibm-1160
ibm-1164_P100-1999		ibm-1164_P100-1999 ibm-1164
ibm-5123_P100-1999		ibm-5123_P100-1999 ibm-5123
ibm-8482_P100-1999		ibm-8482_P100-1999 ibm-8482
ibm-4899_P100-1998		ibm-4899_P100-1998 ibm-4899
ibm-4971_P100-1999		ibm-4971_P100-1999 ibm-4971
ibm-9067_X100-2005		ibm-9067_X100-2005 ibm-9067
ibm-12712_P100-1998		ibm-12712_P100-1998 ibm-12712 ebcdic-he
ibm-16804_X110-1999		ibm-16804_X110-1999 ibm-16804 ebcdic-ar
ibm-37_P100-1995,swaplfnl		ibm-37_P100-1995,swaplfnl ibm-37-s390
ibm-1047_P100-1995,swaplfnl		ibm-1047_P100-1995,swaplfnl ibm-1047-s390 IBM1047_LF
ibm-1140_P100-1997,swaplfnl		ibm-1140_P100-1997,swaplfnl ibm-1140-s390
ibm-1141_P100-1997,swaplfnl		ibm-1141_P100-1997,swaplfnl ibm-1141-s390 IBM1141_LF
ibm-1142_P100-1997,swaplfnl		ibm-1142_P100-1997,swaplfnl ibm-1142-s390
ibm-1143_P100-1997,swaplfnl		ibm-1143_P100-1997,swaplfnl ibm-1143-s390
ibm-1144_P100-1997,swaplfnl		ibm-1144_P100-1997,swaplfnl ibm-1144-s390
ibm-1145_P100-1997,swaplfnl		ibm-1145_P100-1997,swaplfnl ibm-1145-s390
ibm-1146_P100-1997,swaplfnl		ibm-1146_P100-1997,swaplfnl ibm-1146-s390
ibm-1147_P100-1997,swaplfnl		ibm-1147_P100-1997,swaplfnl ibm-1147-s390
ibm-1148_P100-1997,swaplfnl		ibm-1148_P100-1997,swaplfnl ibm-1148-s390
ibm-1149_P100-1997,swaplfnl		ibm-1149_P100-1997,swaplfnl ibm-1149-s390
ibm-1153_P100-1999,swaplfnl		ibm-1153_P100-1999,swaplfnl ibm-1153-s390
ibm-12712_P100-1998,swaplfnl		ibm-12712_P100-1998,swaplfnl ibm-12712-s390
ibm-16804_X110-1999,swaplfnl		ibm-16804_X110-1999,swaplfnl ibm-16804-s390
ebcdic-xml-us		ebcdic-xml-us

The new alias **x-roman8** is added to **ibm-1051_P100-1995** charset. This alias is applicable to XCOM Data Transport for Unicode transfers on Linux, AIX, Solaris Sparc, Solaris x86, and Linux s390x 64-bit platforms only.

Double-byte character sets (DBCS)

Canonical Name	IANA	All Aliases
ibm-5478_P100-1995	GB_2312-80 chinese iso-ir-58 csISO58GB231280	ibm-5478_P100-1995 ibm-5478 GB_2312-80 chinese iso-ir-58 csISO58GB231280 gb2312-1980 GB2312.1980-0
ibm-971_P100-1995		ibm-971_P100-1995 ibm-971 ibm-971_VPUA x-IBM971
ibm-16684_P110-2003		ibm-16684_P110-2003 ibm-16684 ibm-20780

ISO_2022

Canonical Name	IANA	All Aliases
ISO_2022,locale=ja,version=0	ISO-2022-JP csISO2022JP	ISO_2022,locale=ja,version=0 ISO-2022-JP csISO2022JP x-windows-iso2022jp x-windows-50220
ISO_2022,locale=ja,version=1	JIS_Encoding csJISEncoding	ISO_2022,locale=ja,version=1 ISO-2022-JP-1 JIS_Encoding csJISEncoding ibm-5054 JIS x-windows-50221
ISO_2022,locale=ja,version=2	ISO-2022-JP-2 csISO2022JP2	ISO_2022,locale=ja,version=2 ISO-2022-JP-2 csISO2022JP2
ISO_2022,locale=ja,version=3		ISO_2022,locale=ja,version=3 JIS7
ISO_2022,locale=ja,version=4		ISO_2022,locale=ja,version=4 JIS8
ISO_2022,locale=ko,version=0	ISO-2022-KR csISO2022KR	ISO_2022,locale=ko,version=0 ISO-2022-KR csISO2022KR
ISO_2022,locale=ko,version=1		ISO_2022,locale=ko,version=1 ibm-25546
ISO_2022,locale=zh,version=0	ISO-2022-CN	ISO_2022,locale=zh,version=0 ISO-2022-CN csISO2022CN x-ISO-2022-CN-GB
ISO_2022,locale=zh,version=1	ISO-2022-CN-EXT	ISO_2022,locale=zh,version=1 ISO-2022-CN-EXT
ISO_2022,locale=zh,version=2		ISO_2022,locale=zh,version=2 ISO-2022-CN-CNS x-ISO-2022-CN-CNS

HZ

Canonical Name	IANA	All Aliases
HZ	HZ-GB-2312	HZ HZ-GB-2312

Indian Script Code for Information Interchange (ISCII)

Canonical Name	ANA	All Aliases
ISCII,version=0		ISCII,version=0 x-ISCII91 x-iscii-de windows-57002 iscii-dev ibm-4902
ISCII,version=1		ISCII,version=1 x-iscii-be windows-57003 iscii-bng windows-57006 x-iscii-as
ISCII,version=2		ISCII,version=2 x-iscii-pa windows-57011 iscii-gur
ISCII,version=3		ISCII,version=3 x-iscii-gu windows-57010 iscii-guj
ISCII,version=4		ISCII,version=4 x-iscii-or windows-57007 iscii-ori
ISCII,version=5		ISCII,version=5 x-iscii-ta windows-57004 iscii-tml
ISCII,version=6		ISCII,version=6 x-iscii-te windows-57005 iscii-tlg
ISCII,version=7		ISCII,version=7 x-iscii-ka windows-57008 iscii-knd
ISCII,version=8		ISCII,version=8 x-iscii-ma windows-57009 iscii-mlm

LMBCS_1

Canonical Name	IANA	All Aliases
LMBCS-1		LMBCS-1 lmbcs ibm-65025

EBCDIC_STATEFUL

Canonical Name	IANA	All Aliases
ibm-930_P120-1999		ibm-930_P120-1999 ibm-930 ibm-5026 IBM930 cp930 930 x-IBM930 x-IBM930A
ibm-933_P110-1995		ibm-933_P110-1995 ibm-933 cp933 933 x-IBM933
ibm-935_P110-1999		ibm-935_P110-1999 ibm-935 cp935 935 x-IBM935
ibm-937_P110-1999		ibm-937_P110-1999 ibm-937 cp937 937 x-IBM937
ibm-939_P120-1999		ibm-939_P120-1999 ibm-939 ibm-931 ibm-5035 IBM939 cp939 939 x-IBM939 x-IBM939A
ibm-1364_P110-2007		ibm-1364_P110-2007 ibm-1364 x-IBM1364
ibm-1371_P100-1999		ibm-1371_P100-1999 ibm-1371 x-IBM1371
ibm-1388_P103-2001		ibm-1388_P103-2001 ibm-1388 ibm-9580 x-IBM1388

ibm-1390_P110-2003		ibm-1390_P110-2003 ibm-1390 x-IBM1390
ibm-1399_P110-2003		ibm-1399_P110-2003 ibm-1399 x-IBM1399

Additional Resources

Includes educational resources, product references, and support information.

This article provides resources to assist you in maximizing your product experience, including information about education, webcasts, product support resources, and product maintenance.

Education and Training

Use the following links to learn more about XCOM Data Transport:

- Mainframe Training
- Mainframe eLearning Library
- Learning Paths for XCOM Data Transport
- Mainframe Product Roadmap Webcasts
- Educate YouTube Channel (requires a Google account)

Product Support

To view product details, go to Broadcom Support. Under **Software** select **Mainframe Software**, **All Products**, search by product name, and select **Product Details**.

Maintenance

Use the following resources to obtain information about maintaining XCOM Data Transport:

- Mainframe Installation and Maintenance Tools
- Security Advisories - Mainframe Software (login required)
- Broadcom Mainframe Products Solutions List
- Broadcom Mainframe Products Fix Category Solutions List
- Recommended Service for z/OS (CARS)
- Migrate SMP/E Environments into z/OSMF

  NOTE
  For migration assistance and access to z/OSMF trainings from Broadcom, see z/OSMF Migration.
- Mainframe Essentials: SYSVIEW Essentials, Software Toolkit Plug-in for z/OSMF, Mainframe Resource Intelligence
- Broadcom Mainframe Product Lifecycle Page
- XCOM Data Transport Release and Support Lifecycle Dates
- Mainframe Compatibilities
- Broadcom Support Network Details

User Communities

Consult your peers, reach out to subject matter experts, and read the latest technical insights and information in our global communities:

- Broadcom Mainframe Software Division (MSD) Microsite
- Broadcom Mainframe Software Communities
- XCOM Data Transport Community

Social Media

Through the following Broadcom Mainframe Software channels, we share information that provides value to our mainframe community, including events, blog posts, eBooks, analyst reports, and more:

- LinkedIn
- X

Documentation Legal Notice

This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the “Documentation”) is for your informational purposes only and is subject to change or withdrawal by Broadcom at any time. This Documentation is proprietary information of Broadcom and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of Broadcom.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all Broadcom copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to Broadcom that all copies and partial copies of the Documentation have been returned to Broadcom or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, BROADCOM PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL BROADCOM BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF BROADCOM IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice

The manufacturer of this Documentation is Broadcom Inc.

Provided with “Restricted Rights.” Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

BROADCOM®
MAINFRAME SOFTWARE