# CA XCOM™ Data Transport® for UNIX and Linux

## Release Notes

### Release 11.6 Second Edition

# CA Technologies Product References

This document references the following CA Technologies products:

- CA XCOM™ Data Transport® for Gateway Version 12.0

# Contact CA Technologies

**Contact CA Support**

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

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

**Providing Feedback About Product Documentation**

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

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

# Documentation Changes

The following documentation updates have been made since the last release of this documentation:

- CA XCOM Data Transport for Linux PC x64 (see page 13) - Provides information about CA XCOM Data Transport for Linux PC x64, a 64-bit application
- CA Licensing for Linux x64 bit (see page 13) - Provides licensing information
- CA XCOM Data Transport for AIX 64 (see page 13) - Provides information about CA XCOM Data Transport for AIX 64, a 64-bit application
- CA Licensing for AIX 64 (see page 13) - Provides licensing information
- CA XCOM Data Transport for Oracle Solaris 64 (see page 14) - Provides information about CA XCOM Data Transport for Oracle Solaris 64, a 64-bit application
- CA Licensing (Oracle Solaris 64) (see page 14) - Provides licensing information
- CA XCOM Data Transport for Oracle Solaris x86 64 (see page 14) - Provides information about CA XCOM Data Transport for Oracle Solaris x86 64, a 64-bit application
- CA Licensing (Oracle Solaris x86 64) (see page 14) - Provides licensing information
- CA XCOM Data Transport for Linux s390x (see page 15) - Provides information about CA XCOM Data Transport for Linux s390x, a 64-bit application
- CA Licensing (Linux s390x) (see page 15) - Provides licensing information
- SNA Support (see page 15) - SNA transfer protocol support discontinued

# Contents

# Chapter 1: Enhanced Features

The *Release Notes* for CA XCOM Data Transport for Unix and Linux documents both new features and changes to existing features for r11.6.

**Note:** The Release Notes Release 11.6 Second Edition documents changes due to 64-bit support.

This section contains the following topics:



## UNICODE and Multi-Byte Character Set Support for Data Transfer

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.

The ICU (International Components for Unicode) toolkit is utilized for performing data conversion operations by CA XCOM Data Transport.

More information:

- Unicode and Multi-Byte Character Set Support for Data Transfer (see page 17)

# Transmission Password Encryption Cipher Selection

CA XCOM Data Transport always protects the password during transmission by using encryption. Prior to r11.6, the password was encrypted using a CA XCOM Data Transport proprietary password encryption algorithm. It consists of 2 key elements (a fixed and a variable element) which are combined to form a symmetric encryption key that is used to encrypt the transmitted password.

With CA XCOM Data Transport r11.6, the encryption cipher used to protect the password during transmission can now be selected using parameters. Permitted ciphers include DES, 3DES, AES, RC2, RC4 and the existing CA XCOM proprietary algorithm.

More information:

- Transmission Password Encryption Cipher Selection (see page 35)

# History Search Enhancements

The Get History Records screen has been enhanced with several new features including an auto-refresh option and additional search fields.

More information:

- History Search Enhancements (see page 41)

# PAM-Based Authentication

The Pluggable Authentication Modules (PAM) library is a generalized API for authentication-related services. With PAM-based authentication in CA XCOM Data Transport, users can choose different authentication mechanisms (such as SYSTEM, EEM, or LDAP) without having to change the application.

# New Parameters

This section describes the new parameters used for PAM-based authentication.

**AUTH_TYPE**

Specifies the type of authentication (PAM or SYSTEM) to be used for transfers.

**PAM_PATH**

Specifies the path to your PAM library for your current UNIX or Linux platform.

For more information about these parameters, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# Removal of Java Dependency for XML Parsing

The CA XCOM Data Transport GUI allows users to create configuration files which can later be reloaded into the GUI or used at the command line. Configuration files created by the GUI are saved in XML format. In CA XCOM Data Transport r11.5, when XML format configuration files are processed Java based routines were used to parse the file.

The use of Java based routines required CA XCOM Data Transport to load the Java executables and create a Java virtual machine for each command line transfer. This resulted in increased overhead and memory usage required for the Java virtual machine.

For CA XCOM Data Transport r11.6, the Java Dependency has been removed for XML parsing thus eliminating the overhead and memory usage required for the Java virtual machine.

# Cross Platform Additional Parameters

CA XCOM Data Transport r11.6 includes support for additional parameters used by other CA XCOM Data Transport partner systems.

More information:

- Cross Platform Additional Parameters (see page 45)

# Updated Database SQL file

A new history update SQL file, histupdateDB.sql, is installed during installation of CA XCOM and is required to be run for those History database users upgrading from r11.5 to r11.6. This needs to be done in order to support the new enhancements made with this version of CA XCOM.

# New XCOMTCP Option

A new option has been added to the XCOMTCP command for Transmission Password Encryption.

**-ping**

CA XCOM Data Transport tests reachability of the remote CA XCOM Data Transport Server and displays information about CA XCOM Data Transport Server.

XCOMTCP –ping command example:

```
xcomtcp -ping REMOTE_SYSTEM=XX PORT=XX TRNENCRL_CIPHER=ALL
XCOMN0882I PING INFO FOR <SYSTEMNAME>
XCOMN0882I RELEASE=r11.6 SP00 GEN LEVEL <LEVEL> SYSTEM NAME=<SYSNAME> SYSTEM ID=<SYSID>
XCOMN0882I NEGOTIATED CIPHER=<CIPHER>
```

# CA XCOM Gateway Compatibility

CA XCOM Data Transport r11.6 maintenance level 13011 or higher is compatible with:

- CA XCOM Gateway r11.5
- CA XCOM Gateway r11.6

If your CA XCOM Data Transport server is allied to CA XCOM Gateway, specify the CA XCOM Gateway version using the new GATEWAY_VERSION Global Parameter.

More information:

Using CA XCOM Data Transport with [set to your product name], see the section About CA XCOM Gateway in the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# CA XCOM Data Transport r11.5 Enhanced Features

Sites upgrading from CA XCOM Data Transport r11.0 should also reference the enhanced features that were first introduced in CA XCOM Data Transport r11.5.

More information:

- CA XCOM Data Transport r11.5 Enhanced Features

# New Operating Systems Support

## CA XCOM Data Transport for Linux PC x64

CA XCOM Data Transport for Linux PC x64 is built as a 64-bit application. The CA XCOM Data Transport for Linux PC x64 API must be compiled and linked as a 64-bit application.

CA XCOM Data Transport for Linux PC x64 no longer supports xcomtool and the xcomtool utility is not shipped with the product.

**Note:** For more information about software requirements, hardware requirements, and operating system support, see the section *Installing CA XCOM Data Transport for Linux PC x64* in the *Installation Guide*.

### CA Licensing (Linux PC x64)

This release uses CA Licensing to ensure that the installed version of CA XCOM Data Transport is properly licensed.

**Note:** For more information about using CA Licensing, see the section *Installing CA XCOM Data Transport for Linux PC x64* in the *Installation Guide*.

## CA XCOM Data Transport for AIX 64

CA XCOM Data Transport for AIX 64 is built as a 64-bit application. The CA XCOM Data Transport for AIX 64 API must be compiled and linked as a 64-bit application.

CA XCOM Data Transport for AIX 64 no longer supports xcomtool and the xcomtool utility is not shipped with the product.

**Note:** For more information about software requirements, hardware requirements, and operating system support, see the section *Installing CA XCOM Data Transport for AIX 64* in the *Installation Guide*.

### CA Licensing (AIX 64)

This release uses CA Licensing to ensure that the installed version of CA XCOM Data Transport is properly licensed.

**Note:** For more information about using CA Licensing, see the section *Installing CA XCOM Data Transport for AIX 64* in the *Installation Guide*.

# CA XCOM Data Transport for Oracle Solaris 64

CA XCOM Data Transport for Oracle Solaris 64 is built as a 64-bit application. The CA XCOM Data Transport for Oracle Solaris 64 API must be compiled and linked as a 64-bit application.

CA XCOM Data Transport for Oracle Solaris 64 no longer supports xcomtool and the xcomtool utility is not shipped with the product.

**Note:** For more information about software requirements, hardware requirements, and operating system support, see the section Installing CA XCOM Data Transport for Oracle Solaris 64 in the *Installation Guide*.

## CA Licensing (Oracle Solaris 64)

This release uses CA Licensing to ensure that the installed version of CA XCOM Data Transport is properly licensed.

**Note:** For more information about using CA Licensing, see the section Installing CA XCOM Data Transport for Oracle Solaris 64 in the *Installation Guide*.

# CA XCOM Data Transport for Oracle Solaris x86 64

CA XCOM Data Transport for Oracle Solaris x86 64 is built as a 64-bit application. The CA XCOM Data Transport for Oracle Solaris x86 64 API must be compiled and linked as a 64-bit application.

CA XCOM Data Transport for Oracle Solaris x86 64 no longer supports xcomtool and the xcomtool utility is not shipped with the product.

**Note:** For more information about software requirements, hardware requirements, and operating system support, see the section Installing CA XCOM Data Transport for Oracle Solaris x86 64 in the *Installation Guide*.

## CA Licensing (Oracle Solaris x86 64)

This release uses CA Licensing to ensure that the installed version of CA XCOM Data Transport is properly licensed.

**Note:** For more information about using CA Licensing, see the section Installing CA XCOM Data Transport for Oracle Solaris x86 64 in the *Installation Guide*.

# CA XCOM Data Transport for Linux s390x

CA XCOM Data Transport for Linux s390x is built as a 64-bit application. The CA XCOM Data Transport for Linux s390x API must be compiled and linked as a 64-bit application.

CA XCOM Data Transport for Linux s390x no longer supports xcomtool and the xcomtool utility is not shipped with the product.

**Note:** For more information about software requirements, hardware requirements, and operating system support, see the section Installing CA XCOM Data Transport for Linux s390x in the *Installation Guide.*

## CA Licensing (Linux s390x)

This release uses CA Licensing to ensure that the installed version of CA XCOM Data Transport is properly licensed.

**Note:** For more information about using CA Licensing, see the section Installing CA XCOM Data Transport for Linux s390x in the *Installation Guide*.

# SNA Support

This release does not support SNA transfer protocol.

# Chapter 2: 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.

This section contains the following topics:



# New Global Parameters

Global parameters added for Unicode transfers.

## DEFAULT_CHARSET

This parameter specifies the default character set CA 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 characters that cannot be converted because they 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.

Additionally, 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 not consistent with the specified input character set.

### XCOM_ICUPATH

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

## New Configuration Parameters

Configuration parameters added for Unicode transfers.

### LOCAL_CHARSET

This parameter specifies the local character set CA 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.

Additionally, 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).

## MBCS_CONVERROR

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

## 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 CA 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 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) have been added.

Fields added to the Edit Transfer Record screen for Unicode transfers.

**Local System Parameters Character-set**

Specifies the local character set that the CA 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 CA 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 modified on the Detail History Record screen for Unicode transfers.

**Encoding**

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

Fields 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 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 CA 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.

# New and Changed Messages

This section describes the new and changed messages to support this enhancement.

## 0898I

**Unicode conversion detected malformed characters.**

**Reason:**

Malformed characters were detected during Unicode conversion.

**Action:**

No action is required.

## 0899I

**Unicode conversion performed character substitution.**

**Reason:**

Malformed characters detected during Unicode conversion were substituted.

**Action:**

None. The detected malformed characters can be found in the transfer trace.

## 0900I

**Unicode conversion skipped malformed characters.**

**Reason:**

Malformed characters detected during Unicode conversion were skipped.

**Action:**

None. The detected malformed characters can be found in the transfer trace.

## 0901I

**CARRIAGE_FLAG=XPACK has been set for the Unicode transfer.**

**Reason:**

CARRIAGE_FLAG=XPACK is automatically set for Unicode transfer.

**Action:**

None

## 0902I

**Unicode conversion summary: Descriptive message.**

**Reason:**

Summary of Unicode conversion is written in the trace.

**Action:**

None

## 0903I

**Block I/O has been disabled for Unicode transfer.**

**Reason:**

Block I/O is not supported for Unicode transfer. Hence Block I/O has been disabled internally.

**Action:**

None

## 0904I

**Unicode converter opened successfully for charset=Descriptive message**

**Reason:**

Unicode converter is opened successfully for Unicode transfer.

**Action:**

None. The name of the charset, canonical name and CCSID (if available) of the charset opened for Unicode transfer can be found in the trace.

## 0905E

**Remote XCOM server does not support Unicode. Transfer terminated.**

**Reason:**

The remote partner returned an error indicating that the Unicode transfer (CODE_FLAG=UTF8 or CODE_FLAG=UTF16) is not supported by that version of CA XCOM Data Transport.

**Action:**

Upgrade the remote CA XCOM Data Transport.

## 0906E

**Could not load ICU Library.**

**Reason:**

Failed to load ICU (International Components for Unicode) Library required for Unicode transfer.

**Action:**

Check if your CA XCOM Data Transport installation is valid. The ICU library will be installed by CA XCOM Data Transport installer. Retry the operation. If the problem persists contact CA Technical Support.

## 0907E

**LOCAL_DELIM contains invalid value.**

**Reason:**

CA XCOM Data Transport API (XcomAPI) returns this error code when the local_delim contains an invalid value.

**Action:**

Correct the value of the local_delim XCOM API member passed to XcomAPI routine. See the list of supported delimiters for local_delim in Application Programming Interface chapter in this guide.

## 0908E

**REMOTE_DELIM contains invalid value.**

**Reason:**

CA XCOM Data Transport API (XcomAPI) returns this error code when the remote_delim parameter contains an invalid value.

**Action:**

Correct the value of the remote_delim XCOM API member passed to the XcomAPI routine. See the list of supported delimiters for remote_delim in the Application Programming Interface chapter in this guide.

## 0909E

**MBCS_INPUTERROR contains REPLACE# succeeded by invalid value.**

**Reason:**

CA XCOM Data Transport API (XcomAPI) returns this error code when the mbcs_inputerror parameter contains an invalid value.

**Action:**

Correct the value of the mbcs_inputerror XCOM API member passed to the XcomAPI routine. See the list of correct formats for mbcs_inputerror in the Application Programming Interface chapter in this guide.

## 0910E

**MBCS_INPUTERROR contains REPLACE# succeeded by value out of range(0-1114111).**

**Reason:**

CA XCOM Data Transport API (XcomAPI) returns this error code when the mbcs_inputerror parameter contains an invalid value.

**Action:**

Correct the value of the mbcs_inputerror XCOM API member passed to the XcomAPI routine. See the list of correct formats for mbcs_inputerror in the Application Programming Interface chapter in this guide.

## 0911E

**MBCS_INPUTERROR value is invalid.**

**Reason:**

CA XCOM Data Transport API (XcomAPI) returns this error code when the mbcs_inputerror parameter contains an invalid value.

**Action:**

Correct the value of the mbcs_inputerror XCOM API member passed to the XcomAPI routine. See the list of correct formats for mbcs_inputerror in the Application Programming Interface chapter in this guide.

## 0912E

**MBCS_CONVERROR contains REPLACE# succeeded by invalid value.**

**Reason:**

CA XCOM Data Transport API (XcomAPI) returns this error code when the mbcs_converror parameter contains an invalid value.

**Action:**

Correct the value of the mbcs_converror XCOM API member passed to the XcomAPI routine. See the list of correct formats for mbcs_converror in the Application Programming Interface chapter in this guide.

## 0913E

**MBCS_CONVERROR contains REPLACE# succeeded by value out of range(0-1114111).**

**Reason:**

CA XCOM Data Transport API (XcomAPI) returns this error code when the mbcs_converror parameter contains an invalid value.

**Action:**

Correct the value of the mbcs_converror XCOM API member passed to the XcomAPI routine. See the list of correct formats for mbcs_converror in the Application Programming Interface chapter in this guide.

## 0914E

**MBCS_CONVERROR value is invalid.**

**Reason:**

CA XCOM Data Transport API (XcomAPI) returns this error code when the mbcs_converror parameter contains an invalid value.

**Action:**

Correct the value of the mbcs_converror XCOM API member passed to the XcomAPI routine. See the list of correct formats for mbcs_converror in the Application Programming Interface chapter in this guide.

## 0915E

**Failed to open Unicode converter.**

**Reason:**

Failed to open Unicode converter for the Unicode transfer (CODE_FLAG=UTF8 or CODE_FLAG=UTF16).

**Action:**

Check the LOCAL_CHARSET, REMOTE_CHARSET or DEFAULT_CHARSET values. See the list of charsets supported in the Appendix chapter in this guide.

## 0916E

**Failed to open Unicode converter for [Descriptive Message].**

**Reason:**

Failed to open Unicode converter for the Unicode transfer (CODE_FLAG=UTF8 or CODE_FLAG=UTF16).

**Action:**

Check the LOCAL_CHARSET, REMOTE_CHARSET or DEFAULT_CHARSET values. See the list of charsets supported in the Appendix chapter in this guide.

## 0917E

**Failed to open internal Unicode converter for [Descriptive Message].**

**Reason:**

Failed to open the Unicode converter for translating the Unicode data to the negotiated Unicode encoding internally.

**Action:**

Retry the operation. If the problem persists, contact CA Technical Support.

## 0920E

**Failed to allocate memory for Unicode conversion.**

**Reason:**

Failed to allocate the memory required for Unicode conversion.

**Action:**

Retry the operation. If the problem persists, contact CA Technical Support.

## 0921E

**Invalid MBCS_CONVERROR replacement Unicode character.**

**Reason:**

The replacement character specified with the MBCS_CONVERROR=REPLACE#nnnnnn option is not a valid Unicode character.

**Action:**

Specify a valid Unicode character in the decimal value for the MBCS_CONVERROR=REPLACE#nnnnnn option.

## 0922E

**Invalid MBCS_INPUTERROR replacement Unicode character.**

**Reason:**

The replacement character specified with the MBCS_INPUTERROR=REPLACE#nnnnnn option is not a valid Unicode character.

**Action:**

Specify a valid Unicode character in the decimal value for the MBCS_INPUTERROR=REPLACE#nnnnnn option.

## 0923E

**MBCS_CONVERROR replacement character cannot be converted to [Descriptive Message].**

**Reason:**

The replacement character specified with the MBCS_CONVERROR=REPLACE#nnnnnn option cannot be converted to the specified charset.

**Action:**

Specify a valid Unicode character in the decimal value for the MBCS_CONVERROR=REPLACE#nnnnnn option which can be converted to the charset.

## 0924E

**MBCS_INPUTERROR replacement character cannot be converted to [Descriptive Message].**

**Reason:**

The replacement character specified with the MBCS_INPUTERROR=REPLACE#nnnnnn option cannot be converted to the specified charset.

**Action:**

Specify a valid Unicode character in the decimal value for the MBCS_INPUTERROR=REPLACE#nnnnnn option which can be converted to the charset.

## 0925E

**Unicode conversion failed.**

**Reason:**

An error occurred during Unicode conversion.

**Action:**

Check the actual error message in the transfer trace.

## 0926E

**Failed to convert to Unicode. rc=[Descriptive message]**

**Reason:**

An error occurred during converting from the specified charset to Unicode encoding.

**Action:**

Retry the operation. If the problem persists, contact CA Technical Support.

## 0927E

**Failed to convert from Unicode. rc=[Descriptive message]**

**Reason:**

An error occurred during converting from the Unicode encoding to specified charset.

**Action:**

Retry the operation. If the problem persists, contact CA Technical Support.

## 0929E

**Unicode conversion detected malformed characters. Transfer terminated.**

**Reason:**

An error occurred during Unicode conversion.

**Action:**

None. The detected malformed characters can be found in the transfer trace.

## 3689E

**UNICODE Input Character Error Action is Invalid.**

**Reason:**

The entered action is invalid. The valid actions are "FAIL","SKIP","REPLACE", or “REPLACE#nnnnnnn”. Where nnnnnnn is a number in the range of 0 through 1114111.

**Action:**

Correct the entered action and repeat the process.

## 3690E

**UNICODE Convert Character Error Action is Invalid.**

**Reason:**

The entered action is invalid. The valid action is "FAIL","SKIP","REPLACE", or “REPLACE#nnnnnnn”. Where nnnnnnn is a number in the range of 0 through 1114111.

**Action:**

Correct the entered action and repeat the process.

## 3691E

**UNICODE Default Delimiter is Invalid.**

**Reason:**

The entered default delimiter list contains invalid characters and/or delimiter values and/or mutually exclusive values. The valid delimiter values are “EBCDIC", "ASCII", "CR", "LF", "NL", "VT", "FF", "LS", "PS", "CRLF", "LFCR", "CRNL" separated by (:) character. For mutually exclusive values and the other rules related to the delimiter values, see DEFAULT_DELIM parameter description in the XCOM.GLB Parameters section of the Administration Guide.

**Action:**

Correct the list and repeat the action.

## 3692E

**Character-Set Input Error is Invalid.**

**Reason:**

The entered value is invalid. The valid value is "FAIL","SKIP","REPLACE", or “REPLACE#nnnnnnn”. Where nnnnnnn is a number in the range of 0 through 1114111.

**Action:**

Correct the entered value and repeat the process.

## 3693E

**Character-Set Convert Error is Invalid.**

**Reason:**

The entered value is invalid. The valid value is "FAIL","SKIP","REPLACE", or “REPLACE#nnnnnnn”. Where nnnnnnn is a number in the range of 0 through 1114111.

**Action:**

Correct the entered value and repeat the process.

## 3694E

**Local Record Delimiter is Invalid.**

**Reason:**

The entered default delimiter list contains invalid characters and/or delimiter values and/or mutually exclusive values. The valid delimiter values are: “EBCDIC", "ASCII", "CR", "LF", "NL", "VT", "FF", "LS", "PS", "CRLF", "LFCR", "CRNL" separated by (:) character. For mutually exclusive values and the other rules related to the delimiter values, see LOCAL_DELIM parameter description in the List of Parameters section of this guide.

**Action:**

Correct the list and repeat the action.

## 3695E

**Remote Record Delimiter is Invalid.**

**Reason:**

The entered default delimiter list contains invalid characters and/or delimiter values and/or mutually exclusive values. The valid delimiter values are: “EBCDIC", "ASCII", "CR", "LF", "NL", "VT", "FF", "LS", "PS", "CRLF", "LFCR", "CRNL" separated by (:) character. For mutually exclusive values and the other rules related to the delimiter values, see REMOTE_DELIM parameter description in the List of Parameters section of this guide.

**Action:**

Correct the list and repeat the action.

# Chapter 3: 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.

In order to use Transmission Password Encryption Cipher Selection, the following must be true:

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

A COMPAT option is provided that can be used select the CA XCOM Data Transport proprietary password encryption algorithm to provide backward compatibility if Transmission Password Encryption Cipher Selection cannot be used.

This section contains the following topics:



## 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.

### TRNENCRR_CIPHER

This parameter specifies the permitted list of ciphers that are used to encrypt the password fields for remotely initiated transfers. The permitted list of ciphers is matched against the requested list of ciphers provide by the local system by the TRNENCRL_CIPHER parameter. The common cipher with the highest ranking is selected to encrypt the password fields.

### 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 CA 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 which 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 which are used to encrypt the password fields for locally initiated transfers.

**Remote Permitted Cipher List**

Specifies the permitted list of ciphers which 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 CA XCOM Data Transport header.

# New and Changed Messages

This section describes the new and changed messages to support this enhancement.

New messages for Transmission Password Encryption Cipher Selection.

## 0886E

**PING action cannot be performed with TRNENCRL_CIPHER=COMPAT.**

**Reason:**

This message is issued if the XCOM PING action is performed with TRNENCRL_CIPHER value set to COMPAT.

**Action:**

Retry XCOM Ping action with TRNENCRL_CIPHER value set to cipher(s) other than COMPAT. See the list of ciphers allowed for TRNENCRL_CIPHER in the List of Parameters in this guide.

## 1317E

**Local discover request failed.**

**Reason:**

An error was detected while attempting to send a discovery request. Possible reason could be the failure of Xrcvwt.

**Action:**

Check the Network connectivity and retry the transfer.

## 1325E

**Remote Discover Request failed: reason**

**Reason:**

Possible reason could be one of the following:

- Xrcvwt FAILED
- Xrcvwt/Xcnfrmed FAILED
- Xsnddta FAILED
- Xprprcv FAILED
- Cipher match not found
- Error occurred while matching the cipher
- Cannot initialize ETPKI library
- Error code : code

**Action:**

Depends on the reason that is returned in the message.

For the first four reasons, check the network connectivity and retry the transfer.

If the cipher match is not found, recheck the Local and Remote cipher preferences set in xcom.glb. Refer to How Transmission Password Encryption Works for more information about how to set the (STC)TRNENCRL_CIPHER/TRNENCRR_CIPHER parameters.

If ETPKI (CAPKI) Initialization fails, check whether the ETPKI (CAPKI) Library is installed properly and that the CAPKIHOME global parameter is set to the correct value.

## 1326E

**Local ping failed.**

**Reason:**

xcomtcp returns this error code when the required arguments are not supplied to the XCOM Ping request.

**Action:**

Provide the required arguments to the XCOM Ping request.

## 1327E

**Transfer is not allowed without Password cipher negotiation.**

**Reason:**

This error is produced when TRNENCRR_CIPHER on remote XCOM Server does not have COMPAT in the list of ciphers and

- Transfer is initiated with TRNENCRL_CIPHER=COMPAT.
- Transfer is initiated from CA XCOM Data Transport version which does not support Password cipher negotiation.

**Action:**

Add COMPAT to the TRNENCRR_CIPHER global parameter on remote XCOM server to permit the CA XCOM Data Transport proprietary cipher without cipher negotiation that is required for backward password compatibility with CA XCOM Data Transport versions before 11.6

or

Choose different cipher for TRNENCRL_CIPHER other than COMPAT, in order to encrypt the password when in flight.

## 1427E

**Password cipher negotiation failed: reason**

**Reason:**

Possible reason could be one of the following:

- Cipher match not found
- Partner returned cipher error
- Cannot initialize ETPKI library

**Action:**

Depends on reason.

If the cipher match is not found, recheck the Local and Remote cipher preferences set in xcom.glb. Refer to How Transmission Password Encryption Works for more information.

Recheck the remote partner cipher preferences. Refer to How Transmission Password Encryption Works for more information about how to set the (STC)TRNENCRL_CIPHER/TRNENCRR_CIPHER parameters.

Check whether the ETPKI (CAPKI) Library is installed properly and that the CAPKIHOME global parameter is set to the correct value.

# Chapter 4: 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

This section contains the following topics:



## 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.

# Chapter 5: Cross Platform Additional Parameters

CA XCOM Data Transport r11.6 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.

This section contains the following topics:



## 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 Configuration 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 of a FILE_OPTION=CREATE transfer.

### 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 of a 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).

# Chapter 6: CA XCOM Data Transport r11.5 Enhanced Features

This section documents the enhanced features that were first introduced in CA XCOM Data Transport r11.5 for sites upgrading from CA XCOM Data Transport r11.0.

This section contains the following topics:

# CA XCOM Data Transport GUI

CA XCOM Data Transport r11.5 introduced the CA XCOM Data Transport Graphical User Interface (GUI), providing a common user interface for CA XCOM Data Transport on the following platforms:

- UNIX and Linux
- Windows

The GUI provides access to all of the features of CA XCOM Data Transport. Using the GUI you can do:

- Create and edit CA XCOM Data Transport transfer configuration files
- Initiate transfers
- Monitor the log and trace functions
- Update the status of active and pending transfers
- Get the history of transfer records
- To produce an Easytrieve report or export to Excel, save the history of transfer records in the tab delimited form

Users in the UNIX and Linux Administrator Group can also do:

- Trusted transfer configuration
- Global parameter configuration

The CA XCOM Data Transport GUI is the front end for CA XCOM Data Transport. You use this GUI to communicate with the CA XCOM Data Transport server. All requests, such as scheduling transfers and getting history records, are passed to the CA XCOM Data Transport server using the TCP/IP interface. Results that are obtained are returned to the GUI.

The CA XCOM Data Transport GUI runs standalone with minimal configuration.

For more information about the CA XCOM Data Transport GUI, see:

- *The CA XCOM Data Transport for UNIX and Linux User Guide*
- *The GUI online help*

# Metatransfers

CA XCOM Data Transport r11.5 introduced metatransfers, that is, transfers containing control information.

Using metatransfers allows you to do the following:

- Schedule transfers to a remote CA XCOM Data Transport server running r11.5 or above, using the –c5 option.
- Schedule inquire transfers to a remote CA XCOM Data Transport server running r11.5 or above, using the –c6 option.
- Retrieve history records to a remote CA XCOM Data Transport server running r11.5 or above, using the –c7 option. For more information, see History Command Option (-c7) in this chapter.

**Note:** If the remote server is z/OS, it can be either r11.5 or r11.6.

You can perform metatransfers across the following mainframe and distributed platforms:

- z/OS
- UNIX and Linux
- Windows

The following parameters have been added or updated to support metatransfers and to allow z/OS and UNIX parameters to be used in Windows:

- AGE—Specifies the number of days of history records that are retained when a purge procedure is executed.
- CKPT (alias of CHECKPOINT_COUNT)—Defines how often (based on record count) the sending system requests a checkpoint to be taken.
- CODE (alias of CODE_FLAG)—Identifies the type of data being transferred.
- DROPSESS—Indicates whether CA XCOM Data Transport drops an LU-to-LU session at the conclusion of a scheduled file transfer.
- EPRTY—Indicates the execution priority for the request on the remote z/OS system.
- FILE (alias of REMOTE_FILE)—The name of the file on the remote system that is being transferred.
- FILEDATA—Indicates how a remote USS file is allocated on the remote z/OS system when performing a receive file transfer on the local Windows system.
- GROUP—Valid only on a metatransfer (-c5); the group ID to use on the CA XCOM Data Transport system receiving a metatransfer.
- HIST_FILE—The name of the file to contain the history records returned by the inquire function (-c6) or when retrieving history records (-c7).

- HOLD_TRANSFER—Prevents a QUEUE=YES transfer from starting until explicitly released.
- INQ_FILE—Specifies the complete path and file name that will contain the information required to do an inquire on a transfer (-c6).
- INQ_WAIT—Specifies how long CA XCOM Data Transport should wait (in hours (hh), minutes (mm), and seconds (ss)) for transfers to complete when doing an inquire on a transfer (-c6).
- IPNAME (alias of REMOTE_SYSTEM)—For TCP/IP protocols, the name of the remote system that receives a file, job, or report.
- IPPORT (alias of PORT)—The number of the TCP/IP port on the remote CA XCOM Data Transport server.
- LDOMAIN—The domain associated with LUSERID and LPASSWORD, if the target system is Windows, when handling a metatransfer (-c5) or a history record retrieval transfer (-c7).
- LFILE (alias of LOCAL_FILE)—The name of the file on the local system that is being transferred.
- LIST—Valid only on a metatransfer (-c5); the list ID to use on the CA XCOM Data Transport system receiving a metatransfer.
- LPASS (alias of LPASSWORD)—The password of the local user to be validated on the CA XCOM Data Transport server handling a metatransfer (-c5) or a history record retrieval transfer (-c7).
- LU—Valid only on a metatransfer (-c5); the LU to use on the CA XCOM Data Transport system receiving a metatransfer.
- LUSER (alias of LUSERID)—The user ID to use on the CA XCOM Data Transport system receiving a metatransfer (-c5) or a history record retrieval transfer (-c7).
- PROGLIB— Specifies the PDSE program library for a transfer to a CA XCOM Data Transport for z/OS system.
- RELEASE—Valid for both normal schedules and metatransfers when the remote partner is z/OS. Tells CA XCOM Data Transport for z/OS to release unused DASD space.
- RNOTIFY (alias of NOTIFYR and NOTIFY)—Specifies the remote user notification type when sending data to a remote system.
- RNOTIFYNAME (alias of NOTIFY_NAME and NOTIFYNAME)—Specifies the remote user notification type when sending data to a remote system.
- SECURE_SCHEDULE—Specifies whether the metatransfer ( c5,  c6, or  c7) uses SSL.
- SPRTY—Indicates the scheduling priority of a metatransfer request.
- STARTDATE (alias of START_DATE, but with extra functionality)—Specifies the date when the transfer is to begin. Used only when QUEUE=YES.
- STARTTIME (alias of START_TIME, but with extra functionality)—Specifies the time when the transfer is to begin. Used only when QUEUE=YES.

- STCIP—Specifies the IP address of the CA XCOM Data Transport server to handle the metatransfer ( c5,  c6, or  c7).
- STCPORT—Specifies the TCP/IP port number of the CA XCOM Data Transport server to handle the metatransfer request ( c5,  c6, or  c7).
- SYSUDATA (alias of USER_DATA)—An open field where a user can specify any text associated with the transfer.
- TDUDATA (alias of TRANSFER_USR_DATA)—An open field where a user can specify any text associated with the transfer.
- TRANSFER_ID (alias of TRANSFER_NAME)—Allows the user to enter information to identify the file transfer request.
- TRANSFER_NAME (alias of TRANSFER_ID)—Allows the user to enter information to identify the file transfer request.
- TRUNCATE (alias of TRUNCATION)—Indicates whether CA XCOM Data Transport truncates excess characters in the source file if the record exceeds the maximum record length.
- USER—Valid only on a metatransfer (-c5); the user ID to use on the CA XCOM Data Transport system receiving a metatransfer.

For more information, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# History Records

CA XCOM Data Transport r11.5 introduced history records, produced at the end of a transfer and written to a database. History records allow you to view transfer details even after transfers are completed and removed from the queue.

The history database feature allows you to specify a database management system to which CA XCOM Data Transport is to write history records.

This feature also provides for getting history records and returning them to the requestor from the GUI with the use of history filter fields. In addition, the GUI allows the history records that are returned to be exported into a comma separated (CSV) format or into a format that a CA Easytrieve report can be generated from.

Support for two new groups, XCOMADM and XCOMSADM, has also been added to control what history records a user is authorized to view.

For more information about history records, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# History Command Option (-c7)

The new **-c7** command option allows you 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 CA Easytrieve report can be generated from.

To generate the history filter file in XML format, use the Export button on the Get History Records page of the GUI.

To create a history filter file in CNF format, you can specify the following new parameters:

- OEDATE—Limits the history request to only those file transfers that were scheduled or completed on or before the end date and time.
- OETIME—Limits the history request to only those file transfers that were scheduled or completed on or before the end date and time.
- OFILETYPE—Limits the history request to those transfers with the specified FILETYPE.
- 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.
- 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.
- 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.
- OINIT—Limits the history request to only locally initiated transfers or only remotely initiated transfers.
- OLIMIT—Sets the maximum number of history records that can be returned.
- 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
- OLMSG—Limits the history request by the transfer's last message.
- OLU—Limits the history request to only those file transfers with a specific remote LU name.
- OREQ—Limits the history request to only those file transfers that contain this specific request number.
- OSDATE—Limits the history request to only those file transfers that were scheduled or completed on or after the start date and time.
- OSTIME—Limits the history request to only those file transfers that were scheduled or completed on or after the start date and time.

- OSYSID—Limits the history request to only those file transfers executed on the specified system ID.
- OSYSNAME—Limits the history request to only those file transfers executed on the specified system name.

  **Note:** OSYSNAME and OSYSID together uniquely identify a CA XCOM Data Transport r11.6 server
- OTNAME—Limits the history request to only those file transfers with a specific remote TCP/IP name or TCP/IP address.
- OTYPE—Specifies if the history request should include active transfer requests, inactive transfer requests, or completed transfers.
- OTYPEREQ—Limits the history request to only send transfers or only receive transfers.
- OUSER—Limits the history request to only those file transfers submitted by a specific user.

For more information about using the -c7 option, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

For more information about using the Export button, see the *CA XCOM Data Transport GUI Online Help*.

# Symbolic Variables

Symbolic variables let you store transfer parameters in control files with variable data that is resolved to other values at schedule time or at transfer time, depending on the specific parameter.

The following standard symbolic parameters are supplied with CA XCOM Data Transport:

- &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.
- &ID—Causes the value entered for ID to be substituted dynamically in the current keyword value.
- &IPNAME—Causes the value entered for IPNAME to be substituted dynamically in the current keyword value.
- &LU—Causes the value entered for LU to be substituted dynamically in the current keyword value.
- &LUSER—Causes the current local logged on user ID 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.
- &USERID—Causes the current remote user ID to be substituted dynamically in the current keyword value.

No setup is required to use the pre-defined variables that are integrated as part of CA XCOM Data Transport. If you place these variables in the parameter data set for a transfer, CA XCOM Data Transport resolves them when performing that transfer.

For more information, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# CA Easytrieve Reports

You can run CA Easytrieve reports that are prebuilt and supplied by the CA XCOM Data Transport product install. The input records for the reports are the common history records.

The use of CA Easytrieve reports is limited to the following platforms:

- CA XCOM Data Transport for Windows
- CA XCOM Data Transport for AIX
- CA XCOM Data Transport for AIX 64
- CA XCOM Data Transport for Linux s390x

For more information, see the sample code in the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# Transfer Control (XTC)

CA XCOM Data Transport Transfer Control (XTC) 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.

The following XTC parameters have been added:

- HOLD—Prevents a TYPE=SCHEDULE transfer from starting until explicitly released.
- 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.
- XTCERRDECR—Specifies the transfer requests for which the HOLDCOUNT parameter value is decremented when the current transfer completes unsuccessfully.
- XTCERRINCR—Specifies the transfer requests for which the HOLDCOUNT parameter value is incremented if the current file transfer fails.
- XTCERRPURGE—Specifies the transfer requests to be purged if the transfer concludes unsuccessfully.
- XTCERRREL—Specifies the transfers to be released if the current transfer completes unsuccessfully.
- XTCGOODDECR—Indicates an XTCNET job whose hold count decrements if the file transfer completes successfully.
- XTCGOODINCR—Indicates the transfer requests whose HOLDCOUNT parameter is incremented when the current file transfer completes successfully.
- XTCGOODPURGE—Specifies the transfer requests to be purged when the current file transfer completes successfully.
- XTCGOODREL—Specifies the transfer requests to be released if the current file transfer concludes successfully.
- XTCJOB—Defines the name of a transfer request belonging to the group of interdependent transfer requests named through the XTCNET parameter.
- XTCNET—Defines the name of a group of interdependent transfer requests.

For more information, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# SMTP Email Notification

You can now use SMTP instead of MAPI for email notification of transfer completion.

The following parameters have been introduced to support SMTP email notification:

- MAIL_TYPE—Specifies the type of MAIL server used for sending mail notifications.
- SMTP_SERVER —Specifies the name of the SMTP server.

For more information, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# Block Level I/O

CA XCOM Data Transport uses record level I/O to read and write records to/from disk. To improve read/write I/O times, the I/O routines now perform block level I/O whenever possible.

The following new parameters have been introduced to support block level I/O:

- CACHE_READ_SZ
- CACHE_WRITE_SZ

For more information, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# Encryption at Rest

CA XCOM Data Transport currently provides the ability to encrypt data during transmission (SSL). However, when the data reaches the receiving CA XCOM Data Transport system the resulting file is written in plain text. This release provides encryption at rest, which works as follows:

- The local CA XCOM Data Transport system reads an encrypted input file, decrypts the file, and transfers the data to the partner.
- The remote CA XCOM Data Transport system receives a data transfer and writes the output file as an encrypted file.

The following new local/remote pairs of parameters have been added for encryption at rest:

- LEAR_CIPHER/EAR_CIPHER—Indicates which encryption algorithm is to be used.
- LEAR_DIGEST/EAR_DIGEST—Indicates the digest value to be matched with the generated digest. You must have already generated a digest value on the input file.
- LEAR_HASH/EAR_HASH—Indicates which digest algorithm is to be used.
- LEAR_KEY/EAR_KEY—Indicates the symmetric encryption key to be used to encrypt/decrypt.

For more information about these parameters, see the *CA XCOM Data Transport for UNIX and Linux) User Guide*.

# Trusted Access Database

In r11.6, the CA XCOM Data Transport Trusted Access facility has been changed, so that it works the same way across platforms. The trusted database feature allows you to specify a database management system in which CA XCOM Data Transport is to store the trusted records. You can use either of the following products for this database management system:

- MySQL freeware
- IBM's DB2

Two relational database tables are used to store trusted user authentication information:

**XCOM_TRUSTED_SYS table**

The XCOM_TRUSTED_SYS table contains the name of the remote SYSTEM_NAME from which the trusted transfer will be initiated.

**XCOM_TRUSTED_USERS table**

The XCOM_TRUSTED_USERS table contains the remote SYSTEM_NAME and the USER_NAME.

For more information about the Trusted Access Database, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# PAM-Based Authentication

The Pluggable Authentication Modules (PAM) library is a generalized API for authentication-related services. CA XCOM Data Transport r11.5 introduced an option to select PAM-based authentication, such as SYSTEM or LDAP on a UNIX server.

## New Parameters

This section describes the parameters used for PAM-based authentication.

**AUTH_TYPE**

Specifies the type of authentication (PAM or SYSTEM) to be used for transfers.

**PAM_PATH**

Specifies the path to your PAM library for your current UNIX or Linux platform.

For more information about these parameters, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

# New Cipher Suites for SSL Transfers

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
```

You can use various other suites, depending on whether you are using TLSv1 or SSLv3.

For more information about using cipher suites, see the *CA XCOM Data Transport for UNIX and Linux User Guide*.

## Cipher Suites Supported in TLSv1 when FIPS_MODE=NO

The following cipher suites are supported in TLSv1 when FIPS_MODE=NO:

- DHE-RSA-AES256-SHA
- DHE-DSS-AES256-SHA
- AES256-SHA
- EDH-RSA-DES-CBC3-SHA
- EDH-DSS-DES-CBC3-SHA
- DES-CBC3-SHA
- DHE-RSA-AES128-SHA
- DHE-DSS-AES128-SHA
- AES128-SHA
- RC4-SHA
- RC4-MD5
- EDH-RSA-DES-CBC-SHA
- EDH-DSS-DES-CBC-SHA
- DES-CBC-SHA
- EXP-EDH-RSA-DES-CBC-SHA
- EXP-EDH-DSS-DES-CBC-SHA
- EXP-DES-CBC-SHA
- EXP-RC2-CBC-MD5
- EXP-RC4-MD5

## Cipher Suites Supported in TLSv1 when FIPS_MODE=YES

The following cipher suites are supported in TLSv1 when FIPS_MODE=YES:

- DHE-RSA-AES256-SHA
- DHE-DSS-AES256-SHA
- AES256-SHA
- EDH-RSA-DES-CBC3-SHA
- EDH-DSS-DES-CBC3-SHA
- DES-CBC3-SHA
- DHE-RSA-AES128-SHA
- DHE-DSS-AES128-SHA
- AES128-SHA

## Cipher Suites Supported in SSLv3

SSLv3 can be used only with FIPS_MODE=NO. It uses the same cipher suites as TLSv1 with FIPS_MODE=NO.

## User Interfaces Stabilized

CA XCOM Data Transport r11.5 introduced the CA XCOM Data Transport Graphical User Interface (GUI), providing a common user interface for CA XCOM Data Transport on the following platforms:

- Windows
- UNIX and Linux

The GUI provides access to all of the features of CA XCOM Data Transport, and the former user interfaces for these platforms are no longer necessary.

The following user interfaces are still included in CA XCOM Data Transport r11.5:

- xcomtool (for UNIX and Linux)

However, they have been stabilized, ready to be removed from a future release.

# Appendix A: Acknowledgements

The following license agreements are available in the \Bookshelf Files\TPSA folder in the CA Bookshelf:

Adaptive Communication Environment (ACE) 5.7.8

AIX JRE 1.6.0 SR11

Aleksey XML Security Library 1.2.9

Ant 1.7

ant-contrib 1.0b3

Apache Commons FileUpload 1.2.1

apache woden 1.0M8

AspectJ 1.7.1

Axiom 1.2.7

Axis2 1.4.1

Commons Codec 1.3

Commons Codec 1.4

Commons httpclient 3.1

Commons IO 1.4

Commons Logging 1.1.1

EXPAT 2.0.1

geronimo annotation 1.0 spec 1.1.1

geronimo stax-api 1.0.1

HP-UX JRE 6.0.02

httpclient 4.2

httpcore 4.0-alpha2

IBM 64 bit SDK for AIX, Java(TM) Technology Edition, Version 7.0 Redistributables

ICU4C 4.6.1

ICU4C 51.1

JAXB 2.1.12

JAXB 2.2.1

Jaxen 1.1

JRE v.1.6

Libcurl 7.21.1

libcurl 7.25.0

Libxml2 2.7.7

Libxml2 2.9.0

Libxml2 2.9.1

Libxml2 2.6.27

Log4cplus 1.0.2

log4cplus v.1.0.3

Log4j 1.2.16

MIT Kerberos V5 Release 1.11

myfaces 1.1.4

neethi 2.0.4

not-yet-commons-ssl 0.3.10

OpenLDAP 2.3.20

OpenSSL 0.9.8h

OpenSSL 1.0.1e

Oracle Java Runtime Environment (JRE)

pam_userpass 1.0.2

PCRE 6.3

PCRE 8.1

Pluggable Authentication Modules ( PAM ) Linux-PAM-1.1.3

Pluggable Authentication Modules ( PAM ) Linux-PAM-1.1.6

tomahawk 1.1.5

WSDL4J 1.6.2

Xerces-C 3.1.1

xercesImpl 2.8.1

XMLBeans 2.3.0

xml-commons xml-api 1.3.04

XMLSchema 1.4.7

Zlib 1.0.2

Zlib 1.2.3

Zlib 1.2.5

Zthread 2.3.2

# Index