iSeries

Security -- Network Security APIs

Version 5 Release 3

iSeries

Security -- Network Security APIs

Version 5 Release 3

Note

Before using this information and the product it supports, be sure to read the information in "Notices," on page 43.

Sixth Edition (August 2005)

This edition applies to version 5, release 3, modification 0 of Operating System/400 (product number 5722-SS1) and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CISC models.

Contents

Network Security APIs

The network security APIs include:

- "Server Authentication Entry APIs"
- "NetWare Authentication Entry APIs" on page 11

Security APIs | APIs by category

APIs

These are the APIs for this category.

Server Authentication Entry APIs

The Server Authentication Entry APIs can be used by a client requesting connection to a server.

The server authentication entry APIs are:

- "Add Server Authentication Entry (QsyAddServerEntry) API" on page 2 (QsyAddServerEntry) adds server authentication information for use by application requesters in connecting to application servers.
- "Change Server Authentication Entry (QsyChangeServerEntry) API" on page 4 (QsyChangeServerEntry) changes server authentication information for use by application requesters in connecting to application servers.
- "Remove Server Authentication Entry (QsyRemoveServerEntry) API" on page 6 (QsyRemoveServerEntry) removes server authentication information for use by application requesters in connecting to application servers.
- "Retrieve Server Authentication Entries (QSYRTVSE, QsyRetrieveServerEntries) API" on page 7 (QSYRTVSE, QsyRetrieveServerEntries) returns a list of server authentication entries for a user profile.

Top | Security APIs | APIs by category

Add Server Authentication Entry (QsyAddServerEntry) API

Syntax for QsyAddServerEntry:

```
#include <qsysvrfn.h>

void QsyAddServerEntry
       (char            User_profile[10],
        char           *Server_name,
        int             Length_of_server_name,
        char           *User_ID,
        int             Length_of_user_ID,
        char           *Password,
        int             Length_of_password,
        void           *Error_code);
```

Service Program: QSYSVRFN

Default Public Authority: *USE

Threadsafe: No

The Add Server Authentication Entry (QsyAddServerEntry) API adds server authentication information for use by application requesters in connecting to application servers.

When adding a server authentication entry for a Distributed Relational Database Architecture (DRDA) application that uses TCP/IP, the server name must be entered in upper case.

Authorities and Locks

If the user profile parameter is not *CURRENT or the user profile currently running, then the user profile that calls this API must have *SECADM special authority and *OBJMGT and *USE authorities to the user profile.

Required Parameter Group

User profile
INPUT; CHAR(10)

The user profile for which the server authentication entry will be added. The special value *CURRENT may be specified to add an entry for the user profile that calls this API.

Server name
INPUT; CHAR(*)

The name of the application server.

Length of server name
INPUT; BINARY(4)

The length of the server name that is specified in the server name parameter. The length of the server name must be a value from 1 to 200.

User ID
INPUT; CHAR(*)

The user name for which requests will be made to the implementation server.

Length of user ID
INPUT; BINARY(4)

The length of the user ID that is specified in the user ID parameter. The length of the user ID must be a value from 0 to 1000. If the length is 0, the user ID will be the same as the name that is specified in the user profile parameter.

Password
INPUT; CHAR(*)

The password to be used to authenticate the user when the client attempts to connect to the server.

Length of password
INPUT; BINARY(4)

The length of the password that is specified in the password parameter. The length of the password must be a value from 0 to 696. If the length is 0, then no password is supplied on the connection request. If the retain server security data (QRETSVRSEC) system value is set to 0 (do not retain data), then the length of the password is assumed to be 0.

Error code
I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Error Messages

Message ID	Error Message Text
CPF2204 E	User profile &1 not found.
CPF2213 E	Not able to allocate user profile &1.
CPF2222 E	Storage limit is greater than specified for user profile &1.
CPF224F E	Server authentication entry already exists.
CPF225F E	Not all information stored.
CPF226C E	Not authorized to perform function.
CPF3CF1 E	Error code parameter not valid.
CPF3CF2 E	Error(s) occurred during running of &1 API.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C90 E	Literal value cannot be changed.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.

API introduced: V4R2

Top | Security APIs | APIs by category

Change Server Authentication Entry (QsyChangeServerEntry) API

Syntax for QsyChangeServerEntry:

```
#include <qsysvrfn.h>

void QsyChangeServerEntry
       (char           User_profile[10],
        char          *Server_name,
        int            Length_of_server_name,
        char          *User_ID,
        int            Length_of_user_ID,
        char          *Password,
        int            Length_of_password,
        void          *Error_code);
```

Service Program: QSYSVRFN

Default Public Authority: *USE

Threadsafe: No

The Change Server Authentication Entry (QsyChangeServerEntry) API changes the server authentication information for use by application requesters in connecting to application servers.

Authorities and Locks

If the user profile parameter is not *CURRENT or the user profile currently running, then the user profile that calls this API must have *SECADM special authority and *OBJMGT and *USE authorities to the user profile.

Required Parameter Group

User profile
INPUT; CHAR(10)

The user profile for which the server authentication entry will be changed. The special value *CURRENT may be specified to change an entry for the user profile that calls this API.

Server name
INPUT; CHAR(*)

The name of the application server.

Length of server name
INPUT; BINARY(4)

The length of the server name that is specified in the server name parameter. The length of the server name must be a value from 1 to 200.

User ID
INPUT; CHAR(*)

The user name for which requests will be made to the implementation server.

Length of user ID
INPUT; BINARY(4)

The length of the user ID that is specified in the user ID parameter. The length of the user ID must be a value from -1 to 1000. If -1 is specified, the user ID value is not changed. If 0 is specified. the user ID will be the same as the name that is specified in the user profile parameter.

Password
INPUT; CHAR(*)

The password to be used to authenticate the user when the client attempts to connect to the server.

Length of password
INPUT; BINARY(4)

The length of the password that is specified in the password parameter. The length of the password must be a value from -1 to 696. If -1 is specified, the password value does not change. If 0 is specified, then no password is supplied on the connection request. If the retain server security data (QRETSVRSEC) system value is set to 0 (do not retain data), then the length of the password is assumed to be 0.

Error code
I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Error Messages

Message ID	Error Message Text
CPF2204 E	User profile &1 not found.
CPF2213 E	Not able to allocate user profile &1.
CPF225E E	Server authentication entry does not exist.
CPF225F E	Not all information stored.
CPF226C E	Not authorized to perform function.
CPF3CF1 E	Error code parameter not valid.
CPF3CF2 E	Error(s) occurred during running of &1 API.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C90 E	Literal value cannot be changed.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.

API introduced: V4R2

Top | Security APIs | APIs by category

Remove Server Authentication Entry (QsyRemoveServerEntry) API

Syntax for QsyRemoveServerEntry:

```
#include <qsysvrfn.h>

void QsyRemoveServerEntry(
             char     User_profile[10],
             char    *Server_name,
             int      Length_of_server_name,
             void    *Error_code);
```

Service Program: QSYSVRFN

Default Public Authority: *USE

Threadsafe: No

The Remove Server Authentication Entry (QsyRemoveServerEntry) API removes server authentication information for use by application requesters in connecting to application servers.

Authorities and Locks

If the user profile parameter is not *CURRENT or the user profile currently running, then the user profile that calls this API must have *SECADM special authority and *OBJMGT and *USE authorities to the user profile.

Required Parameter Group

User profile
INPUT; CHAR(10)

The user profile for which the server authentication entry will be removed. The special value *CURRENT may be specified to remove an entry for the user profile that calls this API.

Server name
INPUT; CHAR(*)

The name of the application server. The special value *ALL may be specified to indicate that all server authentication entries for the user profile that is specified in the user profile parameter are to be removed.

Length of server name
INPUT; BINARY(4)

The length of the server name that is specified in the server name parameter. The length of the server name must be a value from 1 to 200.

Error code
I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Error Messages

Message ID	Error Message Text
CPF2204 E	User profile &1 not found.
CPF2213 E	Not able to allocate user profile &1.
CPF225E E	Server authentication entry does not exist.
CPF226C E	Not authorized to perform function.
CPF3CF1 E	Error code parameter not valid.
CPF3CF2 E	Error(s) occurred during running of &1 API.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C90 E	Literal value cannot be changed.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.

API introduced: V4R2

Top | Security APIs | APIs by category

Retrieve Server Authentication Entries (QSYRTVSE, QsyRetrieveServerEntries) API

»

Required Parameter Group for QSYRTVSE:

1	Receiver variable
Output	Char(*)
2	Length of receiver variable
Input	Binary(4)
3	Return records feedback information
Output	Char(12)
4	Format name
Input	Char(8)
5	Starting server name
Input	Char(*)
6	Length of starting server name
Input	Binary(4)
7	Starting server option
Input	Char(1)
8	User profile
Input	Char(10)
9	Error code
I/O	Char(*)

Default Public Authority: *USE

Threadsafe: No

Syntax for QsyRetrieveServerEntries:

```
#include <qsysvrfn.h>

void QsyRetrieveServerEntries
       (void                        *Receiver_variable,
        int                          Length_of_receiver_variable,
        Qsy_RSVRE_Feedback_Info_T   *Return_records_feedback_information,
        char                        *Format_name,
        char                        *Starting_server_name,
        int                          Length_of_starting_server_name,
        char                         Starting_server_option,
        char                         User_profile[10],
        void                        *Error_code);
```

Service Program: QSYSVRFN

Default Public Authority: *USE

Threadsafe: No

«

The Retrieve Server Authentication Entries (OPM, QSYRTVSE; ILE, QsyRetrieveServerEntries) API returns a list of server authentication entries for a user profile.

Authorities and Locks

User Profile Authority
*READ

Required Parameter Group

Receiver variable
OUTPUT; CHAR(*)

The receiver variable that receives the information requested. You can specify the size of the area to be smaller than the format requested as long as you specify the length parameter correctly. As a result, the API returns only the data that the area can hold.

Length of receiver variable
INPUT; BINARY(4)

The length of the receiver variable provided. The length of receiver variable parameter may be specified up to the size of the receiver variable specified in the user program. If the length of receiver variable parameter specified is larger than the allocated size of the receiver variable specified in the user program, the results are not predictable.

Returned records feedback information
OUTPUT; CHAR(12)

Information about the entries that are returned in the receiver variable.

See "Format of Returned Records Feedback Information" on page 10 for details.

Format name
INPUT; CHAR(8)

The name of the format that is used to retrieve server authentication entries for the user profile.

You can specify this format:

SVRE0100	For a detailed description of this format, see "SVRE0100 Format."

Starting server name
INPUT; CHAR(*)

The server name at which to start listing server authentication entries. The server authentication entries are listed in hexadecimal sort sequence by server name.

Possible values follow:

**FIRST*	Server authentication entries are returned starting with the server that has the smallest hexadecimal value.
server name	If an exact match for the starting server name is found, the starting server option parameter indicates whether that server authentication entry is returned. If an exact match for the starting server name is not found, the listing begins with the first existing server authentication entry for the server name whose hexadecimal value would follow the hexadecimal value of the specified starting server name.

Length of starting server name
INPUT; BINARY(4)

The length of the starting server name. The length of the starting server name may be from 1 to 200.

Starting server option
INPUT; CHAR(1)

This parameter indicates whether the starting server authentication entry is returned when an exact match for the starting server name is found.

Possible values follow:

0	Server authentication entries for server names whose hexadecimal value is greater than the hexadecimal value for the starting server name are returned.
1	Server authentication entries for server names whose hexadecimal value is equal to or greater than the hexadecimal value for the starting server name are returned.

User profile
INPUT; CHAR(10)

The name of the user profile for which the server authentication entries are returned.

Error code
I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Receiver Variable Description

The following tables describe the order and format of the data returned in the receiver variable. For detailed descriptions of the fields in the tables, see "Field Descriptions" on page 10.

SVRE0100 Format

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Length of entry

Offset			
Dec	Hex	Type	Field
4	4	BINARY(4)	Length of server name
8	8	BINARY(4)	CCSID of server name
12	C	CHAR(200)	Server name
212	D4	BINARY(4)	Displacement to user ID
216	D8	BINARY(4)	Length of user ID
220	DC	BINARY(4)	CCSID of user ID
224	E0	CHAR(1)	Password stored indicator
		CHAR(*)	User ID

Format of Returned Records Feedback Information

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Bytes returned
4	4	BINARY(4)	Bytes available
8	8	BINARY(4)	Number of server authentication entries

Field Descriptions

Bytes available. The number of bytes of data available to be returned to the user in the receiver variable. If all data is returned, bytes available is the same as the number of bytes returned. If the receiver variable was not large enough to contain all of the data, this value is estimated based on the total number of server authentication entries for the user profile and the format specified.

Bytes returned. The number of bytes of data returned to the user in the receiver variable. This is the lesser of the number of bytes available to be returned or the length of the receiver variable.

CCSID of server name. The CCSID of the server name. This will be the default job CCSID of the job that added the server authentication entry.

CCSID of user ID. The CCSID of the user ID. This will be the default job CCSID of the job that last changed the user ID field in the server authentication entry.

Displacement to user ID. The displacement in the entry to the start of the user ID.

Format name. The name of the format that is used to return server authentication entries for a user profile.

Length of entry. The length (in bytes) of the current entry. This length can be used to access the next entry.

Length of server name ID. The length (in bytes) of the server name.

Length of user ID. The length (in bytes) of the user ID.

Number of server authentication entries. The number of complete entries returned in the list of server authentication entries. A value of zero is returned if the list is empty.

Password specified. Indicates whether the server authentication entry has a password associated with it.

Possible values follow:

0	The server authentication entry does not have a password associated with it.
1	The server authentication entry does have a password associated with it.

Server name. The name of the server that the entry is for.

User ID. The user ID that is used on requests to the server.

User profile. The name of the user profile for which the list of server authentication entries is returned.

Error Messages

Message ID	Error Message Text
CPFA0AA E	Error occurred while attempting to obtain space.
CPF2204 E	User profile &1 not found.
CPF2213 E	Not able to allocate user profile &1.
CPF2217 E	Not authorized to user profile &1.
CPF2222 E	Storage limit is greater than specified for user profile &1.
CPF3CF1 E	Error code parameter not valid.
CPF3CF2 E	Error(s) occurred during running of &1 API.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C21 E	Format name &1 is not valid.
CPF3C3C E	Value for parameter &1 not valid.
CPF3C90 E	Literal value cannot be changed.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.

API introduced: V4R2

Top | Security APIs | APIs by category

NetWare Authentication Entry APIs

The NetWare Authentication Entry APIs provide a means that automatically logs you on to a server when you request a NetWare function (for example, the QNETWARE file system or NetWare administration commands). You can create authentication entries for each NetWare Directory Services (NDS) tree or NetWare 3.x server to which you are authorized. The entry identifies the tree or server, your name on that server, and (optionally) your password. The system saves the authentication entries as part of the user profile. When you request a NetWare function, the system attempts to start a connection to the server by using the data stored in the authentication entries.

Alternatively, you can explicitly start and end a connection to a server, such as when the system administrator has disabled storing passwords through the Retain Server Security Data (QRETSVRSEC) system value.

Note: To use these APIs, you need NetWare on iSeries(TM). See Netware on iSeries in the Information Center for more information on this topic.

The NetWare Authentication Entry APIs are:

- “Add NetWare Authentication Entry (QfpzAddNtwAutE) API” on page 12 (QfpzAddNtwAutE) stores user authentication information that is used to access the specified server.

- “Change NetWare Authentication Entry (QfpzChgNtwAutE) API” on page 17 (QfpzChgNtwAutE) changes the authentication information in the specified authentication entry.
- “End NetWare Connection (QfpzEndNtwCnn) API” on page 22 (QfpzEndNtwCnn) ends a connection to a NetWare server.
- “List NetWare Authentication Entries (QfpzListNtwAutE) API” on page 24 (QfpzListNtwAutE) returns a list of authentication entries in a user profile.
- “Remove NetWare Authentication Entry (QfpzRmvNtwAutE) API” on page 31 (QfpzRmvNtwAutE) removes an authentication entry from the user profile.
- “Start NetWare Connection (QfpzStrNtwCnn) API” on page 34 (QfpzStrNtwCnn) establishes the caller as an authenticated user of the specified server and starts the connection with the server.
- “Verify NetWare Authentication Entry (QfpzVfyNtwAutE) API” on page 39 (QfpzVfyNtwAutE) returns a list of authentication entries for the specified user profile.

Top | Security APIs | APIs by category

Add NetWare Authentication Entry (QfpzAddNtwAutE) API

Required Parameter Group:

1	Entry identifier data
Input	Char(*)
2	Length of entry identifier
Input	Binary(4)
3	Entry identifier format name
Input	Char(8)
4	Authentication entry data
Input	Char(*)
5	Length of authentication entry
Input	Binary(4)
6	Entry data format name
Input	Char(8)
7	Error code
I/O	Char(*)

Default Public Authority: *USE

Library Name/Service Program: QFPNTWE/QFPZAAPI

Threadsafe: No

The Add NetWare Authentication Entry (QfpzAddNtwAutE) API stores user authentication information that is used to access the specified server. This information can be used at a later time to start an authenticated connection to the server without requiring the user to enter the data.

Authorities and Locks

User Profile Authority

The user profile must be the current user profile, or the caller must have *USE and *OBJMGT authority to the user profile and *SECADM special authority.

Required Parameter Group

Entry identifier data

INPUT; CHAR(*)

The authentication entry to be added. The content and format of this structure are determined by the format name. See “Format of Authentication Entry Identifier” on page 14 for a description of these formats.

Length of entry identifier

INPUT; BINARY(4)

The length of the entry identifier data structure.

Entry identifier format name

INPUT; CHAR(8)

The content and format of the authentication entry identifier data.

The possible format names follow:

“AUTE0100 Format” on page 14	NetWare Version 3.x server authentication entry identifier
“AUTE0200 Format” on page 14	NetWare Directory Services tree authentication entry identifier

See “Format of Authentication Entry Identifier” on page 14 for a description of these formats.

Authentication entry data

INPUT; CHAR(*)

The authentication entry to be added. The content and format of this structure are determined by the format name. See “Format of Authentication Entry Data” on page 14 for a description of these formats.

Length of authentication entry

INPUT; BINARY(4)

The length of the authentication entry data structure.

Entry data format name

INPUT; CHAR(8)

The content and format of the authentication entry input data. Each format corresponds to a type of authentication entry.

The possible format names follow:

“AUTD0100 Format” on page 14	NetWare Version 3.x server authentication entry data
“AUTD0200 Format” on page 15	NetWare Directory Services tree authentication entry data

See “Format of Authentication Entry Data” for a description of these formats.

Error code
I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Format of Authentication Entry Identifier

For details about the format of the authentication entries, see the following sections. For details about the fields in each format, see “Field Descriptions” on page 15.

AUTE0100 Format

This format is used to identify a NetWare Version 3.x server authentication entry.

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to server name
4	4	BINARY(4)	Length of server name
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	Server name

AUTE0200 Format

This format is used to identify a NetWare Directory Services tree authentication entry.

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to NDS tree name
4	4	BINARY(4)	Length of NDS tree name
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	NDS tree name

Format of Authentication Entry Data

For details about the format of the authentication entries, see the following sections. For details about the fields in each format, see “Field Descriptions” on page 15.

AUTD0100 Format

This format is used to specify the authentication entry data for a NetWare Version 3.x server authentication entry. This format must be used when entry identifier format AUTE0100 is used.

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to NetWare user name
4	4	BINARY(4)	Length of NetWare user name
8	8	BINARY(4)	Offset to password

Offset			
Dec	Hex	Type	Field
12	C	BINARY(4)	Length of password
16	10	BINARY(4)	Reserved
		CHAR(*)	NetWare user name
		CHAR(*)	Password

AUTD0200 Format

This format is used to specify the authentication entry data for a NetWare Directory Services tree authentication entry. This format must be used when entry identifier format AUTE0200 is used.

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Offset to NDS context
4	4	BINARY(4)	Length of NDS context
8	8	BINARY(4)	Offset to NetWare user name
12	C	BINARY(4)	Length of NetWare user name
16	10	BINARY(4)	Offset to password
20	14	BINARY(4)	Length of password
24	18	BINARY(4)	Reserved
		CHAR(*)	NDS context
		CHAR(*)	NetWare user name
		CHAR(*)	Password

Field Descriptions

Length of NDS context. The length, in bytes, of the NDS context.

Length of NDS tree name. The length, in bytes, of the NDS tree name.

Length of NetWare user name. The length, in bytes, of the NetWare user name.

Length of password. The length, in bytes, of the password.

Length of server name. The length, in bytes, of the server name.

NDS context. The directory context in which the user is defined.

NDS tree name. For NDS trees, the name of the tree to which the authentication entry applies.

NetWare user name. The NetWare user name that is used to authenticate the user to the server.

The following special value may be used:

**USRPRF*	The NetWare user name is the same as the user profile name.

Offset to NDS context. The offset, in bytes, from the start of the input data area to the NDS context.

Offset to NDS tree name. The offset, in bytes, from the start of the input data area to the NDS tree name.

Offset to NetWare user name. The offset, in bytes, from the start of the input data area to the user name.

Offset to password. The offset, in bytes, from the start of the input data area to the password.

Offset to server name. The offset, in bytes, from the start of the input data area to the server name.

Password. The password that is used to authenticate the user to the server.

The following special values may be used:

**NONE*	The system does not need password information to authenticate the user.
**STRNTWCNN*	The system does not store password information in the authentication entry. Use the Start NetWare Connection (STRNTWCNN) command, with the correct password, to start a connection to a server.

Reserved. Set this field to binary zeros.

Server name. The name of the server to which the authentication entry applies.

User profile name. The name of the user profile to which the authentication entry is to be added.

You can use the following special values for the user profile name:

**CURRENT*	The current user profile.

Error Messages

Message ID	Error Message Text
CPF2204 E	User profile &1 not found.
CPF2217 E	Not authorized to user profile &1.
CPF24B4 E	Severe error while addressing parameter list.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C21 E	Format name &1 is not valid.
CPF3C90 E	Literal value cannot be changed.
CPF3CF1 E	Error code parameter not valid.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.
FPE0211 E	Parameter length &1 not valid for field &2.
FPE0212 E	Field offset and length not within data.
FPE0216 E	Authentication entry already exists for &2.
FPE021F E	Data format &1 not valid with identifier format &2.
FPE0255 E	PASSWORD(*STRNTWCNN) required when QRETSVRSEC is 0.

API introduced: V3R7

Top | Security APIs | APIs by category

Change NetWare Authentication Entry (QfpzChgNtwAutE) API

Required Parameter Group:

1	Entry identifier data
Input	Char(*)
2	Length of entry identifier
Input	Binary(4)
3	Entry identifier format name
Input	Char(8)
4	Authentication entry data
Input	Char(*)
5	Length of authentication entry
Input	Binary(4)
6	Entry data format name
Input	Char(8)
7	Error code
I/O	Char(*)

Default Public Authority: *USE

Library Name/Service Program: QFPNTWE/QFPZAAPI

Threadsafe: No

The Change NetWare Authentication Entry (QfpzChgNtwAutE) API changes the authentication information in the specified authentication entry. The format that is specified for the authentication entry must correspond to the server type of the existing entry.

Authorities and Locks

User Profile Authority

The user profile must be the current user profile, or the caller must have *USE and *OBJMGT authority to the user profile and *SECADM special authority.

Required Parameter Group

Entry identifier data

INPUT; CHAR(*)

The authentication entry to be changed. The content and format of this structure are determined by the format name. See "Format of Authentication Entry Identifier" on page 18 for a description of these formats.

Length of entry identifier

INPUT; BINARY(4)

The length of the entry identifier data structure.

Entry identifier format name

INPUT; CHAR(8)

The content and format of the authentication entry identifier data.

The possible format names follow:

“AUTE0100 Format”	NetWare Version 3.x server authentication entry identifier
“AUTE0200 Format” on page 19	NetWare Directory Services tree authentication entry identifier

See “Format of Authentication Entry Identifier” for a description of these formats.

Authentication entry data

INPUT; CHAR(*)

The changed authentication entry data. The content and format of this structure are determined by the format name. See “Format of Authentication Entry Data” on page 19 for a description of these formats.

Length of authentication entry

INPUT; BINARY(4)

The length of the authentication entry data structure.

Entry data format name

INPUT; CHAR(8)

The content and format of the authentication entry input data. Each format corresponds to a type of authentication entry.

The possible format names follow:

“AUTD0100 Format” on page 19	NetWare Version 3.x server authentication entry data
“AUTD0200 Format” on page 19	NetWare Directory Services tree authentication entry data

See “Format of Authentication Entry Data” on page 19 for a description of these formats.

Error code

I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Format of Authentication Entry Identifier

For details about the format of the entry identifier data, see the following sections. For details about the fields in each format, see “Field Descriptions” on page 20.

AUTE0100 Format

This format is used to identify a NetWare Version 3.x server authentication entry.

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to server name

Offset			
Dec	Hex	Type	Field
4	4	BINARY(4)	Length of server name
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	Server name

AUTE0200 Format

This format is used to identify a NetWare Directory Services tree authentication entry.

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Offset to NDS tree name
4	4	BINARY(4)	Length of NDS tree name
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	NDS tree name

Format of Authentication Entry Data

For details about the format of the authentication entries, see the following sections. For details about the fields in each format, see "Field Descriptions" on page 20.

AUTD0100 Format

This format is used to specify the authentication entry data for a NetWare Version 3.x server authentication entry. This format must be used when entry identifier format AUTE0100 is used.

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Offset to NetWare user name
4	4	BINARY(4)	Length of NetWare user name
8	8	BINARY(4)	Offset to password
12	C	BINARY(4)	Length of password
16	10	BINARY(4)	Reserved
		CHAR(*)	NetWare user name
		CHAR(*)	Password

AUTD0200 Format

This format is used to specify the authentication entry data for a NetWare Directory Services tree authentication entry. This format must be used when entry identifier format AUTE0200 is used.

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Offset to NDS context
4	4	BINARY(4)	Length of NDS context
8	8	BINARY(4)	Offset to NetWare user name
12	C	BINARY(4)	Length of NetWare user name
16	10	BINARY(4)	Offset to password
20	14	BINARY(4)	Length of password
24	18	BINARY(4)	Reserved
		CHAR(*)	NDS context
		CHAR(*)	NetWare user name
		CHAR(*)	Password

Field Descriptions

Length of NDS context. The length, in bytes, of the NDS context.

Length of NDS tree name. The length, in bytes, of the NDS tree name.

Length of NetWare user name. The length, in bytes, of the NetWare user name.

Length of password. The length, in bytes, of the password.

Length of server name. The length, in bytes, of the server name.

NDS context. For NDS trees, the directory context in which the user is defined.

The following special value may be used:

**SAME*	The value remains the same.

NDS tree name. For NDS trees, the name of the directory tree to which the authentication entry applies.

The selected NDS tree name must be the same as the NDS tree name of an existing NetWare Directory Services tree authentication entry.

NetWare user name. The NetWare user name that is used to authenticate the user to the server.

The following special values may be used:

**USRPRF*	The NetWare user name is the same as the user profile name.
**SAME*	The value remains the same.

Offset to NDS context. The offset, in bytes, from the start of the input data area to the NDS context.

Offset to NDS tree name. The offset, in bytes, from the start of the input data area to the NDS tree name.

Offset to NetWare user name. The offset, in bytes, from the start of the input data area to the NetWare user name

Offset to password. The offset, in bytes, from the start of the input data area to the password.

Offset to server name. The offset, in bytes, from the start of the input data area to the server name.

Password. The password that is used to authenticate the user to the server.

The following special values may be used:

**NONE*	The system does not need password information to authenticate the user.
**STRNTWCNN*	The system does not store the password information in the authentication entry. Use Start NetWare Connection (STRNTWCNN) command, with the correct password, to start a connection to a server.
**SAME*	The value remains the same.

Reserved. Set this field to binary zeros.

Server name. The name of the server to which the authentication entry applies.

The selected server name must be the same as the server name of an existing NetWare Version 3.x server authentication entry.

User profile name. The name of the user profile that contains the authentication entry.

You can use the following special value for the user profile name:

**CURRENT*	Use the current user profile.

Error Messages

Message ID	Error Message Text
CPF2204 E	User profile &1 not found.
CPF2217 E	Not authorized to user profile &1.
CPF24B4 E	Severe error while addressing parameter list.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C21 E	Format name &1 is not valid.
CPF3C90 E	Literal value cannot be changed.
CPF3CF1 E	Error code parameter not valid.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.
FPE0211 E	Parameter length &1 not valid for field &2.
FPE0212 E	Field offset and length not within data.
FPE0215 E	Could not find authentication entry for &1.
FPE021F E	Data format &1 not valid with identifier format &2.
FPE0255 E	PASSWORD(*STRNTWCNN) required when QRETSVRSEC is 0.

API introduced: V3R7

Top | Security APIs | APIs by category

End NetWare Connection (QfpzEndNtwCnn) API

Required Parameter Group:

1	Input data
Input	Char(*)
2	Length of input data
Input	Binary(4)
3	Format name
Input	Char(8)
4	Error code
I/O	Char(*)

Default Public Authority: *USE

Library Name/Service Program: QFPNTWE/QFPZAAPI

Threadsafe: No

The End NetWare Connection (QfpzEndNtwCnn) API ends a connection to a NetWare server. Connections to a server that was started from this iSeries server system, other iSeries servers, or workstations can be ended. Use the Work with NetWare Connections (WRKNTWCNN) command to view a list of connections.

Authorities and Locks

Special Authority

The connection must have been started by the current job, or the caller must have job control (*JOBCTL) special authority.

NetWare Authority

If this iSeries server did not start the connection being ended, the caller must have operator privileges for the server.

Note: To have operator privileges, the caller must have an authenticated connection to the server. The caller must be a NetWare user who is a member of the OPERATORS property for the server.

Required Parameter Group

Input data

INPUT; CHAR(*)

The input data that identifies the connections to be ended. The format name determines the format and content of this data. For detailed descriptions of the data’s format and content, see “Input Data Formats” on page 23. For detailed descriptions of the fields in this structure, see “Field Descriptions” on page 23.

Length of input data

INPUT; BINARY(4)

The length of the input data structure.

Format name

INPUT; CHAR(8)

The format and content of the input data structure.

The following formats are supported:

"ENDC0100 Format"	End connections for the current job on the specified servers.
"ENDC0200 Format"	End the specified connection number on the specified server.

Error code

I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Input Data Formats

ENDC0100 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to server name
4	4	BINARY(4)	Length of server name
8	8	BINARY(4)	Reserved
		CHAR(*)	Server name

ENDC0200 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to server name
4	4	BINARY(4)	Length of server name
8	8	BINARY(4)	Connection number
12	C	BINARY(4)	Reserved
		CHAR(*)	Server name

Field Descriptions

Connection number. The connection number by which the connection is known on the server.

Length of server name. The length, in bytes, of the server name.

Offset to server name. The offset, in bytes, from the start of the input data to the server name.

Reserved. Set this field to binary zeros.

Server name. The name of the server. If format ENDC0100 is specified, the following special value may be used:

*ALL End connections to all servers.

Error Messages

Message ID	Error Message Text
CPF24B4 E	Severe error while addressing parameter list.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C21 E	Format name &1 is not valid.
CPF3C90 E	Literal value cannot be changed.
CPF3CF1 E	Error code parameter not valid.
CPF90FF E	*JOBCTL special authority required to do requested operation.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.
FPE0211 E	Parameter length &1 not valid for field &2.
FPE0212 E	Field offset and length not within data.
FPE021B E	Connection &1 on server &2 not found.
FPE0230 E	Connection type -1 must be single value.
FPE0231 E	Value &1 not valid for connection type.
FPE0232 E	User &1 not connected to server.
FPE023F E	Server operator privileges required.

API introduced: V3R7

Top | Security APIs | APIs by category

List NetWare Authentication Entries (QfpzListNtwAutE) API

Required Parameter Group:

1	Qualified user space name
Input	Char(20)
2	Entry identifier data
Input	Char(*)
3	Length of entry qualifier
Input	Binary(4)
4	Entry identifier format name
Input	Char(8)
5	Error code
I/O	Char(*)

Default Public Authority: *USE

Library Name/Service Program: QFPNTWE/QFPZAAPI

Threadsafe: No

The List NetWare Authentication Entries (QfpzListNtwAutE) API returns a list of authentication entries in a user profile. All entries may be returned, or only those entries that match specified criteria can be requested.

Authorities and Locks

User Profile Authority
The user profile must be the current user profile, or the caller must have *USE and *OBJMGT authority to the user profile and *SECADM special authority.

User Space Authority
*CHANGE

User Space Library Authority
*USE

User Space Lock
*EXCLRD

Required Parameter Group

Qualified user space name
INPUT; CHAR(20)

The user space that receives the information, and the library in which it is located. The first 10 characters contain the user space name, and the second 10 characters contain the library name.

You can use these special values for the library name:

**CURLIB*	The job's current library
**LIBL*	The library list

Entry identifier data
INPUT; CHAR(*)

The authentication entries to be retrieved. The content and format of this structure are determined by the format name. See "Format of Authentication Entry Identifier" on page 26 for a description of these formats.

Length of entry identifier
INPUT; BINARY(4)

The length of the authentication entry identifier structure.

Entry identifier format name
INPUT; CHAR(8)

The content and format of the authentication entry identifier data. Each format corresponds to a method of identifying an authentication entry.

The possible format names follow:

"AUTE0100 Format" on page 26	The NetWare Version 3.x server entry is identified by the server name.
"AUTE0200 Format" on page 26	The NetWare Directory Services tree entry is identified by the NDS tree name.

“AUTE0900 Format”	All authentication entries in a user profile are retrieved.

Error code
I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Format of Authentication Entry Identifier

For details about the format of the entry identifier data, see the following sections. For details about the fields in each format, see “Field Descriptions” on page 28.

AUTE0100 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to selected server name
4	4	BINARY(4)	Length of selected server name
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	Selected server name

AUTE0200 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to selected NDS tree name
4	4	BINARY(4)	Length of selected NDS tree name
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	Selected NDS tree name

AUTE0900 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	User profile name
10	A	CHAR(6)	Reserved

Format of Authentication Entry Lists

The authentication entry list consists of:

- A user area
- A generic area

- An input parameter section
- A header section
- A list data section:
  - Server authentication entry

For details about the user area and generic header, see User Space Format for List APIs. For details about the remaining items, see the following sections. For detailed descriptions of the fields in the list that is returned, see "Field Descriptions" on page 28.

When you retrieve list entry information from a user space, do not use the entry size that is returned in the generic header. Instead, use the displacement to next entry field that is returned in each list entry. If you do not use the displacement to next entry field, the results may not be valid. For examples of how to process lists, see Examples.

Input Parameter Section

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	User space name specified
10	A	CHAR(10)	User space library name specified
20	14	CHAR(8)	Format name specified
28	1C	BINARY(4)	Offset to identifier data specified
32	20	BINARY(4)	Length of identifier data specified
		CHAR(*)	Identifier data specified

Header Section

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	User space name used
10	A	CHAR(10)	User space library name used
20	14	CHAR(10)	User profile name used

Server Authentication Entry

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Displacement to next entry
4	4	BINARY(4)	Displacement to server specific data
8	8	CHAR(10)	Server type
18	12	CHAR(2)	Reserved
		CHAR(*)	Server-type specific data

NetWare 3.x Server Specific Data

If the server type is *NETWARE3, the format of the server-type specific data is as follows:

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Displacement to server name
4	4	BINARY(4)	Length of server name
8	8	BINARY(4)	Displacement to NetWare user name
12	C	BINARY(4)	Length of NetWare user name
		CHAR(*)	Server name
		CHAR(*)	NetWare user name

NetWare Directory Services Tree Specific Data

If the server type is *NDS, the format of the server-type specific data is as follows:

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Displacement to NDS tree name
4	4	BINARY(4)	Length of NDS tree name
8	8	BINARY(4)	Displacement to NDS context
12	C	BINARY(4)	Length of NDS context
16	10	BINARY(4)	Displacement to NetWare user name
20	14	BINARY(4)	Length of NetWare user name
		CHAR(*)	NDS tree name
		CHAR(*)	NDS context name
		CHAR(*)	NetWare user name

Field Descriptions

Displacement to NDS context. The displacement, in bytes, from the start of the list entry to the NDS context.

Displacement to NDS tree name. The displacement, in bytes, from the start of the list entry to the NDS tree name.

Displacement to NetWare user name. The displacement, in bytes, from the start of the list entry to the NetWare user name.

Displacement to next entry. The displacement, in bytes, from the start of the current list entry to the start of the next entry in the list.

Displacement to server name. The displacement, in bytes, from the start of the list entry to the server name.

Displacement to server specific data. The displacement, in bytes, from the start of the list entry to the server specific data.

Format name specified. The entry identifier format name that the caller of this API specifies.

Identifier data specified. The authentication entry identifier data that is specified when this API is called.

Length of identifier data specified. The length, in bytes, of the identifier data that is specified on the call to this API.

Length of NDS context. The length, in bytes, of the NDS context.

Length of NDS tree name. The length, in bytes, of the NDS tree name.

Length of NetWare user name. The length, in bytes, of the NetWare user name.

Length of selected NDS tree name. The length, in bytes, of the selected NDS tree name.

Length of selected server name. The length, in bytes, of the selected server name.

Length of server name. The length, in bytes, of the server name.

NDS context name. For NDS trees, the directory context in which the user is defined.

NDS tree name. For NDS trees, the name of the directory tree to which the authentication entry applies.

NetWare user name. The user name for which requests are made to the server.

Offset to identifier data specified. The offset, in bytes, from the start of the input parameter header to the identifier data that is specified on the call to this API.

Offset to selected NDS tree name. The offset, in bytes, from the start of the identifier data to the selected NDS tree name.

Offset to selected server name. The offset, in bytes, from the start of the identifier data to the selected server name.

Reserved. Set this field to binary zeros.

Selected NDS tree name. The name of the NDS tree for which authentication entries are to be listed. A generic name can be used.

The following special value may be specified:

*ALL All NDS tree authentication entries.

Selected server name. The name of the server for which authentication entries are to be listed. A generic name can be used.

The following special value may be specified:

*ALL All NetWare 3.x server authentication entries.

Server name. The name of the server to which the authentication entry applies.

Server type. The kind of server. The content and format of the server-specific data areas are dependent on the server type.

The following values may be returned:

**NETWARE3*	The entry is for a NetWare Version 3.x server.
**NDS*	The entry is for a NetWare Directory Services tree.

Server-type specific data. The name of the server and the user name that are used to start a connection to the server. The format and content of this data are dependent on the server type.

User profile name. The name of the user profile for which entries are to be listed.

The following special value may be specified:

**CURRENT*	The current job's user profile.

User profile name used. The actual user profile name that is used for the authentication entries listed.

User space library name specified. The name that is specified for the library that contains the user space to receive the generated list.

User space library name used. The actual name of the library that is used to contain the user space that received the list.

User space name specified. The name that is specified for the user space that is to receive the generated list.

User space name used. The actual name of the user space that received the list.

Error Messages

Message ID	Error Message Text
CPF2204 E	User profile &1 not found.
CPF2217 E	Not authorized to user profile &1.
CPF24B4 E	Severe error while addressing parameter list.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C21 E	Format name &1 is not valid.
CPF3C90 E	Literal value cannot be changed.
CPF3CAA E	List is too large for user space &1.
CPF3CF1 E	Error code parameter not valid.
CPF9801 E	Object &2 in library &3 not found.
CPF9802 E	Not authorized to object &2 in &3.
CPF9803 E	Cannot allocate object &2 in library &3.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.
FPE0211 E	Parameter length &1 not valid for field &2.
FPE0212 E	Field offset and length not within data.
FPE0213 E	Special value &1 not valid for field &2.

API introduced: V3R7

Top | Security APIs | APIs by category

Remove NetWare Authentication Entry (QfpzRmvNtwAutE) API

Required Parameter Group:

1	Entry identifier data
Input	Char(*)
2	Length of entry identifier
Input	Binary(4)
3	Entry identifier format name
Input	Char(8)
4	Error code
I/O	Char(*)

Default Public Authority: *USE

Library Name/Service Program: QFPNTWE/QFPZAAPI

Threadsafe: No

The Remove NetWare Authentication Entry (QfpzRmvNtwAutE) API removes an authentication entry from the user profile.

Authorities and Locks

User Profile Authority

The user profile must be the current user profile, or the caller must have *USE and *OBJMGT authority to the user profile and *SECADM special authority.

Required Parameter Group

Entry identifier data

INPUT; CHAR(*)

The authentication entry to be removed. The content and format of this structure is determined by the format name. See “Format of Authentication Entry Identifier” on page 32 for a description of these formats.

Length of entry identifier

INPUT; BINARY(4)

The length of the authentication entry identifier structure.

Entry identifier format name

INPUT; CHAR(8)

The content and format of the authentication entry identifier data. Each format corresponds to a method of identifying an authentication entry.

The possible format names follow:

"AUTE0100 Format"	The NetWare Version 3.x server entry is identified by the server name.
"AUTE0200 Format"	The NetWare Directory Services tree entry is identified by the NDS tree name.
"AUTE0900 Format"	All authentication entries are removed from a user profile.

Error code

I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Format of Authentication Entry Identifier

For details about the format of the entry identifier data, see the following sections. For details about the fields in each format, see "Field Descriptions" on page 33.

AUTE0100 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to server name
4	4	BINARY(4)	Length of server name
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	Server name

AUTE0200 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to NDS tree name
4	4	BINARY(4)	Length of NDS tree name
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	NDS tree name

AUTE0900 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	CHAR(10)	User profile name
10	A	CHAR(6)	Reserved

Field Descriptions

Length of NDS tree name. The length, in bytes, of the NDS tree name.

Length of server name. The length, in bytes, of the server name.

NDS tree name. For *NDS tree entries, the name of the directory tree.

Offset to NDS tree name. The offset, in bytes, from the start of the input data area to the NDS tree name.

Offset to server name. The offset, in bytes, from the start of the input data area to the server name.

Reserved. Set this field to binary zeros.

Server name. For *NETWARE3 entries, the name of the server.

User profile name. The name of the user profile that contains the authentication entry.

You can use the following special value for the user profile name:

**CURRENT*	The current user profile.

Error Messages

Message ID	Error Message Text
CPF2204 E	User profile &1 not found.
CPF2217 E	Not authorized to user profile &1.
CPF24B4 E	Severe error while addressing parameter list.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C21 E	Format name &1 is not valid.
CPF3C90 E	Literal value cannot be changed.
CPF3CF1 E	Error code parameter not valid.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.
FPE0211 E	Parameter length &1 not valid for field &2.
FPE0212 E	Field offset and length not within data.
FPE0215 E	Could not find authentication entry for &1.

API introduced:

Top | Security APIs | APIs by category

Start NetWare Connection (QfpzStrNtwCnn) API

Required Parameter Group:

1	Connection data
Input	Char(*)
2	Length of connection data
Input	Binary(4)
3	Connection data format name
Input	Char(8)
4	Error code
I/O	Char(*)

Default Public Authority: *USE

Library Name/Service Program: QFPNTWE/QFPZAAPI

Threadsafe: No

The Start NetWare Connection (QfpzStrNtwCnn) API establishes the caller as an authenticated user of the specified server. This API could be used to:

- Start a connection to a server for which a user has no authentication entry
- Start a connection when the authentication entry has the password special value *STRNTWCNN

  An authentication entry may have the value *STRNTWCNN because the QRETSVRSEC system value does not allow retrieving security data such as passwords.
- Start a connection to a server by using a different NetWare user name than is specified in the authentication entry for that server

If an authentication entry for the server exists and contains a valid password and user name, the system starts a connection to the server when needed, and this API need not be used.

Connections may only be used by the specified user and job. The job may be either the current job (value 1 for the authorized job field) or any job (value 2 for the authorized job field) on the system. If any job is specified, all jobs that are running under the specified user profile can use the connection. If a connection for a user profile and job exists, that connection is used. Otherwise, an any-job connection for the user profile is used. A current-job connection might be used, for example, when a job uses a connection with a different NetWare user name than other jobs on the system. An any-job connection is required for certain functions, for example, printing to a NetWare print server.

NetWare backup services require a separate authentication. To support this, the connection type field specifies whether a normal login (*USER connection) or a backup services authentication (*SAVRST connection) is to be performed.

A user profile can only have one connection per job to a server (either NetWare 3.x server or a server within a NetWare Directory Services (NDS) tree) at any given time. Connections may be started to all servers within an NDS tree, for example, when copying from one server to another. If connections to multiple servers are open, operations that do not require a specific server use the first connection started.

Authorities and Locks

User Profile Authority
*USE

Required Parameter Group

Connection data
INPUT; CHAR(*)

The server and user name for the connection to be started. The content and format of this structure are determined by the format name. See "Format of Connection Data" for a description of these formats.

Length of connection data
INPUT; BINARY(4)

The length of the connection data structure.

Connection data format name
INPUT; CHAR(8)

The content and format of the input data. Several formats are provided. Each format corresponds to a type of server.

The possible format names follow:

"SVRC0100 Format (NetWare Version 3.x Server)"	Start a connection to a NetWare Version 3.x server.
"SVRC0200 Format (NetWare Directory Services Tree)" on page 36	Start a connection to a server in a NetWare Directory Services tree.

Error code
I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Format of Connection Data

For details about the format of the connection data, see the following sections. For details about the fields in each format, see "Field Descriptions" on page 37.

SVRC0100 Format (NetWare Version 3.x Server)

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to server name
0	0	BINARY(4)	Offset to server name
4	4	BINARY(4)	Length of server name
8	8	BINARY(4)	Offset to NetWare user name
12	C	BINARY(4)	Length of NetWare user name
16	10	BINARY(4)	Offset to password
20	14	BINARY(4)	Length of password

Offset		Type	Field
Dec	Hex		
24	18	BINARY(4)	Offset to connection type list
28	1C	BINARY(4)	Number of entries in connection type list
32	20	BINARY(4)	Authorized job
36	24	BINARY(4)	Connection idle time
40	28	CHAR(10)	User profile name
50	32	CHAR(6)	Reserved
		CHAR(*)	Server name
		CHAR(*)	NetWare user name
		CHAR(*)	Password
Note: The following field is repeated for each connection type requested.			
		BINARY(4)	Connection type

SVRC0200 Format (NetWare Directory Services Tree)

Offset		Type	Field
Dec	Hex		
0	0	BINARY(4)	Offset to NDS tree name
4	4	BINARY(4)	Length of NDS tree name
8	8	BINARY(4)	Offset to server name
12	C	BINARY(4)	Length of server name
16	10	BINARY(4)	Offset to NDS context
20	14	BINARY(4)	Length of NDS context
24	18	BINARY(4)	Offset to NetWare user name
28	1C	BINARY(4)	Length of NetWare user name
32	20	BINARY(4)	Offset to password
36	24	BINARY(4)	Length of password
40	28	BINARY(4)	Offset to connection type list
44	2C	BINARY(4)	Number of entries in connection type list
48	30	BINARY(4)	Authorized job
52	34	BINARY(4)	Connection idle time
56	38	CHAR(10)	User profile name
66	42	CHAR(6)	Reserved
		CHAR(*)	NDS tree name
		CHAR(*)	Server name
		CHAR(*)	NDS context
		CHAR(*)	NetWare user name
		CHAR(*)	Password
Note: The following field is repeated for each connection type requested.			
		BINARY(4)	Connection type

Field Descriptions

Authorized job. The jobs authorized to use the connection.

The following values may be specified:

1 The connection can be used only by the *current job*. The connection ends when the current job is ended, or when the connection is ended by an end NetWare connection request.

2 The connection can be used by *any job* on the system that is running under the specified user profile. The connection must be explicitly ended by an end NetWare connection request.

Notes:

1. The QNETWARE file system requires a connection for the current job.
2. If no authentication entry exists, or the password in the authentication entry is *STRNTWCNN, printing functions require that a connection for any job (authorized job value is 2) be started.
3. If the authorized job value is 1 and user reclaims the QFPZAUT activation group (RCLACTGRP command), the connection ends.
4. For either authorized job value, you may use the connection idle time value to control when a connection ends.

Connection idle time. The idle time (in minutes) before a connection is closed by the system. Idle time is the time between subsequent requests. Valid values range from 1 minute through 9999 minutes.

The following special value may be specified:

-1 There is no idle time. A connection is closed by the user, or it is closed when the job has ended (if the authorized job value is 1).

Connection type. The type of authentication to be performed. NetWare backup services require a separate authentication in addition to a normal user authentication.

The following values may be specified:

1 A normal user authentication is done. The user can use administrative and file system functions other than save and restore.

2 Authentication to NetWare backup services is performed. This option requires that the appropriate NetWare Loadable Module software (TSA312.NLM, TSA410.NLM, or TSANDS.NLM, and SMDR.NLM) be loaded on the server.

-1 Perform both user and backup services authentications. If -1 is specified, it must be the only connection type value specified.

Length of NDS context. The length, in bytes, of the NDS context.

Length of NDS tree name. The length, in bytes, of the NDS tree name.

Length of NetWare user name. The length, in bytes, of the NetWare user name.

Length of password. The length, in bytes, of the password.

Length of server name. The length, in bytes, of the server name.

NDS context. For NetWare Directory Services trees, the directory context in which the user is defined.

The following special value may be used:

*AUTE Use the NDS context from the user profile authentication entry for this NDS tree.

NDS tree name. The name of the NetWare Directory Services tree to which the connection is to be started.

NetWare user name. The NetWare user name that is used to authenticate the user to the server.

The following special values may be used:

*USRPRF Use the authorized OS/400 user profile name as the NetWare user name.
*AUTE Use the NetWare user name from the user profile authentication entry for this server or NDS tree.

Number of entries in connection type list. The number of connection type values in the connection type list. At least one value must be specified.

Offset to connection type list. The offset, in bytes, from the start of the data area to the connection type values.

Offset to NDS context. The offset, in bytes, from the start of the data area to the NDS context.

Offset to NDS tree name. The offset, in bytes, from the start of the data area to the NDS tree name.

Offset to NetWare user name. The offset, in bytes, from the start of the data area to the NetWare user name.

Offset to password. The offset, in bytes, from the start of the data area to the password.

Offset to server name. The offset, in bytes, from the start of the data area to the server name.

Password. The password that is used to authenticate the user to the server.

The following special values may be used:

*AUTE Use the password from the user profile authentication entry for this server or NDS tree.
*NONE No password is used to authenticate the user.

Reserved. Set this field to binary zeros.

Server name. The name of the server to which the connection is to be started.

When you are connecting to a NetWare Version 3.x server, the server name must be specified.

When you are connecting to a NetWare Directory Services tree, the connection must be started on a server within the tree.

Either the server name or one of the following special values must be specified:

*ANY Use any server within the specified NDS tree.
*ALL Start connections to all servers in the specified tree.

User profile name. The OS/400 user profile name that is authorized to use this connection.

The following special value may be used:

CURRENT Jobs running under the current user profile are authorized to use the connection.

Error Messages

Message ID	Error Message Text
CPF2144 E	Not authorized to &1 in &2 type *&3.
CPF2217 E	Not authorized to user profile &1.
CPF24B4 E	Severe error while addressing parameter list.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C21 E	Format name &1 is not valid.
CPF3C90 E	Literal value cannot be changed.
CPF3CF1 E	Error code parameter not valid.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.
FPE0211 E	Parameter length &1 not valid for field &2.
FPE0212 E	Field offset and length not within data.
FPE0215 E	Could not find authentication entry for &1.
FPE021C E	Not able to contact server &1.
FPE0227 E	Server name *ANY or *ALL not allowed for server type.
FPE0230 E	Connection type -1 must be single value.
FPE0235 E	Error &2 connecting to server &3.
FPE0236 E	Value &1 not valid for authorized job.
FPE0237 E	Value &1 not valid for connection idle time.
FPE023C E	Server is not a &2 server. Server: &1.
FPE023D E	NDS tree not known to system. Tree: &1.
FPE023E E	Server &1 not found in tree &2.

API introduced: V3R7

Top | Security APIs | APIs by category

Verify NetWare Authentication Entry (QfpzVfyNtwAutE) API

Required Parameter Group:

1	Entry identifier data
Input	Char(*)
2	Length of entry identifier
Input	Binary(4)
3	Entry identifier format name
Input	Char(8)
4	Error code
I/O	Char(*)

Default Public Authority: *USE

Library Name/Service Program: QFPNTWE/QFPZAAPI

Threadsafe: No

The Verify NetWare Authentication Entry (QfpzVfyNtwAutE) API verifies that the specified authentication entry can be used to connect to a server. The user name, password, and other data are sent to the server, where they are used to attempt to start an authenticated connection to the server. This API might be used, for example, to verify that the password is valid before submitting a batch job that would use this entry.

Authorities and Locks

User Profile Authority
The user profile must be the current user profile, or the caller must have *USE and *OBJMGT authority to the user profile and *SECADM special authority.

Required Parameter Group

Entry identifier data
INPUT; CHAR(*)

The server and user name for the connection to be started. The content and format of this structure are determined by the format name. See "Format of Authentication Entry Identifier" for a description of these formats.

Length of entry identifier
INPUT; BINARY(4)

The length of the entry identifier structure.

Entry identifier format name
INPUT; CHAR(8)

The content and format of the input data. Each format corresponds to a method of identifying an authentication entry.

The possible format names follow:

"AUTE0100 Format"	The NetWare Version 3.x server entry is identified by the server name.
"AUTE0200 Format" on page 41	The NetWare Directory Services tree entry is identified by the NDS tree name.

Error code
I/O; CHAR(*)

The structure in which to return error information. For the format of the structure, see Error Code Parameter.

Format of Authentication Entry Identifier

For details about the format of the entry identifier, see the following sections. For details about the fields in each format, see "Field Descriptions" on page 41.

AUTE0100 Format

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Offset to server name
4	4	BINARY(4)	Length of server name

Offset			
Dec	Hex	Type	Field
8	8	CHAR(10)	User profile name
18	12	CHAR(6)	Reserved
		CHAR(*)	Server name

AUTE0200 Format

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Offset to NDS tree name
4	4	BINARY(4)	Length of NDS tree name
8	8	BINARY(4)	Offset to server name
12	C	BINARY(4)	Length of server name
16	10	CHAR(10)	User profile name
26	1A	CHAR(6)	Reserved
		CHAR(*)	NDS tree name
		CHAR(*)	Server name

Field Descriptions

Length of NDS tree name. The length, in bytes, of the NDS tree name.

Length of server name. The length, in bytes, of the server name.

NDS tree name. For *NDS trees, the name of the directory tree.

Offset to NDS tree name. The offset, in bytes, from the start of the input data area to the NDS tree name.

Offset to server name. The offset, in bytes, from the start of the input data area to the server name.

Server name. For *NETWARE3 servers, the name of the server.

For *NDS servers, the name of the server on which to verify the authentication entry.

For *NDS servers, the following special value may be specified:

**ANY*	Any server in the specified NDS tree.

Reserved. Set this field to binary zeros.

User profile name. The name of the user profile that contains the authentication entry.

You can use the following special values for the object name:

**CURRENT*	The current user profile.

Error Messages

Message ID	Error Message Text
CPF2204 E	User profile &1 not found.
CPF2217 E	Not authorized to user profile &1.
CPF24B4 E	Severe error while addressing parameter list.
CPF3C1D E	Length specified in parameter &1 not valid.
CPF3C21 E	Format name &1 is not valid.
CPF3C90 E	Literal value cannot be changed.
CPF3CF1 E	Error code parameter not valid.
CPF9872 E	Program or service program &1 in library &2 ended. Reason code &3.
FPE0211 E	Parameter length &1 not valid for field &2.
FPE0212 E	Field offset and length not within data.
FPE0215 E	Could not find authentication entry for &1.
FPE021C E	Not able to contact server &1.
FPE0235 E	Error &2 connecting to server &3.
FPE023C E	Server is not a &2 server. Server: &1.
FPE023D E	NDS tree not known to system. Tree: &1.
FPE023E E	Server &1 not found in tree &2.

API introduced: V3R7

Top | Security APIs | APIs by category

Appendix. Notices

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

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

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

```
IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY  10504-1785
U.S.A.
```

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

```
IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan
```

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION ″AS IS″ WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

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

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

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

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

```
IBM Corporation
Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.
```

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

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

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:
Advanced 36
Advanced Function Printing
Advanced Peer-to-Peer Networking
AFP
AIX
AS/400
COBOL/400
CUA
DB2
DB2 Universal Database
Distributed Relational Database Architecture
Domino
DPI

DRDA
eServer
GDDM
IBM
Integrated Language Environment
Intelligent Printer Data Stream
IPDS
iSeries
Lotus Notes
MVS
Netfinity
Net.Data
NetView
Notes
OfficeVision
Operating System/2
Operating System/400
OS/2
OS/400
PartnerWorld
PowerPC
PrintManager
Print Services Facility
RISC System/6000
RPG/400
RS/6000
SAA
SecureWay
System/36
System/370
System/38
System/390
VisualAge
WebSphere
xSeries

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

Terms and conditions for downloading and printing publications

Permissions for the use of the information you have selected for download are granted subject to the following terms and conditions and your indication of acceptance thereof.

Personal Use: You may reproduce this information for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of this information, or any portion thereof, without the express consent of IBM$^{(R)}$.

Commercial Use: You may reproduce, distribute and display this information solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of this information, or reproduce, distribute or display this information or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the information or any data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the information is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations. IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THIS INFORMATION. THE INFORMATION IS PROVIDED ″AS-IS″ AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

All material copyrighted by IBM Corporation.

By downloading or printing information from this site, you have indicated your agreement with these terms and conditions.

Code disclaimer information

This document contains programming examples.

SUBJECT TO ANY STATUTORY WARRANTIES WHICH CANNOT BE EXCLUDED, IBM$^{(R)}$, ITS PROGRAM DEVELOPERS AND SUPPLIERS MAKE NO WARRANTIES OR CONDITIONS EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT, REGARDING THE PROGRAM OR TECHNICAL SUPPORT, IF ANY.

UNDER NO CIRCUMSTANCES IS IBM, ITS PROGRAM DEVELOPERS OR SUPPLIERS LIABLE FOR ANY OF THE FOLLOWING, EVEN IF INFORMED OF THEIR POSSIBILITY:

1. LOSS OF, OR DAMAGE TO, DATA;
2. SPECIAL, INCIDENTAL, OR INDIRECT DAMAGES, OR FOR ANY ECONOMIC CONSEQUENTIAL DAMAGES; OR
3. LOST PROFITS, BUSINESS, REVENUE, GOODWILL, OR ANTICIPATED SAVINGS.

SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO SOME OR ALL OF THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.

Printed in USA