ACTIVEDIRECTORY (LDAP) CONFIGURATION

TECHNOTE 16/14

Date: November 7th, 2016 (Updated October 13th, 2017)

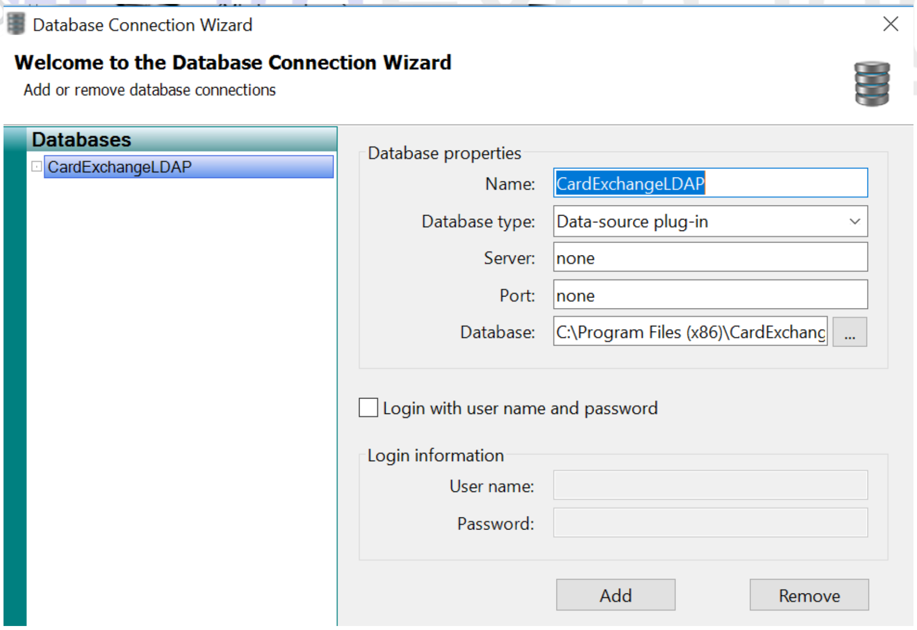
Author: Craig Bechelli

Applicable Products: CardExchange® Producer

ACTIVEDIRECTORY (LDAP) CONFIGURATION

Overview

The LDAPExchange data-source plug-in allows CardExchange® Producer to connect to Active Directory (LDAP) as though it is one of the available databases.
The plug-in allows you to read and update entries as well as display and add photos to existing entries.



Database Connection Wizard

Welcome to the Database Connection Wizard

Add or remove database connections

Databases

- CardExchangeLDAP

Database properties

Name: CardExchangeLDAP

Database type: Data-source plug-in

Server: none

Port: none

Database: C:\Program Files (x86)\CardExchang

☐ Login with user name and password

Login information

User name:

Password:

It is not possible to add new entries into Active Directory or to delete entries from Active Directory.

Configuration

To use the LDAP data-source plug-in, you must have either CardExchange® Producer Ultimate V9 or CardExchange® Producer Business/Enterprise V10.
If using CardExchange® Producer Ultimate V9 or CardExchange® Producer Business V10 you also need an additional license for the plug-in, you can activate this through the activation wizard.
CardExchange® Producer Enterprise V10 has this plug-in included as standard.

Then proceed as follows:

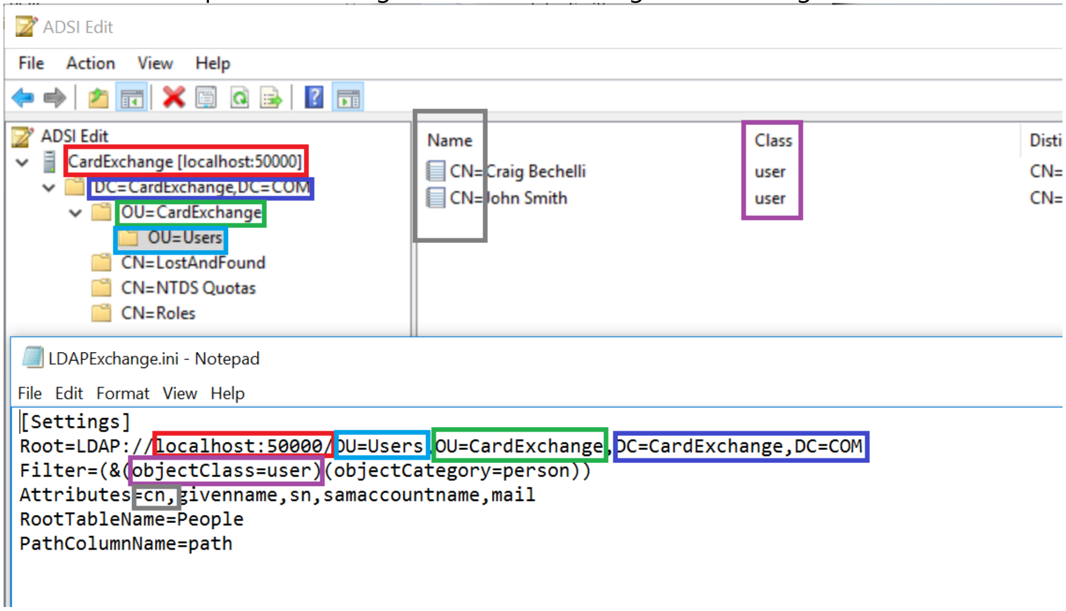
1. Start Notepad with administrator rights and open the file LDAPExchange.ini which is located in the CardExchange® Producer installation folder.

   For CardExchange® Producer V9 this is: (C:\Program Files (x86)\CardExchange Solutions\CardExchange Producer).

   For CardExchange® Producer V10 this is: (C:\Program Files (x86)\CardExchange Solutions\CardExchange Gateway).

2. Specify the desired root node and the attributes (fields) that you want to load from the LDAP.

   The filter is set to user by default, but can be changed to other object types if desired. Leave the root-table name as is.
   Leave the path-column name as is unless the name "path" also appears in your attribute list, in which case you should choose a unique name.
   The path column will contain the full distinguished name of the LDAP entries and can serve as primary key.
   Below is an example LDAP configuration and matching LDAPExchange.ini file:



```
[Settings]
Root=LDAP://localhost:50000/OU=Users,OU=CardExchange,DC=CardExchange,DC=COM
Filter=(&(objectClass=user)(objectCategory=person))
Attributes=cn,givenname,sn,samaccountname,mail
RootTableName=People
PathColumnName=path
```

3. Start CardExchange and create a new card or modify an existing one.

4. In the database setup, chose Data-source plug-in as database type, enter “none” for the server and the port (these fields are not used) and select the LDAPExchange.dll file located in the installation folder as the database.

5. Check the Login option and supply the correct login details (This may not be needed for a domain user)

6. On the second page, select the People table and choose a primary key. If none of the attributes specified in LDAPExchange.ini can serve as primary key, you can use the path column. This is a long text column and hence not very efficient as primary key, but it works if needed.

7. On the fourth page, if you allow editing at all, disable the options to add and delete records, as this is not supported by LDAPExchange.

8. On the storage items page, you can create storage items, but again, it will only be possible to update existing records of the LDAP and not inserting new records.

Photos

If you want to store photos in the LDAP, you will need to indicate that the photo field should be treated as a binary field by adding a line Binary=photo to the LDAPExchange.ini file, which now will look something like this:

```
[Settings]

Root=LDAP://localhost/CN=Partition
Filter=(&(objectClass=user)(objectCategory=person))
Attributes=cn,department,revision,sn,photo
RootTableName=People
PathColumnName=path
AuthenticationType=0
Binary=photo
```

Please, note that the photo field is mentioned both in the Attributes list as in the Binary list. If you also want to store the signature or finger print, you can add more field to the Binary list (e.g. Binary=photo,signature,fingerprint).

In order to store the photos, you will have to create a storage item in the database setup to store the photo when it is captured.

Encryption

If you need to use an encrypted connection to the LDAP server then CardExchange® Producer supports several methods which can be set by changing the AuthenticationType value in the LDAPExchange.ini file.

Please note that the values are powers of two and that they can hence be combined. A value of 3 would for example stand for Secure and SecureSocketsLayer.

Below are the possible values with an explanation from Microsoft

None = 0

Summary:
Equates to zero, which means to use basic authentication (simple bind) in the LDAP provider.

Secure = 1

Summary:
Requests secure authentication. When this flag is set, the WinNT provider uses NTLM to authenticate the client. Active Directory Domain Services uses Kerberos, and possibly NTLM, to authenticate the client.

When the user name and password are a null reference, ADSI binds to the object using the security context of the calling thread, which is either the security context of the user account under which the application is running or of the client user account that the calling thread is impersonating.

Encryption = 2

Summary:
Attaches a cryptographic signature to the message that both identifies the sender and ensures that the message has not been modified in transit.

SecureSocketsLayer = 2

Summary:
Attaches a cryptographic signature to the message that both identifies the sender and ensures that the message has not been modified in transit.

Active Directory Domain Services requires the Certificate Server be installed to support Secure Sockets Layer (SSL) encryption.

ReadonlyServer = 4

Summary:
For a WinNT provider, ADSI tries to connect to a domain controller. For Active Directory Domain Services, this flag indicates that a writable server is not required for a serverless binding.

Anonymous = 16

Summary:
No authentication is performed.

FastBind = 32

Summary:
Specifies that ADSI will not attempt to query the Active Directory Domain Services objectClass property. Therefore, only the base interfaces that are supported by all ADSI objects will be exposed. Other interfaces that the object supports will not be available.

A user can use this option to boost the performance in a series of object manipulations that involve only methods of the base interfaces. However, ADSI does not verify if any of the request objects actually exist on the server.

For more information, see the topic "Fast Binding Option for Batch Write/Modify Operations" in the MSDN Library at http://msdn.microsoft.com/library.

For more information about the objectClass property, see the "Object-Class" topic in the MSDN Library at http://msdn.microsoft.com/library.

Signing = 64

Summary:
Verifies data integrity to ensure that the data received is the same as the data sent. The System.DirectoryServices.AuthenticationTypes.Secure flag must also be set to use signing.

Sealing = 128

Summary:
Encrypts data using Kerberos. The System.DirectoryServices.AuthenticationTypes.Secure flag must also be set to use sealing.

Delegation = 256

Summary:
Enables Active Directory Services Interface (ADSI) to delegate the user's security context, which is necessary for moving objects across domains.

ServerBind = 512

Summary:
If your ADsPath includes a server name, specify this flag when using the LDAP provider. Do not use this flag for paths that include a domain name or for serverless paths. Specifying a server name without also specifying this flag results in unnecessary network traffic.