AMX Account Management Tutorial 1

This tutorial is for IT staff who are experienced in identity management, it requires insight into how the ActiveDirectory works, and a working knowledge of Windows.

This tutorial will demonstrate some of the features of AMX, specifically:

- Using identityReport to generate a test file of identities from your Active Directory. In a production situation this identity source would be obtained from an HR system.
- Using identitySync with the test file of identities to synchronise the Active Directory. With no changes to the identities, no changes will be reported for the Active Directory.
- Making some changes to the identities, and synchronising with the Active Directory. Review the outputs of identitySync.
- Re-running identitySync in the update mode to have identitySync make an update to The Active Directory.
- Re-running identity Sync in the undo mode to restore the Active Directory to its original configuration.
- Using the DSMOD do and undo files created by identitySync, to update the Active Directory with the DSMODDo file, and undo the change with DSMODUndo.
- Other updates of CNs, distinguishedNames and Managers.

AMX runs on Windows and must be setup as shown in the AMX Tutorial Setup document. In this tutorial identityReport and identitySync are run from the Command Line using AMXRun which sets the environment variables, in production it is expected to be run by the Task Scheduler.

1. identityReport

identityReport will read the Active Directory and produce a test file of accounts and their attributes. identityReport can be run with a non-privileged account.

Update ActiveDirectory1 Properties

Edit the Properties file ActiveDirectory1.properties, it is in the <installDirectory>\Tutorial1 directory. identityReport properties are consistent with identitySync and the same properties file is used for both applications.

The parameters that need to be updated are:

```
// Systems
ActiveDirectoryIdentityResource1 = dc1.corp.example.com
ActiveDirectoryIdentityAccountContainer1= OU=accounts,DC=corp,DC=example,DC=com
ActiveDirectoryIdentityUser1 =
ActiveDirectoryIdentityFilterAttribute1 = distinguishedName
ActiveDirectoryIdentityFilterValue1 = OU=EDN:ou=lon
```

- ActiveDirectoryIdentityResource1 can be the resolvable DNS name of the target Domain Controller.
- ActiveDirectoryIdentityAccountContainer1is domain specific. For better performance during the tutorial specify the container where the accounts of interest are located. Use the MMC Console Users and Computers to identify a suitable container. In production the account container must have all the accounts otherwise the check for account uniqueness will fail. For the tutorial a test domain or a test OU in a production domain are suitable.
- ActiveDirectoryIdentityUser1 can be left blank in situations where a domain account is used on the PC running AMX. The account does not need to be an administrator to read the Active Directory accounts. If an alternative account is needed, create an ActiveDirectoryPasswd1.txt file adding the password in the first line, it will be encrypted when identityReport.exe first runs.

```
ActiveDirectoryIdentityPasswd1 = ActiveDirectoryPasswd1.txt
```

- ActiveDirectoryIdentityFilterAttribute1 is the Active Directory attribute that is used to select accounts that are to be reported, ActiveDirectoryIdentityFilterValue1 is the value that the filtered attribute must contain to be included. Any metaverse attribute that is defined in the ActiveDirectorySchema1.txt file can be used as a filter. Blank includes all the accounts.

In this example the distinguishedName attribute is used and the ActiveDirectoryIdentityFilterValue1 defines values that the attribute must contain to be included in the identityReport report. The value is not case sensitive, and the ":"character allows multiple filter values to be entered. For example:

```
OU=EDN:ou=lon
```

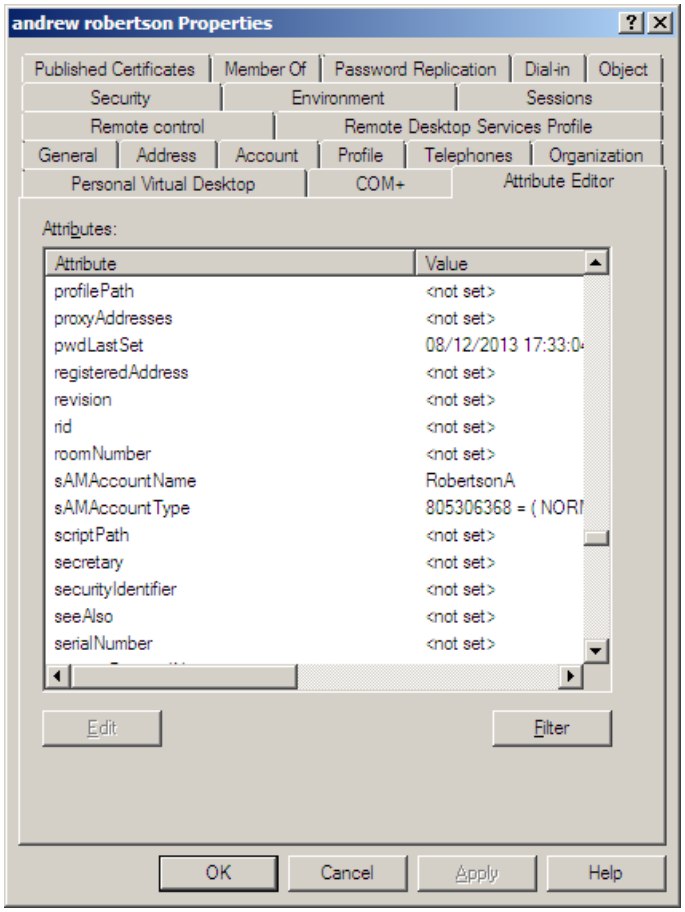
The combination of AccountContainer and filtering defines which accounts will be managed by identitySync. For tutorial purposes you may want to use test accounts. In this case an alternative ActiveDirectoryIdentityFilterAttribute could be description, with an ActiveDirectoryIdentityFilterValue1 of "test". The ActiveDirectoryIdentityFilterAttribute must be defined in the Active Directory schema file ActiveDirectorySchema1.txt, see the next section for details of how to add attributes to the schema.

Update ActiveDirectory Schema File

The file ActiveDirectorySchema1.txt does not need any modification unless a FilterAttribute is used that is not defined in the schema file. The format of the schema file is:

```
[Staging Attribute Name],[Metaverse Attribute Name];Attribute Flags and Modifiers
```

Staging is the attribute name in the Active Directory, this can be checked in the MMC Console Users and Computers, using the attribute editor. If metaverse attribute name is blank it defaults to the staging attribute name. When the staging attribute name is blank it is derived from other attributes during the transform stage.



Note that the sAMAccountName is the staging name for the user logon name.

The metaverse attribute names are used by AMX to build its internal database.

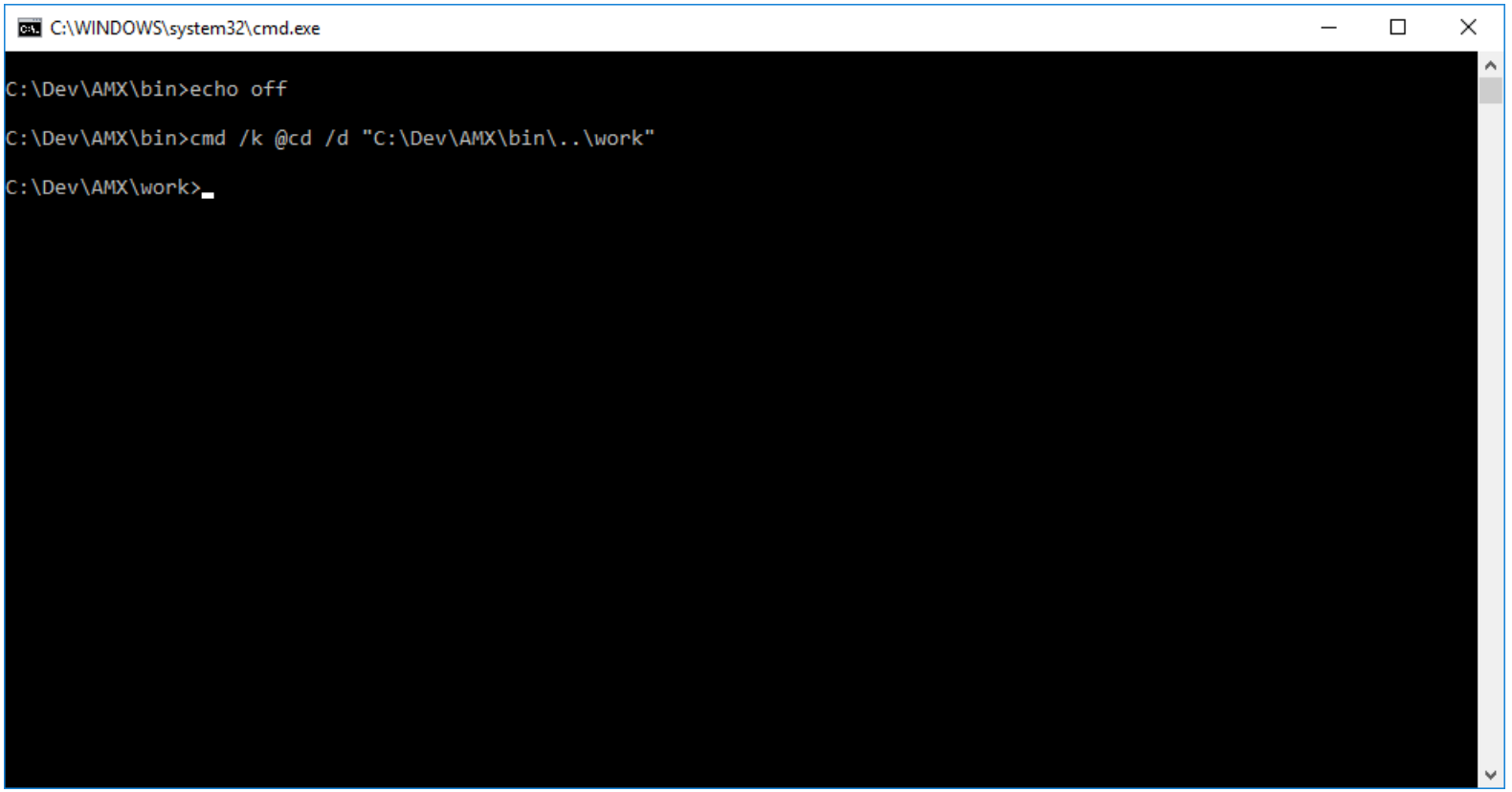
To add additional attributes, they must be defined in the Active Directory Schema.

Update ActiveDirectory Passwd

When ActiveDirectoryIdentityUser1 is defined, enter the password in the first line of the file, when a domain account is used, the password is not required. The password will be encrypted when identityReport first runs and the clear text password will be removed.

Run identityReport

Right click on AMX Run in the Start Programs menu or AMXRun.bat in the installation directory bin, and Run as Administrator.



This will open a Command Prompt.

Change directory to tutorial1 and run identityReport.exe which is in the parent directory

```
C:\AMX\Tutorial1>identityReport.exe ActiveDirectory1.properties
Begins Mon, 31 Oct 2016 12:24:30 GMT
ActiveDirectoryIdentity1 corpID
Extracted 102 Identities
ActiveDirectoryIdentity Finished Mon, 31 Oct 2016 12:24:32 GMT
Total of 102 Identities

Finished Mon, 31 Oct 2016 12:24:32 GMT

C:\AMX\Tutorial1>
```

Check that the expected number of accounts were returned. In situations where the number of accounts is incorrect, open the debug file and check for errors.

After a successful run this will create:

- identityReportAD1.csv containing the accounts and their attributes. This will be used by identitySync later as the source of identities.
- Debug.txt
- Info.txt

Failed to connect to the Active Directory

The console may show a message such as

```
Error: ActiveDirectory Extract A referral was returned from the server.
 for LDAP:// dc1.example.com/DC=corpz,DC=example,DC=com
Error: Exception of type 'AMX.ActiveDirectory+CustomException' was thrown.
```

This is caused when the domain controller does not recognize the domain.

Bad credentials

```
Error: ActiveDirectory Extract The username or password is incorrect.
 for LDAP://192.168.121.61/DC=corp,DC=example,DC=com
Error: Exception of type 'AMX.ActiveDirectory+CustomException' was thrown.
```

ActiveDirectory Extract Filter Attribute company not in Metaverse

A filter attribute defined in the property ActiveDirectoryFilterAttribute1 is not defined in the right hand side of the ActiveDirectorySchema. See Update ActiveDirectory Schema

Filter failed to match any accounts

Search for “Skipped” in the debug.txt file

```
ActiveDirectory ExtractAttribute filter distinguishedName value = cn=alpin
thomson,ou=accounts,dc=corp,dc=example,dc=com filter ou=accountsz

ActiveDirectory ExtractAttribute Skipped ..... Thomson, Alpin
```

In this example the filter is miss-spelled ou=accountsz and is not found in the distinguishedName, so the record is skipped.

2. identitySync

identitySync can be run in the following modes:

- Analyse mode, the default, report the changes that the synchronisation identified
- Do mode, report and implement the changes, intended for production use in a scheduled task

- Undo mode, undo the changes reported by analyse or do
- Redo mode, do the changes reported by analyse or do.

To make changes to the Active Directory an account with Account Operator privileges is required, Analyse mode can use a non-privileged account.

For this exercise a second properties file ActiveDirectory2.properties is used rather than updating the originals. Comments identify the differences. ActiveDirectory2.properties defines a trip-wire, currently set at 1, in this case if identitySync finds more than 1 change it will process none of them. This can be set to 0. In production situations where the authoritative source of identities is HR or payroll the recommendation would be to set it to 5% of the total number of accounts that are being managed.

```
MaximumUpdates = 1
MaximumDisables = 1
MaximumDeletes = 1
```

identitySync never deletes anything except when it's undoing an account create.

Update ActiveDirectory2 Properties

Edit the ActiveDirectory2.properties file for identitySync, it is in the <installDirectory>\Tutorial1 directory.

The identities are read from the CSV file IdentityReportAD1.csv just created by identityReport, note the parameters CSVidentityResource in ActiveDirectory2.properties are un-commented to use this file. The accounts are read from the Active Directory, copy the values of the ActiveDirectoryIdentity properties to the ActiveDirectory ones.

ActiveDirectory2.properties uses some additional parameters that were not used by identityReport, the Active Directory name must be the NETBIOS name of the domain:

```
ActiveDirectoryName1 = corp
```

The ActiveDirectoryLoadMode controls the behaviour of the identitySync load process (Create, Update, Disable, Delete) CRUDD. U is update only. Note that identitySync never deletes anything, a delete caused by the identity source deleting an entry is a move to a recycle bin in the Active Directory.

```
ActiveDirectoryLoadMode1 = U
```

Review the CSV Schema

The IdCSVschemaAD1.txt file does not need any modification.

Some explanation of the CSV schema supplied with the tutorial is useful. The attribute pairs define the attribute names in the resource, for example the column name in CSV and the Active Directory attribute name, followed by the attribute name in the metaverse. The metaverse is identitySync's database of attribute values, and the only metaverse attribute names that match between the CSV and ActiveDirectory are synchronised. If a filter attribute was added to the Active Directory schema, do not necessary to add it to the CSV Schema file and it then will not be updated by identitySync.

Review ActiveDirectorySchema1.txt

The file ActiveDirectorySchema1.txt defines the schema, a key attribute is the join. This is used to match the identity to the account and is defined in the account Schema. For the purposes of this tutorial sAMAccountName is used for the join. In production employeeID or fullName would be used. The join must be unique and should be able to match all the accounts, otherwise they are reported as accounts with no owners, or ghosts. The full AMX package has tools to assist with matching accounts to identities.

The schema has flags ManagerID, ManagerJoin, Name and ManagerName, these are only used when the schema is for an identity source. They are explained in step 5.

Run identitySync

Use the same console window as identityReport to run identitySync.exe in the analyse mode.

```
C:\AMX\Tutorial1>identitySync.exe ActiveDirectory2.properties
Begins Mon, 31 Oct 2016 12:27:06 GMT
Warning: Not run as administrator
CSVidentity1 C:\Dev\AMX\Tutorial1\IdentityReportAD1.csv
Extracted 102 Identities
CSVidentity Finished Mon, 31 Oct 2016 12:27:06 GMT
Total of 102 Identities

ActiveDirectory1 corp
Extracted 102 Accounts
Account joins    102
Account creates  0
Account updates  0
Account disables 0
Account deletes  0
ActiveDirectory Finished Mon, 31 Oct 2016 12:27:07 GMT
Finished Mon, 31 Oct 2016 12:27:07 GMT

C:\AMX\Tutorial1>
```

Ignore the warning that the ActionFile.txt cannot be found, it will be created when identitySync first runs. Check that the same number of accounts were returned from the identityReportAD1.csv and the Active Directory, this will be the same number as found by identityReport.

Review the Action File

identitySync writes any creates, updates, disables or deletes to the file ActionFile.txt in the current directory. The previous identitySync result should contain an empty Action file containing only the word “Ends”.

identitySync also creates a DSMODDo.txt and a DSMODUndo.txt files these should be empty containing only a comment with today's date. These files are intended to make batch updates to the Active Directory using Microsoft's DSMOD tool.

Total of Zero Identities

The property CSVidentityResource1 has not been un-commented.

Failed to open identityReportAD1.csv

If the file is currently opened by Excel or some other application, an error is reported:

```
Error: CSV Extract Read of 'C:\AMX\Tutorial1\ identityReportAD1.csv init The process cannot access the
file 'C:\AMX\Tutorial1\ identityReportAD1.csv' because it is being used by another process.
```

Close the file and re-run identitySync.exe

3. Run identitySync in update mode

Make a change to the Identity Source

This exercise shows a simple attribute being synchronised by identitySync between the source of identities and the ActiveDirectory. Updates can also be done automatically by identitySync or manually using DSMOD with the batch files that identitySync produces.

1. Open identityReportAD1.csv in Excel, or a text editor of your choice. Change one attributes of one identity. identitySync property MaxUpdates is set to 1 and so identitySync will not change more than one account. You can change this and change more accounts. Note that identitySync can change any attribute, DSMOD unfortunately cannot. See the DSMOD command line help for details of which attributes can be updated. In this exercise schema DSMOD cannot be used to change the following attributes:
   - Location

- Street
- Postcode

For example it can change: First Name, Initials, Last Name, title, description or department

2. If Excel was used to update identityReportAD1.csv, close it to release file access for identitySync.

Update ActiveDirectory1 Properties

Edit the ActiveDirectory1.properties Properties file, it is in the <installDirectory>\Tutorial1 directory.

- Check that the Active Directory account is an Account Operator in the property ActiveDirectoryUser. If another account is required set the account name and add its password in the ActiveDirectoryPassword1.txt file. Open it in a text editor and note the encrypted password on line 2. Enter the password on line 1, delete line 2 and save the file. It will be encrypted when identitySync first runs.

Run identitySync

1. Use the same console window to run identitySync.exe in the analyse mode.
   ```
   C:\AMX\Tutorial1>..\identitySync.exe ActiveDirectory2.properties
   ```

2. This will create a new transaction file ActionFile.txt which could be used to update the Active Directory. Check that the ActionFile.txt has the expected update in the form:
   `02/01/2016|corp;SmithA|Update|location=London|location=Glasgow.`
   The first field after Update is the original attribute and value, the second is the new one.

3. Run identitySync.exe in redo mode to actually make the changes and avoid reloading accounts from the Active Directory. Check that the account in the Active Directory has changed.

```
C:\AMX\Tutorial1>..\identitySync.exe ActiveDirectory2.properties redo
```

identitySync do mode will re-read the Active Directory, recreate the transaction file and then make the changes.

Failed to make updates

```
Error: ActionFile ProcessFile ActionFile.txt has 2 Account Updates. Greater than max updates 1
```

The identitySync tripwire is set to a maximum of 1 update in identitySync.properties. Change it to a more realistic value if you actually want to make the changes.

```
// Trip wire checked in do, redo phase
MaximumUpdates = 1
MaximumDisables = 1
MaximumDeletes = 1
```

```
Error: ActiveDirectory Load  Update of StevensonA Failed: Access is denied.
```

The account specified in ActiveDirectoryUser in identitySync.properties does not have Account Operator privileges in the Active Directory.

4. Run identitySync in undo mode

Use the undo parameter to reverse the update and leave the Active Directory in its original configuration.

1. Run identitySync.exe in the undo mode. Check the account in the Active Directory has reverted back to its original value.
   ```
   C:\AMX\Tutorial1>identitySync.exe ActiveDirectory2.properties undo
   ```

2. Change identityReportAD1.csv back to its original state.

3. Run identitySync.exe in the analyse mode. Check the ActionFile.txt has no updates.

Failed to undo an update

The Action File has an activity date for each action, the action will only be performed when the date is today. Note that for security reasons AMX does not allow the transaction file to be manually altered, but it can be deleted.

Use DSMOD to update the Active Directory

identitySync creates two batch files for manual updates of the Active Directory if the properties ActiveDirectoryManualDo and Undo are defined. When identitySync is first used to manage accounts, there are often a large number of changes required. Using manual updates allows this to be done in a controlled and safe manner.

Use DSMOD on a Domain Controller or a system with the Active Directory tools installed, use the DSMODdo.txt file to update and DSMODundo.txt to undo the changes.

5. Other Updates

DistinguishedName

The distinguishedName includes the OU structure, changing an OU will move the account to the new OU.

Note that the DN, CN and the object name are linked in the Active Directory, changing the CN will update the object name and the RDN of the distinguishedName. Changing the object name will update the CN and the RDN. In a production scenario either a CN or a DN should be used as part of the identity record, not both.

1. Edit identityReportAD1.csv, update the distinguishedName of a record changing an OU or similar

2. Run identitySync.exe in the analyse mode. Check the Action.txt file has the old and the new distinguishedName

3. Run identitySync.exe in the redo mode to make the change in the Active Directory. Check the Active Directory reflects the move of OU. A refresh will be necessary in the Users & Computers MMC to see the change.

4. Run identitySync in the undo mode. Check the Active Directory has the account in its original OU.

5. Edit identityReportAD1.csv, undo the change to the distinguishedName.

CN

AMX can change the object name and the CN, and we advise that only one of them is synchronised. This exercise will show a change of CN.

1. Edit identityReportAD1.csv, update the CN of a record

2. Run identitySync.exe in the analyse mode. Check the Action.txt file has the old and the new CN

3. Run identitySync.exe in the redo mode to make the change in the Active Directory. Check the Active Directory has a new object name. Check the canonical name of the object has the new distinguishedName and that the CN has changed. A refresh will be necessary in the Users & Computers MMC to see the change.

4. Run identitySync in the undo mode. Check the Active Directory has the object with its original name.

5. Edit identityReportAD1.csv, undo the change to the CN.

Manager

The exercise shows the manager attribute in the ActiveDirectory being synchronised with the value in the source of identities, in this case identityReportAD1.csv. The value of the manager attribute in the ActiveDirectory is the distinguishedName.

The CSV schema IdCSVschemaAD1.txt defines the attribute containing an individual's manager using the "managerID" flag. For each identity:

- The value of the individual's attribute tagged with "managerID" is used to search for their manager's record using the "ManagerJoin" tagged attribute. In this case it is the attribute ManagerAccount which in IdentityReportAD1.csv contains account names if manager was defined in the Active Directory.
- When the search is successful, the attribute tagged with "Name" which is the distinguished name of the individual's manager is used to update the value of the attribute flagged as "managerName" in the CSV Schema. In this case the attribute flagged "managerName" is manager.
- The value of the manager attribute is synchronised between the identity and the Active Directory during the load phase.

In this exercise the managerID in identityReportAD1.csv contains the account name of the individual's manager which is used to search for the manager's record. Then from the manager's record the distinguished name is used to update the person's manager attribute value. These do not need to be changed in the IdCSVSchemaAD1.txt file for this exercise:

```
accountName,;ManagerJoin
managerAccount,;managerID

distinguishedName,distinguishedName;Name
,manager;ManagerName
```

1. Update a managerAccount in the identityReportAD1.csv using the account name of another record

2. Run identitySync.exe ActiveDirectory2.properties using the analyse mode. Check the transaction file ActionFile.txt to see that the new manager Distinguished Name is updated, and no others.

```
23/04/2016|corp;CN=sean
o'brian,OU=EDN,OU=accounts,DC=corp,DC=example,DC=com|Update|manager=CN=bonnie
burns,OU=EDN,OU=accounts,DC=corp,DC=example,DC=com|manager=CN=ewan
watson,OU=EDN,OU=accounts,DC=corp,DC=example,DC=com
```

3. Run identitySync.exe in the redo mode. Check the manager is changed in the Active Directory in this case Ewan Watson

4. Run identitySync.exe in the undo mode to replace the original manager Bonnie Burns

5. Change identityReportAD1.csv back to its original value

If the manager's record cannot be found, the manager will be blanked.

1. Update a manager in the identityReportAD1.csv using a non-existent account name

2. Run identitySync.exe in the analyse mode. Check the Action.txt file to see that the manager Distinguished Name is blanked.

3. Run identitySync.exe in the redo mode. Check the manager has been removed in the Active Directory

4. Run identitySync.exe in the undo mode to replace the original manager

5. Change identityReportAD1.csv back to its original value

In a production situation the manager's employee ID might be used by the source of identities to indicate a person's manager. In this case:

1. The attribute flagged by the ManagerID flag in the CSV schema indicates the attribute that contains the employeeID of the person's manager.

2. The ManagerJoin is employeeID which is used to search for the manager's record with an employeeID matching managerID.

3. The attribute flagged by the Name attribute is updated by identitySync with the distinguishedName from manager's record and then used to synchronise with the manager in the Active Directory.

The identity schema would contain:

```
manager,managerEmployeeID;managerID
employeeID,employeeID;managerJoin
distinguishedName,distinguishedName;Name
,manager;ManagerName
```

The production identity source may not have the distinguishedName values. In this case identitySync copies the manager's distinguishedName from the manager's ActiveDirectory record, and the identity schema would contain:

```
,distinguishedName;Name
```