

Implementing Identity Management

AMX, an Access Management system can extract the identities from all these authoritative sources to find the joiners, movers and leavers, and transform them to a suitable format such that they can be used to load accounts into target resources such as the Active Directory, Exchange, LDAP, databases, Unix Systems, the Cloud etc. In its simplest form identity management is not an IT function and access management, which is, is a synchronisation process.

What you end up with

The end point of an AMX Identity and Access Management project is accounts on most of your infrastructure, in-house and cloud based being created, updated and disabled as individuals join, move and leave the organisation. The on-going effort is minimal, it consists of monitoring the processes by reading periodic emails to ensure the process is working as expected. Some Service Desk calls may need the AMX processes to be investigated, for example during an audit, the logs show when a recent leaver was disabled.

When roles are implemented and operating in strict mode, new responsibilities are added when a person changes roles and old responsibilities are removed at some predefined time later. This is role hysteresis which allows a short period when a person has access to resources of his old role as well as his new, subject to separation of duties rules.

Implementation Process

Every situation is different, the major difference when implementing AMX compared with other IdM systems is that no development environment is needed. The implementation is done immediately on the live environment because AMX is safe, predictable and reversible. It is safe because it creates a transaction file which it inspects and will not proceed if changes exceed pre-set limits. It is predictable because the transaction file can be reviewed to check any updates that AMX has found. Initially there may be a lot of changes if account management was done manually, with inconsistent formatting and typos. Most adapters can create batch files to manually update managed resources, and at least initially using them may be preferable to having AMX make the changes.

The basic steps are:

Setup

1. Obtain high level management sponsorship for the project.
2. Survey the environment. Identify the authoritative sources of identities and the managed systems.
3. Arrange for access to these systems with unprivileged read-only accounts. This often non trivial when the systems are managed by diverse parts of the organisation. If this fails, start again at step one.
4. Identify the system that will run AMX, it does not need to be a massively powerful system and it does not need to be dedicated to AMX. AMX is trivial to install because it does not need databases or web application servers so it can be easily moved between systems.
5. Check the firewall settings between the system running AMX and the systems it is managing. In cases where the ports are blocked identityServer can be used as a proxy if an unused port is opened in the appropriate firewall. In the meantime install AMX on a system the other side of the firewall and use remote desktop to run it. Arrange for a port to be opened in the firewall, and if this fails, start again at step one.
6. If WSDL or Web API cloud based resources are used, check if an account for a web proxy is required.

Survey

7. Use tools like dbVisualizer and Apache Directory Studio to debug connections and identify attributes that have useful values to create a schema for the resource. Use the default Metaverse name which is the Staging name so that the schemas can be combined later. For example

```
, resource;nosync
sAMAccountName,
displayName,
employeeID,
distinguishedName,
name,
CN,
givenName,
initials,
sn,
l,
lastlogon,
mail,
manager,
memberOf,
```

8. Configure properties files for identityReport on each identity source and managed system individually. Use the same connection details as in the tools in step 7.
9. Run identityReport and resolve any errors and then review the reports for completeness.
10. Adjust the adapter filter attribute and value to extract the objects of interest, for example filter out service accounts.

Identify Unused Accounts

11. Review the reports in Excel or similar, sort by lastlogon or password last changed and identify accounts which are potentially no longer in use. Investigate and potentially move them aside.

Design the Metaverse

12. Load the schema files as columns in an Excel Spreadsheet by cutting and pasting. Choose a common Metaverse name, note that only attributes that are common in the Metaverse are synchronised.

AD corp	MetaVerse		HRIS	MetaVerse
	resource;nosync			resource;nosync
userAccountControl	active		termination_id	active
sAMAccountName	accountName		accountName	
displayName				displayName
employeeID			employee_id	employeeID
distinguishedName				distinguishedName
name				name
CN				CN
givenName			emp_firstname	firstName
initials			emp_middle_name	
sn			emp_lastname	lastName
	preferredName		emp_nick_name	preferredName
l	location			location
lastlogon				lastlogon

mail				mail
manager			mgr.erep_sup_emp_number	manager
memberOf				memberOf

13. Update the schemas with the Metaverse names and re-run identityReport.

Check Joins

Identity Management’s dirty little secret is that to manage accounts the identities of their owners must be established. In some situations this is already in place. AMX has tools to check the joins and report any ghost accounts, that is accounts that are active but have no owners.

If account matching is in place skip this step, otherwise run accountMatch recursively to join accounts to their owners by matching sets of attributes. See the AccountMatch reference document for further information.

- 14. Use the pairs of reports, from the identity and managed resources with accountMatch. Transform attributes and test for matches, for example compare the last name and work address derived from phone number. accountMatch outputs a match file and two new files with the persons and accounts that it failed to match.
- 15. Recurse by changing the attributes or their transforms and repeat the previous step.
- 16. AccountMatch produces batch update files and a lookup file for identitySync to use.

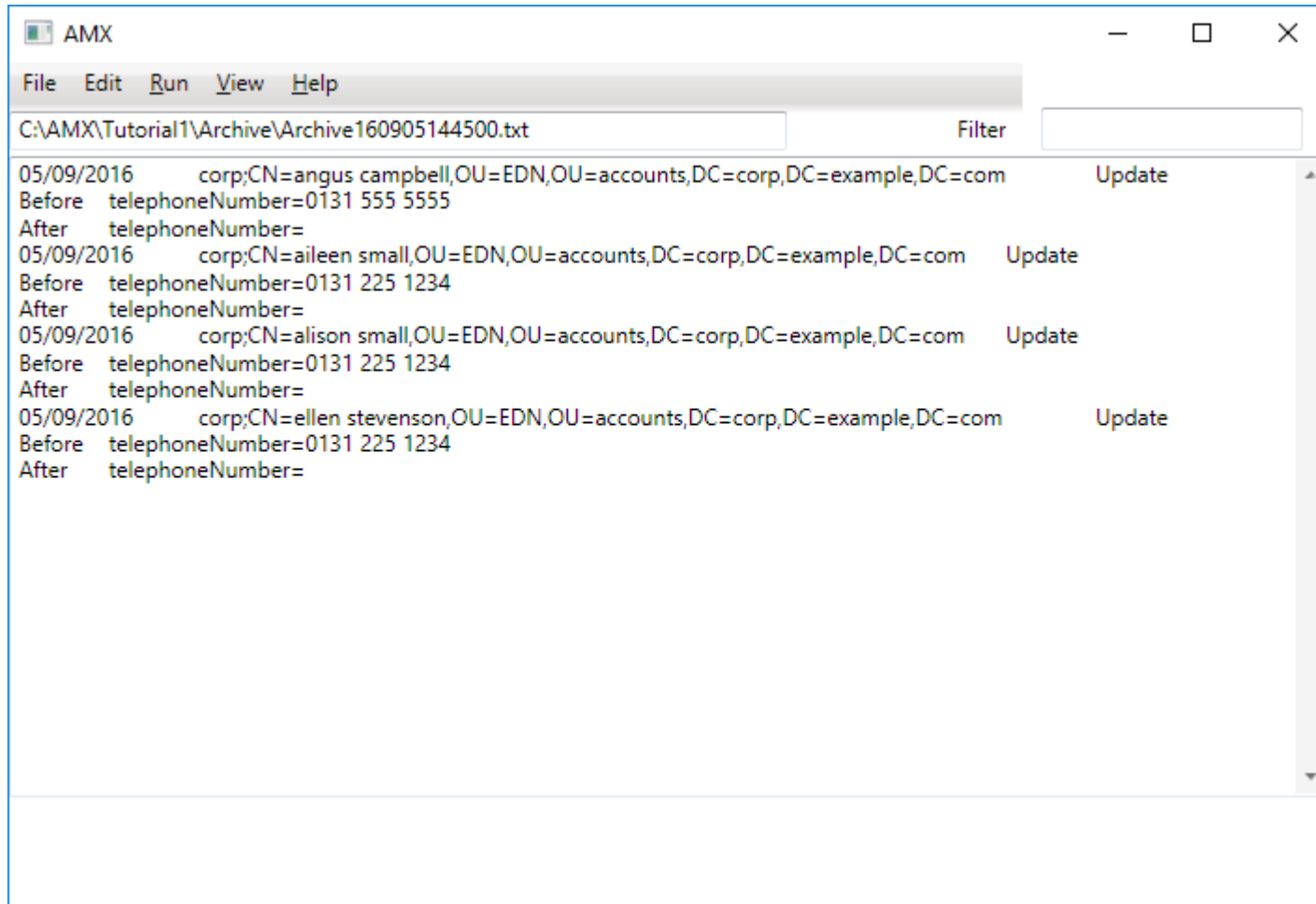
Develop Transforms

- 17. Start to add a resources into the main identityReport properties file for the source of identities and then each managed resource. Add the join attribute to the managed resource schema, this will be used as the sort for the report. Review groups of a person’s accounts and add transforms to the identity schemas to match the values of all the attributes. See the AMX Reference document for details of Transforms.
- 18. Not all attributes can be made to match with transforms, for example a person’s location may be flat wrong in a managed resource.

Check Differences

The difference engine uses the transforms developed when designing the Metaverse. At this point identitySync will report attributes that have values that are different and that you want to align.

19. Add the additional identitySync properties to the properties file, configure the manual update output files, and run identitySync in info mode. The transaction file will report the differences. To review the transaction file, either open ActionFile.txt or use AMXUser for a simpler view.



```
AMX
File Edit Run View Help
C:\AMX\Tutorial1\Archive\Archive160905144500.txt Filter
05/09/2016 corp;CN=angus campbell,OU=EDN,OU=accounts,DC=corp,DC=example,DC=com Update
Before telephoneNumber=0131 555 5555
After telephoneNumber=
05/09/2016 corp;CN=aileen small,OU=EDN,OU=accounts,DC=corp,DC=example,DC=com Update
Before telephoneNumber=0131 225 1234
After telephoneNumber=
05/09/2016 corp;CN=alison small,OU=EDN,OU=accounts,DC=corp,DC=example,DC=com Update
Before telephoneNumber=0131 225 1234
After telephoneNumber=
05/09/2016 corp;CN=ellen stevenson,OU=EDN,OU=accounts,DC=corp,DC=example,DC=com Update
Before telephoneNumber=0131 225 1234
After telephoneNumber=
```

Update Managed Resources

20. Review the manual update batch files created by identitySync and execute some or all of them as appropriate. The updates may include updates that make improvements to the join process by replacing the lookup file created by accountMatch. If account management was inconsistent, the updates may be a simple reformatting exercise, for instance normalizing the format of phone numbers.
21. Repeat this phase until updates are stable. This is the bulk of the implementation effort.

Identify and Create Roles

Skip this step if Role Based Access Control is out of scope. RBAC may be very high level to begin with and be developed to fine grained later. For example just manage roles such as Manager and non-Manager, or fine grained roles such as Sales Development, Customer Service Representative etc.

22. Create a Separation of Duties table for any responsibilities that should never be given to an individual. Run SODAnalyser on the managed resource reports to check that all existing responsibilities are correct. Resolve any discrepancies before proceeding.
23. Identify attributes in the sources of identity which defines the role a person plays in the organisation. For example a London based Sales Person.
24. Run RoleAnalyzer, see separate documentation for details. The resultant report will show all the well-defined responsibilities for the role, that is responsibilities that most of the persons performing the role have been assigned. If a small number of persons in the role have a missing responsibility, interview them to find out how they manage without it or what would be the security risk if they were given it.
25. Use the output from RoleAnalyzer to create template users in the managed resource or as lookup tables with an Excel spreadsheet of roles and their responsibilities for each managed resource.
26. Run identitySync in info mode, with roles in loose mode for each managed resource with RBAC. Loose mode adds but never removes responsibilities. See AMX Reference document for further details about role management.
27. Change either the roles by removing responsibilities or update individual account adding responsibilities using the batch files until responsibilities are in sync with the roles.
28. Repeat the last steps converting each role to strict mode as required. Strict mode will remove responsibilities

Run identitySync in production mode

29. When the managed resource updates are under control, increase the trip-wires in the properties file. Configure email reports and if required use accounts with administrative rights on the managed resources. Setup identitySync as a scheduled task and let it run, it will send emails when it finishes each run.

30. The first run should be a no-op.

31. Subsequent runs can be configured as:

- Cautious. identitySync is configured to run in info mode, it reports all updates by email but makes no changes to the managed resource. The administrator makes the changes indicated by identitySync either manually using the native tools or using the batch files created by identitySync.
- Careful. The tripwires defining the maximum number of changes are configured to a very low level. When a tripwire is activated identitySync makes no changes to the managed resource and the administrator can investigate the changes before committing any.
- Confident. The tripwires are configured to a realistic level, and AMX makes all the changes automatically.