Integrating ADMT with Priasoft

Priasoft

ADMT?

What is ADMT and why use it?

ADMT is short for Active Directory Migration Toolkit. It is a free, supported toolset from Microsoft for migration specific Active Directory settings from one Domain to another. During the development of an Exchange migration plan, there are often drivers to look at the migration of User account details beyond those of the Exchange Address book. A user account is also the security principal for an environment and holds the password and Security Identifier (SID) of a user. There can be a desire or need to then migrate the SID or Password of a user to a target environment.

While there are 3rd party tools that offer support for such activities, it may not be the wisest choice. Given that passwords and SIDs are part of the core security framework of an organization, proper protection and manipulation of such things is of great importance, and not just for protecting the end-user experience, but also for staying compliant for many government regulations and corporate or organizational policies.

As such, Microsoft's ADMT is usually the best choice for migrating and managing things that interact or can affect the security framework. If an issue arises, the fact that Microsoft's tool has been used means that support can be raised with the true "owner" of that framework – Microsoft. If a 3rd party tool is used, it can place a customer in between a "finger pointing" session between Microsoft and the 3rd party vendor.

Furthermore, given that ADMT can be scripted and can easily be integrated into Priasoft tools, there is a financial savings available since such 3rd party tools (specific to AD migration work) is not required.

Core Idea

Priasoft has long believed that success of tools in an enterprise is by its ability to be flexible inside the environment. This flexibility is shown by many options and the ability to execute scripts on certain events. Flexibility is further enhanced by our choice to not dictate the type or language of the script to use. As long as a script can be called, and optional parameters passed to it, our tools will be able to execute.

Specifically in the case of migrating passwords (or other options available with ADMT), an opportunity exists to create an atomic process that includes the migration of the mailbox, address book details, and the user's security attributes.

Prerequisites

There are some requirements to be able to create this atomic process with ADMT, the core of which will be outlined here.

1. ADMT must be installed and working in the target environment.
    a. In order to directly call ADMT command line tasks, ADMT will need to be installed on the same host(s) as the Priasoft migration tools.
    b. ADMT has a requirement of a Sever OS (2008 R2) in order for it to be installed. Server 2008 R2 is a supported platform for the Priasoft tools as well.
    c. ADMT has a requirement of a SQL Database instance which it uses for logging and tasking. SQL Express can be used, and for simplicity of deployment, SQL Express 2005, SP3 should be used. Any version of SQL server (2005 or greater) can be used, but the 2008 version has some installation issues. Full SQL server can also be used, and such allows for remote connections. SQL Express cannot be accessed by ADMT running on other hosts – such is a limitation of SQL Express.
2. Password Export Service (PES)
    a. This is a separate component of ADMT specifically designed to migrate passwords from one domain to another
    b. This service, another no-cost Microsoft product, must be installed on at least one writable DC in the source domain(s).
    c. A requirement for proper use of this component is a 2-way trust between the source and target domains.
    d. An encryption key file must be created from the ADMT host that is used by the Password Export Service. This ensures protection of the password details as it is pulled from the source domain and written to target objects. Note that the PES does not actually reverse the hashed value to a plain-text value; the service help provide the necessary data needed to create a compatible password in the target domain.
    e. There is a registry value that must be set after installing the PES as well as a required reboot (the service is supported by a special DLL that is loaded into the LSA when the computer starts up).
    f. After install and reboot, the PES service must be started. By default the service is set to “Manual” and so does not startup automatically.
3. A script should be used to do post-ADMT work on the target AD object(s)
    a. ADMT will set the target user properties such that after a successful password migration, the user will be marked as “Require new password at first logon” and “Password does not expire” will be explicitly unchecked.
    b. If those attributes should be adjusted, such must occur after ADMT processing.
    c. ADMT sets the values this way because it is not able to determine whether the current password will meet the complexity requirement of the target domain (again, it is not working with the actual password, but a special hash value).
    d. Validation must be made then to ensure that the target domain’s password policy or policies match or are less restrictive than the source.

Execution

There are several options on how execution can occur and how integrations are made between the Priasoft tools and ADMT. The first focus will be on atomic operations and such can be handled from 2 of Priasoft's core tools sets:

- PCS: Priasoft Collaboration Suite (a directory sync solution specific to Exchange migration efforts)
- MMM: Mailbox Migration Manager

Both of the toolsets above provide events at which scripts can be called to do additional work. Execution of scripts by these tools is simple and involves merely executing the script (or executable) and passing one or more command-line arguments to be consumed by the script or application. Due to the fact that the Priasoft tools are merely launching a program (all scripts have some executable program that does the work), ANY scripting platform is available for use. Common scripting languages available, and often pre-installed with Windows are: batch files (*.bat and *.cmd), VBScript (*.vbs), PowerShell (*.ps1). As already mentioned, executables can also be launched with command-line parameters so that, if needed, a custom application or unique scripting tool could also be used without special code from Priasoft.

PCS allows for scripts to be called before and/or after each object is processed and optionally once final event after all objects have been processed. MMM allows for scripts to be called before and/or after each mailbox is migrated. With either product, pre-work scripts always execute while post-work scripts execute only if the task (sync or migration) completed successfully. The Priasoft products support several replaceable tokens that can represent actual data which can be passed to a script or program. The attributes available come from either the source or target AD object, most commonly a unique value like 'distinguishedName' or 'Primary SMTP Address' is used. However, there is some ability to retrieve almost any attribute from Active Directory for the object being processed.

Creating an atomic process for use with ADMT is rather easy and would likely be a script that performs the following operations:

1. Parses command-line parameters
2. Creates an input file for ADMT identifying the source and target user account to process. Such is a simple text file with a header row and one line for the account to process.
    a. Header row: Sourcename,TargetSAM
    b. First row: sourceusername,targetusername
3. ADMT is called with necessary command-line parameters. An internally tested example of this is:
    a. ADMT user /sd:sourceDomain /td:targetDomain /po:copy /ps:sourceDCrunningPES /co:merge /ux:* /ix:* /f:input.file
    b. It should be noted that use of ADMT requires an account with high privileges, typically a Domain Administrator.
    c. In order to facilitate this requirement without adjusting the security of the current logged in user (which can cause issues for mailbox migrations), the "RUNAS" command can be used to run ADMT with a proper account.
    d. Prior to using RUNAS in the script, the command should be used once in a standalone capacity so that the "/savecred" option can be employed. This option will cache the password associated with the RUNAS command such that future executions of RUNAS will not prompt for credentials. This is a key attribute of developing the atomic process.

   e. The "/savecred" option above securely saves the password and is only used by RUNAS. Protections can be made by using a specific Domain Admin account for which can either be enabled/disabled at will or for which the password can be changed and re-changed. Either task will be sufficient to prevent unauthorized use of the cached credential.
4. After ADMT's processing completes, the next line in the script runs and would be used to update/modify the target user account, if necessary, to support business needs.
   a. By default, ADMT, when migrating a password, will set the target user with:
      i. Require new password at first logon will be "checked"
      ii. Password never expires will be "unchecked"
   b. If such details are undesirable, scripting can be used to update those settings. Powershell and VBScript are easy languages that can be used to manipulate AD attributes. ADModify.Net (another MS tool) is another that can easily make changes in AD.

When building any script, it is always a good idea to create some amount of logging so that auditing can be made on actions and troubleshooting of issues, if any occur.

Non-Atomic Execution

The other option to the use of ADMT, is by having scripted events generate another script that can be run once at the end of a batch or job. Since Priasoft tools are easy to call scripts, it is also not difficult to have a script that creates and appends to a text file with a long list of commands.

If, for instance, ADMT is found to be required to run on the target DC (migrating SIDs for example), then there may not be an ability to call ADMT directly. Having scripts create a script means simply copying the final script file (or having it write directly to the target DC's file system) to the DC running ADMT and then running from that host.