Windows Server Platforms at MIT - Containers

Windows Server Platforms at MIT - Containers

Introduction

In WIN, the win.mit.edu Domain, a computer container, or organizational unit, membership is defined within the MIT Moira system. A container defined by Moira may be either Private or Public. To add a machine to a Private container you must be a container administrator or ownership administrator for that computer container. In the case of a Public container, virtually anyone at MIT can add their machine to the container.

A machine owner, as defined by Moira, may always remove the machine from a container. It does not matter if the container is Public or Private. For most Institute-owned machines, only the container administrator or a Moira administrator may remove a machine from a container.

Before reading this document, you should first familiarize yourself with joining Clients to win.mit.edu. Reach out to ops-help@mit.edu to request a new container.

- Add or Remove a Machine from a Container
  - Stella (available on UNIX Athena and WIN machines)
  - Moira (available on UNIX Athena and WIN machines)
  - AD Container Management (available on WIN machines)
- Resources
  - Details on the AD Container Management Snap-In Interface
  - Quickstation Logon Script
  - Description of the OS Groups Service
  - Scripting Some Common Operations Using wscript
    - Sample mylogin.bat:
    - Sample CreateShortCuts.vbs:

Add or Remove a Machine from a Container

There are several tools you can use to add a machine to a container. They include stella, Moira and AD Container Management. They are available on many OSs, so use the most convenient - if you are closer to a UNIX Athena command line, use stella or moira. If you are closer to a WIN machine, use stella, moira or AD Container Manager, adcontmgr.

Stella (available on UNIX Athena and WIN machines)

To add a machine to a container using stella you must use the "-acn" options. For example, to add the machine "foo" to the container "test" you would type the following at the command line:

```
stella foo -acn Machines/test
```

Note: You must include the top level container name "Machines" in addition to the common container name "test." The container hierarchy is delimited by a forward slash "/." Also note that the container names are case-sensitive.

To confirm the container assignment you may use stella to list the container information as well:

```
stella foo -lcn
```

See more stella examples in the Joining Clients document.

Moira (available on UNIX Athena and WIN machines)

Moira is a curses-based client that displays text menu choices. The container menu is available from the top level menu, currently it is choice number 12. From the container menu you should choose "Machine to Container Mappings menu", currently choice 7. From this menu you can choose to add a machine to a container.

AD Container Management (available on WIN machines)

The AD Container Management tool is a GUI application written for Windows 2000 and XP, based on "Active Directory Users and Computers." It is an MMC snap-in tool, normally found under the "Administrative Tools" menu, or invoked with the command `adcontmgr.msc`. Note that not all user's profiles display the "Administrative Tools" menu by default. You can enable the display of this menu by adjusting the properties of your Windows Start Menu to show the "Administrative Tools" menu in the "Programs" menu.

If you have container administrative privileges, you run AD Container Management and do the following:

1. Log in to a WIN machine using your Athena Kerberos credentials.
2. Start the AD Container Management console.
3. In the left tree-like scope display pane, open the WIN.MIT.EDU domain.
4. Also in the scope pane open the **Machines** container.
5. Select the container to which the new machine will be added.
6. Right click the container name and select **Add Computer...**
7. Type the hostname machine_name and click **OK**.

Resources

Details on the AD Container Management Snap-In Interface

AD Container Management has extensive on-line help - see that material for ideas on Moira, AD, and the adcontmgr.msc user interface. There, learn about the scope pane, container items, computer items, actions such as add and delete, and associated properties like groups and policies.

Quickstation Logon Script

Type "quickstation" at a WIN command prompt for usage information.

This WIN program is meant run as a user logon script. It sits in the background and automatically logs off the calling user under certain circumstances. This can prevent users from tying up workstations for excessive amounts of time, useful in a multiuser environment like a cluster.

The usage info is fairly detailed, but here is a typical example:
`quickstation /run /idle:120 /sess:1440 /warn:5 /clock`
This logs off the user if the user is idle (no keyboard or mouse input) for two hours (120 minutes). Also, the user will be logged off after being logged on for 24 hours (1440 minutes), regardless of idle time. Five minutes before this 24-hour mark, a warning message will display telling the user that there are five minutes left, and to use this time to save data. Finally, a countdown window will tell the user how much time is left; this window can be minimized but not closed.

In getting this to run as a logon script keep in mind:

An easy way to run a logon script in your container is to use group policy, and enable `Computer Configuration\Administrative Templates\System\Logon\Run these programs at user logon...` and add a line like `quickstation.exe /run /idle:120 /sess:1440 /warn:5 /clock`. Keep in mind that group policy takes a few hours to propagate to machines, unless the machines are rebooted or you manually refresh GP. The real caveat here is that you should use only one Group Policy Object to enable this setting. If you use two, only the logon scripts specified in the one which takes precedence will run. You cannot concatenate the two script lists.

Alternatively, one can place a batch file in the AllUsers' Startup folder on each machine in your container... for instance, a file named "quick.cmd" which contains the line `start quickstation.exe /run /idle:120 /sess:1440 /warn:5 /clock`. It is important to include "start" so that the batch file will exit while quickstation runs in the background. The caveat here is that you have to do this to each computer in your container, either manually, or through some other automatic process (like a machine startup script).

Description of the OS Groups Service

A service named "OS Groups" runs in the domain. It automatically populates the following groups in Active Directory:

- Win2KPro.group : Machines running Windows 2000 Professional
- Win2KSrv.group : Machines running Windows 2000 Server
- Win2K.group : Machines running Windows 2000 Professional or Server
- WinXPPro.group : Machines running Windows XP Professional
- WinSrv2003.group : Machines running Windows Server 2003 (note, this OS is supported yet under test in the domain)
- WinOther.group : Machines running another OS or an unknown OS

Note: These are not Moira groups. They exist only in the Active Directory

When a new machine enters the domain or an existing machine upgrades its OS, it is automatically added to the proper group. These groups can therefore be placed on access control lists in Active Directory. This is especially useful for GPO application and MSI software installation, and it eliminates the need for separate containers for XP Professional, 2000 Professional, and 2000 Server machines.

As examples in win.mit.edu:

- Access to the existing Windows 2000 Service Pack 3 MSI deployment is denied for machines in WinXPPro.group and WinSrv2003.group.
- We deploy the Windows XP Service Pack 1 MSI to the whole domain, but access to this deployment is denied for machines in Win2K.group and WinSrv2003.group.

Using these examples means that Windows XP machines no longer download (and fail to install) Win2KSP3, and that Windows 2000 machines will never download (and fail to install) WinXPSP1.

Scripting Some Common Operations Using wscript

Users have long complained about the limitations of the Microsoft batch language. Windows 2000 and later operating systems include the Windows scripting host which provides much more power to users and system administrators interested in automating tasks.

The following script demonstrates the use of wscript to create some drive and shortcut mappings.

Sample mylogin.bat:

```
REM This script calls some .vbs (Visual Basic) scripts to

REM 1- create a shortcut to M:\My files the desktop

REM 2- create a shortcut to the MIT library home page on the desktop

REM 3- create a shortcut to L:\My links in Internet explorer favorites.

REM ----------------------------------------

REM call the .vbs script:

H:\WinData\MyScripts\CreateShortCuts.vbs
```

Sample CreateShortCuts.vbs:

```
'--------------------

'Create a ShortCut to M:\My files

'Create a WshShell Object

Set WshShell = Wscript.CreateObject("Wscript.Shell")

'Create a WshShortcut Object

strDesktop = WshShell.SpecialFolders("Desktop")

Set oShellLink = WshShell.CreateShortcut(strDesktop + "\My files.lnk")

'Set the Target Path for the shortcut

oShellLink.TargetPath = "Explorer"

'Set the HotKey to activate this link

oShellLink.HotKey = "ALT+CTRL+M"

'Set the icon ( path to your favorite icon)

oShellLink.IconLocation = "\\afs\athena.mit.edu\ \

 project\pismere\support\for-users\sample\icons\pismere.ico"

'Set the additional parameters for the shortcut

oShellLink.Arguments = "M:\My files"

'Save the shortcut

oShellLink.Save

'Clean up the WshShortcut Object

Set oShellLink = Nothing

'------------------------------------------------------------
```

```
' Create a ShortCut to the MIT Libraries on desktop:

'Create a WshShell Object

Set WshShell = Wscript.CreateObject("Wscript.Shell")

'Create a WshShortcut Object

strDesktop = WshShell.SpecialFolders("Desktop")

Set oURLLink = WshShell.CreateShortcut(strDesktop + "\MIT Libraries.URL")

'Set the Tragte URL

oURLLink.TargetPath = "http://libraries.mit.edu/"

'Save the shortcut

oURLLink.Save

'------------------------------------------------------

'Create a Short Cut to L:\My Links in the Internet explore favorites.

'Create a WshShell Object

Set WshShell = Wscript.CreateObject("Wscript.Shell")

'Create a WshShortcut Object

strFavorites = WshShell.SpecialFolders("Favorites")

Set oShellLink = WshShell.CreateShortcut(strFavorites + "\My Links.lnk")

'Set the Target Path for the shortcut

oShellLink.TargetPath = "L:\My Links\"

'Save the shortcut

oShellLink.Save

'Clean up the WshShortcut Object
```

```
Set oShellLink = Nothing
```

Related Links

- The win.mit.edu Domain
- Windows Server Platforms
- WIN.MIT.EDU Landing Page
- Windows Server Software Landing Page