Learn System Center Configuration Manager IN A MONTH OF LUNCHES

COVERS SCCM 1511 AND WINDOWS 10

JAMES BANNAN

SAMPLE CHAPTER

MANNING

Learn System Center Configuration Manager in a Month of Lunches

by James C. Bannan

Chapter 4

brief contents

4 Managing ConfigMgr devices and users

At the heart of ConfigMgr are devices: systems (either physical or virtual) that run an operating system supported by ConfigMgr and that have the ConfigMgr client installed. The client communicates regularly with the hierarchy of ConfigMgr servers, and performs tasks such as downloading and processing policies, reporting on hardware and software inventory, running operating system deployments, installing applications and system updates, and many more. As you can imagine, the health and reliability of your ConfigMgr environment is directly correlated with the overall health of the ConfigMgr agents deployed across your environment, so it's critically important to get this right.

But when you're immersed in the world of managing desktop and laptop systems, deploying applications, scheduling patches, and running inventories, it's remarkably easy to forget that most systems tend to have fleshy attachments called users who are (often surprisingly) the main source of productivity in your business. ConfigMgr can work with users directly, enabling some extremely advanced and intelligent management scenarios.

This raises the question: what's the importance of being able to manage users in ConfigMgr? Isn't that what Active Directory (AD) is for? Yes, but ConfigMgr doesn't seek to become a replacement user-management platform for AD, but rather to bring a much richer and more intuitive user-centric focus to the realm of systems management, using AD as a foundation.

As shown in figure 4.1, in this chapter you'll use the tasks undertaken in chapter 3 to successfully work with the ConfigMgr client and devices, as well as bring the lab users into the mix and establish formal relationships between users and devices via User Device Affinity (UDA). By the end of this chapter, you'll be ready to perform some advanced administration with your lab environment.

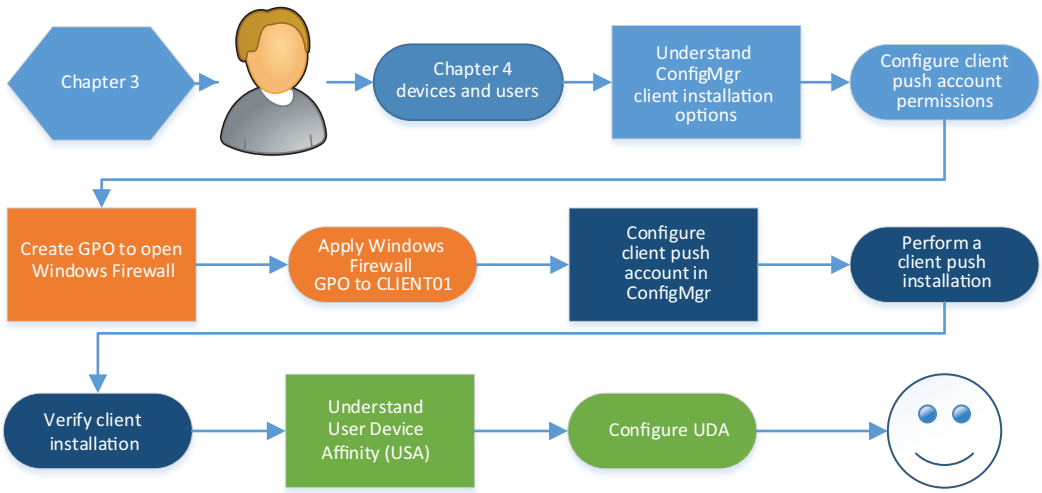

Figure 4.1 By the end of this chapter, the secrets of devices and users will be yours!

4.1 Understanding devices and the ConfigMgr client

In the world of ConfigMgr, a device can be a variety of systems, such as the following (not a definitive list!):

- A Windows 10 tablet
- A virtual Windows server
- A Windows 10 Virtual Desktop Infrastructure (VDI) instance
- A MacBook Pro running OS X
- A physical Ubuntu server
- An Android phone
- An Apple iPad

The platforms and operating systems are vastly different, but the principle is the same with all of them: once a management agent has been deployed to them, you can use ConfigMgr to manage them directly.

4.1.1 Installing the ConfigMgr client on remote systems

As already mentioned, the ConfigMgr client is a software package that needs to be installed on remote systems in order for them to be managed directly within the ConfigMgr environment. The client can be deployed and installed in a variety of ways,

which gives you a lot of flexibility about how to get the software out to the systems in your environment.

To install the ConfigMgr client on a remote system, follow these steps:

1. Log in to CLIENT01 as MOL\Administrator (make sure you use the full domain username) and then open File Explorer.
2. In the File Explorer address bar, navigate to \\CM01\SMS_PS1\Client. You should see the same content as in figure 4.2; this is the location of the ConfigMgr client source files and prerequisite files.

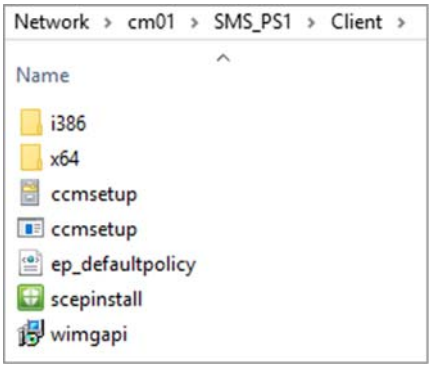

Figure 4.2 The ConfigMgr client software repository on CM01

This share maps to a local folder on the ConfigMgr server and is automatically created and populated when the server is installed. On any ConfigMgr server installation, there should always be a share at \\<SERVERNAME>\SMS_SITECODE\Client. This is one of those consistencies of ConfigMgr, which means that you always know where to find the client software. It's also a good troubleshooting tip: if you can't access that share from a remote machine, something's wrong and needs investigating.

Table 4.1 gives an overview of the most common methods of deploying the ConfigMgr client throughout your environment.

Table 4.1 ConfigMgr client installation methods

Installation method	Description
Client push	The client installer CCMSetup.exe is copied across from the ConfigMgr server to systems that have been discovered.
Group Policy installation	CCMSetup.msi is used to trigger the client installation via a Group Policy Object (GPO). This can be targeted at either systems or users (or both).
Logon script installation	CCMSetup.exe is called from within a logon script to install or reinstall the client.
Manual client installation	CCMSetup.exe is launched with the necessary installation properties from a local machine. All requisite files are automatically copied across.

Table 4.1 ConfigMgr client installation methods *(continued)*

Installation method	Description
Software update-based client installation	Publishes the client to a software update point. Remote clients use the built-in Windows update agent to retrieve or upgrade the client installation files.

Each method has its own pros and cons, and there's no one completely correct way of deploying the client. Most enterprise environments rely on a mixture of multiple methods to ensure that the client is successfully deployed throughout the organization. Time, experience, and familiarity with the processes will allow you to work out which is the best mixture for your organization.

For the purposes of this book, you'll walk through how to do the most common installation method, the client push.

4.2 Preparing for a client push

A client push installation requires the ConfigMgr server to be able to authenticate to the remote client in order to copy the software across and then trigger the installation. To enable this, the ConfigMgr server must be able to do the following:

1. Find the target computer on the network
2. Communicate directly to the target computer through the firewall (if any)
3. Authenticate to the target computer

The ability of one server to find, communicate with, and authenticate against a target computer isn't a process specific to ConfigMgr; it's one of the most common scenarios in enterprise computing. In the context of ConfigMgr, you need to specify a client push account that the server will use for the authentication process. The prerequisites in the lab environment are therefore as follows:

1. Set the client push account permissions in Active Directory.
2. Open the Windows firewall on the target system (CLIENT01).
3. Configure the client push account in ConfigMgr.

These prerequisites generally need to be done only once in your organization; they don't need to be repeated every time you want to perform a client push installation.

4.2.1 Set the client push account permissions

To assign the client push account the necessary permissions, perform the following steps:

1. On the Domain Controller DC01, open Active Directory Users and Computers, and then navigate to mol.sccmlab.net > MoL > Service Accounts. In this container are specialized user accounts, one of which is called CM_CP. The description given is the ConfigMgr client push account. This is the account you'll configure within the ConfigMgr console.

2. Right-click the CM_CP account and select "Add to a group."
3. In the "Enter the object names to select" field, type `Domain Admins` and then click "OK." A dialog box should pop up stating that the operation was successful.
4. Double-click the CM_CP account to open the properties and go to the Member Of tab. Two groups should be listed: Domain Admins and Domain Users.

Now that the account has sufficient permissions, it's time to configure it in the ConfigMgr console.

More on client push account permissions

In order to authenticate to remote systems, the CM_CP account needs to have local administrator access on remote systems. You can do this in various ways, but the easiest is to make the account a member of the Domain Admins group. On each domain-joined system, this group is automatically added to the local administrators group, which is why it's such a powerful AD group, and why access to it must be rigidly controlled. It's also the reason that many organizations don't like adding the Configuration Manager client push account to the Domain Admins group, preferring instead to use Group Policy Preferences to add the account to the local administrators group explicitly. For the sake of your lab environment, however, using the Domain Admins group is perfectly acceptable.

If you don't want to make the client push account a member of Domain Admins, but you do want to be able to push the client to Domain Controllers, you'll have to make the account a member of the BUILTIN\Administrators group in order for the account to have sufficient permissions, since Domain Controllers don't have local security groups.

Try It Now—Assign AD permissions

Follow the steps in section 4.2.1 to assign client push rights to MOL\CM_CP, in preparation for using this account for client push.

Consider your own organization. Would making the client push account a member of the Domain Admins group be acceptable? Is an internal security model in place that might preclude this? If so, how would you go about giving the push account sufficient local access to install the ConfigMgr client?

4.2.2 Open the Windows Firewall

Performing a client push requires that the ConfigMgr server can talk directly to the Windows operating system on the client. By default, the local Windows Firewall settings on the remote system will block the communication attempt, and the client push will fail.

You can get around this in various ways, but the approach you'll take is to create a Group Policy Object (GPO) in AD that will disable the Windows Firewall for domain-joined systems:

1. Log in to DC01 as MOL\Administrator and open the Group Policy Management utility.
2. Navigate to Forest > Domains > mol.sccmlab.net. Right-click MoL and select "Create a new GPO in this domain, and Link it here." Call the new GPO `SEC-Windows Firewall` (the *SEC* stands for *Security*, so if you create other security-based GPOs, they'll all appear in the GPO list in order). Click "OK."
3. Expand Group Policy Objects, right-click "SEC-Windows Firewall," and select "Edit."
4. In the Group Policy Management Editor, navigate to Computer Configuration > Policies > Windows Settings > Security Settings > Windows Firewall with Advanced Security.
5. Select "Windows Firewall with Advanced Security" again, and then in the right-hand side of the editor select "Windows Firewall Properties," as shown in figure 4.3.

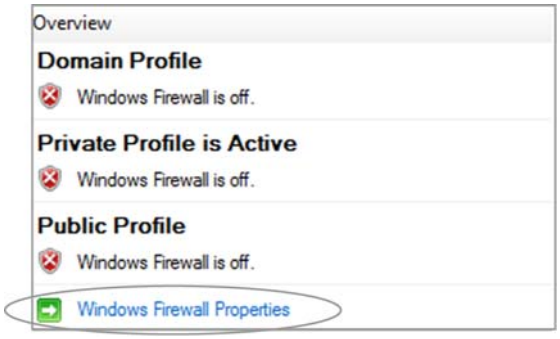

Figure 4.3 Create a Group Policy Object to change Windows Firewall settings.

6. On the Domain Profile tab, select "Off" from the "Firewall State" drop-down list.
7. Repeat this process on the Private Profile and Public Profile tabs. Click "OK" and then close the GPO Editor.

> **NOTE** Turning off all the firewall profiles is fine in your lab environment but would be unacceptable for most companies, unless they're using a third-party firewall product. Many enterprises disable the domain profile but leave the private and public profiles enabled, in which case you'll be able to deploy the ConfigMgr client via client push without having to worry about firewall configuration or Group Policy.

8. Next, log on to CLIENT01 as MOL\Administrator and open an elevated command prompt.

9 At the command prompt, type `gpupdate /force`. This forces the client to query AD for the latest Group Policy.

10 To verify that the changes have taken effect, navigate to Control Panel > Administrative Tools > Windows Firewall with Advanced Security. The firewall properties should look like those in figure 4.4, with the firewall turned off for all network profiles.

Figure 4.4 Windows Firewall turned off using Group Policy

Now that the environment is ready, you can move on to the ConfigMgr component of the push.

> **Try It Now—Configure Windows Firewall**
>
> Follow the steps in section 4.2.2 to create a GPO that will disable the Windows Firewall, and then apply the GPO to CLIENT01 and verify that the changes were successful.
>
> Think about the approach taken in your own organization. Is the Windows Firewall enabled or disabled? If disabled, are all network profiles turned off or only the Domain profile? How are the firewall settings distributed to managed machines?

4.2.3 Configure the client push account

To configure the client push account, in the ConfigMgr console, do the following:

1 Navigate to Administration > Site Configuration > Sites.

2 Right-click the PS1 site and select Client Installation Settings > Client Push Installation.

3 Navigate to the Accounts tab. Click the sunburst icon next to "Client Push Installation accounts," and then choose "New Account."

4 In the Windows User Account window, type in `MOL\CM_CP` for the username, and `P@ssw0rd` for the "Password" and "Confirm Password" fields.

5 Click the "Verify" button to check the details. Leave the "Data Source" field as "Network Share." In the "Network Share" field, type `\\mol.sccmlab.net\SYSVOL` (don't click "OK" just yet). This is a default share that's created in an AD environment, and that should be accessible to any authenticated user. The window should look like figure 4.5.

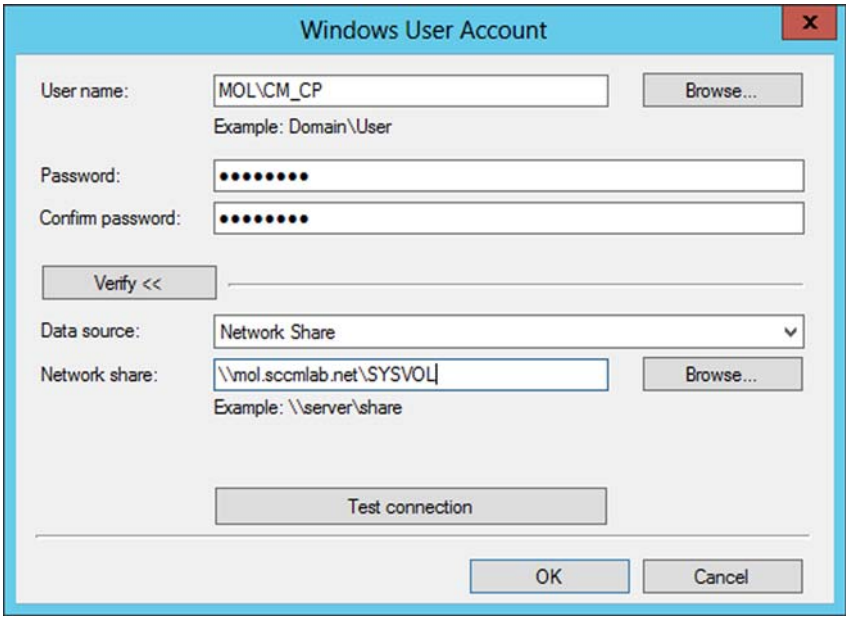

Figure 4.5 The new ConfigMgr client push account

6 Click the "Test connection" button, and a message should pop up stating that the connection was successfully verified.

7 Click "OK" and "OK" again to save the new client push account.

Now that the prerequisites are in place, you can perform the push installation of the ConfigMgr client.

Try It Now—Configure the client push account

On CM01, go through the process of configuring the client push account.

4.3 Performing a client push

What happens when you perform a client push? Figure 4.6 details what's happening behind the scenes, and where the process can fail.

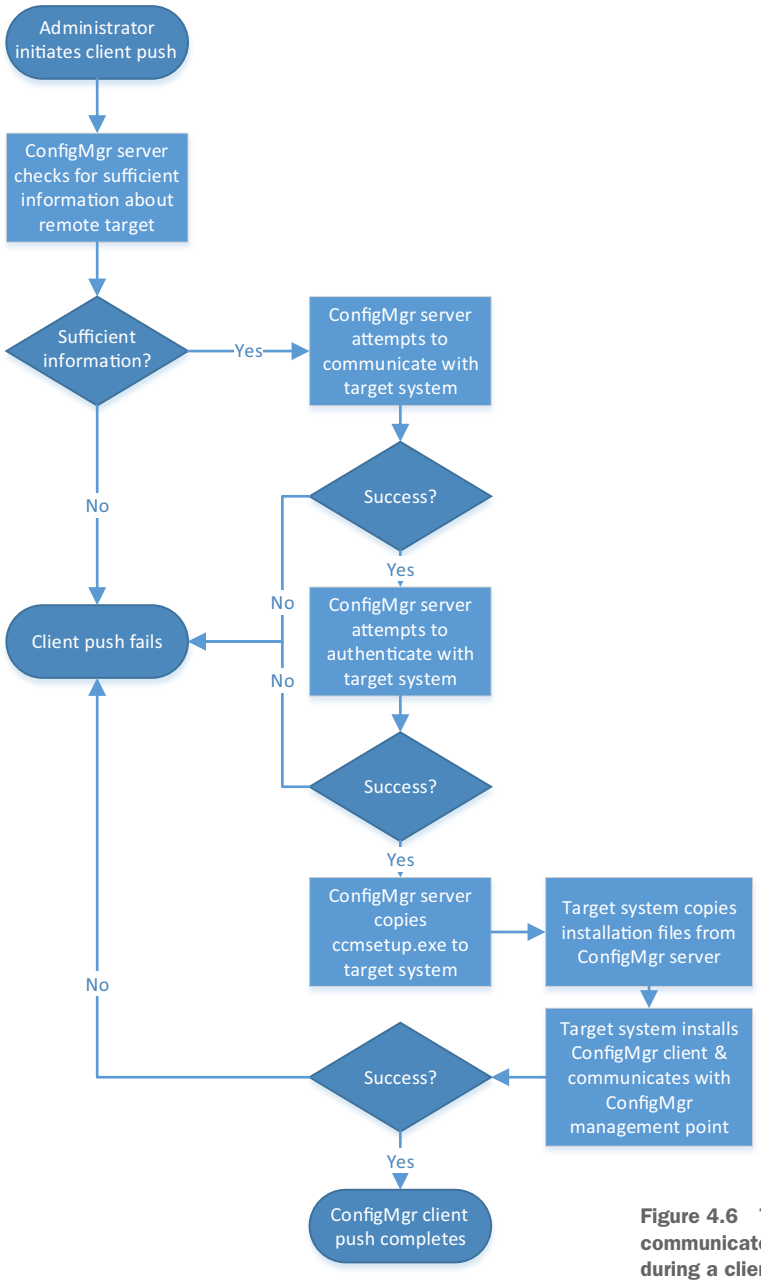

Figure 4.6 The ConfigMgr server communicates with target systems during a client push.

Now for the moment of truth. Because this is an important process, you'll want to have a look at the inner workings to see it in action, so let's quickly set that up:

1. Log on to CLIENT01 as MOL\Administrator. Right-click the taskbar and select "Task Manager."
2. In the Task Manager window, click the down arrow next to "More Details" to expand the view. The default tab is the Processes view; this is the one you're interested in.
3. On CM01, open ccm.log with CMTrace. This is the log file that will be written to during the client push, so you can watch the process happening in real time, as the push process is largely hidden when using the ConfigMgr console.
4. Navigate to Assets and Compliance > Devices. Right-click "CLIENT01" and select "Install Client."
5. The Install ConfigMgr Client window opens. Click "Next." On the following screen, select "Install the client software from a specified site." Click "Next" again.
6. Click "Next" again and then switch to CMTrace. You'll see the ConfigMgr server authenticate to CLIENT01 using the CM_CP client push account. As shown in figure 4.7, the server will copy across the ccmsetup.exe installer and the MobileClient.tcf file. This last file contains the installation properties that the ccmsetup.exe file will look for during the installation.

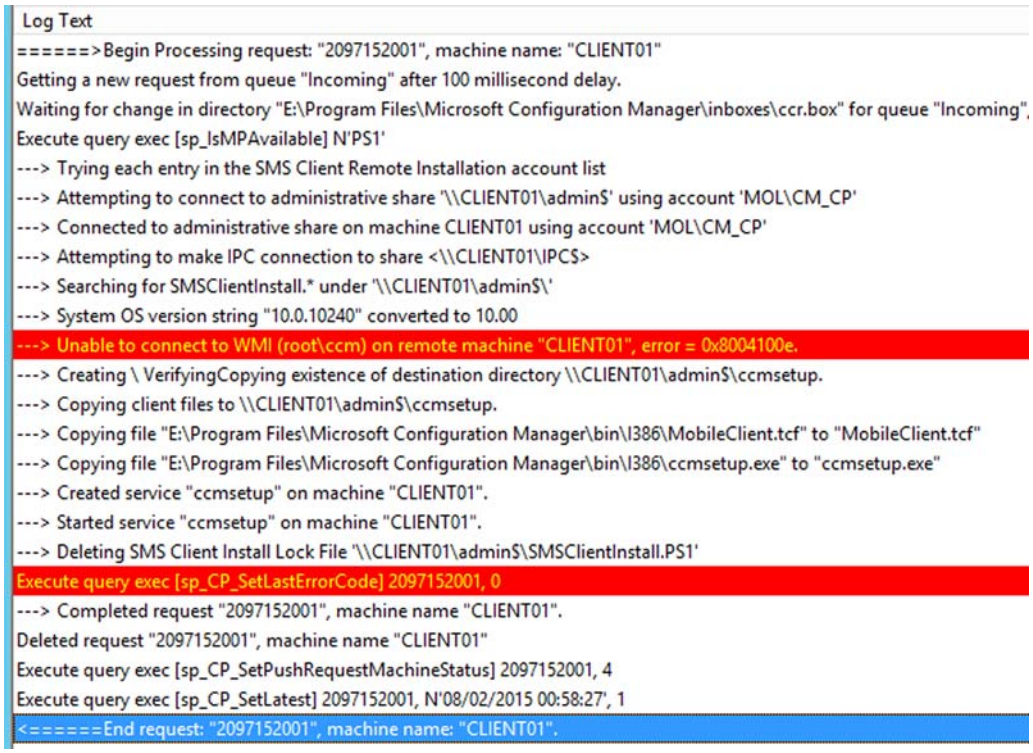

Figure 4.7 The ConfigMgr server pushing the client installer to a remote machine

7 Now switch back to CLIENT01 and take a look at the Task Manager. You'll see that under Background Processes is a new entry called ccmsetup.exe (32 bit). This is the ConfigMgr client installer that has been copied to CLIENT01 and has been triggered to install silently.

WHICH LOG? The client installation process will be logged to ccmsetup.log, which is located in %WINDIR%\ccmsetup\logs on any machine on which the client is installed.

Depending on the speed of the system, the installation will run for several minutes before the ccmsetup.exe entry disappears from the list of background processes to be replaced by a new one called "Host Process for Microsoft Configuration Manager," as shown in figure 4.8. Expand the process and you'll see that the process is underpinned by a Windows Service called SMS Agent Host.

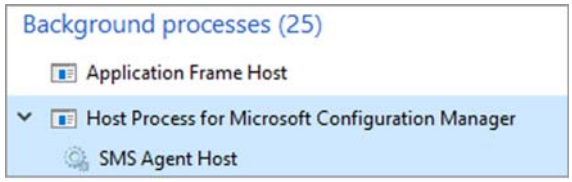

Figure 4.8 The ConfigMgr client has been successfully installed and is now running.

8 Navigate to Control Panel and launch the ConfigMgr utility. You should see that the client is installed, is attached to the PS1 site, and is looking at CM01 as the assigned Management Point, as shown in figure 4.9.

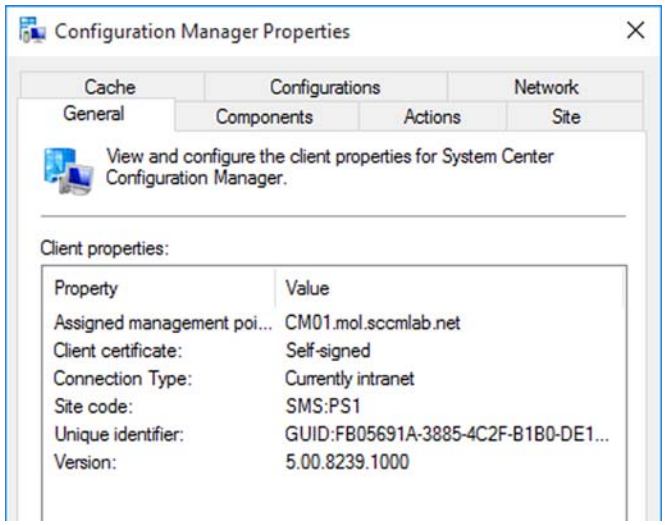

Figure 4.9 The ConfigMgr client on CLIENT01

9 Finally, switch back to CM01 and navigate to Assets and Compliance > Devices. You'll see that ConfigMgr is now reporting that CLIENT01 has a client, is assigned to site PS1, and that the client is active.

Congratulations, you've just performed a successful client push and now have a managed system!

4.4 Working with users in the ConfigMgr world

ConfigMgr 2012 introduced the concept of user and device equivalence. In the world of ConfigMgr, users are now as important as devices when it comes to targeted management. This concept needs to be taken with a grain of salt. The vast majority of your targeted administration will be to devices, because that's where the active agent sits. As tempting as it sometimes might be, you can't deploy an agent on a user—not yet, anyway.

But the concept is sound in principle, and is borne out by the way that users and devices are organized in the ConfigMgr console. Open the console and navigate to Assets and Compliance. As in figure 4.10, you'll see that users and user collections are placed at the same level as devices and device collections. It might seem like a minor thing, but in ConfigMgr 2007, users didn't receive the same level of prominence, and were generally swamped by device collections.

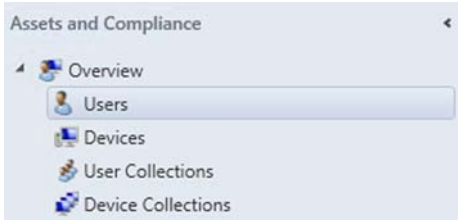

Figure 4.10 Users and devices at the same level in the ConfigMgr console

Let's look at an example of why this equivalence is now so important and powerful. Say that a user has requested access to an application that isn't available by default throughout your organization, such as Microsoft Visio. This application normally isn't installed everywhere because it's licensed separately from the rest of Microsoft Office, so installing it incurs an additional cost.

As the ConfigMgr administrator, you have the Microsoft Visio application ready to be deployed, but which is the best way to target the deployment? Should you target the user or the user's machine? A device-centric management approach introduces some awkward compromises:

- It makes sense to target the user, given that was where the request originated. But the deployment will follow the user around regardless of where that user logs on, which means you could have multiple, unlicensed copies of Microsoft Visio being distributed throughout your environment. Not good.

- You can avoid the preceding situation by targeting the user's system, even though doing so isn't reflective of the business nature of the original request. The downside of this approach is that you have to know which system belongs to the user, and if that system is lost, stolen, or in any other way has to be replaced, you have to keep track of which additional software packages had been deployed to it so that you can replicate them on the user's new machine. This is all feasible, but it adds a level of management and administrative overhead that's time-consuming and, as you'll see, now unnecessary.

A better option is one that reflects the business-centric nature of the request, while eliminating the chance of spreading software beyond where it's needed. You can achieve this in ConfigMgr by focusing on the user and defining the relationships between users and the devices they use. This is called User Device Affinity, or UDA.

4.5 Creating relationships between users and devices

UDA allows you to define Primary users for devices, Primary devices for users, or any combination of the two. For example, as shown in figure 4.11, you can assign a primary user to one particular device: the machine that the user always logs on to and that's assigned to that user in your company's asset register. Or you could assign multiple primary devices to a single user, such as a designer who has a desktop, a laptop, and a slate all assigned to them. Alternatively, you could assign multiple primary users to multiple primary devices, such as a team of interns who don't have machines assigned to them, but who rather make use of a pool of systems.

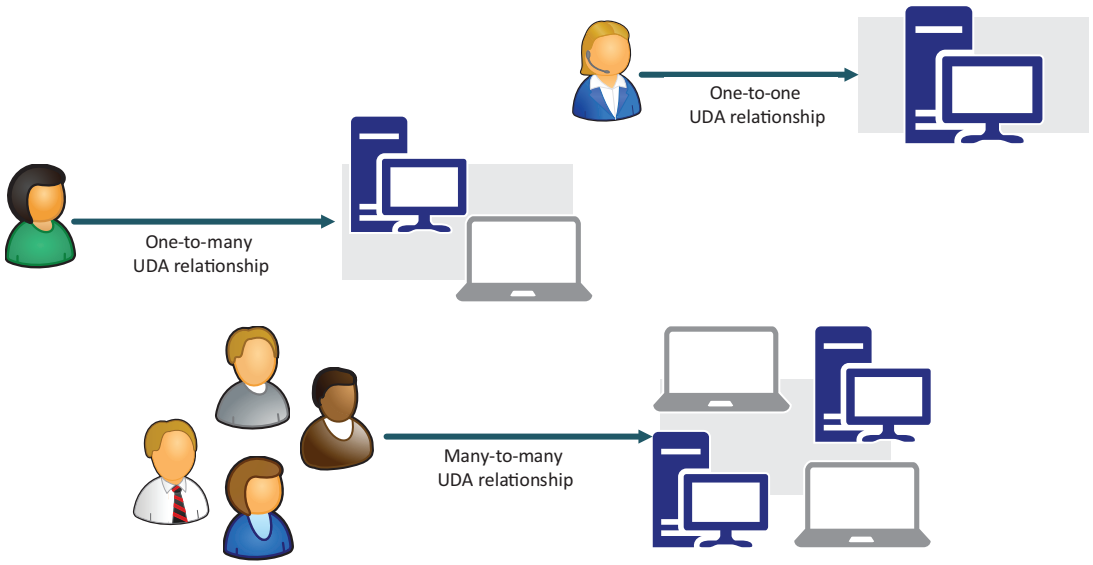

Figure 4.11 You can define a variety of relationships between users and devices.

The importance of UDA is that it allows you to accurately reflect how your managed systems are used by real-life users. Let's go back to the example you looked at earlier concerning Microsoft Visio, except this time let's flesh it out a little.

4.5.1 Understanding a UDA scenario

You receive a request from Alex, a manager in the business. Alex wants access to Microsoft Visio 2016, and has already gone through the necessary processes to purchase a license. As the ConfigMgr administrator, you already have access to Visio 2016 as an application that can be deployed. You want to ensure that Alex gets the software he's asked for in a quick, user-friendly way, so you deploy the Visio application directly to Alex and direct him to open the Application Catalog. Alex now sees Microsoft Visio in his list of available software, and clicks "Install." This triggers an instant policy refresh on the ConfigMgr client, and Visio downloads and is installed.

The following day, Alex forgets his laptop at home, and so logs on to his colleague's system, as his colleague is on training all day. This system doesn't have Visio installed, so Alex goes back to the Application Catalog and installs Visio. This time, Alex is presented with an error, and the software doesn't download or install. What happened?

As the administrator, you assigned Alex's user object in ConfigMgr as the primary user of the laptop that has been allocated to him, and made the primary user/primary device relationship a prerequisite of the software deployment. The ConfigMgr deployment process understands UDA, and because Alex isn't logging on to his primary device, the software installation fails.

Now it's time to look at how to configure UDA.

4.5.2 Creating UDA relationships

ConfigMgr provides various ways to establish a relationship between a user and a device, and all of them have their place for the ongoing maintenance of UDA. Table 4.2 details each of the options.

Table 4.2 Ways to establish User Device Affinity relationships

UDA method	Description
Administrator-defined	The ConfigMgr administrator manually defines the primary user(s) or primary device(s) on a per user/device basis via the console.
Bulk import	The ConfigMgr administrator maintains a CSV file of users and devices, and imports this data into the console.
Client settings	The ConfigMgr client tracks who logs on to the local system, how often, and for how long. When a usage threshold is reached, the client automatically makes the user a primary user of the machine.
OSD variable	During an operating system deployment (OSD) task sequence, the variable `SMSTSUdaUsers` can be called to create one or more primary users of the machine being deployed.

Table 4.2 Ways to establish User Device Affinity relationships *(continued)*

UDA method	Description
User-defined	If the user has the necessary permissions (which are denied by default), they can nominate themselves to be a primary user of the system that they're currently logged onto.

The method you'll focus on for now is administrator-defined, since this best exposes the inner workings of UDA.

Navigate to Assets and Compliance > Devices, right-click "CLIENT01," and select "Edit Primary Users." The window that opens shows you some of the potential candidates for primary users—users who have logged on to the machine recently, as shown in figure 4.12.

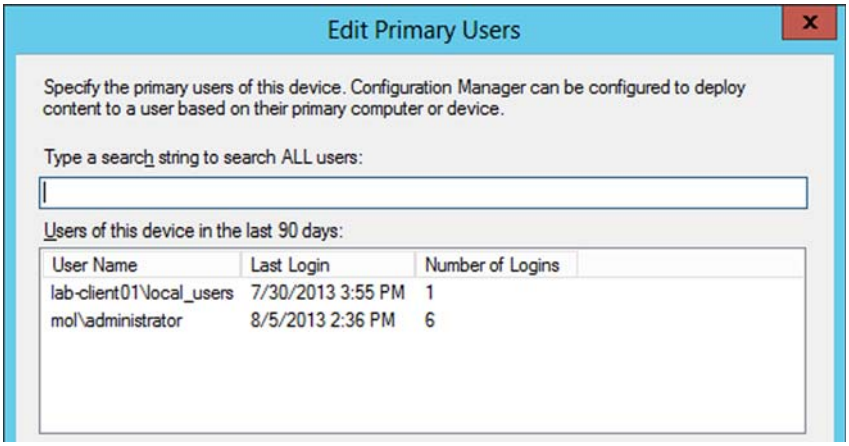

Figure 4.12 Potential primary users of CLIENT01

Rather than select any of the available options, in the field under "Type a search string…" type in `MOL\Mike`. This will perform a live search against all ConfigMgr users, and the entries in the user list will be replaced with MOL\Mike. Select this user and choose "Add." The user is now added as an Administrator Defined Primary User of CLIENT01, as shown in figure 4.13.

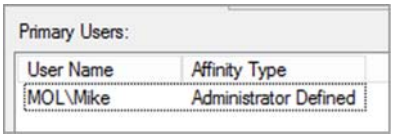

Figure 4.13 An Administrator Defined Primary User of CLIENT01

Creating a primary user relationship for a device is the same as creating a primary device relationship for a user. To verify this, go to users, right-click "MOL\Mike," and select "Edit Primary Devices." As shown in figure 4.14, by defining a primary user for CLIENT01, you've also assigned a primary device to Lab User 01.

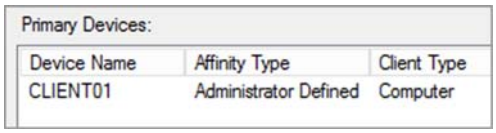

Figure 4.14 Defining a primary user automatically creates a primary device.

You can perform the same administrative tasks with PowerShell. To create a user affinity directly to a device, use the following command:

```
Add-CMUserAffinityToDevice -DeviceName 'CLIENT01' -UserName 'MOL\Mike'
```

Alternatively, to add a device affinity directly to a user, run the following:

```
Add-CMDeviceAffinityToUser -DeviceName 'CLIENT01' -UserName 'MOL\Mike'
```

Congratulations! You've successfully configured user and group discovery as well as established a relationship between a user and a device. You'll be able to use this relationship later in the book for some fancy application deployment.

4.6 Lab

ConfigMgr isn't just used to manage systems running client versions of Microsoft Windows; it's also used to manage servers, particularly in the area of inventory and patch management.

In this lab, you're going to deploy the ConfigMgr client to the Domain Controller DC01 by using a client push. By default, a client push stops if it detects that the remote system is a domain controller, as many customers don't like the idea of having a management agent on a critical piece of AD infrastructure. But in this lab environment, it's perfectly safe, and as the book progresses you'll be making changes and implementing new configurations that will serve to isolate and protect DC01 from incurring any accidental changes or deployments.

To successfully push the ConfigMgr client to DC01, here are the steps you'll need to perform.

4.6.1 Discover DC01

Navigate to Assets and Compliance > Devices, and you'll see that DC01 isn't there. That's because when you configured the Active Directory System Discovery method, the AD container that you specified (mol.sccmlab.net/MoL/Workstations) doesn't contain DC01.

To have the discovery method find and import DC01 into ConfigMgr, you need to add the container that DC01 resides in as a second discovery path. All domain

controllers are created by default in the domain controllers container, so in your lab the container is mol.sccmlab.net/Domain Controllers.

Add this path to the Active Directory System Discovery method and ensure that DC01 appears in devices.

4.6.2 Disable the Windows Firewall on DC01

Just as with CLIENT01, the Windows Firewall settings on DC01 prevent ConfigMgr from copying across the client installation files, so in this lab you need to disable the firewall.

You can use the same GPO that you created earlier in this chapter. But at the moment, the GPO is linked to the mol.sccmlab.net/MoL container in which DC01 doesn't reside.

So, using the Group Policy Management tool on DC01, you need to link the SEC-Windows Firewall policy to the mol.sccmlab.net/Domain Controllers container as well.

After this is done, run `gpupdate /force` on DC01 and open the control panel to verify that the Windows Firewall has been turned off.

4.6.3 Push the ConfigMgr client to DC01

Using the same client push method that you used to deploy the client to CLIENT01, it's time to deploy the client to DC01.

You've already configured the client push account, and because it's a member of Domain Admins, you won't need to do anything there.

The one thing that's different with this client push is that by default, the client won't deploy to a domain controller. The Client Push Wizard includes an option to override this behavior, so look at all the options carefully and choose the right one. Then, push the client out and watch the ccm.log on CM01 and the processes in the task manager on DC01 to ensure that the copy and installation is processed correctly.

Finally, check devices to ensure that the agent on DC01 is talking back happily to CM01.

MICROSOFT/WINDOWS

Learn System Center Configuration Manager in a Month of Lunches

COVERS SCCM 1511 AND WINDOWS 10

James Bannan

Businesses rely on a complex patchwork of client computers, physical and virtual servers, middleware, mobile devices, and cloud services. Microsoft System Center Configuration Manager sits in the middle of this mix, providing a single administrative control center to deploy and manage Windows servers and applications across your entire infrastructure, including cross-platform management of Mac OS X, Linux, and UNIX. To get up to speed with the day-to-day tasks of managing a system with ConfigMgr, all you need is this book—and a quiet place to eat your lunch.

Learn System Center Configuration Manager in a Month of Lunches is a super-practical guide to Microsoft System Center Configuration Manager. In this book, you'll cut to the chase and learn the administrative procedures and techniques that will keep your systems humming smoothly. Whether you're a new sysadmin or you already understand the inner workings of Active Directory and Windows Server, you'll be productive immediately as you work through the 22 self-contained lessons in this handy tutorial.

WHAT'S INSIDE

- Covers the latest build of Configuration Manager
- How to simplify updates, operating system deployment, and reporting
- Cross-platform and mobile management including Linux, OS X, and Windows
- Smart application delivery

No prior experience with System Center Configuration Manager needed.

James Bannan is a Cloud and Datacenter Management MVP based in Australia.

"The single most useful consolidated source of SCCM guidance I've found."
—Francis Setash
US Department of State

"The best book that explains SCCM with the least amount of theoretical jargon."
—Nasir Naeem, Interserve Plc

"A one-stop shop for ConfigMgr admins."
—David Moravec
Mainstream Technologies

"A nice, gradual, and hands-on introduction to SCCM ... will bring you to operational status in no time."
—Alain Couniot, STIB-MIVB

"The quickest way to learn SCCM!"
—Joseph Moody
DeployHappiness.com

To download their free eBook in PDF, ePub, and Kindle formats, owners of this book should visit manning.com/books/learn-system-center-configuration-manager-in-a-month-of-lunches

ISBN-13: 978-1-61729-168-5
ISBN-10: 1-61729-168-4
54499
9 781617 291685

MANNING US $44.99 / Can $51.99 [INCLUDING eBOOK]