2017

System Center Configuration Manager

ADMINISTRATION E-BOOK

BY RAPHAEL PEREZ, MICROSOFT MVP IN ENTERPRISE MOBILITY
MVP PROFILE: https://mvp.microsoft.com/en-us/PublicProfile/4027143
TWITTER: @DOTRAPHAEL
LINKEDIN: https://uk.linkedin.com/in/dotraphael
BLOG: http://thedesktopteam.com/raphael
COMPANY: http://www.tucandata.com

Version: 2.00 | Date: February 2017

Table of Contents

1. Document Change Control Sheet 9
1.1. Document History 9
2. About 10
2.1. Raphael Perez (Author) 10
2.2. Niall Brady (Reviewer) 10
2.3. Panu Saukko (Reviewer) 10
3. Introduction 11
4. Lab Information 12
4.1. PowerShell 14
4.2. Installing a Hyper-V Server 15
4.3. Installing Hyper-V Role 16
4.4. Downloading Software 16
4.5. Creating Windows Virtual Machines 17
4.6. CLASSROOM-ROUTER01 17
4.7. CLASSROOM-SRV0001 18
4.8. CLASSROOM-SRV0002 18
4.9. CLASSROOM-SRV0003 18
4.10. CLASSROOM-WKS0001 19
4.11. CLASSROOM-WKS0002 19
4.12. CLASSROOM-WKS0004 20
5. Active Directory 21
5.1. Extending Active Directory Schema 21
5.2. Creating the System Management Container 21
6. Pre-Requirements 23
6.1. Installing .NET Framework 3.5 23
6.2. Validating .NET Framework 4.6 Installation 23
6.3. Installing Microsoft Remote Differential Compression 24
6.4. Installing Assessment and Deployment Kit (ADK) for Windows 10 1607 24
6.5. Logs Reading 25
7. SQL Server 27
7.1. Creating Firewall Rules for SQL 27

7.2. Installing SQL Server 2016 28
7.3. Installing SQL Server Management Studio 30
7.4. Validating Installation 30
7.5. SQL Server Max Memory 31
7.6. SQL Server Recovery Model for SQL Server Reporting Services Database 32
7.7. Validating Static Port and Services 32
8. Installation of SCCM Primary Site 34
8.1. Downloading SCCM Setup Files 34
8.2. Pre-requirements Check 34
8.3. Creating Firewall Rules for SCCM Site Server Console Communication 35
8.4. Site Server Installation 35
8.5. Installation Status 37
8.6. Console Overview 39
8.7. Validating Installation 42
8.8. Configuration Manager Toolkit 46
9. Upgrade to Current Branch 1610 47
9.1. Before you begin 47
9.2. Run the prerequisite check 48
9.3. Install an update 48
9.4. Turning on Features 50
10. Basic Site Configuration 51
10.1. Installation of basic Roles (DP, MP, FSP, SRS) 51
10.1.1. Distribution Point 51
10.1.1.1. Creating Firewall Rules 51
10.1.1.2. Install Requirements 52
10.1.1.3. Installing Site System Role 53
10.1.2. Management Point 55
10.1.2.1. Creating Firewall Rules 55
10.1.2.2. Install Requirements 56
10.1.2.3. Installing Site System Role 57
10.1.3. Fallback Status Point 59
10.1.3.1. Creating Firewall Rules 59

10.1.3.2. Install Fallback Requirements 60
10.1.3.3. Installing Site System Role 60
10.1.4. Reporting Services Point 62
10.1.4.1. Creating Firewall Rules 62
10.1.4.2. Installing Site System Role 62
10.2. Boundaries 65
10.3. Boundary Group 66
10.4. Distribution Point Group 66
10.5. Network Access Account 67
10.6. Discovery 67
10.6.1. Active Directory Forest Discovery 67
10.6.2. Active Directory System Discovery 68
10.6.3. Active Directory User Discovery 69
10.6.4. Active Directory Group Discovery 70
11. Basic Client Settings 72
11.1. Changing Default Client Settings 72
12. Client Installation 74
12.1. Windows Client Installation 74
12.1.1. Push Configuration 74
12.1.2. Manual Push 75
12.1.3. Validating the Installation and Installation Process on the Client 75
12.1.4. Validating the Installation and Installation Process on the SCCM Console 76
12.2. Linux Client Installation 77
12.3. Client Properties 78
12.4. Resource Explorer 78
13. Collections 80
13.1. Windows 10 Workstation Collection 80
13.2. Windows 8 Workstation Collection 81
13.3. CentOS Servers Collection 82
13.4. Collection Membership Incremental Evaluation 83
14. Remote Control 85
14.1. Creating Device Settings for Remote Tools 85

14.2. Checking the Policies that should be applied 86
14.3. Validating Remote Control 86
14.4. Starting Remote Control 87
14.5. Monitoring Remote Access from Client 88
14.6. Monitoring Remote Access from Server 88
15. Hardware Inventory 90
15.1. Changing Default Client Settings 90
15.2. Updating Client Policies 92
15.3. Starting Hardware Inventory 92
15.4. Resource Explorer 93
16. Client Health 94
16.1. Client Status Settings 94
16.2. Client Health Configuration 94
16.3. Executing CCMEVAL manually 95
16.4. Forcing CCMEVAL failure 96
16.5. Monitoring Client Health 97
17. Software Metering 100
17.1. Changing Default Client Settings 100
17.2. Updating Default Software Metering Settings and Clearing existing rules 100
17.3. Creating Rule 101
17.4. Starting Validation Software Metering 101
17.5. Summarization Software Metering Data Manually 102
17.6. Monitoring Software Metering Data via Reports 103
17.7. Monitoring Software Metering Data via Collections 103
18. Site Roles for User Centric Management 105
18.1. Application Catalog WebService Role 105
18.1.1. Creating Firewall Rules 105
18.1.2. Install Requirements 106
18.1.3. Installing Site System Role 106
18.2. Application Catalog WebSite Role 109
18.2.1. Creating Firewall Rules 109
18.2.2. Install Requirements 109
18.2.3. Installing Site System Role 110

18.3. Enable the new SCCM Software Center 113
18.4. Starting validation of the new Software Center 113
19. Primary Users 115
19.1. Changing Default Client Settings 115
19.2. Manual association by the Administrator 116
19.3. Manual association by the User 116
20. Application Management - Basic 118
20.1. Creating Application 118
20.2. Adding Requirements to Applications 119
20.3. Return Codes 119
20.4. Distributing Application Content to Distribution Point Group 120
20.5. Monitoring Application Content Distribution via Console 120
20.6. Monitoring Application Content Distribution via Reports 121
21. Deploying and monitoring Applications 122
21.1. Deploying Application 122
21.2. Installing available application 123
21.3. Monitoring Application Deployment via Console 124
21.4. Monitoring Application Deployment via Client Logs 124
21.5. Monitoring Application Deployment via Reports 125
22. Application Management – App-V 5 Applications 126
22.1. Creating an Application App-V Client 5.0 126
22.2. Creating a Virtual Application Robocopy 128
22.3. Visualizing Application Relationship 129
23. Deploying and monitoring available Virtual Applications to a User 131
23.1. Enable App-V in-box 131
23.2. Deploying Application 132
23.3. Installing available application 132
24. Application Management – Advanced 133
24.1. Preparing the Environment 133
24.2. Creating an Application Supersedence 137
24.3. Creating an Application Supersedence Deployment 138
24.4. Installing an Application Supersedence 139
24.5. Creating a Global Condition 139

24.6. Using a Custom Global Condition 140
24.7. Deploying an Approval Required Application 140
24.8. Requesting application via Software Center 141
24.9. Approving/Denying application requests 141
24.10. Installing an approved application 141
24.11. Simulate Application Deployment 142
25. Application Management for Linux 143
25.1. Creating a Package and Program 143
25.2. Deploying Application to a Linux 144
25.3. Installing Required Applications 145
25.4. Monitoring Application Deployment via Console 145
25.5. Monitoring Package Deployment via Reports 146
26. Software Update 147
26.1. Installation of Windows Server Update Services 147
26.2. Installation of the SCCM Software Update Point 148
26.3. Manual Synchronization 152
26.4. Changing the List of Products 152
26.5. Changing Default Client Settings 153
26.6. SCCM Client Scan 154
26.7. Reporting Compliance via Console for All Updates 155
26.8. Deploying a Patch 155
26.9. Installing a Patch 157
26.10. Reporting Compliance via Console for an Update Group 158
26.11. Reporting Compliance via Reports 158
27. Endpoint Protection 160
27.1. Configuring Software Update Point 160
27.2. Installing Endpoint Protection Point 161
27.3. Automating the definition updates delivery 163
27.4. Changing Default Client Settings 164
27.5. Testing Malware activity 165
27.6. Monitoring Malware Activity via Console 166
27.7. Monitoring Malware Activity via Reports 166
28. Compliance Settings 168

28.1. Changing Default Client Settings....168
28.2. Registry Configuration Items with Auto-remediation....168
28.3. Application Settings Configuration Items without Auto-remediation....170
28.4. Creating Baselines....171
28.5. Deploying Baselines....171
28.6. Starting Validation Compliance Settings....172
28.7. Monitoring Baselines from Client....173
28.8. Monitoring Baselines via SCCM Console....173
28.9. Monitoring Baselines via Reports....174
28.10. Monitoring Baselines via Collections....174
29. Role Based Access Control....176
29.1. Creating Security Scope....176
29.2. Using Security Scope....176
29.3. Creating an Application Administrator....177
29.4. Testing new Security Rights....177
30. Cloud Distribution Point (Windows Azure Integration)....179
30.1. Preparing the Environment....179
30.2. Windows Azure Calculator....180
30.3. Windows Azure Subscription....180
30.4. Changing Default Client Settings....180
30.5. Windows Azure Self Signed Certificate....180
30.6. Upload Certificate to Windows Azure....181
30.7. Adding Certificate to Enterprise Trusted store....181
30.8. Cloud Distribution Point Creation....181
30.9. DNS Configuration....183
30.10. Distribute Content to Cloud DP....183
30.11. Deploying an Application....183
30.12. Installing an Application using Cloud DP....184
31. Backup via SCCM....185
31.1. Configuring Backup....185
31.2. Starting Backup Manually....185
31.3. Monitoring Backup....186
32. Restore....187

32.1. Stop SCCM Site 187
32.2. Delete SCCM Database 187
32.3. Restore SCCM Site 188
32.4. Post-Restore Tasks 190
32.4.1. Accounts Password Reset 190
32.4.2. Distribution Point Self-Signed Certificate Reset 190
33. Appendix A – Tools 192
33.1. DataExplorer 192
33.2. HealthCheck Toolkit 192
33.3. CM12Automation 192
33.4. ConfigMgrRegistrationRequest 192
33.5. SCCM Client Center 192
33.6. Mark Cochrane RegkeytoMof 3.3a 192
33.7. System Center Update Publisher 2011 193
33.8. OSD WebPortal 193
33.9. Recast RCT Free 2.4 193
33.10. RuckZuck 193
33.11. Clean Software Update Groups console extension for ConfigMgr 193
33.12. Cireson Remote Manage app 193
33.13. Reg2CI 194
33.14. Collection Commander 194
33.15. SQL Server Index and Statistics Maintenance 194
33.16. PowerShell – SQL Audit Script 194
33.17. 1E's Free Tools 194
33.18. ConfigMgr Task Sequence Monitor 195
33.19. ConfigMgr OSD FrontEnd 195
33.20. Configuration Manager Web Frontend 195
33.21. Windows-Nood OSD FrontEnd 195
34. Appendix B – Unmissable Sites 196

1. Document Change Control Sheet

1.1. Document History

Date	Author	Version	Change/Reference
April/2016	Raphael Perez	1.00	Initial Release
February/2017	Raphael Perez	2.00	Updated to Windows Server 2016, SQL 2016 and SCCM 1610

2. About

2.1. Raphael Perez (Author)

Raphael is a 8 times Microsoft MVP (https://mvp.microsoft.com/en-us/PublicProfile/4027143) with over 20 years of experience in IT, of which 14 years have been dedicated to System Center and Automation.

One of three MVPs in Enterprise Client Management in the UK, Raphael holds more than 25 Microsoft certifications and is a MCT (Microsoft Certified Trainer). Since 2008, Raphael has been providing Microsoft training from basic to advanced levels in several categories.

Throughout his career, Raphael has participated as a speaker in well-known events such as TechEd and Gartner Security Risk Management. He also organised community events and lectured around the world, sharing best practices and knowledge within the industry.

Bilingual in English and Portuguese, Raphael has authored diverse articles published in Microsoft's TechEd, served as the editor-in-chief of a magazine focused on System Center in Brazil and wrote two books: "Understanding System Center 2012 SP1 Configuration Manager: The walkthrough book" (https://wp.me/p3ttD0-am and https://wp.me/p3ttD0-8S) and "System Center 2012 R2 Configuration Manager: Automation from Zero to Hero" (https://wp.me/p3ttD0-pd).

He is a Community leader, attending physical and virtual meetings and engaging with the community across several forums, twitter (http://twitter.com/dotraphael), LinkedIn (http://www.linkedin.com/in/dotraphael) and his blog (http://www.thedesktopteam.com/).

Raphael is Technical Director at TucanData Ltd (http://www.tucandata.com/), a company that provides extensions to enterprise applications, enhancing reports and data visualisation capabilities as well as consultancy and training services within the United Kingdom and has been working in several different System Center Configuration Manager and OS Deployment projects from small to enterprise environments across the UK.

2.2. Niall Brady (Reviewer)

Niall is an Irishman living in Sweden with 3 kids. He blogs about System Center Configuration Manager and Microsoft Intune. He's the guy behind https://www.windows-noob.com.

2.3. Panu Saukko (Reviewer)

Panu from Finland has trained and consulted Microsoft management products about 20 years. He has been MVP for 13 years. His Twitter account is http://tuwitter.com/panusaukko.

3. Introduction

This e-book has been created to provide you with a step by step information, so you can start understand a System Center Configuration Manager (SCCM) world. The intended audience of this e-book are technical people that want to learn or improve their understanding of SCCM. Minimum knowledge of the following software and technologies is assumed, including but not limited to Active Directory, SQL Server, Windows Server, Microsoft Deployment Toolkit, BitLocker, Hyper-V, and Windows Client (i.e. Windows 7, Windows 8, Windows 10). Knowledge of SCCM 2012 or early versions (including SCCM 2007 and SMS 2003) is beneficial.

It's recommended to use this e-book as it has been written because there are dependencies between the chapters.

Please note that the terms System Center Configuration Manager, ConfigMgr, Configuration Manager, CM and SCCM all refer to the same Microsoft product, and the terms are used interchangeably.

4. Lab Information

The System Center Configuration Manager lab environment was created using Hyper-V 2016 Virtual Machines connected to its own virtual network. The lab has seven (7) virtual machines installed on one (1) Hyper-V host, installed with default configuration, as per following configuration:

Virtual Machine	Hardware	Description	Base OS
HYPER-V	RAM: 24GB Drive 01 (C): 500GB Drive 02 (D): DVD Processor/Core: 4 Network Adapter	Hyper-V Server	Windows Server 2016 IP Address: DHCP
ROUTER01	RAM: 512MB Drive 01: 2GB Processor/Core: 1 Network Adapter Network Adapter	Linux router used to connect VMs to the internet	VyOS 1.1.3 External IP: DHCP Internal IP: 192.168.3.254 Internal Subnet 255.255.255.0 Internal DNS 192.168.3.1
SRV0001	RAM: 2048MB Drive 01 (C): 127GB Drive 02 (D): DVD Processor/Core: 1 Network Adapter	Domain Controller for domain called classroom.intranet (netbios name classroom), DNS, DHCP and Enterprise CA	Windows Server 2016 IP Address: 192.168.3.1 Subnet 255.255.255.0 Default Gateway: 192.168.3.254 DNS 192.168.3.1
SRV0002	RAM: 8192MB Drive 01 (C): 127GB Drive 02 (D): DVD Processor/Core: 2 Network Adapter	Site Server for ConfigMgr	Windows Server 2016 IP Address: 192.168.3.2 Subnet 255.255.255.0 Default Gateway: 192.168.3.254 DNS 192.168.3.1
SRV0003	RAM: 1024MB Drive 01: 127GB Processor/Core: 1 Network Adapter	Linux CentOS Server	CentOS 7 x64 IP Address: DHCP
WKS0001	RAM: 2048MB Drive 01 (C): 127GB Processor/Core: 1 Network Adapter	Windows 10 Enterprise Edition x64 – Workstation	Windows 10 x64 IP Address: DHCP
WKS0002	RAM: 2048MB Drive 01 (C): 127GB Processor/Core: 1 Network Adapter	Windows 10 Enterprise Edition x64 – Workstation	Windows 10 x64 IP Address: DHCP
WKS0004	RAM: 2048MB	Windows 8.1 Enterprise Edition x64 – Workstation	Windows 8.1 x64 IP Address: DHCP

	Drive 01 (C): 127GB Processor/Core: 1 Network Adapter		

All user accounts have the password set to Pa$$w0rd and the below list explains its utilization:

Account	Objective
CLASSROOM\administrator	Domain admin account
CLASSROOM\admworkstation	Domain user account used to demonstrate RBA settings.
CLASSROOM\sccmadmin	Account with full rights on the SCCM Servers
CLASSROOM\sccmpush	Account used for client push. This account has admin rights on all workstations
CLASSROOM\svc_sccmna	Account used as network account
CLASSROOM\svc_ssrsea	Account used as SSRS execution account
CLASSROOM\svc_sccmjoin	Account used to join computers to the domain
CLASSROOM\User01	Account used to deploy software to
CLASSROOM\User02	Account used to deploy software to
SRV0003\administrator	Local account on the SRV0003 Linux box
SRV0003\root	Root account on the SRV0003 Linux box

The following table shows the groups created to be used on this training and its objective:

Group	Objective
CLASSROOM\SCCM Admins	Contain all users with Full Access to the SCCM Infrastructure and it is a member of the SCCM Remote Tools
CLASSROOM\Workstation Admins	Contain the Admworkstation user
CLASSROOM\SCCM Remote Tools	Contain users with rights to remote access client machines
CLASSROOM\SCCM Servers	Contain all SCCM Servers

The following table shows the group policies used on this training and its objective:

Group Policy	Objective	Link	Enabled
Disable Windows Service	Set the BITS Window Service as disabled	Workstations Disabled OU	YES
SCCM Local Administrators	Set the Local Administrators membership group for the SCCM Servers	SCCM Servers OU	YES
Workstation Local Administrators	Set the Local Administrators group membership for the Desktops	Workstations OU	YES
Workstation Local Firewall	Set the Workstations Firewall Exclusion for the Client Push	Workstations OU	YES

4.1. PowerShell

Automation is a key skill for IT Professionals in today's world and everything can be automated. Within Windows and System Center Configuration Manager this is also true, so I have created over 150 scripts that can help you when using this e-book. The collection of scripts can be downloaded from http://www.tucandata.com/TrainingFiles/TrainingFilesv2.zip.

Some of the scripts are used to create the entire lab environment using Hyper-V. It is recommended to use PowerShell ISE instead of a normal PowerShell console as it is richer environment. While many PowerShell scripts are expected to run without any user intervention, they have not been created to log or show results easily. Some scripts require you to run few lines at a time as a reboot of the machine may be necessary.

Note: To be able to run the PowerShell scripts, you need to change the PowerShell Execution Policy accordingly. This is necessary because the scripts are not signed.

This can also be achieved via an elevated PowerShell console using the commands below:

```
Set-ExecutionPolicy Unrestricted -Force
```

Finally, whenever possible, I have created scripts to automate SCCM tasks, they are located under the Course Scripts folder, however, tasks not related to SCCM (i.e. Automation on Linux client, export certificate, integration with Microsoft Azure, DNS configuration etc.) may not have an associated script.

Most of the time, these scripts run from the Configuration Manager Drive. To achieve this, you can start the PowerShell via the SCCM Console or use the following PowerShell commands to enable the PowerShell SCCM environment from a normal PowerShell:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$ModulePath = $env:SMS_ADMIN_UI_PATH
if ($ModulePath -eq $null) {
      $ModulePath = (Get-ItemProperty -Path
"Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session
Manager\Environment").SMS_ADMIN_UI_PATH
}

$ModulePath =
$ModulePath.Replace("bin\i386","bin\ConfigurationManager.psd1")

$Certificate = Get-AuthenticodeSignature -FilePath "$ModulePath" -
ErrorAction SilentlyContinue
$CertStore = New-Object
System.Security.Cryptography.X509Certificates.X509Store("TrustedPublisher")
$CertStore.Open([System.Security.Cryptography.X509Certificates.OpenFlags]::
MaxAllowed)
$Certexist = ($CertStore.Certificates | where {$_.thumbprint -eq
$Certificate.SignerCertificate.Thumbprint}) -ne $null

if ($Certexist -eq $false)
```

```
{
    $CertStore.Add($Certificate.SignerCertificate)
}

$CertStore.Close()

import-module $ModulePath -force
if ((get-psdrive $SiteCode -erroraction SilentlyContinue | measure).Count -
ne 1)
{
    new-psdrive -Name $SiteCode -PSProvider "AdminUI.PS.Provider\CMSite" -
Root $servername
}
cd "$($SiteCode):"
```

Note: To be able to run some of the PowerShell scripts on the workstations, Domain Users Group will be added to the Local Administrators Group on the WKS0001, WKS0002 and WKS0004.

4.2. Installing a Hyper-V Server

Before we start, we need to build a Hyper-V Server that will host our Virtual Environment.

Note: Hyper-V server requires Virtualization capability in the host hardware, for details see https://technet.microsoft.com/en-us/windows-server-docs/compute/hyper-v/system-requirements-for-hyper-v-on-windows.

To create a Hyper-V Server, perform the following actions:

01. Download Windows Server 2016 Evaluation from Microsoft website https://www.microsoft.com/en-us/evalcenter/evaluate-windows-server-2016 and burn a DVD
02. Insert the Windows Server 2016 DVD-ROM and turn on your computer. After a few minutes, you receive the Windows Server 2016 screen shown. Select the correct Language, Time and Currency Format and Keyboard or input method and Click Next.
03. On the next Install Windows screen, click Install now.
04. On the Select the Operating System you want to install, select Windows Server 2016 Standard Evaluation (Server with a GUI) and click Next.
05. Under License terms, select I accept the license terms and click Next
06. Under Which type of installation do you want? Click Custom: Install Windows only (advanced)
07. Under Where do you want to install Windows? Click Next
08. The Installation will start and it will take some time to complete (15-30 minutes depending on your hardware).
09. Once the installation is completed, On the Settings, you must change the password before logging on for the first time. Once completed, click Finish.
10. Perform a full windows update until there is no other update to be applied
11. Download the collection of Scripts from http://www.tucandata.com/TrainingFiles/TrainingFilesv2.zip and extract to C:\

4.3. Installing Hyper-V Role

Perform this task on the Hyper-V server logged on as administrator
01. In Server Manager, on the Manage menu, click Add Roles and Features.
02. On the Before you begin page, verify that your destination server and network environment are prepared for the role and feature you want to install. Click Next.
03. On the Select installation type page, select Role-based or feature-based installation and then click Next.
04. On the Select destination server page, select a server from the server pool and then click Next.
05. On the Select server roles page, select Hyper-V.
06. To add the tools that you use to create and manage virtual machines, click Add Features, and click Next.
07. On the Features page, click Next.
08. On the Hyper-V page, click Next
09. On the Create Virtual Switches page, click Next
10. On the Virtual Machine Migration page, click Next
11. On the Default Stores page, click Next
12. On the Confirm installation selections page, select Restart the destination server automatically if required.
13. On the Add Roles and Features Wizard message, click Yes and them Install
14. When the server reboots, open the Server Manager so the installation can finish. Once done, click close

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name Hyper-V | Install-WindowsFeature -
IncludeManagementTools -Restart
```

4.4. Downloading Software

Once we have our Hyper-V host configured, it is time to download the required software and create the virtual machines, so Perform this task on the the Hyper-V server logged on as administrator:

01. Open PowerShell (run as administrator) and navigate to C:\Trainingfiles\Scripts
02. Execute .\DownloadSoftware.ps1 **Note:** If anti-virus software has been installed on the Hyper-V host, it is recommended to add an exclusion for C:\TrainingFiles otherwise it will identify the C:\TrainingFiles\Source\Eicar\eicar test file.txt as a Virus. This file is not a virus, it is an industry standard test file for antivirus engines. More information can be found at http://www.eicar.org/
03. Copy the App-v5.0 SP3 Client (appv_client_setup.exe)[1] to C:\Trainingfiles\Source\App-v5 Client

[1] To have access to the App-V 5.0 SP3 client access to MDOP package is needed. Access the MSDN, TechNet or Microsoft volume license website to download MDOP 2014 R2

4.5. Creating Windows Virtual Machines

Perform this task on the Hyper-V server logged on as administrator
01. Open PowerShell (run as administrator) and navigate to C:\Trainingfiles\Scripts
02. Execute .\CreateVMs.ps1

4.6. CLASSROOM-ROUTER01

Perform this task on the router01 virtual machine
01. Boot Virtual Machine CLASSROOM-ROUTER01
02. Log in using **vyos** as login and password
03. Type **install image** and press enter
04. On Would you like to continue, press enter
05. On Partition, press enter
06. On Install the image on, press enter
07. On Continue, type **Yes** and press enter
08. On How big of a root partition should I create, press enter
09. On What would you like to name this image, press enter
10. On Which one should I copy to sda, press enter
11. On Enter password for user 'vyos', type **Pa$$w0rd** and press enter
12. On Retype password for user 'vyos' type **Pa$$w0rd** and press enter
13. On Which drive should grub modify the boot partition on, press enter
14. Type **poweroff** and press enter
15. On Proceed with poweroff, type **Yes** and press enter
16. Select Media -> DVD Drive -> Eject vyos-1.1.3-amd64.iso and power on the virtual machine
17. Log on with login **vyos** and password **Pa$$w0rd**
18. Type **configure** and press enter
19. Type **set interface ethernet eth0 address dhcp** and press enter
20. Type **set interface ethernet eth0 description 'External'** and press enter
21. Type **set interface ethernet eth1 address 192.168.3.254/24** and press enter
22. Type **set interface ethernet eth1 description 'Internal'** and press enter
23. Type **set system name-server 8.8.8.8** and press enter
24. Type **set system name-server 8.8.4.4** and press enter
25. Type **set system host-name router01** and press enter
26. Type **set nat source rule 100 outbound-interface 'eth0'** and press enter
27. Type **set nat source rule 100 source address '192.168.3.0/24'** and press enter
28. Type **set nat source rule 100 translation address masquerade** and press enter
29. Type **commit** and press enter
30. Type **save** and press enter
31. Type **exit** and press enter
32. Type **show interfaces** and press enter
33. Type **ping www.google.com** and press enter

4.7. CLASSROOM-SRV0001

Perform this task on the srv0001 virtual machine
01. Confirm the Virtual Machine CLASSROOM-ROUTER01 is up and is providing internet connectivity
02. Boot Virtual Machine CLASSROOM-SRV0001
03. Log on as administrator
04. Open PowerShell (run as administrator) and navigate to C:\Trainingfiles\Scripts
05. Type.**\SRV0001.ps1** and press Enter
06. Type **.\SRV0001-01-InstallDC.ps1** and press Enter **Note:** The computer will restart automatically
07. Log on as administrator, Open PowerShell (run as administrator) and navigate to C:\Trainingfiles\Scripts
08. Type **.\SRV0001-02-ConfigureDC.ps1** and press Enter

4.8. CLASSROOM-SRV0002

Perform this task on the srv0002 virtual machine as administrator
01. Confirm the Virtual Machine CLASSROOM-ROUTER01 is up and is providing internet connectivity
02. Confirm the Virtual Machine CLASSROOM-SRV0001 is up and has been configured as Domain Controller
03. Boot Virtual Machine CLASSROOM-SRV0002
04. Log on as classroom\administrator
05. Open PowerShell (run as administrator) and navigate to C:\Trainingfiles\Scripts
06. Type **.\SRV0002.ps1** and press Enter **Note:** Computer will shutdown

4.9. CLASSROOM-SRV0003

Perform this task on the srv0003 virtual machine
01. Confirm the Virtual Machine CLASSROOM-ROUTER01 is up and is providing internet connectivity
02. Confirm the Virtual Machine CLASSROOM-SRV0001 is up and has been configured as Domain Controller
03. Boot Virtual Machine CLASSROOM-SRV0003
04. On CentOS 7, press Enter
05. On Welcome to Centos 7, select Language and keyboard type and click Continue
06. On Installation Summary, under System, click Network & Host Name
07. On Ethernet (eth0) click On and under hostname, type SRV0003.classroom.intranet, click Apply and then click Done
08. On Installation Summary, under System, click Installation Destination and click Done
09. On Installation Summary, under Software, click Software Selection
10. On Software Selection, select GNOME Desktop and click Done

11. On Installation Summary, click Begin Installation
12. On User Settings, click Root Password
13. On Root Password, type Pa$$word on root password and confirm and click Done twice
14. On User Settings, click User Creation
15. On Create user, type Administrator on Full Name and Name, Pa$$w0rd on password and confirm password and click Done twice
16. Wait the installation to finish, click Reboot
17. On Initial Setup, click License Information
18. Under License Agreement, click I accept the License Agreement and then Done.
19. Back on Initial Setup click Finish Configuration
20. On Logon, select Administrator, type Pa$$w0rd and click Sign In
21. On Welcome, click Next
22. On Typing, click Next
23. On Online Accounts, click Skip
24. On Ready to Go click Start using CentOS Linux
25. Click Applications, System Tools, and Software Update
26. Click Install Updates
27. Right click the desktop and click Open in terminal
28. Log on as a root using the command **su** and when asked the password type **Pa$$w0rd**
29. Type yum install lsb libXss* and press enter
30. Type yes to download the dependencies
30. Once yum has completed the installation, power the machine off

4.10. CLASSROOM-WKS0001

Perform this task on the wks0001 virtual machine as administrator
01. Confirm the Virtual Machine CLASSROOM-ROUTER01 is up and is providing internet connectivity
02. Confirm the Virtual Machine CLASSROOM-SRV0001 is up and has been configured as Domain Controller
03. Boot Virtual Machine CLASSROOM-WKS0001
04. Log on as classroom\administrator
05. Open PowerShell (run as administrator)
06. Type **Set-ExecutionPolicy Unrestricted -force** and press Enter
07. Type **\\srv0001\Trainingfiles\Scripts\WKS0001.ps1** and press Enter **Note:** Computer will shutdown

4.11. CLASSROOM-WKS0002

Perform this task on the wks0002 virtual machine as administrator
01. Confirm the Virtual Machine CLASSROOM-ROUTER01 is up and is providing internet connectivity
02. Confirm the Virtual Machine CLASSROOM-SRV0001 is up and has been configured as Domain Controller
03. Boot Virtual Machine CLASSROOM-WKS0002

04. Log on as classroom\administrator
05. Open PowerShell (run as administrator)
06. Type **Set-ExecutionPolicy Unrestricted -force** and press Enter
07. Type **\\srv0001\Trainingfiles\Scripts\WKS0002.ps1** and press Enter **Note:** Computer will shutdown

4.12. CLASSROOM-WKS0004

Perform this task on the wks0004 virtual machine as administrator
01. Confirm the Virtual Machine CLASSROOM-ROUTER01 is up and is providing internet connectivity
02. Confirm the Virtual Machine CLASSROOM-SRV0001 is up and has been configured as Domain Controller
03. Boot Virtual Machine CLASSROOM-WKS0004
04. Log on as classroom\administrator
05. Open PowerShell (run as administrator)
06. Type **Set-ExecutionPolicy Unrestricted -force** and press Enter
07. Type **\\srv0001\Trainingfiles\Scripts\WKS0004.ps1** and press Enter **Note:** Computer will shutdown

5. Active Directory

Computers used in this Lab	SRV0001
More information	Schema extensions for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/plan-design/network/schema-extensions
Description	In this chapter, we will extend the active directory schema for SCCM. The Extension adds a new container and new attributes to the AD schema, allowing the SCCM environment to publish information to the AD and domain-joined machines to query these information. Extending the schema is not required task, however, it can simplify the deployment, configuration and management of a SCCM environment.

5.1. Extending Active Directory Schema

Perform this task on the srv0001 virtual machine logged on as administrator
01. Open Windows Explorer
02. Navigate to **C:\trainingfiles\Source\SCCMCB\Extract\SMSSETUP\BIN\X64**
03. Execute **extadsch.exe** (run as administrator)
03. Review the extending of the schema by examining the content of the C:\extadsch.log file **Note:** If the schema has already been extended, the C:\extadsch.log will print lines with **already exists** information.

This can also be achieved via PowerShell using the commands below:

```
#Extend Schema
Start-Process -Filepath
('C:\trainingfiles\Source\SCCMCB\Extract\SMSSETUP\BIN\X64\extadsch.exe') -
wait
Start-Sleep 30

#Confirm Schema Extension
$schema =
[DirectoryServices.ActiveDirectory.ActiveDirectorySchema]::GetCurrentSchema
()
start-sleep 5
$schema.RefreshSchema()
$schema.FindClass("mSSMSSite")
```

5.2. Creating the System Management Container

Perform this task on the srv0001 virtual machine logged on as administrator
01. Open **ADSI Edit**.
02. Right Click ADSI Edit and click **Connect to…**
03. On Connection Settings, click **Ok**

04. Expand **Default naming context**, **DC=classroom,DC=intranet**, **CN=System**
05. right click **CN=System** and choose **New -> Object**
06. Choose object class as **Container**, click **Next**
07. In Value, type **System Management** and click **Next**
08. Click **Finish**
09. Back on the ADSI Edit screen, right click **CN=System Management** and choose **Properties** **Note:** A refresh of the list of objects may be necessary
10. Select the **Security tab**
11. Click on the **Advanced** button
12. On Advanced Security Settings for System Management, Click **Add**
13. On Permission Entry for System Management, click **Select a Principal**
14. On Select Users, Computers, Service Account, or Group, type **SCCM Servers** and click **Check Names**. Click **OK**
15. On Permission Entry for System Management, select Allow **Full Control** under permissions, and **This object and all descendant objects** under Apply to. Click Ok three (3) times.

This can also be achieved via PowerShell using the commands below:

```
Import-Module ActiveDirectory
$root = (Get-ADRootDSE).defaultNamingContext

if (!([adsi]::Exists("LDAP://CN=System Management,CN=System,$root"))) {
    $smcontainer = New-ADObject -Type Container -name "System Management" -
Path "CN=System,$root" -Passthru
}

$acl = get-acl "ad:CN=System Management,CN=System,$root"

$objGroup = Get-ADGroup -filter {Name -eq "SCCM Servers"}
$All =
[System.DirectoryServices.ActiveDirectorySecurityInheritance]::SelfAndChild
ren
$ace = new-object System.DirectoryServices.ActiveDirectoryAccessRule
$objGroup.SID, "GenericAll", "Allow", $All
$acl.AddAccessRule($ace)
Set-acl -aclobject $acl "ad:CN=System Management,CN=System,$root"
```

6. Pre-Requirements

Computers used in this Lab	SRV0001 SRV0002
More information	Prepare Windows Servers to support System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/plan-design/network/prepare-windows-servers Log files in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/plan-design/hierarchy/log-files
Description	In this chapter, we will prepare the environment for the installation of the SCCM as well as register the default log viewer for CMTrace

6.1. Installing .NET Framework 3.5

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next
06. Under select server roles, keep the default, and click Next
07. Under features, expand .NET Framework 3.5 Features and select .NET Framework 3.5 (includes .NET 2.0 and 3.0). Click Next
08. Under Confirm installation selections, click Specify an alternate source path
09. Under Specify alternate source path, type \\srv0001\Trainingfiles\Source\WS2016\sources\sxs and click Ok. One back, Click Install
10. Once the installation is succeeded. Click Close

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name Net-Framework-Core | Install-WindowsFeature -source '\\srv0001\Trainingfiles\Source\WS2016\sources\sxs'
```

6.2. Validating .NET Framework 4.6 Installation

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next

06. Under select server roles, keep the default, and click Next
07. Under features, expand .NET Framework 4.6 Features and Confirm that .NET Framework 4.6 is installed. Click Cancel

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name NET-Framework-45-Features
```

6.3. Installing Microsoft Remote Differential Compression

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next
06. Under select server roles, keep the default, and click Next
07. Under features, Select Remote Differential Compression. Click Next
08. Under Confirm installation selections click Install
09. Once the installation is succeeded. Click Close

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name RDC | Install-WindowsFeature
```

6.4. Installing Assessment and Deployment Kit (ADK) for Windows 10 1607

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Execute adksetup.exe (run as administrator) from \\srv0001\Trainingfiles\Source\AdkW10
02. Under Specify location, click Next
03. Under Windows Kits Privacy, click Next
04. Under License Agreement, click Accept
05. Under Select the features you want to install only the following: • Deployment Tools • Windows Preinstallation Environment (Windows PE) • Imaging and Configuration Designer (ICD) • Configuration Designer • User State Migration Tool (USMT) Click Install
06. Once the installation is completed, click Close

This can also be achieved via PowerShell using the commands below:

```
Start-Process -Filepath
("\\srv0001\TrainingFiles\Source\AdkW10\adksetup.exe") -ArgumentList
("/quiet /ceip off /norestart /Features OptionId.DeploymentTools
OptionId.WindowsPreinstallationEnvironment
OptionId.ImagingAndConfigurationDesigner OptionId.ICDConfigurationDesigner
OptionId.UserStateMigrationTool") -wait
```

6.5. Logs Reading

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Copy CMTrace.exe from \\srv0001\Trainingfiles\Source\SCCMCB\Extract\SMSSETUP\TOOLS to C:\Windows
02. Execute C:\Windows\CMTrace.exe
03. When asked Do you want to make this program the default viewer for log files?, click Yes.
04. Close Configuration Manager Trace Log Tool **Note:** Repeat the process on every machine that you want to read SCCM (or other) logfiles using CMTrace

This can also be achieved via PowerShell using the commands below:

```
Copy-Item
'\\srv0001\Trainingfiles\Source\SCCMCB\Extract\SMSSETUP\TOOLS\CMTrace.exe'
'C:\windows\cmtrace.exe'

New-PSDrive -Name HKCR -PSProvider Registry -Root HKEY_CLASSES_ROOT | Out-
Null
$executecmtrace = """" + $env:windir + '\CMTrace.exe' + """ " + """%1"""
New-Item -Path "hkcr:Local
Settings\Software\Microsoft\Windows\Shell\MuiCache" -Force | Out-Null
Set-ItemProperty -Path "hkcr:Local
Settings\Software\Microsoft\Windows\Shell\MuiCache" -Name ($env:windir +
'\CMTrace.exe') -Value 'Configuration Manager Trace Log Tool' | Out-Null
New-Item -Path "hkcr:.lo_" -Force | Out-Null
Set-ItemProperty -Path "hkcr:.lo " -Name '(Default)' -Value 'Log.File' |
Out-Null

New-Item -Path "hkcr:.log" -Force | Out-Null
Set-ItemProperty -Path "hkcr:.log" -Name '(Default)' -Value 'Log.File' |
Out-Null

New-Item -Path "hkcr:Log.File" -Force | Out-Null
New-Item -Path "hkcr:Log.File\Shell" -Force | Out-Null
New-Item -Path "hkcr:Log.File\Shell\Open" -Force | Out-Null
New-Item -Path "hkcr:Log.File\Shell\Open\Command" -Force | Out-Null
Set-ItemProperty -Path "hkcr:\Log.File\shell\open\command" -Name
'(Default)' -Value $executecmtrace | Out-Null

New-Item -Path "hkcu:Software\Classes\Local
Settings\Software\Microsoft\Windows\Shell\MuiCache" -Force | Out-Null
```

```
Set-ItemProperty -Path "hkcu:Software\Classes\Local
Settings\Software\Microsoft\Windows\Shell\MuiCache" -Name ($env:windir +
'\CMTrace.exe') -Value 'Configuration Manager Trace Log Tool' | Out-Null
New-Item -Path "hkcu:Software\Classes\.lo_" -Force | Out-Null
Set-ItemProperty -Path "hkcu:Software\Classes\.lo_" -Name '(Default)' -
Value 'Log.File' | Out-Null

New-Item -Path "hkcu:Software\Classes\.log" -Force | Out-Null
Set-ItemProperty -Path "hkcu:Software\Classes\.log" -Name '(Default)' -
Value 'Log.File' | Out-Null

New-Item -Path "hkcu:Software\Classes\Log.File" -Force | Out-Null
New-Item -Path "hkcu:Software\Classes\Log.File\Shell" -Force | Out-Null
New-Item -Path "hkcu:Software\Classes\Log.File\Shell\Open" -Force | Out-
Null
New-Item -Path "hkcu:Software\Classes\Log.File\Shell\Open\Command" -Force |
Out-Null
Set-ItemProperty -Path "hkcu:Software\Classes\Log.File\shell\open\command"
-Name '(Default)' -Value $executecmtrace | Out-Null

Start-Process -Filepath ('C:\windows\cmtrace.exe')
```

7. SQL Server

Computers used in this Lab	SRV0001 SRV0002
More information	Support for SQL Server versions for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/plan-design/configs/support-for-sql-server-versions International support in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/plan-design/hierarchy/international-support How to determine the version and edition of SQL Server and its components https://support.microsoft.com/en-gb/kb/321185
Description	In this chapter, we will install and configure the SQL Server and SQL Reporting Services to be used with SCCM. Basic configuration like max memory and recovery model will also be performed

7.1. Creating Firewall Rules for SQL

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Windows Firewall with Advanced Security and click Inbound Rules
02. Click New Rule
03. On New Inbound Rule Wizard, select Port and click Next
04. On Protocol and Ports select TCP and type 1433 under specify local ports and click Next
05. On Action, click Next
06. On Profile, click Next
07. On Name, type SQL Server (TCP 1433) Inbound and click Finish
08. Click New Rule
09. On New Inbound Rule Wizard, select Port and click Next
10. On Protocol and Ports select TCP and type 4022 under specify local ports and click Next
11. On Action, click Next
12. On Profile, click Next
13. On Name, type SQL Server SSB (TCP 4022) Inbound and click Finish

This can also be achieved via PowerShell using the commands below:

```
New-NetFirewallRule -DisplayName "SQL Server (TCP 1433) Inbound " -Action
Allow -Direction Inbound -LocalPort 1433 -Protocol TCP
New-NetFirewallRule -DisplayName "SQL Server (TCP 4022) Inbound " -Action
Allow -Direction Inbound -LocalPort 4022 -Protocol TCP
```

7.2. Installing SQL Server 2016

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Execute setup.exe from \\srv0001\Trainingfiles\Source\SQL2016
02. On the SQL Server Installation Center, Click Installation
03. Under the Installation, click New SQL Server stand-alone installation or add features to an existing installation
04. Under Product Key, select Specify a free version, select Evaluation, and click Next
05. Under License Terms, click I accept the license terms and click Next
06. Under License Terms, click I accept the license terms and click Next
07. Under Microsoft Update, click Next
08. Under Product Updates uncheck Include SQL Server product updates and Click Next **Note:** if there is an Error screen, it can be safety ignored as this is because the machine does not have internet access or was unable to connect to the Microsoft Servers
09. Under Install Rules, Click Next
10. Under Feature Selection select • Database Engine Services • Reporting Service – Native Change the Instance root directory to C:\SQLServer\, Shared feature directory to C:\SQLServer\ and Shared feature directory (x86) to C:\SQLServer (x86)\ and click Next
11. Instance Configuration select Default Instance. Click Next
12. Under Server Configuration, Account Name for SQL Server Agent, click Browser and type SYSTEM. Click Check Names. and click ok times. Repeat same steps for SQL Server Database Engine and SQL Server Reporting Services **Note:** Using LocalSystem as Service Account for SQL is not a best practice, however, it is easy to use (and this is the reason I'm using it on this lab). For more information about why this is not best practices, refer to https://www.mssqltips.com/sqlservertip/2384/why-system-account-is-a-bad-idea-for-sql-server-service-account/
13. Under Server Configuration, click Collation
14. Click Customize and select SQL Collation
15. Select SQL_Lating1_General_CP1_CI_AS. Click Ok and then Next
16. Under Database Engine Configuration click Add and type SCCM Admins. Click Check Names and OK.
17. Under Database Engine, click TempDB and configure the Initial Size as 1024. Click next **Note:** The TempDB initial size should be approximately 25% of the size of the estimated SCCM database.
18. Under Reporting Services Configuration, make sure Install and configure is selected under Reporting Service Native Move. Click Next
19. Under Ready to Install, click Install
20. Once the setup is completed, click Close

This can also be achieved via PowerShell using the commands below:

```
$inifile = @"
[OPTIONS]
ACTION="Install"
SUPPRESSPRIVACYSTATEMENTNOTICE="False"
IACCEPTROPENLICENSETERMS="False"
ENU="True"
QUIET="False"
QUIETSIMPLE="True"
;UIMODE="Normal"
UpdateEnabled="False"
USEMICROSOFTUPDATE="False"
FEATURES=SQLENGINE,RS
UpdateSource="MU"
HELP="False"
INDICATEPROGRESS="False"
X86="False"
INSTANCENAME="MSSQLSERVER"
INSTALLSHAREDDIR="C:\SQLServer"
INSTALLSHAREDWOWDIR="C:\SQLServer (x86)"
INSTANCEID="MSSQLSERVER"
RSINSTALLMODE="DefaultNativeMode"
SQLTELSVCACCT="NT Service\SQLTELEMETRY"
SQLTELSVCSTARTUPTYPE="Automatic"
INSTANCEDIR="C:\SQLServer"
AGTSVCACCOUNT="NT AUTHORITY\SYSTEM"
AGTSVCSTARTUPTYPE="Manual"
COMMFABRICPORT="0"
COMMFABRICNETWORKLEVEL="0"
COMMFABRICENCRYPTION="0"
MATRIXCMBRICKCOMMPORT="0"
SQLSVCSTARTUPTYPE="Automatic"
FILESTREAMLEVEL="0"
ENABLERANU="False"
SQLCOLLATION="SQL_Latin1_General_CP1_CI_AS"
SQLSVCACCOUNT="NT AUTHORITY\SYSTEM"
SQLSVCINSTANTFILEINIT="False"
SQLSYSADMINACCOUNTS="CLASSROOM\SCCM Admins"
SQLTEMPDBFILECOUNT="2"
SQLTEMPDBFILESIZE="1024"
SQLTEMPDBFILEGROWTH="64"
SQLTEMPDBLOGFILESIZE="1024"
SQLTEMPDBLOGFILEGROWTH="64"
ADDCURRENTUSERASSQLADMIN="False"
TCPENABLED="1"
NPENABLED="0"
BROWSERSVCSTARTUPTYPE="Disabled"
RSSVCACCOUNT="NT AUTHORITY\SYSTEM"
RSSVCSTARTUPTYPE="Automatic"
"@

$inifile | Out-File -FilePath "\\srv0001\TempFiles\installsql2016.ini"

Start-Process -Filepath
("\\srv0001\TrainingFiles\Source\SQL2016\setup.exe") -ArgumentList
```

```
('/ConfigurationFile="\\srv0001\TempFiles\\installsql2016.ini"
/IAcceptSQLServerLicenseTerms') -wait
Start-sleep 30

$web = New-Object -ComObject msxml2.xmlhttp
$url = "http://localhost:80/reports"
try
{
      $web.open('GET', $url, $false)
      $web.send()

      Write-host "HTTP Return $($web.status)"
}
catch
{
      Write-host "ERROR: $($_)"
}
```

7.3. Installing SQL Server Management Studio

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Execute SSMS-Setup-ENU.exe from \\srv0001\Trainingfiles\Source\SQLMgmt
02. On the Welcome, click Install
03. On Setup Completed, click Close

This can also be achieved via PowerShell using the commands below:

```
Start-Process -Filepath ("\\srv0001\Trainingfiles\Source\SQLMgmt\SSMS-
Setup-ENU.exe") -ArgumentList ('/install /quiet /norestart') -wait
```

7.4. Validating Installation

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open SQL Server Management Studio
02. Connect to the SRV0002 server
03. Click New query and type **select @@version** and click execute **Note:** The result should be something similar to **Microsoft SQL Server 2016 (RTM) - 13.0.1601.5 (X64) Apr 29 2016 23:23:58 Copyright (c) Microsoft Corporation Enterprise Evaluation Edition (64-bit) on Windows Server 2016 Standard Evaluation 6.3 <X64> (Build 14393:) (Hypervisor)**
04. Now, type **SELECT SERVERPROPERTY('productversion'), SERVERPROPERTY('productlevel'), SERVERPROPERTY('edition')** and click Execute **Note:** the result should be something like **13.0.1601.5 RTM Enterprise Evaluation Edition (64-bit)**
05. Open Internet Explorer and navigate to http://SRV0002/ReportServer
06. Navigate to http://SRV0002/Reports

This can also be achieved via PowerShell using the commands below:

```
$conn = New-Object System.Data.SqlClient.SqlConnection
$conn.ConnectionString = "Data Source=SRV0002;Initial
Catalog=Master;trusted_connection = true;"
$conn.Open()

$SqlCommand = $Conn.CreateCommand()
$SqlCommand.CommandTimeOut = 0
$SqlCommand.CommandText = "select @@version"
$DataAdapter = new-object System.Data.SqlClient.SqlDataAdapter $SqlCommand
$dataset = new-object System.Data.Dataset
$DataAdapter.Fill($dataset)

$SqlCommand2 = $Conn.CreateCommand()
$SqlCommand2.CommandTimeOut = 0
$SqlCommand2.CommandText = "SELECT SERVERPROPERTY
('productversion'),SERVERPROPERTY ('productlevel'), SERVERPROPERTY
('edition')"
$DataAdapter2 = new-object System.Data.SqlClient.SqlDataAdapter
$SqlCommand2
$dataset2 = new-object System.Data.Dataset
$DataAdapter2.Fill($dataset2)

$dataset.Tables[0] | select Column1
$dataset2.Tables[0] | select Column1,Column2,Column3

$conn.close()

$webclient = new-object System.Net.WebClient
$webclient.Credentials = new-object
System.Net.NetworkCredential("sccmadmin", 'Pa$$w0rd', "classroom")
$webclient.DownloadString("http://SRV0002/ReportServer")
```

7.5. SQL Server Max Memory

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open SQL Server Management Studio
02. Connect to the SRV0002 server
03. On Object Explorer, right SRV0002 (SQL Server) and click Properties
04. Under Server Properties – SRV0002, click Memory and set the minimum and maximum memory for the server to 4096. Click Ok **Note:** It is recommended to allow 4GB for the Windows OS and set the memory to a minimum of 8GB. If the server has more memory free, the values should be 80% for the SQL (if it is running on its own server) and 50% if the server is co-hosted on the SCCM Server

This can also be achieved via PowerShell using the commands below:

```
$maxMem = 4096
$minMem = 4096
```

```
[reflection.assembly]::LoadWithPartialName("Microsoft.SqlServer.Smo") |
Out-Null
$srv = new-object
Microsoft.SQLServer.Management.Smo.Server($SQLInstanceName)
$srv.ConnectionContext.LoginSecure = $true

$srv.Configuration.MaxServerMemory.ConfigValue = $maxMem
$srv.Configuration.MinServerMemory.ConfigValue = $minMem

$srv.Configuration.Alter()
```

7.6. SQL Server Recovery Model for SQL Server Reporting Services Database

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open SQL Server Management Studio
02. Connect to the SRV0002 server
03. Expand Databases, select ReportServer and click Properties
04. Under Database Properties – ReportServer, click Options and change Recovery Model from Full to Single. Click Ok

This can also be achieved via PowerShell using the commands below:

```
$Server="SRV0002"
$db = "ReportServer"
[System.Reflection.Assembly]::LoadWithPartialName("Microsoft.SqlServer.SMO"
) | out-null
$SMOserver = New-Object ('Microsoft.SqlServer.Management.Smo.Server') -
argumentlist $Server
$SMOserver.Databases["$db"] | select Name, RecoveryModel | Format-Table
$SMOserver.databases["$db"].recoverymodel = "Simple"
$SMOserver.databases["$db"].alter()
$SMOserver.Databases["$db"] | select Name, RecoveryModel | Format-Table
```

7.7. Validating Static Port and Services

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open SQL Server 2016 Configuration Manager
02. Expand SQL Server Network Configuration and click Protocols for MSSQLSERVER
03. Select with right click TCP/IP and click Properties
04. Click IP Addresses tab and confirm that a static TCP Port is used. **Note:** SCCM does not support TCP Dynamic Ports

This can also be achieved via PowerShell using the commands below:

```
foreach ($item in (Get-Item -Path
"Registry::HKEY LOCAL MACHINE\Software\Microsoft\Microsoft SQL
Server\Instance Names\SQL" | select-object -ExpandProperty Property)) {
    $instance = (Get-ItemProperty -Path
"Registry::HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL
Server\Instance Names\SQL\").$item
    $info = Get-ItemProperty -Path
"Registry::HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL
Server\$instance\$item\SuperSocketNetLib\Tcp\IpAll" | select
TcpDynamicPorts, TcpPort
    "{0} - {1} - {2}" -f $item, $info.TcpDynamicPorts, $info.TcpPort }
```

8. Installation of SCCM Primary Site

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	Supported configurations for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/plan-design/configs/supported-configurations List of Prerequisite Checks for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/servers/deploy/install/list-of-prerequisite-checks Install System Center Configuration Manager sites https://docs.microsoft.com/en-us/sccm/core/servers/deploy/install/installing-sites
Description	In this chapter, we will install SCCM as well as validate if the installation has occurred correctly. We will also be installing the Configuration Manager Toolkit, a set of extra tools to help administration of the SCCM environment.

8.1. Downloading SCCM Setup Files

Perform this task on the SRV0001 virtual machine logged on as administrator
01. Execute setupdl.exe from C:\Trainingfiles\Source\SCCMCB\Extract\SMSSETUP\BIN\X64 **Note:** This step was already executed on section 4.4 and can be safely ignored
02. Once the Configuration Manager Setup download is loaded, type C:\Trainingfiles\Source\SCCMCB\Redist and click download
03. Once the download is completed, examine the \\srv0001\Trainingfiles\Source\SCCMCB\Redist folder

This can also be achieved via PowerShell using the commands below:

```
Start-Process -Filepath
("C:\Trainingfiles\Source\SCCMCB\Extract\SMSSETUP\BIN\X64\SETUPDL.exe") -
ArgumentList ("c:\Trainingfiles\Source\SCCMCB\Redist") -wait
```

8.2. Pre-requirements Check

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Command Prompt as administrator
02. type \\srv0001\Trainingfiles\Source\SCCMCB\Extract\SMSSETUP\BIN\X64\prereqchk.exe /pri /sql SRV0002.classroom.intranet /sdk SRV0002.classroom.intranet and press enter
03. Once the Installation prerequisite check completed, confirm that there are no errors. Click OK

Note: You should see the following warnings: • WSUS on site Server • Verify site server permission to publish to active directory • SQL Server process memory allocation
04. You can also review the errors and warnings by examining the content of the C:\ConfigMgrPrereq.log file

This can also be achieved via PowerShell using the commands below:

```
Start-Process -Filepath ("\\srv0001\Trainingfiles\Source\SCCMCB\Extract\SMSSETUP\BIN\X64\prereqchk.exe") -ArgumentList ("/pri /sql SRV0002.classroom.intranet /sdk SRV0002.classroom.intranet") -wait
```

8.3. Creating Firewall Rules for SCCM Site Server Console Communication

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Windows Firewall with Advanced Security and click Inbound Rules
02. Click New Rule
03. On New Inbound Rule Wizard, select Port and click Next
04. On Protocol and Ports select TCP and type 135 under specify local ports and click Next
05. On Action, click Next
06. On Profile, click Next
07. On Name, type SCCM Console (TCP 135) Inbound and click Finish

This can also be achieved via PowerShell using the commands below:

```
New-NetFirewallRule -DisplayName "SCCM Console (TCP 135) Inbound" -Action Allow -Direction Inbound -LocalPort 135 -Protocol TCP
```

8.4. Site Server Installation

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Execute splash.hta from \\srv0001\Trainingfiles\Source\SCCMCB\Extract
02. Under Microsoft System Center Configuration Manager, click Install
03. On Before You Begin, click Next
04. On Available Setup Options, click Install a Configuration Manager Primary Site and click Next
05. Select Install the evaluation edition of this product and click Next
06. Under Product License Terms, select I accept for all products and click Next
07. Under Prerequisite Licenses, select I accept these License Terms for: • Microsoft SQL Server Express • Microsoft SQL Server Native Client • Microsoft Silverlight 5

Click Next

08. Under Prerequisite Downloads, select Use previously downloaded files and in path type \\srv0001\Trainingfiles\Source\SCCMCB\Redist and click Next

09. Under server language selection, leave the default selection and click Next

10. Under Client Language selection leave the default selection and click Next

11. Under Site and Installation Settings, use the following information:

- Site Code: 001
- Site Name: Training Lab
- Installation Folder: C:\ConfigMgr
- Install the Configuration Manager Console: Checked

Click Next

Note: In production, SCCM should not be installed on the System Drive. It is only being installed here on the System Drive because it is a lab environment.
Note: A file called NO_SMS_ON_DRIVE.SMS should be used to exclude SCCM from using a specific drive. Fore more information, refer to https://technet.microsoft.com/en-us/library/bb632890.aspx
Note: Site Code, Site Name and Installation Folder cannot be changed after the installation.

12. Under Primary Site Installation select Install the primary site as a stand-alone site and click Next

13. On the Configuration Manager question window, click Yes

14. Under Database Information leave the default and click Next

15. Under Database Information (Path to the SQL files) leave the default and click Next

16. Under SMS Provider Settings, leave the default and click Next

17. Under Client Computer Communication Settings select Configure the communication method on each site system role and click Next

18. Under Site System Roles uncheck Install a management point and Install a distribution Point and click Next

Note: These site system roles are going to be installed later

19. Under Usage Data, click Next

20. Under Service Connection Point Setup, leave the default and click Next

21. Under Settings summary, review the settings and click Next

22. The prerequisite check will validate the system. Once it is done, click Begin Install

23. Once the installation is completed, click Close.

Note: Installation takes about half an hour. During the installation process, you can press the View Log button multiple times and examine the progress of the installation by reviewing the log file status.

24. At the root of C-partition, multiple log files are created that tell the status of the installation:

- ConfigMgrAdminUISetup.log: SCCM console installation log
- ConfigMgrPrereq.log: Prerequisites review log
- ConfigMgrSetup.log: site server installation log
- ConfigMgrSetupWizard.log: installation wizard log
- smstsvc.log: installation program log (the errors in it can be ignored)

This can also be achieved via PowerShell using the commands below:

```
$inifile = @"
[Identification]
Action=InstallPrimarySite

[Options]
ProductID=EVAL
SiteCode=001
SiteName=Training Lab
SMSInstallDir=c:\ConfigMgr
SDKServer=SRV0002.classroom.intranet
RoleCommunicationProtocol=HTTPorHTTPS
ClientsUsePKICertificate=0
PrerequisiteComp=1
PrerequisitePath=\\srv0001\Trainingfiles\Source\SCCMCB\Redist
MobileDeviceLanguage=0
AdminConsole=1
JoinCEIP=0

[SQLConfigOptions]
SQLServerName=SRV0002.classroom.intranet
DatabaseName=CM_001
SQLSSBPort=4022
SQLDataFilePath=C:\SQLServer\MSSQL13.MSSQLSERVER\MSSQL\DATA\
SQLLogFilePath=C:\SQLServer\MSSQL13.MSSQLSERVER\MSSQL\DATA\

[CloudConnectorOptions]
CloudConnector=1
CloudConnectorServer=SRV0002.classroom.intranet
UseProxy=0
ProxyName=
ProxyPort=

[SystemCenterOptions]
SysCenterId=x2ctA79R2ED90XmgV4SwzfKYYJ6lr5gT8IVycMy7Jpg=

[HierarchyExpansionOption]

"@

$inifile -replace "`n", "`r`n"| Out-File -FilePath
"\\srv0001\TempFiles\installcmcb.ini"

Start-Process -Filepath
("\\srv0001\TrainingFiles\Source\SCCMCB\Extract\SMSSETUP\BIN\X64\setup.exe"
) -ArgumentList ('/script "\\srv0001\TempFiles\installcmcb.ini"') -wait
```

8.5. Installation Status

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Services console
02. Services for SCCM are still named with SMS-prefix.

Note the existence of:

- SMS_EXECUTIE
- SMS_SITE_BACKUP
- SMS_SITE_COMPONENT_MANAGER
- SMS_SITE_SQL_BACKUP
- SMS_SITE_VSS_WRITER

Note: All services should be with Status running and the startup type as Automatic. The only exception is the SMS_SITE_BACKUP, that will set to Manual and will be started only when needed by SCCM

03. Open Windows Explorer and navigate to C:\ConfigMgr.

04. Open Local Users and groups and navigate to groups

05. Multiple groups exist for SCCM, and they have a prefix of ConfigMgr or SMS.

Note: The following groups should be created:

- ConfigMgr_CollectedFilesAccess
- ConfigMgr_DViewAccess
- SMS Admins
- SMS_SiteSystemToSiteServerConnection_MP_<SITECODE>
- SMS_SiteSystemToSiteServerConnection_SMSProv_<SITECODE>
- SMS_SiteSystemToSiteServerConnection_Stat_<SITECODE>
- SMS_SiteToSiteConnection_<SITECODE>

06. On SRV0001 (Domain Controller), open ADSI Edit

07. Expand Default naming context, DC=CLASSROOM,DC=intranet, CN=System, CN=System Management

Note: Existence of CN=SMS-Site-001 record from mSSMSSite class

This can also be achieved via PowerShell using the commands below:

```
Get-CimInstance win32_service | where-object {$_.Name -in
("SMS EXECUTIVE","SMS SITE BACKUP","SMS SITE COMPONENT MANAGER","SMS SITE S
QL_BACKUP","SMS_SITE_VSS_WRITER")} | select Name,StartMode,State,Status

$dn = New-Object System.DirectoryServices.DirectoryEntry
$dsLookFor = new-object System.DirectoryServices.DirectorySearcher($dn)
$dsLookFor.Filter = ("CN=SMS-SITE-001")
$dsLookFor.SearchScope = "subtree";
$dsLookFor.findOne()

Get-ChildItem -Path C:\ConfigMgr

$Groups = Gwmi win32_group | where { $_.Name -in
("ConfigMgr CollectedFilesAccess", "ConfigMgr DViewAccess", "SMS Admins",
"SMS SiteSystemToSiteServerConnection MP 001",
"SMS_SiteSystemToSiteServerConnection_SMSProv_001",
"SMS_SiteSystemToSiteServerConnection_Stat_001",
"SMS_SiteToSiteConnection_001") } | select Name
```

```
if ($Groups.Count -ne 7) { Write-Host "Should be 7 Groups" } else { "Groups OK" }
```

8.6. Console Overview

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console. As the Management Console starts, it is good to notice that the new console is not MMC-based.

02. Let's look at few main points:
• Console uses new Ribbon-user interface (Originally introduced in Office 2007).
• Bottom left corner has a list of main views:
  ○ Assets and Compliance: here is information on machines and users as well as their settings
  ○ Software Library: includes application management, software updates and operating systems deployment related tasks
  ○ Monitoring: information on the status of SCCM and reporting related settings
  ○ Administration: SCCM environment and configuration related settings

03. Click Assets and Compliance. Here are the settings for: • Users: Manage users and user groups for the hierarchy • Devices: Manage devices for the hierarchy • User Collections: Manage user collection for the hierarchy • Device Collections: Manage device collection for the hierarchy • User State Migration: Manage user state migration for when you deploy operating system • Asset Intelligence: Manage the Asset Intelligence catalog, import license files, and synchronize with System Center Online to reconcile software licenses. • Software Metering: Configure rules to monitor software application usage. • Compliance Settings: Manage configuration items and configuration baselines to assess and remediate the compliance of settings on devices. • Endpoint Protection: Manage Antimalware and Firewall policies. • All Corporate-owned Devices: Manage Corporate-owned Devices and Device Enrollment Profiles
04. Click Devices. Note that ribbon bar functions update to relevant tasks. This view shows all devices know to SCCM. Right-click Devices and look at the available options. From the right side of the window pane, right-click SRV0002 and notice the options. Ribbon displays options from both left side selection e.g. Devices and right-side selection SRV0002.
05. Click Software Library. There are 4 main levels: • Application Management: Manage application deployments for users and devices, and configure global conditions for all applications in the hierarchy. • Software Updates: Manage software updates, software update groups, deployment packages for software updates, and automatic deployment rules • Operating Systems: Manage drivers, operating system images, upgrade packages, boot images, and task sequences to deploy operating systems and virtual hard disks. • Windows 10 Servicing: Manage Servicing for Windows 10 • Office 365 Client Management: Monitor amd Manage Office 365 clients (available in SCCM 1610 and later)

06. Expand Application Management. There are 9 sublevels:

- Applications: Manage and deploy applications to users and devices, and configure rules to install and uninstall applications.
- License Information for Store Apps: Manage Licensed Store Applications from Windows Store for Business and Apple's VPP.
- Packages: Manage packages that contain the files and instructions to deploy programs to users and devices.
- Approval Requests: Manage application requests from users for Software Center applications that require approval.
- Global Conditions: Manage global conditions for all applications in the site hierarchy.
- App-V Virtual Environments: Virtual Environment
- Windows Sideloading Key: Windows Sideloading Keys
- Application Management Policies: Configure application management policies for the hierarchy.
- App Configuration Policies: Manage app configuration policies.

07. Expand Software Update. There are 4 sublevels:

- All Software Updates: Synchronize, configure, download, and deploy software updates.
- Software Update Groups: Manage software updates as a group
- Deployment Packages: Manage software update deployment packages
- Automatic Deployment Rules: Manage rules that automatically identify, download, add to a software update group, and optionally deploy software updates that meet specific criteria.

08. Expand Operating Systems. There are 7 sublevels:

- Drivers: Manage device drivers and device driver catalogs to deploy operating systems
- Driver Packages: Manage device driver packages.
- Operating System Images: Manage Windows image files for operating system deployment
- Operating System Upgrade Packages: Manage operating system upgrade packages
- Boot Images: Manage boot images for operating system deployment.
- Task Sequences: Manage task sequences that automate steps or tasks on client computers.
- Virtual Hard Disks: Manage Virtual Hard Disks.

09. Expand Windows 10 Servicing. There are 2 sublevels:

- All Windows 10 Updates: Manage Updates for Windows 10.
- Servicing Plans: Manage servicing plans for Windows 10.

10. Click Monitoring. There are 13 levels here:

- Alerts: View and manage alerts.
- Queries: View and manage Configuration Manager queries.
- Reporting: View and manage reports and report subscriptions, and configure report options.
- Site Hierarchy: View and manage the status of all sites in the hierarchy by using a hierarchy diagram or a geographical view. The geographical view of the hierarchy requires a web browser and access to the Internet.
- System Status: View and manage site status, component status, conflicting records, and status message queries.
- Deployments: View information about the status of deployed software.
- Client Operations: View client operation details.
- Client Status: View and configure options for client status.
- Database Replication: View site-to-site link status.

- Distribution Status: View content status, distribution point status, and distribution point configuration status.
- Software Update Point Synchronization Status: View software update point synchronization status across the hierarchy.
- Site Servicing Status: View the status of Configuration Manager updates you've installed in your hierarchy.
- Security: View Endpoint Protection and Health Attestation details. (available in SCCM 1610 and later)
- Upgrade Analytics: Analyze device compatibility with Windows 10 to facilitate upgrades. (available in SCCM 1610 and later)
- Compliance Settings: View Compliance Settings details. (available in SCCM 1610 and later)
- Endpoint Protection Status: View Endpoint Protection status details. It has been moved under security in SCCM 1610 and later.

Note: In my opinion, Reporting should have a separate workspace like in other System Center products.

11. Click Administration. There are 8 main levels:

- Hierarchy Configuration: Manage boundaries, site-to-site communication, discovery methods, Active Directory forest and Exchange Server connection settings.
- Cloud Services: Manage subscriptions to cloud services in your hierarchy.
- Site Configuration: Manage servers and site system roles, components, site maintenance, and status configuration
- Client Settings: Configure default and custom client settings.
- Security: Manage administrative users, security roles, security scopes, certificates, and accounts that you configure in the Configuration Manager console.
- Distribution Points: Manage individual distribution points and configuration properties, and view disk space capacity.
- Distribution Point Groups: Manage distribution points as a group
- Migration: Manage migration of data from sites in a Configuration Manager hierarchy to sites in this Configuration Manager hierarchy.

Note: Only the user that performed the installation will have permissions in SCCM by default. You need to remember to give other users permissions.

12. Click on File->Connect via PowerShell

13. On Do you want to run Software from this untrusted publisher? Type A and enter

14. Type **Get-Module -Name ConfigurationManager | select Version** and confirm the version is 5.0.8412.1000

This can also be achieved via PowerShell using the commands below:

```
#Step 12, 13 and 14 only
$ModulePath = $env:SMS_ADMIN_UI_PATH
if ($ModulePath -eq $null) {
      $ModulePath = (Get-ItemProperty -Path
"Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session
Manager\Environment").SMS_ADMIN_UI_PATH
}
```

```
$ModulePath =
$ModulePath.Replace("bin\i386","bin\ConfigurationManager.psd1")

$Certificate = Get-AuthenticodeSignature -FilePath "$ModulePath" -
ErrorAction SilentlyContinue
$CertStore = New-Object
System.Security.Cryptography.X509Certificates.X509Store("TrustedPublisher")
$CertStore.Open([System.Security.Cryptography.X509Certificates.OpenFlags]::
MaxAllowed)
$Certexist = ($CertStore.Certificates | where {$_.thumbprint -eq
$Certificate.SignerCertificate.Thumbprint}) -ne $null

if ($Certexist -eq $false)
{
    $CertStore.Add($Certificate.SignerCertificate)
}

$CertStore.Close()

import-module $ModulePath -force

Get-Module -Name ConfigurationManager | select Version
```

8.7. Validating Installation

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Under monitoring, expand System Status and click Component Status
03. Search for SMS_DATABASE_NOTIFICATION_MONITOR
04. Right Click SMS_DATABASE_NOTIFICATION_MONITOR, Show Messages and click All
05. Under Status Messages: Set Viewing Period, click OK
06. Verify the existence of Message ID 2420 **Note:** During the installation, these messages are normal, however, it should not occur after the installation
07. Double click any 2420 messages to see its details. Once done, click Ok
08. Search for SMS_SITE_SQL_BACKUP
09. Right Click SMS_SITE_SQL_BACKUP, Show Messages and click All
10. Under Status Messages: Set Viewing Period, click OK
11. Verify the existence of Message ID 4959 **Note:** If this Message ID exist, there is a problem with the SQL Server Account SPN. Refer to https://docs.microsoft.com/en-us/sccm/core/servers/manage/modify-your-infrastructure#bkmk_SPN for more information
12. Search for SMS_SITE_COMPONENT_MANAGER
13. Right Click SMS_SITE_COMPONENT_MANAGER, Show Messages and click All
14. Under Status Messages: Set Viewing Period, click OK
15. Verify the existence of Message ID 1027
16. Double click the message to see its details. Once done, click Ok

17. Search for SMS_HIERARCHY_MANAGER
18. Right Click SMS_HIERARCHY_MANAGER, Show Messages and click All
19. Under Status Messages: Set Viewing Period, click OK
20. Verify the existence of Message ID 3306
21. Double click any of the messages to see its details. Once done, click Ok
22. Verify the existence of Message ID 3323 **Note:** If this Message ID exist, the SCCM Server will accept HTTP or HTTPS connections.
23. Double click any of the messages to see its details. Once done, click Ok
24. Verify the existence of Message ID 4911 **Note:** If this Message ID exist, the container System Management have the correct permissions.
25. Double click any of the messages to see its details. Once done, click Ok
26. Verify the existence of Message ID 4913 **Note:** If this Message ID exist, there is a problem with the security of the System Management container.
27. Double click any of the messages to see its details. Once done, click Ok
28. Search for SMS_REPLICATION_CONFIGURATION_MONITOR
29. Right Click SMS_REPLICATION_CONFIGURATION_MONITOR, Show Messages and click All
30. Under Status Messages: Set Viewing Period, click OK
31. Verify the existence of Message ID 4629
32. Double click any of the messages to see its details. Once done, click Ok
33. Verify the existence of Message ID 620 **Note:** If found, check the performance of your SQL Server.
34. Search for SMS_DMP_DOWNLOADER
35. Right Click SMS_DMP_DOWNLOADER, Show Messages and click All
36. Under Status Messages: Set Viewing Period, click OK
37. Verify the existence of Message ID 4629
38. Double click any of the messages to see its details. Once done, click Ok
39. Verify the existence of Message ID 9700 **Note:** If found check the network and/or internet. Also, this message is also normal to be seen during the installation.
40. Verify the existence of Message ID 1104 **Note:** During the installation, this message is normal, however, it should not occur after the installation
41. Search for SMS_WINNT_SERVER_DISCOVERY_AGENT
42. Right Click SMS_WINNT_SERVER_DISCOVERY_AGENT, Show Messages and click All
43. Under Status Messages: Set Viewing Period, click OK
44. Verify the existence of Message ID 4202
45. Double click any of the messages to see its details. Note the number of system roles found, it should be 5. Once done, click Ok
46. Click Administration.

47. Expand Site Configuration and click Servers and Site System Roles

Note: Confirm that the Count of Roles for all listed servers match the number at message ID 4202

This can also be achieved via PowerShell using the commands below:

```
$component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS StatMsg stmsg inner join SMS StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_DATABASE_NOTIFICATION_MONITOR' and stmsg.MessageID = 2420 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '001'"
if ($component -ne $null)
{
    Write-Host "Warning: Found SMS_DATABASE_NOTIFICATION_MONITOR 2420 id's"
-ForegroundColor Yellow
}

$component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_SITE_SQL_BACKUP' and stmsg.MessageID = 4959 and stmsgin.InsStrIndex =
0 and stmsgin.SiteCode = '001'"
if ($component -ne $null)
{
    Write-Host "Error: Missing SQL SPN Information -
https://technet.microsoft.com/en-
us/library/hh427336.aspx#BKMK_ManageSPNforDBSrv" -ForegroundColor Red
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS SITE COMPONENT MANAGER' and stmsg.MessageID = 1027 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '001'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_SITE_COMPONENT_MANAGER 2017 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS StatMsg stmsg inner join SMS StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_HIERARCHY_MANAGER' and stmsg.MessageID = 3306 and stmsgin.InsStrIndex
= 0 and stmsgin.SiteCode = '001'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS HIERARCHY MANAGER 3306 id's"
        break
    } else { Start-Sleep 10 }
```

```
}

$component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS HIERARCHY MANAGER' and stmsg.MessageID = 4911 and stmsgin.InsStrIndex
= 0 and stmsgin.SiteCode = '001'"
if ($component -ne $null)
{
    Write-Host "Found SMS_HIERARCHY_MANAGER 4911 id's"
} else {
    Write-Host "ERROR: Not Found SMS_HIERARCHY_MANAGER 4911 id's" -
ForegroundColor Red
}

$component = gwmi -Namespace ("root\sms\site 001") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_HIERARCHY_MANAGER' and stmsg.MessageID = 4913 and stmsgin.InsStrIndex
= 0 and stmsgin.SiteCode = '001'"
if ($component -ne $null)
{
    Write-Host "ERROR: Found SMS_HIERARCHY_MANAGER 4913 id's" -
ForegroundColor Red
}

$component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_REPLICATION_CONFIGURATION_MONITOR' and stmsg.MessageID = 4629 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '001'"
if ($component -ne $null)
{
    Write-Host "Found SMS_REPLICATION_CONFIGURATION_MONITOR 4629 id's"
}

$component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS StatMsg stmsg inner join SMS StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_REPLICATION_CONFIGURATION_MONITOR' and stmsg.MessageID = 620 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '001'"
if ($component -ne $null)
{
    Write-Host "Error: Found SMS_REPLICATION_CONFIGURATION_MONITOR 620
id's" -ForegroundColor Red
}

$component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS DMP DOWNLOADER' and stmsg.MessageID = 4629 and stmsgin.InsStrIndex = 0
and stmsgin.SiteCode = '001'"
if ($component -ne $null)
{
    Write-Host "Found SMS_DMP_DOWNLOADER 4629 id's"
}
```

```
$component = gwmi -Namespace ("root\sms\site 001") -query "select
stmsg.RecordID from SMS_StatMsg stmsg where stmsg.Component =
'SMS_DMP_DOWNLOADER' and stmsg.MessageID = 9700 and stmsg.SiteCode = '001'"
if ($component -ne $null)
{
    Write-Host "Error: Found SMS DMP DOWNLOADER 9700 id's" -ForegroundColor
Red
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_001") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS WINNT SERVER DISCOVERY AGENT' and stmsg.MessageID = 4202 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '001'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS WINNT SERVER DISCOVERY AGENT 4202 id's"
        break
    } else { Start-Sleep 10 }
}

$Total = $component.InsStrValue
$roles = gwmi -Namespace ("root\sms\site_001") -query "select * from
SMS_SCI_SysResUse where FileType=2"
if ($roles.count -ne $total)
{
    Write-Host "ERROR: Found $($roles.count). expected $Total"
} else {
    Write-Host "All $Total roles have been created"
}
```

8.8. Configuration Manager Toolkit

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Execute ConfigMgrTools.msi from \\srv0001\Trainingfiles\Source\SCCMCB-Toolkit
02. On ConfigMgr 2012 Toolkit R2 click Next
03. Under Software License Terms, click I accept the license agreement and click Next
04. Under Select Components, click Next
05. Under Setup Complete, click Finish

This can also be achieved via PowerShell using the commands below:

```
Start-Process -Filepath ("msiexec.exe") -ArgumentList ('/i
"\\srv0001\Trainingfiles\Source\SCCMCB-Toolkit\ConfigMgrTools.msi" /qb /l*v
c:\ConfigMgrTools.msi.log') -wait
```

9. Upgrade to Current Branch 1610

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	What's new in version 1610 of Configuration Manager https://docs.microsoft.com/en-us/sccm/core/plan-design/changes/whats-new-in-version-1610 Install in-console updates for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/servers/manage/install-in-console-updates Upgrade checklists https://docs.microsoft.com/en-us/sccm/core/servers/deploy/install/upgrade-to-configuration-manager Checklist for installing update 1610 for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/servers/manage/checklist-for-installing-update-1610
Description	In this chapter, we will be upgrading the existing installation (version 1606) to the latest (version 1610) and turning on some of the extra available features.

9.1. Before you begin

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console
02. if a new version is available, a message will appear with the text: A new update is available for Configuration Manager. You can view and enable available updates in the Administration workspace from the Cloud Services > Updates and Servicing node.
03. Click Administration.
04. Expand Cloud Services and then click Updates and Servicing **Note:** All updates will be listed and in the State column, it will show as Available **Note:** All downloaded files are going to be saved to C:\ConfigMgr\EasySetupPayload
05. You can also review the following logs: • ConfigMgr\Logs\Dmpdownloader.log: Records details on downloads from Microsoft and Intune.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"
```

```
gwmi -Namespace "root\SMS\site $($SiteCode)" -query "select * from
SMS_CM_UpdatePackages where Name like 'Configuration Manager 1610%' AND
UpdateType = 0"
```

9.2. Run the prerequisite check

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Cloud Services and then click Updates and Servicing
03. Select Configuration Manager 1610 and click Run prerequisite check
04. Click Monitoring
05. Click Site Servicing Status
06. Click Show Status to monitor the prerequisite check
07. You can also review the errors and warnings by examining the content of the C:\ConfigMgrPrereq.log file

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

while ($true)
{
      $SiteUpdate = gwmi -Namespace "root\SMS\site_$($SiteCode)" -query
"select * from SMS_CM_UpdatePackages where Name like 'Configuration Manager
1610%' AND UpdateType = 0"
      if ($SiteUpdate -ne $null) {
            if ($SiteUpdate.State -ne 262146) {
                  Write-Host " Update is in Downloading state..."
                  Start-Sleep 30
            } else {
                  Write-Host "Update is ready, executing pre-req"
                  $SiteUpdate.UpdatePrereqAndStateFlags(1,2)
                  break
            }
      }
}
```

9.3. Install an update

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Cloud Services and then click Updates and Servicing
03. Select Configuration Manager 1610 and click Install Update Pack
04. On Configuration Manager Updates Wizard, click Next
05. On Features, unselect all and click next **Note:** We will enable see how to enable new features later

06. On Options for Client Update, click Next
07. On Review and accept the terms for this update pack, select I accept the license terms and Privacy statement and click Next
08. On Summary, click Next
09. On Completion, click Close **Note:** The Installation will start and it will take some time to complete (60-90 minutes depending on your hardware). The State of the update is now show as Installed
10. Once the installation is completed, restart the console and the message “A new version of the console is available (5.00.8548.1500). Click ok to close the console and install the new version now. Click cancel to continue working with the old console (5.0.8412.1313). Working in the old console might corrupt data.”. Click Ok
11. Once completed, the console will restart
12. Click Administration.
13. Expand Cloud Services and then click Updates and Servicing **Note:** The State of the update is now show as Installed
14. You can also review the following logs: • C:\ConfigMgr\Logs\CMUpdate.log: Records details of the upgrade process. • C:\ConfigMgr\Logs\hman.log: Records information about site configuration changes, and the publishing of site information in Active Directory Domain Services. • C:\ConfigMgrAdminUISetup.log: SCCM console installation log

This can also be achieved via PowerShell using the commands below:

```
while ($true)
{
      $SiteUpdate = gwmi -Namespace "root\SMS\site_$($SiteCode)" -query
"select * from SMS_CM_UpdatePackages where Name like 'Configuration Manager
1610%' AND UpdateType = 0"
      if ($SiteUpdate -ne $null) {
            if ($SiteUpdate.State -ne 131074) {
                  Write-Host "Pre-Check is still happening..."
                  Start-Sleep 30
            } else {
                  Write-Host "Pre-Req done, starting update"
                  $SiteUpdate.UpdatePrereqAndStateFlags(2,2)
                  Get-Process -Name Microsoft.ConfigurationManagement |
Stop-Process
                  break
            }
      }
}

while ($true)
{
      $SiteUpdate = gwmi -Namespace "root\SMS\site_$($SiteCode)" -query
"select * from SMS_CM_UpdatePackages where Name like 'Configuration Manager
1610%' AND UpdateType = 0" -ErrorAction SilentlyContinue
      if ($SiteUpdate -ne $null) {
            if ($SiteUpdate.State -ne 196612) {
```

```
                  Write-Host "Installation is still happening..."
                  Start-Sleep 30
            } else {
                  Write-Host "Installation done"
                  Start-Process -Filepath
("C:\ConfigMgr\AdminConsole\bin\Microsoft.ConfigurationManagement.exe")
                  break
            }
      } else {
            Write-Host "Installation is still happening..."
            Start-Sleep 30
      }
}
```

9.4. Turning on Features

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Cloud Services, Updates and Servicing and then click Features
03. Select Windows Store for Business Integration and click Turn On
04. On the Confirmation Manager question, click Yes.
05. Restart the Configuration Manager Console

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$FeatureName = "Windows Store for Business Integration"
(gwmi -Namespace "root\SMS\site_$($SiteCode)" -query "select * from
SMS_CM_UpdateFeatures where Name
='$FeatureName'").UpdateFeatureExposureStatus(1)
```

10. Basic Site Configuration

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	Prepare Windows Servers to support System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/plan-design/network/prepare-windows-servers Deploy and manage content management infrastructure for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/servers/deploy/configure/deploy-and-manage-content Define site boundaries and boundary groups for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/servers/deploy/configure/define-site-boundaries-and-boundary-groups Run discovery for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/servers/deploy/configure/run-discovery
Firewall Rules	Many firewall rules are being created in this lab for the same set of TCP Ports (80 and 443). They are only being created multiple times to show what ports are required per Site Role. In a production environment, when co-hosting site roles, there is no need to create multiple rules for the same set of TCP Ports
Description	In this chapter, we will be performing the basic configuration of the SCCM. Installation and Configuration of the basic infrastructure for the SCCM to work, including Distribution Point, Management Point, FallBack Status Point, SQL Reporting Services Point, Boundaries, Boundary Group, Distribution Point Group, Network Access Account and Discovery. **Note:** Distribution Point and Management Point can be installed during the SCCM Installation. It has not been done in this lab because we wanted to show you how to perform the installation at later stage, allowing you to be prepared to perform the same steps when installing in a remove server.

10.1. Installation of basic Roles (DP, MP, FSP, SRS)

10.1.1. Distribution Point

10.1.1.1. Creating Firewall Rules

Perform this task on the SRV0002 virtual machine logged on as sccmadmin

01. Open Windows Firewall with Advanced Security and click Inbound Rules

02. Click New Rule
03. On New Inbound Rule Wizard, select Port and click Next
04. On Protocol and Ports select TCP and type 80 under specify local ports and click Next
05. On Action, click Next
06. On Profile, click Next
07. On Name, type IIS Distribution Point (TCP 80) Inbound and click Finish
08. Click New Rule
09. On New Inbound Rule Wizard, select Port and click Next
10. On Protocol and Ports select TCP and type 443 under specify local ports and click Next
11. On Action, click Next
12. On Profile, click Next
13. On Name, type IIS Distribution Point (TCP 443) Inbound and click Finish

This can also be achieved via PowerShell using the commands below:

```
New-NetFirewallRule -DisplayName "IIS Distribution Point (TCP 80) Inbound"
-Action Allow -Direction Inbound -LocalPort 80 -Protocol TCP
New-NetFirewallRule -DisplayName "IIS Distribution Point (TCP 443) Inbound"
-Action Allow -Direction Inbound -LocalPort 443 -Protocol TCP
```

10.1.1.2. Install Requirements

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next
06. Under select server roles, select Web Server (IIS)
07. On Add Roles and Features Wizard, click Add Features and click Next
08. Under Select Features, select Remote Differential Compression and click Next
09. Under Web Server Role (IIS), click Next
10. On Select role service, select: • Security->Windows Authentication • Application Deployment->ISAP Extensions • Management Tools->IIS 6 Management Compatibility->IIS 6 Metabase Compatibility • Management Tools->IIS 6 Management Compatibility->IIS 6 WMI Compatibility • Management Tools->IIS Management Scripts and Tools Click Next
11. Under Confirm installation selections, click Install
12. Once the installation is succeeded. Click Close

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name RDC | Install-WindowsFeature
Get-WindowsFeature -Name Web-Server | Install-WindowsFeature
Get-WindowsFeature -Name Web-ISAPI-Ext | Install-WindowsFeature
Get-WindowsFeature -Name Web-Metabase | Install-WindowsFeature
Get-WindowsFeature -Name Web-Windows-Auth | Install-WindowsFeature
```

10.1.1.3. Installing Site System Role

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Site System Roles
03. Right click \\SRV0002.classroom.intranet and click Add Site System Roles
04. On Add Site System Roles Wizard, General, click Next
05. Under proxy, click Next
06. Under Specify roles for this server, select Distribution Point and click Next
07. Under Specify distribution point settings select "Install and configure IIS if required by Configuration Manager". Leave the other default options and click Next
08. Under Specify drive settings for this distribution point click Next
09. Under Specify settings to install operating systems by using PXE boot click Next
10. Under specify multicast settings for operating system deployment click Next
11. Under specify the content validation settings, click Next
12. Under specify the boundary groups associate with this site system click Next
13. Under confirm the settings, click Next
14. Under You have successfully completed the Add Site System Roles wizard with the following settings click close
15. Click Monitoring
16. Expand System Status and click Component Status
17. Search for SMS_DISTRIBUTION_MANAGER
18. Right Click SMS_DISTRIBUTION_MANAGER, Show Messages and click All
19. Under Status Messages: Set Viewing Period, click OK
20. Verify the existence of Message ID 2302 **Note:** When installing a new Distribution Point, it is normal see this message for the default SCCM Client packages (<SITECODE>00002 and <SITECODE>00003) and SCCM will retry the package again every 30 minutes. Once SCCM successfully distribute the package, you will see the Message ID 2301
21. Double click any 2302 messages to see its details. Once done, click Ok
22. Verify the existence of Message ID 2399
23. Double click this message to see its details. Once done, click Ok
24. Verify the existence of Message ID 2362 **Note:** This message will appear if the Install and Configure IIS checkbox was selected
25. Double click this message to see its details. Once done, click Ok
26. Verify the existence of Message ID 9501

Note: This message will appear if the PXE options was not selected
27. Double click this message to see its details. Once done, click Ok
28. Verify the existence of Message ID 9503 **Note:** This message will appear if the Multicast option was not selected
29. Double click this message to see its details. Once done, click Ok
30. You can also review the following logs: • C:\ConfigMgr\Logs\DistMgr.log **Note:** As the Install and Configure IIS checkbox was selected, the DISM command line will run

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

if ((Get-CMSiteSystemServer -SiteSystemServerName "$servername") -eq $null)
{ New-CMSiteSystemServer -SiteCode $SiteCode -UseSiteServerAccount -
ServerName $servername }
Add-CMDistributionPoint -CertificateExpirationTimeUtc "$((Get-
Date).AddYears(20).ToString())" -SiteSystemServerName $servername -SiteCode
$siteCode -ClientConnectionType Intranet

start-sleep 90

$component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_DISTRIBUTION_MANAGER' and stmsg.MessageID = 2302 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '$SiteCode'"
if ($component -ne $null)
{
    Write-Host "ERROR: Found SMS_DISTRIBUTION_MANAGER 2302 id's" -
ForegroundColor Red
}

$component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_DISTRIBUTION_MANAGER' and stmsg.MessageID = 2391 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '$SiteCode'"
if ($component -ne $null)
{
    Write-Host "ERROR: Found SMS_DISTRIBUTION_MANAGER 2391 id's" -
ForegroundColor Red
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
```

```
'SMS DISTRIBUTION MANAGER' and stmsg.MessageID = 2362 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_DISTRIBUTION_MANAGER 2362 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_DISTRIBUTION_MANAGER' and stmsg.MessageID = 2399 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_DISTRIBUTION_MANAGER 2399 id's"
        break
    } else { Start-Sleep 10 }
}
```

10.1.2. Management Point

10.1.2.1. Creating Firewall Rules

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Windows Firewall with Advanced Security and click Inbound Rules
02. Click New Rule
03. On New Inbound Rule Wizard, select Port and click Next
04. On Protocol and Ports select TCP and type 80 under specify local ports and click Next
05. On Action, click Next
06. On Profile, click Next
07. On Name, type IIS Management Point (TCP 80) Inbound and click Finish
08. Click New Rule
09. On New Inbound Rule Wizard, select Port and click Next
10. On Protocol and Ports select TCP and type 443 under specify local ports and click Next
11. On Action, click Next
12. On Profile, click Next
13. On Name, type IIS Management Point (TCP 443) Inbound and click Finish
14. Click New Rule
15. On New Inbound Rule Wizard, select Port and click Next
16. On Protocol and Ports select TCP and type 10123 under specify local ports and click Next
17. On Action, click Next
18. On Profile, click Next
19. On Name, type IIS Client Notification (TCP 10123) Inbound and click Finish

This can also be achieved via PowerShell using the commands below:

```
New-NetFirewallRule -DisplayName "IIS Management Point (TCP 80) Inbound" -
Action Allow -Direction Inbound -LocalPort 80 -Protocol TCP
New-NetFirewallRule -DisplayName "IIS Management Point (TCP 443) Inbound" -
Action Allow -Direction Inbound -LocalPort 443 -Protocol TCP
New-NetFirewallRule -DisplayName "IIS Client Notification (TCP 10123)
Inbound" -Action Allow -Direction Inbound -LocalPort 10123 -Protocol TCP
```

10.1.2.2. Install Requirements

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next
06. Under select server roles, select Web Server (IIS)
07. On Add Roles and Features Wizard, click Add Features and click Next
08. Under Select Features, select Background Intelligent Transfer Service (BITS)->IIS Server Extension
09. On Add Roles and Features Wizard, click Add Features and click Next
10. Under Web Server Role (IIS), click Next
11. On Select role service, select: • Security->Windows Authentication • Application Deployment->ISAP Extensions • Management Tools->IIS 6 Management Compatibility->IIS 6 Metabase Compatibility • Management Tools->IIS 6 Management Compatibility->IIS 6 WMI Compatibility • Management Tools->IIS Management Scripts and Tools. Click Next
12. Under Confirm installation selections, click Install
13. Once the installation is succeeded. Click Close

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name Web-Server | Install-WindowsFeature
Get-WindowsFeature -Name Web-ISAPI-Ext | Install-WindowsFeature
Get-WindowsFeature -Name Web-Metabase | Install-WindowsFeature
Get-WindowsFeature -Name Web-Windows-Auth | Install-WindowsFeature
Get-WindowsFeature -Name BITS-IIS-Ext | Install-WindowsFeature
```

10.1.2.3. Installing Site System Role

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Site System Roles
03. Right click \\SRV0002.classroom.intranet and click Add Site System Roles
04. On Add Site System Roles Wizard, General, click Next
05. Under proxy, click Next
06. Under Specify roles for this server, select Management Point and click Next
07. Under Specify management point settings, click Next
08. Under specify management point database settings click Next
09. Under confirm the settings, click Next
10. Under You have successfully completed the Add Site System Roles Wizard with the following settings click Close
11. Click Monitoring
12. Expand System Status and click Component Status
13. Search for SMS_MP_CONTROL_MANAGER
14. Right Click SMS_MP_CONTROL_MANAGER, Show Messages and click All
15. Under Status Messages: Set Viewing Period, click OK
16. Verify the existence of Message ID 1013, 1014 and 1015
17. Double click any of the 1013, 1014 and 1015 messages to see its details. Once done, click Ok
18. Verify the existence of Message ID 500
19. Double click on the messages to see its details. Once done, click Ok
20. Verify the existence of Message ID 5460.
21. Double click on the message to see its details. Once done, click Ok
22. Open Internet Explorer and navigate to http://SRV0002.classroom.intranet/sms_mp/.sms_aut?mplist
23. Navigate to http://SRV0002.classroom.intranet/sms_mp/.sms_aut?mpcert
24. You can also review the following logs: • C:\ConfigMgr\Logs\MPSetup.log: Records the installation wrapper process. • C:\ConfigMgr\Logs\mpMSI.log: Records details of installation.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

if ((Get-CMSiteSystemServer -SiteSystemServerName "$servername") -eq $null)
{ New-CMSiteSystemServer -SiteCode $SiteCode -UseSiteServerAccount -
ServerName $servername }

Add-CMManagementPoint -SiteSystemServerName $servername -SiteCode $siteCode

start-sleep 90

while ($true)
{
```

```
    $component = gwmi -Namespace ("root\sms\site $SiteCode") -query "select
stmsgin.InsStrValue from SMS StatMsg stmsg inner join SMS StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_MP_CONTROL_MANAGER' and stmsg.MessageID = 1013 and stmsgin.InsStrIndex
= 0 and stmsgin.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_MP_CONTROL_MANAGER 1013 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS_MP_CONTROL_MANAGER' and stmsg.MessageID = 1014 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_MP_CONTROL_MANAGER 1014 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS_MP_CONTROL_MANAGER' and stmsg.MessageID = 1015 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_MP_CONTROL_MANAGER 1015 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS_MP_CONTROL_MANAGER' and stmsg.MessageID = 500 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS MP CONTROL MANAGER 500 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
```

```
'SMS MP CONTROL MANAGER' and stmsg.MessageID = 5460 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_MP_CONTROL_MANAGER 5460 id's"
        break
    } else { Start-Sleep 10 }
}

$web = New-Object -ComObject msxml2.xmlhttp
$url = "http://$($servername):80/sms mp/.sms aut?mplist"
try
{
      $web.open('GET', $url, $false)
      $web.send()

      Write-host "MPList HTTP Return $($web.status)"
} catch {
      Write-host "MPList ERROR: $($_)" -ForegroundColor Red
}

$web = New-Object -ComObject msxml2.xmlhttp
$url = "http://$($servername):80/sms_mp/.sms_aut?mpcert"
try
{
      $web.open('GET', $url, $false)
      $web.send()

      Write-host "MPCert HTTP Return $($web.status)"
} catch {
      Write-host "MPCert ERROR: $($_)" -ForegroundColor Red
}
```

10.1.3. Fallback Status Point

10.1.3.1. Creating Firewall Rules

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Windows Firewall with Advanced Security and click Inbound Rules
02. Click New Rule
03. On New Inbound Rule Wizard, select Port and click Next
04. On Protocol and Ports select TCP and type 80 under specify local ports and click Next
05. On Action, click Next
06. On Profile, click Next
07. On Name, type IIS Fallback Status Point (TCP 80) Inbound and click Finish

This can also be achieved via PowerShell using the commands below:

```
New-NetFirewallRule -DisplayName "IIS Fallback Status Point (TCP 80)
Inbound" -Action Allow -Direction Inbound -LocalPort 80 -Protocol TCP
```

10.1.3.2. Install Fallback Requirements

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next
06. Under select server roles, select Web Server (IIS)
07. On Add Roles and Features Wizard, click Add Features
08. Under Select Features, click Next
09. On Add Roles and Features Wizard, click Add Features and click Next
10. Under Web Server Role (IIS), click Next
11. On Select role service, select: • Management Tools->IIS 6 Management Compatibility->IIS 6 Metabase Compatibility • Management Tools->IIS 6 Management Compatibility->IIS 6 WMI Compatibility • Management Tools->IIS Management Scripts and Tools. Click Next
12. Under Confirm installation selections, click Install
13. Once the installation is succeeded. Click Close

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name Web-Server | Install-WindowsFeature
Get-WindowsFeature -Name Web-Metabase | Install-WindowsFeature
```

10.1.3.3. Installing Site System Role

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Site System Roles
03. Right click \\SRV0002.classroom.intranet and click Add Site System Roles
04. On Add Site System Roles Wizard, General, click Next
05. Under proxy, click Next
06. Under Specify roles for this server, select Fallback Status Point and click Next
07. Under Specify Fallback Status Point settings leave the default settings and click Next
08. Under confirm the settings, click Next
09. Under You have successfully completed the Add Site System Roles wizard with the following settings click close
10. Click Monitoring
11. Expand System Status and click Component Status
12. Search for SMS_FALLBACK_STATUS_POINT

13. Right Click SMS_FALLBACK_STATUS_POINT, Show Messages and click All
14. Under Status Messages: Set Viewing Period, click OK
15. Verify the existence of Message ID 1013, 1014 and 1015
16. Double click any of the 1013, 1014 and 1015 messages to see its details. Once done, click Ok
17. Verify the existence of Message ID 500
18. Double click on the messages to see its details. Once done, click Ok
19. You can also review the following logs: • C:\ConfigMgr\Logs\fspMSI.log: Records messages generated by the installation. • C:\ConfigMgr\Logs\SMSFSPSetup.log: Records messages generated by the installation.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

if ((Get-CMSiteSystemServer -SiteSystemServerName "$servername") -eq $null)
{ New-CMSiteSystemServer -SiteCode $SiteCode -UseSiteServerAccount -
ServerName $servername }
Add-CMFallbackStatusPoint -SiteSystemServerName $servername -SiteCode
$siteCode

start-sleep 90

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsgin.InsStrValue from SMS StatMsg stmsg inner join SMS StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS_FALLBACK_STATUS_POINT' and stmsg.MessageID = 1013 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_FALLBACK_STATUS_POINT 1013 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_FALLBACK_STATUS_POINT' and stmsg.MessageID = 1014 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_FALLBACK_STATUS_POINT 1014 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
```

```
    $component = gwmi -Namespace ("root\sms\site $SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS_FALLBACK_STATUS_POINT' and stmsg.MessageID = 1015 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS FALLBACK STATUS POINT 1015 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS FALLBACK STATUS POINT' and stmsg.MessageID = 500 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS FALLBACK STATUS POINT 500 id's"
        break
    } else { Start-Sleep 10 }
}
```

10.1.4. Reporting Services Point

10.1.4.1. Creating Firewall Rules

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Windows Firewall with Advanced Security and click Inbound Rules
02. Click New Rule
03. On New Inbound Rule Wizard, select Port and click Next
04. On Protocol and Ports select TCP and type 80 under specify local ports and click Next
05. On Action, click Next
06. On Profile, click Next
07. On Name, type SQL Server Reporting Services (TCP 80) Inbound and click Finish

This can also be achieved via PowerShell using the commands below:

```
New-NetFirewallRule -DisplayName "SQL Server Reporting Services (TCP 80)
Inbound" -Action Allow -Direction Inbound -LocalPort 80 -Protocol TCP
```

10.1.4.2. Installing Site System Role

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Site System Roles
03. Right click \\SRV0002.classroom.intranet and click Add Site System Roles
04. On Add Site System Roles Wizard, General, click Next

05. Under proxy, click Next
06. Under Specify roles for this server, select Reporting Services Point and click Next
07. Under Specify Reporting Services Point settings, click Verify and under User name click Set New Account
08. Under Windows User Account, type the following; • User name: CLASSROOM\svc_ssrsea • Password: Pa$$w0rd • Confirm password: Pa$$w0rd Click Ok
09. Once back to the Add Site System Roles Wizard, click Next
10. Under confirm the settings, click Next
11. Under You have successfully completed the Add Site System Roles wizard with the following settings click close
12. Click Monitoring
13. Expand System Status and click Component Status
14. Search for SMS_SRS_REPORTING_POINT
15. Right Click SMS_SRS_REPORTING_POINT, Show Messages and click All
16. Under Status Messages: Set Viewing Period, click OK
17. Verify the existence of Message ID 1013, 1014 and 1015
18. Double click any of the 1013, 1014 and 1015 messages to see its details. Once done, click Ok
19. Verify the existence of Message ID 500
20. Double click on the messages to see its details. Once done, click Ok
21. Monitor C:\ConfigMgr\Logs\srsrp.log for the status of the reports importation. **Note:** Once the Reporting Services Point is installed, SCCM will import reports into the SQL Server Reporting Service. As there are few hundred reports, this process can take some time.
22. Return to Configuration Manager console, monitoring workspace, expand Reporting and click Reports **Note:** You should see all reports being populated. Once the process of importing the reports are done, there should be over 450 reports.
23. You can also review the following logs: • C:\ConfigMgr\Logs\srsrpsetup.log: Records messages generated by the installation. • C:\ConfigMgr\Logs\srsrpMSI.log: Records messages generated by the installation. • C:\ConfigMgr\Logs\srsrp.log: Records information about the activity and status of the reporting services point.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$Secure = 'Pa$$w0rd'| ConvertTo-SecureString -AsPlainText -Force
$account = "CLASSROOM\svc_ssrsea"
New-CMAccount -Name "$account" -Password $Secure -SiteCode "$SiteCode"
```

```
if ((Get-CMSiteSystemServer -SiteSystemServerName "$servername") -eq $null)
{ New-CMSiteSystemServer -SiteCode $SiteCode -UseSiteServerAccount -
ServerName $servername }
Add-CMReportingServicePoint -ReportServerInstance "MSSQLSERVER" -
SiteSystemServerName "$servername" -UserName "$account" -DatabaseName
"CM $SiteCode" -DatabaseServerName "$servername" -FolderName
"ConfigMgr $SiteCode" -SiteCode "$SiteCode"

start-sleep 90

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
'SMS SRS REPORTING POINT' and stmsg.MessageID = 1013 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS SRS REPORTING POINT 1013 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS SRS REPORTING POINT' and stmsg.MessageID = 1014 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_SRS_REPORTING_POINT 1014 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS SRS REPORTING POINT' and stmsg.MessageID = 1015 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS SRS REPORTING POINT 1015 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_SRS_REPORTING_POINT' and stmsg.MessageID = 500 and stmsg.SiteCode =
'$SiteCode'"
```

```
    if ($component -ne $null)
    {
        Write-Host "Found SMS_SRS_REPORTING_POINT 500 id's"
        break
    } else { Start-Sleep 10 }
}

$web = New-Object -ComObject msxml2.xmlhttp
$url = "http://localhost:80/reportserver/ConfigMgr_$SiteCode"
while ($true)
{
    try
    {
        $web.open('GET', $url, $false)
            $web.send()

            if ($web.status -eq "404") { start-sleep 10 }
            if ($web.status -eq "200")
            {
                        Write-Host "Found ConfigMgr $SiteCode reporting
site"
                  break
            }
    } catch {
      #
    }
}
```

10.2. Boundaries

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Hierarchy Configuration and click Boundaries
03. Click Create Boundary
04. On General select: • Type: IP Address Range • Starting IP Address: 192.168.3.1 • Ending IP Address: 192.168.3.254 Click Ok

This can also be achieved via PowerShell using the commands below:

```
New-CMBoundary -DisplayName "Training Lab Boundary" -BoundaryType IPRange -
Value "192.168.3.1-192.168.3.254"
```

10.3. Boundary Group

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Hierarchy Configuration and click Boundary Group
03. Click Create Boundary Group
04. On General: • Name: Training Lab • Boundaries: 192.168.3.1-192.168.3.254 Change to the References Tab
05. On References: • Select Use this boundary group for site assignment • Site Systems Servers: \\SRV0002.rflsytems.intranet Click OK
06. Right Click Training Lab and click Show Members
07. The 192.168.3.1-192.168.3.254 boundary was added to the boundary group Training Lab

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

New-CMBoundaryGroup -Name "Training Lab" -DefaultSiteCode $SiteCode
Add-CMBoundaryToGroup -BoundaryGroupName "Training Lab" -BoundaryName
"Training Lab Boundary"
$boundarygroup = gwmi -Namespace ("root\sms\site $SiteCode") -query
('select * from SMS_BoundaryGroup where Name="Training Lab"')
$Flag = 0
$NALPath = "[""Display=\\" +  $ServerName +
"\""]MSWNET:[""SMS_SITE=$SiteCode""]\\" + $ServerName + "\"

if ($boundarygroup -ne $null) { $boundarygroup.AddSiteSystem($NALPath,
$Flag) }
```

10.4. Distribution Point Group

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration
02. Click Distribution Point Group
03. Click Create Group
04. On General, type Training Lab under Name and then Click Add
05. On Add Distribution Points, select SRV0002.CLASSROOM.INTRANET and click OK twice
06. Right Click Training Lab and click Show Members

07. The SRV0002.CLASSROOM.INTRANET boundary was added to the distribution point group Training Lab

This can also be achieved via PowerShell using the commands below:

```
$servername = "SRV0002.classroom.intranet"

New-CMDistributionPointGroup -Name "Training Lab"
Add-CMDistributionPointToGroup -DistributionPointGroupName "Training Lab" -
DistributionPointName "$servername"
```

10.5. Network Access Account

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Sites
03. Select 001 – Training Lab site and click Configuration Site Components -> Software Distribution
04. On Software Distribution Component Properties, change to the Network Access Account tab
05. Under Network Access Account tab select Specify the account that accesses network locations and click Set -> New Account
06. Under Windows User account type: • User Name: CLASSROOM\svc_sccmna • Password: Pa$$w0rd • Confirm Password: Pa$$w0rd Click Verify
07. Under verify type \\SRV0002\sms_site for Network Share and click Test Connection
08. Once the connection was successfully verified, click Ok three times

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$Secure = 'Pa$$w0rd'| ConvertTo-SecureString -AsPlainText -Force
$account = "CLASSROOM\svc_sccmna"
New-CMAccount -Name "$account" -Password $Secure -SiteCode $SiteCode
Set-CMSoftwareDistributionComponent -SiteCode $SiteCode -
NetworkAccessAccountNames $account
```

10.6. Discovery

10.6.1. Active Directory Forest Discovery

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Hierarchy Configuration and click Discovery Methods
03. Select Active Directory Forest Discovery and click Properties

04. Under General, select Enabled Active Directory Forest Discovery and select Automatically create IP address range boundaries for IP subnets when they are discovered and Automatically create Active Directory site boundaries when they are discovered. Leave the schedule to run every 1 weeks and click OK
05. When asked if you want to run full discovery as soon as possible, click Yes
06. Select Boundaries and confirm that 3 new boundaries have been populated there
07. Select Active Directory Forests and confirm that the Classroom.intranet is populated there.
08. You can also review the following logs: • C:\ConfigMgr\Logs\ADForestDisc.Log: Records Active Directory Forest Discovery actions.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

Set-CMDiscoveryMethod -ActiveDirectoryForestDiscovery -
EnableActiveDirectorySiteBoundaryCreation $True -Enabled $True -
EnableSubnetBoundaryCreation $True -SiteCode $SiteCode
Invoke-CMForestDiscovery -SiteCode $SiteCode
```

10.6.2. Active Directory System Discovery

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Hierarchy Configuration and click Discovery Methods
03. Select Active Directory System Discovery and click Properties
04. Under General, select Enabled Active Directory System Discovery and add the active directory containers you want to search (using the yellow button) **Note:** In this exercise we will use the classroom.intranet domain
05. Click Polling Schedule to change the default schedule
06. Click Active Directory Attributes to add more attributes to be discovered **Note:** In this exercise, we will add pwdLastSet attribute
07. Click options to add a filter to the discovered machines **Note:** This option requires an active directory functional level of Windows Server 2003 or later **Note:** In this exercise, we will select both options with default value of 90 days
08. Click Ok and When asked if you want to run full discovery as soon as possible, click Yes
09. Click Assets and Compliance and select Devices. **Note:** There are a few more machines discovered now. **Note:** Disabled Computers Accounts in Active Directory and Computers that do not have an IP Address are not going to be discovered
10. Select WKS0001 and click Properties
11. Once the WKS0001 Properties open, note the discovery method used under Agent Name.
12. note the pwdLastSet property was also added to the list. Click Ok

13. You can also review the following logs:

- C:\ConfigMgr\Logs\adsysdis.log: Records Active Directory System Discovery actions.

Note: Look for a line that starts: INFO: Search filter. This line shows the actual search query that was send to AD. There are two types of queries: Full synchronization and Incremental synchronization.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$domainName = "LDAP://DC=$($env:USERDNSDOMAIN.Split(".") -join ",DC=")"
Set-CMDiscoveryMethod -ActiveDirectorySystemDiscovery -
AddActiveDirectoryContainer "$($domainName)" -Enabled $True -
EnableDeltaDiscovery $True -EnableFilteringExpiredLogon $True -
EnableFilteringExpiredPassword $True -SiteCode $SiteCode -
TimeSinceLastLogonDays 90 -TimeSinceLastPasswordUpdateDays 90 -
AddAdditionalAttribute @("pwdLastSet")
Invoke-CMSystemDiscovery -SiteCode $SiteCode
```

10.6.3. Active Directory User Discovery

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Hierarchy Configuration and click Discovery Methods
03. Select Active Directory User Discovery and click Properties
04. Under General, select Enabled Active Directory User Discovery and add the active directory containers you want to search (using the yellow button) **Note:** In this exercise we will use the classroom.intranet domain
05. Click Polling Schedule to change the default schedule
06. Click Active Directory Attributes to add more attributes to be discovered **Note:** In this exercise, we will add physicalDeliveryOfficeName and department attribute
07. Click Ok and When asked if you want to run full discovery as soon as possible, click Yes
08. Click Assets and Compliance and select Users.
09. Select CLASSROOM\User01 (User01) and click Properties
10. Once the CLASSROOM\User01 (User01) Properties open, note the discovery method used under Agent Name.
11. Note the physicalDeliveryOfficeName and department property was also added to the list. Click Ok
09. You can also review the following logs: • C:\ConfigMgr\Logs\adusrdis.log: Records Active Directory User Discovery actions. **Note:** Look for a line that starts: INFO: Search filter. This line shows the actual search query that was send to AD. There are two types of queries: Full synchronization and Incremental synchronization.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$domainName = "LDAP://DC=$($env:USERDNSDOMAIN.Split(".") -join ",DC=")"
Set-CMDiscoveryMethod -ActiveDirectoryUserDiscovery -
AddActiveDirectoryContainer "$($domainname)" -DeltaDiscoveryIntervalMinutes
30 -Enabled $True -EnableDeltaDiscovery $True -SiteCode $SiteCode -
AddAdditionalAttribute @("physicalDeliveryOfficeName")
Invoke-CMUserDiscovery -SiteCode $SiteCode
```

10.6.4. Active Directory Group Discovery

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Hierarchy Configuration and click Discovery Methods
03. Select Active Directory Group Discovery and click Properties
04. Under General, select Enabled Active Directory Group Discovery and click Add -> Location
05. Under Add Active Directory Location add active directory containers you want to search and click Ok **Note:** In this exercise, we will use the classroom.intranet domain
06. One Back to the Active Directory Group Discovery Properties, click Next
07. Click Polling Schedule to change the default schedule
08. Click options to add a filter to the discovered machines **Note:** This option required active directory functional level of Windows Server 2003 or later **Note:** In this exercise, we will leave the default unchecked option
09. Click Ok and When asked if you want to run full discovery as soon as possible, click Yes
10. Click Assets and Compliance and select Users.
11. Select CLASSROOM\Allowed RODC Password Replication Group and click Properties
12. Once the CLASSROOM\Allowed RODC Password Replication Group Properties open, note the discovery method used under Agent Name.
13. You can also review the following logs: • C:\ConfigMgr\Logs\adsgdis.log: Records Active Directory Group Discovery actions. **Note:** Look for a line that starts: INFO: Search filter. This line shows the actual search query that was send to AD. There are two types of queries: Full synchronization and Incremental synchronization.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$domainName = "LDAP://DC=$($env:USERDNSDOMAIN.Split(".") -join ",DC=")"
$discovery = New-CMADGroupDiscoveryScope -LdapLocation "$($domainName)" -
Name "$($domainName)" -SiteCode $SiteCode -RecursiveSearch $true
```

```
Set-CMDiscoveryMethod -ActiveDirectoryGroupDiscovery -
AddGroupDiscoveryScope ($discovery) -Enabled $True -SiteCode $SiteCode -
EnableDeltaDiscovery $true
Invoke-CMGroupDiscovery -SiteCode $SiteCode
```

11. Basic Client Settings

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	About client settings in System Center Configuration Manager https://technet.microsoft.com/en-us/library/mt629384.aspx
Comments	Configuring default client settings apply to every single SCCM client. Depending on the settings it is recommended to create a new client setting and apply only to computers that need the new settings
Description	In this chapter, we will be changing the Default Client Settings with the basic configuration that will be required during this lab. **Note:** Changing the default client settings apply to every single SCCM client. Depending on the settings it is recommended to create a new client setting and apply only to computers that need the new settings

11.1. Changing Default Client Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Client Settings
03. Select the default client settings and click Properties **Note:** The Changing the Default Client Settings should be avoided as it applies to all clients and there is no way to revert it back. My recommendation is to create a new client settings with the changes and deploying to a collection with all clients if needed
04. Under Default Settings, click Computer Agent and change Organization name displayed in Software Center to Training Lab
05. Click Hardware Inventory and change the schedule from 7 days to 1 day
06. Click Software Inventory and change the Schedule from 7 days to 1 day
07. Under software Inventory click Set Types
08. Under Configure Client Settings, add file type *.exe and click Ok **Note:** Adding *.exe for All client hard disk will take long and will fills the database with probably useless information. Validate your environment and use a better path location such as *.exe from %programfiles% or *.sys from %systemroot%\system32\drivers.
09. Back to Default settings **Note:** The authors' personal recommendation is disabling software inventory, as it does not bring any real benefit. As it relies on WMI to search for the files, and WMI is not indexed, it may take longer than expected and thus may affect the client performance
10. Click State Messaging and set the reporting cycle to 2 minutes. Click Ok.

Note: Setting a low value here will allow workstations to send information quicker to the server, however, setting it to a low value is not recommended in a production environment as it will affect Server Performance as well as may increasing network traffic.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$ClientSettingsName = "Default Client Agent Settings"
Set-CMClientSetting -ComputerAgentSettings -Name "$ClientSettingsName" -
BrandingTitle "Training Lab"

$schedule = New-CMSchedule -RecurCount 1 -RecurInterval Days
Set-CMClientSetting -HardwareInventorySettings -Name "$ClientSettingsName"
-EnableHardwareInventory $True -InventorySchedule $Schedule

Set-CMClientSetting -Name "$ClientSettingsName" -SoftwareInventorySettings
-EnableSoftwareInventory $True -SoftwareInventorySchedule $schedule
$settings = gwmi -Namespace "root\sms\site_$SiteCode" -Class
SMS_SCI_ClientComp -Filter "FileType=2 and ItemName='Software Inventory
Agent' and ItemType='Client Component' and SiteCode='$($SiteCode)'"
$values = @("*.exe", "*", "true", "true", "true")
$reg = $settings.RegMultiStringLists
for($i=0;$i -le 4; $i++)
{
    $reg[$i].ValueStrings += $values[$i]
}
$settings.RegMultiStringLists = $reg
$settings.Put()

Set-CMClientSetting -Name "$ClientSettingsName" -StateMessageSettings -
StateMessagingReportingCycleMinutes 2
```

12. Client Installation

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 SRV0003 WKS0001 WKS0002 WKS0004
More information	About client installation properties in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/deploy/about-client-installation-properties How to deploy clients to Windows computers in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/deploy/deploy-clients-to-windows-computers How to deploy clients to UNIX and Linux servers in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/deploy/deploy-clients-to-unix-and-linux-servers
Description	In this chapter, we will be configuring SCCM for the client installation (Windows only), will be installing Windows and Linux client as well as looking at the initial information that is returned from the client once it has been installed

12.1. Windows Client Installation

12.1.1. Push Configuration

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Sites
03. Select 001 – Training Lab, Client Installation Settings and click Client Push Installation
04. On Client Push Installation Properties, change to the Accounts Tab
05. On the Accounts tab, click New (yellow button)
06. Under Windows User Account, type CLASSROOM\svc_sccmpush as UserName and Pa$$word for password and confirm password. Click Verify >>
07. Under verify, select data source network share and network share type \\SRV0002\SMS_SITE and click Test Connection
08. Click Ok twice when the connection is successfully verified
09. Once back to the Client Push Installation Properties, change to the Installation Properties tab
10. On the Installation Properties tab, type FSP=SRV0002 at the end of the Installation Properties and click Ok

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$Secure = 'Pa$$w0rd'| ConvertTo-SecureString -AsPlainText -Force
$account = "CLASSROOM\svc_sccmpush"
New-CMAccount -Name "$account" -Password $Secure -SiteCode "$SiteCode"

Set-CMClientPushInstallation -ChosenAccount "$account" -
EnableAutomaticClientPushInstallation $False -
EnableSystemTypeConfigurationManager $False -EnableSystemTypeServer $True -
EnableSystemTypeWorkstation $True -InstallationProperty
"SMSSITECODE=$($SiteCode) FSP=$($servername)" -
InstallClientToDomainController $False -SiteCode "$($SiteCode)"
```

12.1.2. Manual Push

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Devices
03. Select WKS0001 and click Install Client
04. On Before You Begin, click Next
05. Under Installation Options, select Install the client software from a specific site, confirm the site 001 is selected and click Next
06. Under Summary, click Next
07. Under Completion, click Close
08. You can also review the following logs: • C:\ConfigMgr\Logs\ccm.log: Records client push installation activities. **Note:** Repeat the process for the WKS0002 and WKS0004 machines

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

#note: you may need to force an update of the All System collection if
there is no WKS0001 device
#Invoke-CMDeviceCollectionUpdate -Name "All Systems"
Install-CMClient -DeviceName "WKS0001" -SiteCode "$SiteCode"
Install-CMClient -DeviceName "WKS0002" -SiteCode "$SiteCode"
Install-CMClient -DeviceName "WKS0004" -SiteCode "$SiteCode"
```

12.1.3. Validating the Installation and Installation Process on the Client

Perform this task on the WKS0001 virtual machine logged on as user01
01. On the client WKS0001, open Task manager and confirm that ccmsetup.exe is running
02. Examine the content of C:\windows\ccmsetup folder

03. Once the Installation is completed, the ccmexec.exe process will appear instead of ccmsetup.exe
04. Open Control Panel and confirm that Configuration Manager exist. Open it.
05. On the Configuration Manager Properties, General Tab, confirm the Site Code is SMS:001 and Assigned management point is SRV0002.CLASSROOM.INTRANET. Change to the Components tab
06. Under Components tab, confirm the Components that are installed and enabled. Change to the Actions tab
07. Under Actions tab, confirm that there are 2 actions. Change to the Site tab **Note:** After the installation, it is normal to have only 2 actions until the post-installation process finishes. After this, there will be 10 actions to be used, depending on the options enabled in the client settings
08. Under site, confirm the Configuration manager service location is set to 001. Click Ok
09. Open Software Center under Start -> Microsoft System Center -> Configuration Manager
10. Confirm that the Software Center opens without any problem.
11. You can also review the following client logs: • C:\Windows\CCMSetup\Logs\ccmsetup.log: Records ccmsetup tasks for client setup, client upgrade, and client removal. Can be used to troubleshoot client installation problems. • C:\Windows\CCMSetup\Logs\client.msi.log: Records setup tasks performed by client.msi. Can be used to troubleshoot client installation or removal problems.

This can also be achieved via PowerShell using the commands below:

```
while ($true)
{
      $Process = Get-Process -Name ccmsetup -ErrorAction SilentlyContinue
      if ($Process -ne $null) { Start-Sleep 10 }
      else { Write-host "Process ccmsetup.exe does not exist or already
finished"; break }
}

while ($true)
{
      $Process = Get-Process -Name ccmexec -ErrorAction SilentlyContinue
      if ($Process -eq $null) { Start-Sleep 10 }
      else { Write-host "Process ccmexec exist"; break }
}
start-sleep 60

"Client is assigned to $((Invoke-WMIMethod -Namespace root\ccm -Class
SMS_Client -Name GetAssignedSite).sSiteCode)"
```

12.1.4. Validating the Installation and Installation Process on the SCCM Console

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Devices and confirm that WKS0001 says: • Installed: Yes

- Site Code: 001
- Client Activity: Active

Note: The Client Activity will only show Active after the client post-installations have been completed

This can also be achieved via PowerShell using the commands below:

```
#note: you may need to force an update of the All System collection if
there is no WKS0001 device
#Invoke-CMDeviceCollectionUpdate -Name "All Systems"
#start-sleep 60
Get-CMDevice -Name "WKS000?" | select Name, IsClient, SiteCode,
ClientActiveStatus
```

12.2. Linux Client Installation

Perform this task on the SRV0003 virtual machine logged on as administrator
01. On the Linux desktop, select Places and then Browse Network
02. Click Connect to Server
03. On Connect to Server, type smb://srv0001.classroom.intranet and click connect
04. On Password required for srv0001.classroom.intranet use: • Username: administrator • Domain: classroom.intranet • Password: Pa$$w0rd Click Connect
05. Navigate to TrainingFiles\Source\SCCMCB-LinuxClient and Copy the Extract folder to the Desktop
06. Close the File Explorer
07. Right click Extract and click Open in terminal
08. Execute the command: chmod +x install
09. Log on as a root using the command **su** and when asked the password type **Pa$$w0rd**
10. Type **./install -mp SRV0002.classroom.intranet -sitecode 001 ccm-Universalx64.tar** and press enter
11. Once the installation is completed, return to the SCCM Server (SRV0002), open the Assets and Compliance workspace and click Devices
12. Right click the visible columns and add Approved and Blocked
13. Select the linux machine. **Note:** The machine status is not Approved
14. Right click the Linux machine and click approve
15. Under the configuration manager confirmation window, click Yes
16. Return to the linux machine and type **/opt/microsoft/configmgr/bin/ccmexec -rs policy** and press enter

Note: You may need perform this task twice, but wait few seconds between the attempts
17. Type **/opt/microsoft/configmgr/bin/ccmexec -rs hinv** and press enter
18. Monitor the log file /var/opt/microsoft/scxcm.log **Note:** To get a similar result for logging as you do with CMTrace in a linux machine use: tail -f /var/opt/microsoft/scxcm.log

This can also be achieved via PowerShell using the commands below:

```
#tasks 1 to 10, follow the e-book

#sccm
#tasks 11 to 15
#update All System Client, so you can have the new client appearing
Invoke-CMDeviceCollectionUpdate -Name "All Systems"
start-sleep 60

Get-CMDevice -Name "srv0003.classroom.intranet" | Select Name, IsApproved,
IsBlocked
Approve-CMDevice -DeviceName "srv0003.classroom.intranet"
Get-CMDevice -Name "srv0003.classroom.intranet" | Select Name, IsApproved,
IsBlocked

#tasks 16 to 18, follow the e-book
```

12.3. Client Properties

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Devices
03. Select a Windows Client and click Properties
04. Once the Properties Open note the Agent Edition: Windows desktop or server
05. Operating System Name and version shows Microsoft Windows NT Workstation 10.0 (Tablet Edition). Click Ok
06. Select a Linux client and click Properties
07. Once the Properties Open note the Agent Edition: UNIX and Linux
08. Operating System Name and version shows CentOS Linux 7.0 x64. Click Ok

This can also be achieved via PowerShell using the commands below:

```
Get-CMDevice -Name "WKS000?" | select Name, ClientEdition, DeviceOS
Get-CMDevice -Name "srv0003.classroom.intranet" | select ClientEdition,
DeviceOS
```

12.4. Resource Explorer

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.

02. Click Devices
03. Right click a Windows computer and click Start -> Resource Explorer
04. Once the Resource Explorer open, expand hardware
05. Select Disk Drives
06. Select Logical Disk
07. Select Memory
08. Right click a Linux computer and click Start -> Resource Explorer
09. Once the resource explorer open, expand hardware
10. Select Operating System
11. Select Logical Disk
12. Select Installed Applications

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$ModulePath = $env:SMS_ADMIN_UI_PATH
if ($ModulePath -eq $null) {
      $ModulePath = (Get-ItemProperty -Path
"Registry::HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Control\Session
Manager\Environment").SMS_ADMIN_UI_PATH
}

$ModulePath = $ModulePath.Replace("bin\i386","bin\resourceexplorer.exe")
#windows machine
Start-Process -Filepath ("$ModulePath") -ArgumentList ('-s -
sms:ResExplrQuery="SELECT ResourceID FROM SMS_R_SYSTEM WHERE NAME =
''WKS0001''" -sms:connection=\\' + $servername + '\root\sms\site_' +
$siteCode)

#linux machine
Start-Process -Filepath ("$ModulePath") -ArgumentList ('-s -
sms:ResExplrQuery="SELECT ResourceID FROM SMS R SYSTEM WHERE NAME =
''srv0003.classroom.intranet''" -sms:connection=\\' + $servername +
'\root\sms\site_' + $siteCode)
```

13. Collections

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	Introduction to collections in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/collections/introduction-to-collections How to create collections in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/collections/create-collections Best practices for collections in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/collections/best-practices-for-collections
Collection Rules	Creating collection rules with like are not recommended as this could cause a performance issue on the SCCM and will slow down the collection membership update. In this lab, we are using like only for demonstration purposes. More information about this, refer to http://www.enhansoft.com/blog/configuration-manager-collections-and-collection-evaluation-viewer and http://www.enhansoft.com/blog/how-to-fix-a-poorly-written-wql-query
Description	In this chapter, we will be Creating several collections as well as chaning the default interval of collection membership incremental evaluation. **Note:** In this lab we are changing the collection membership incremental evaluation to speed up the process of updating collections and decreasing the default value is not recommended in a production environment

13.1. Windows 10 Workstation Collection

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Device Collections
03. Select Device Collections and click Create Device Collection
04. Under Specify details for this collection type Windows 10 Workstations under name and select All Systems under Limiting Collection. Click Next
05. Under Define membership rules for this collection, select Add query and click Query Rule
06. Under Query Rule Properties, type Windows 10 for Name and click Edit Query Statement
07. Under Query Statement Properties, change to criteria tab
08. Under criteria tab click New button
09. Under Criterion Properties, click Select
10. Under Select Attribute select: • Attribute Class: System Resource

• Attribute: Operating System Name and Version Click Ok
11. Under Criterion Properties, select Operator is Like and type %Windows NT Workstation 10% under value. Click Ok 3 times
12. Under Create Device Collection Wizard, select Use incremental updates for this collection and click Next **Note:** It is not recommended to have over 250 collections with the Incremental updates enabled
13. Under confirm the settings, click Next
14. Under The Create Device Collection Wizard completed successfully, click Close
15. Under device collections, the new Collection is still under update status. **Note:** Once the collection is created, there is a process to populate it and it may take a while. In this lab, wait 30 seconds or refresh it couple of times until you see the Member Count increment to 2
16. Select the collection and click Show Members
17. The collection will be expanded under Devices and all devices that match the query filter will be displayed.

This can also be achieved via PowerShell using the commands below:

```
$CollUpdate = New-CMSchedule -Start "01/01/2015 9:00 PM" -DayOfWeek
Saturday -RecurCount 1
$Collection = New-CMDeviceCollection -Name "Windows 10 Workstations" -
LimitingCollectionName "All Systems" -RefreshSchedule $CollUpdate -
RefreshType Both
Add-CMDeviceCollectionQueryMembershipRule -CollectionId
$Collection.CollectionID -RuleName "Windows 10" -QueryExpression "select *
from SMS_R_System where OperatingSystemNameandVersion like '%Windows NT
Workstation 10%'"
```

13.2. Windows 8 Workstation Collection

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Device Collections
03. Select Device Collections and click Create Device Collection
04. Under Specify details for this collection type Windows 8 Workstations under name and select All Systems under Limiting Collection. Click Next
05. Under Define membership rules for this collection, select Add query and click Query Rule
06. Under Query Rule Properties, type Windows 8 for Name and click Edit Query Statement
07. Under Query Statement Properties, change to criteria tab
08. Under criteria tab click New button
09. Under Criterion Properties, click Select
10. Under Select Attribute select: • Attribute Class: System Resource • Attribute: Operating System Name and Version

Click Ok
11. Under Criterion Properties, select Operator is Like and type %Windows NT Workstation 6.3% under value. Click Ok 3 times
12. Under Create Device Collection Wizard, select Use incremental updates for this collection and click Next **Note:** It is not recommended to have over 250 collections with the Incremental updates enabled
13. Under confirm the settings, click Next
14. Under The Create Device Collection Wizard completed successfully, click Close
15. Under device collections, the new Collection is still under update status. **Note:** Once the collection is created, there is a process to populate it and it may take a while. In this lab, wait 30 seconds or refresh it couple of times until you see Member Count change to 1
16. Select the collection and click Show Members
17. The collection will be expanded under Devices and all devices that match the query filter will be displayed.

This can also be achieved via PowerShell using the commands below:

```
$CollUpdate = New-CMSchedule -Start "01/01/2015 9:00 PM" -DayOfWeek
Saturday -RecurCount 1
$Collection = New-CMDeviceCollection -Name "Windows 8 Workstations" -
LimitingCollectionName "All Systems" -RefreshSchedule $CollUpdate -
RefreshType Both
Add-CMDeviceCollectionQueryMembershipRule -CollectionId
$Collection.CollectionID -RuleName "Windows 8"  -QueryExpression "select *
from SMS_R_System where OperatingSystemNameandVersion like '%Windows NT
Workstation 6.3%'"
```

13.3. CentOS Servers Collection

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Device Collections
03. Select Device Collections and click Create Device Collection
04. Under Specify details for this collection type CentOS Servers under name and select All Systems under Limiting Collection. Click Next
05. Under Define membership rules for this collection, select Add query and click Query Rule
06. Under Query Rule Properties, type CentOS Servers for Name and click Edit Query Statement
07. Under Query Statement Properties, change to criteria tab
08. Under criteria tab click New button
09. Under Criterion Properties, click Select
10. Under Select Attribute select: • Attribute Class: System Resource • Attribute: Operating System Name and Version

Click Ok
11. Under Criterion Properties, select Operator is Like and type CentOS Linux% under value. Click Ok 3 times
12. Under Create Device Collection Wizard, select Use incremental updates for this collection and click Next **Note:** It is not recommended to have over 250 collections with the Incremental updates enabled
13. Under confirm the settings, click Next
14. Under The Create Device Collection Wizard completed successfully, click Close
15. Under device collections, the new Collection is still under update status. **Note:** Once the collection is created, there is a process to populate it and it may take a while. In this lab, wait 30 seconds or refresh it couple of times until you see Member Count change to 1
16. Select the collection and click Show Members
17. The collection will be expanded under Devices and all devices that match the query filter will be displayed.

This can also be achieved via PowerShell using the commands below:

```
$CollUpdate = New-CMSchedule -Start "01/01/2015 9:00 PM" -DayOfWeek
Saturday -RecurCount 1
$Collection = New-CMDeviceCollection -Name "CentOS Servers" -
LimitingCollectionName "All Systems" -RefreshSchedule $CollUpdate -
RefreshType Both
Add-CMDeviceCollectionQueryMembershipRule -CollectionId
$Collection.CollectionID -RuleName "CentOS Servers"  -QueryExpression
"select * from SMS_R_System where OperatingSystemNameandVersion like
'CentOS Linux%'"
```

13.4. Collection Membership Incremental Evaluation

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Sites
03. Select 001 – Training Lab site and click Configuration Site Components -> Collection Membership Evaluation
04. On Collection Membership Evaluation Component Properties change the default interval (minutes) to 3 **Note:** It is not recommended to decrease the default interval, however, there are situations (normally when there are too many collections – over thousands, or too many with incremental updates – near the recommended 250 limit) where you would need to increase the interval to decrease the intensity of the processor usage.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

Set-CMCollectionMembershipEvaluationComponent -SiteCode $SiteCode -
EvaluationMins 3
```

14. Remote Control

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001
More information	Introduction to remote control in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/remote-control/introduction-to-remote-control Configuring remote control in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/remote-control/configuring-remote-control How to audit remote control usage in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/remote-control/audit-remote-control-usage
Description	In this chapter, we will be creating a Device Client Setting to be used for Remote Tools as well as configuring, validating and monitoring access via SCCM Remote Control

14.1. Creating Device Settings for Remote Tools

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Client Settings
03. Select Client Settings and click Create Custom Client Device Settings
04. Type Remote Control for Windows 10 on Name and select Remote Tools under Select the custom settings to be enforced on client devices
05. Click Remote Tools
06. Click Configure and select Enable Remote Control on client computers and Domain **Note:** Enabling Private and Public should only be used for non-domain machines. Using this option for domain machines may decrease its security depending on the settings you have selected. As example, if you select the Remote Desktop, the connection can be initiated from any network.
07. Under Remote Tools click Set Viewers and add CLASSROOM\SCCM Remote Tools. Click Ok **Note:** This option is only required if the operator does not have administrative rights on target machines or if the Grant Remote Control permission to local Administrators group is set to No
08. Under remote tools perform the following changes: • Prompt user for Remote Control permission: yes • Play a sound on client: None • Manage Remote Desktop settings: Yes • Allow Permitted viewers to connect by using Remote Desktop Connection: Yes

- Require network level authentication on computers that run Windows Vista operating system and later versions: No

Note: These settings are being used in a lab environment where security should not be an issue.

Click Ok

09. Select the Remote Control for Windows 10 and click Deploy

10. Under select collection, click Windows 10 Workstations and click Ok

11. Select Deployments and confirm that the Client Settings has been deployed to the collection

This can also be achieved via PowerShell using the commands below:

```
$ClientSettingsName = "Remote Control for Windows 10"
New-CMClientSetting -Name "$ClientSettingsName" -Type Device

Set-CMClientSetting -Name "$ClientSettingsName" -RemoteToolsSettings -
AccessLevel FullControl -AllowPermittedViewersToRemoteDesktop $True -
AllowRemoteControlOfUnattendedComputer $True -AudibleSignal PlayNoSound -
FirewallExceptionProfile Domain -ManageRemoteDesktopSetting $True -
ManageSolicitedRemoteAssistance $True -ManageUnsolicitedRemoteAssistance
$True -PermittedViewer "CLASSROOM\SCCM Remote Tools" -
RemoteAssistanceAccessLevel FullControl -RequireAuthentication $False -
PromptUserForPermission $True
Start-CMClientSettingDeployment -ClientSettingName "$ClientSettingsName" -
CollectionName "Windows 10 Workstations"
```

14.2. Checking the Policies that should be applied

Perform this task on the SRV0002 virtual machine logged on as sccmadmin

01. Start Configuration Manager Console and Click Assets and Compliance.

02. Click Devices

03. Select a Windows 10 Client and click Client Settings->Resultant Client Settings

04. Once the Resultant Client Settings open, check the Remote Tools and confirm the changes have been made. Click Ok

14.3. Validating Remote Control

Perform this task on the wks0001 virtual machine logged on as user01

01. Open the Configuration Manager Properties

02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now

Note: Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)

03. Under Machine Policy Retrieval & Evaluation Cycle click Ok

Note: Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Open Computer Management
05. Expand Local users and Groups and select Groups
06. Open properties of the ConfigMgr Remote Control Users and confirm that the Permitted viewers have been added to this group

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
Start-Sleep 60

$groupName = "ConfigMgr Remote Control Users"
$LocalGroup = [ADSI]("WinNT://./$groupName,group")
$GMembers = $LocalGroup.psbase.invoke("Members")
$gmembers | foreach { $_.GetType().InvokeMember("Name",'GetProperty',
$null, $_, $null) }
```

14.4. Starting Remote Control

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Assets and Compliance.
02. Click Devices
03. Select the Machine you want to remote access and click Start -> Remote Control
04. The Remote Control will connect to the client
05. On the client, it will need approve or Deny the connection if the client settings -> remote tools -> Prompt user for remote control permissions is set to yes
06. Once the connection is established, the remote user will be able to access the machine
07. The connection will remain active even when the user logs off. **Note:** When using Enhanced session on the virtual machine, once the users logs off, the session will be closed **Note:** Remote Control to a blocked machine has changed from SCCM 2007 where it was not possible

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$ModulePath = $env:SMS_ADMIN_UI_PATH
if ($ModulePath -eq $null) {
      $ModulePath = (Get-ItemProperty -Path
"Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session
Manager\Environment").SMS_ADMIN_UI_PATH
}
$ModulePath += "\CmRcViewer.exe"
```

```
$device = Get-CMDevice -Name "WKS0001"
if ($Device.IsClient -eq $true) { Start-Process -Filepath ("$ModulePath") -
ArgumentList ("$($device.Name) \\$($servername)") } else { "Computer is not
a SCCM Client" }
```

14.5. Monitoring Remote Access from Client

Perform this task on the wks0001 virtual machine logged on as user01
01. On the client, you can also review the following client logs: • C:\Windows\ccm\Logs\cmRcService.log: Records information for the remote control service.

This can also be achieved via PowerShell using the commands below:

```
Start-Process -Filepath ("c:\windows\cmtrace.exe") -ArgumentList
("c:\Windows\ccm\Logs\cmRcService.log")
```

14.6. Monitoring Remote Access from Server

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Reporting and Click Reports
03. Search for remote, select Remote Control – All remote control information and click Run
04. Once the report opens, all information about successfully connections will be presented
05. On the monitoring workspace, expand System Status and click Status Message Queries
06. Search for Remote
07. Select Remote Control Activity Targeted at a Specific System and click Show Messages
08. Under Remote Control Activity Targeted at a Specific System select: • Machine Name: WKS0001 • Time: Select date and time: 6 hours ago Click Ok
09. Verify the existence of Message ID 30076
10. Double click any 30076 messages to see its details. Once done, click Ok
11. Verify the existence of Message ID 30077
12. Double click any 30077 messages to see its details. Once done, click Ok

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

#Open Report
Invoke-CMReport -ReportPath "Status Messages - Audit/Remote Control - All
remote control information" -SiteCode "$SiteCode" -SrsServerName
"$servername"
```

```
#Query:
$Date = (Get-Date).AddHours(-6)
gwmi -Namespace "root\sms\site_$SiteCode" -ComputerName "$servername" -
query "select stat.*, ins.*, att1.*, stat.Time from SMS_StatusMessage as
stat left join SMS_StatMsgInsStrings as ins on stat.RecordID = ins.RecordID
left join SMS StatMsgAttributes as att1 on stat.RecordID = att1.RecordID
inner join SMS StatMsgInsStrings as ins2 on stat.RecordID = ins2.RecordID
where stat.MessageType = 768 and stat.MessageID >= 30069 and stat.MessageID
<= 30087 and ins2.InsStrIndex = 2 and ins2.InsStrValue = 'WKS0001' and
stat.Time >= '$($Date.ToString('yyyy/MM/dd HH:mm:ss.fff'))' order by
stat.Time desc"

$remoteaccesslist = gwmi -Namespace "root\sms\site_$SiteCode" -ComputerName
"$servername" -query "select stat.Time, stat.MessageID, ins.InsStrIndex,
ins.InsStrValue, att1.AttributeID, att1.AttributeTime, att1.AttributeValue
from SMS StatusMessage as stat left join SMS StatMsgInsStrings as ins on
stat.RecordID = ins.RecordID left join SMS_StatMsgAttributes as att1 on
stat.RecordID = att1.RecordID inner join SMS_StatMsgInsStrings as ins2 on
stat.RecordID = ins2.RecordID where stat.MessageType = 768 and
stat.MessageID >= 30069 and stat.MessageID <= 30087 and ins2.InsStrIndex =
2 and ins2.InsStrValue = 'WKS0001' and stat.Time >=
'$($Date.ToString('yyyy/MM/dd HH:mm:ss.fff'))' order by stat.Time desc"
foreach ($remoteaccess in $remoteaccesslist)
{
      $props = @{ 'Time'=$remoteaccess.stat.Time;
                        'MessageID'=$remoteaccess.stat.MessageID
                        'InsStrIndex'=$remoteaccess.ins.InsStrIndex
                        'InsStrValue'=$remoteaccess.ins.InsStrValue
                        'AttributeID'=$remoteaccess.att1.AttributeID
                        'AttributeTime'=$remoteaccess.att1.AttributeTime
                  }
      $obj = new-object -TypeName psobject -Property $props
      write-output $obj
}
```

15. Hardware Inventory

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001
More information	Introduction to hardware inventory in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/inventory/introduction-to-hardware-inventory How to configure hardware inventory in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/inventory/configure-hardware-inventory
Description	In this chapter, we will look at Hardware Inventory, extend collection of data, capturing data from the client as well as looking at the new data from the SCCM Server

15.1. Changing Default Client Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin

01. Start Configuration Manager Console and Click Administration.

02. Click Client Settings

03. Select the default client settings and click Properties

04. Under Default Settings, click Hardware Inventory and then, Set Classes…

05. Make the following changes:

- Under Services (Win32_Service) choose:
  - State
- Under Environment (Win32_Environment) choose:
  - Name
  - User Name
  - Caption
  - Description
  - Install Date
  - Status
  - System Variable
  - Variable Value
- Under Logical Disk (SMS_LogicalDisk) choose:
  - Free Space (MB)

Click Ok twice

06. Navigate to C:\ConfigMgr\inboxes\clifiles.src\hinv

Note: The file sms_def.mof does not exist anymore. All information is now saved into the database

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$class = gwmi -Namespace "root\sms\site_$($SiteCode)" -query 'select * from
SMS_InventoryReport where InventoryReportID = "{00000000-0000-0000-0000-
000000000001}"'
$class.get()
$reportclasses = $class.ReportClasses

$classID = "MICROSOFT|SERVICE|1.0"
for($i=0;$i -lt $reportclasses.count; $i++) {
    if ($reportclasses[$i].SMSClassID -eq $classID) {
            $reportclasses[$i].ReportProperties += "State"
        break
    }
}

$classID = "MICROSOFT|LOGICAL_DISK|1.0"
for($i=0;$i -lt $reportclasses.count; $i++) {
    if ($reportclasses[$i].SMSClassID -eq $classID) {
            $reportclasses[$i].ReportProperties += "FreeSpace"
        break
    }
}

$classID = "MICROSOFT|NETWORK_ADAPTER|1.0"
for($i=0;$i -lt $reportclasses.count; $i++) {
    if ($reportclasses[$i].SMSClassID -eq $classID) {
      $reportclasses[$i].ReportProperties += "NetworkAddresses"
      $reportclasses[$i].ReportProperties += "Speed"
        break
    }
}

$classID = "MICROSOFT|ENVIRONMENT|1.0"
$InventoryReportclass_class = [wmiclass]""
$InventoryReportclass_class.psbase.Path =
"ROOT\SMS\site $($SiteCode):SMS InventoryReportClass"
$InventoryReportclass = $InventoryReportclass_class.createInstance()

$InventoryReportclass.SMSClassid = $classID
$InventoryReportclass.ReportProperties = @("Name", "UserName", "Caption",
"Description", "InstallDate", "Status", "SystemVariable", "VariableValue")

$reportclasses += $InventoryReportclass

$class.ReportClasses = $reportclasses
$class.Put()
```

15.2. Updating Client Policies

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under Machine Policy Retrieval & Evaluation Cycle click Ok **Note:** Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
Start-Sleep 60
```

15.3. Starting Hardware Inventory

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select Hardware Inventory Cycle **Note:** Using this option will force the client to perform the hardware inventory. By default, this happen every once every 7 days and can be changed under Client Settings -> Hardware Inventory -> Hardware Inventory schedule
03. Under Hardware Inventory Cycle click Ok
04. On the client, you can also the following client logs: • C:\Windows\ccm\Logs\InventoryAgent.log: Records activities of hardware inventory, software inventory, and heartbeat discovery actions on the client.

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000001}")

Start-Sleep 60
```

15.4. Resource Explorer

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Devices
03. Right click a Windows computer and click Start -> Resource Explorer
04. Once the resource explorer open, expand hardware
05. Select Environment
06. Select Services and notice the new column
07. Select Logical Disk and notice the new column

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$ModulePath = $env:SMS_ADMIN_UI_PATH
if ($ModulePath -eq $null) {
      $ModulePath = (Get-ItemProperty -Path
"Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session
Manager\Environment").SMS_ADMIN_UI_PATH
}

$ModulePath = $ModulePath.Replace("bin\i386","bin\resourceexplorer.exe")
#windows machine
Start-Process -Filepath ("$ModulePath") -ArgumentList ('-s -
sms:ResExplrQuery="SELECT ResourceID FROM SMS_R_SYSTEM WHERE NAME =
''WKS0001''" -sms:connection=\\' + $servername + '\root\sms\site_' +
$siteCode)
```

16. Client Health

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002
More information	How to configure client status in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/deploy/configure-client-status
Description	In this chapter, we will look at the Client Health monitoring solution build in the SCCM, we will understand how it works, reporting as well as break some client for reporting

16.1. Client Status Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin

01. Start Configuration Manager Console and Click Monitoring.

02. Click Client Status and then Client Status Settings

03. On Client Status Settings Properties, change the default values to:

- Client policy requests during the following days: 1
- Heartbeat discovering during the following days: 1
- Hardware inventory during the following days: 1
- Software inventory during the following days: 1
- Status messages during the following days: 1

Note: These settings are being used in a lab environment. It is not recommended to decrease these numbers, however, sometimes you will need to increase these number to reflect individual company requirements.

Click OK

This can also be achieved via PowerShell using the commands below:

```
Set-CMClientStatusSetting -ClientPolicyDays 1 -HardwareInventoryDays 1 -
HeartbeatDiscoveryDays 1 -SoftwareInventoryDays 1 -StatusMessageDays 1
```

16.2. Client Health Configuration

Perform this task on the wks0001 virtual machine logged on as user01

01. On the client, open the Task Scheduler

02. On the task scheduler, expand Task Scheduler (Local) -> Task Scheduler Library -> Microsoft and click Configuration Manager

Note: This Task is created during the client installation

Note: If you want to control when the client is executed and control the remediation procedures, use the CCMEVALINTERVAL, CCMEVALHOUR and NOTIFYONLY Client Installation Properties
03. Select Configuration Manager Health Evaluation and click Properties
04. On the General tab confirm that the task runs as a System
05. Change to the Actions tab and confirm that it executes a program called ccmeval.exe
06. Change to the Settings tab and confirm that it runs on demand and start if a scheduled start is missed. Click Ok
07. Open C:\windows\ccm\CcmEval.xml on Notepad and examine the validations **Note:** SCCM Client executes up to 30 validations (depending on OS version). **Note:** Editing the ccmeval.xml is not supported.

This can also be achieved via PowerShell using the commands below:

```
$Task = Get-ScheduledTask -TaskName "Configuration Manager Health
Evaluation"
$task.Principal.UserId # should be System
$task.Actions.Execute # should be c:\windows\ccm\ccmeval.exe
$task.Settings.AllowDemandStart # should be true
$task.Settings.StartWhenAvailable # should be true
```

16.3. Executing CCMEVAL manually

Perform this task on the wks0001 virtual machine logged on as user01
01. On the client, open the Task Scheduler
02. On the task scheduler, expand Task Scheduler (Local) -> Task Scheduler Library -> Microsoft and click Configuration Manager
03. Select Configuration Manager Health Evaluation and click Run
04. You can also review the following logs: • C:\Windows\ccm\logs\ccmeval.log: Records Configuration Manager Client status evaluation activities and details for components that are required by the Configuration Manager client. • C:\Windows\ccm\logs\CcmEvalTask.log: Records the Configuration Manager Client status evaluation activities that are initiated by the evaluation scheduled task.

This can also be achieved via PowerShell using the commands below:

```
#Option 1
Get-ScheduledTask -TaskName "Configuration Manager Health Evaluation" |
Start-ScheduledTask

#Option 2
Start-Process -Filepath ("c:\windows\ccm\ccmeval.exe") -wait
```

16.4. Forcing CCMEVAL failure

Perform this task on the srv0001 virtual machine logged on as administrator
01. Open Active Directory User and Computers
02. Navigate to classroom.intranet->Classroom->Workstations->Enabled
03. Move WKS0002 to classroom.intranet->Classroom->Workstations->Disabled

This can also be achieved via PowerShell using the commands below:

```
Get-ADComputer WKS0002 | Move-ADObject -TargetPath
'OU=Disabled,OU=Workstations,OU=Classroom,DC=classroom,DC=intranet'
```

Perform this task on the wks0002 virtual machine logged on as administrator
01. Open command prompt
02. Type gpupdate /force and press Enter and execute it again
03. Start services console
04. Search for Background Intelligent Transfer Service, it will not exist.
05. Open Task Manager and confirm the ccmexec.exe process does not exist
06. Open the Task Scheduler
07. On the task scheduler, expand Task Scheduler (Local) -> Task Scheduler Library -> Microsoft and click Configuration Manager
08. Select Configuration Manager Health Evaluation and click Run
09. You can also review the following logs: • C:\Windows\ccm\logs\ccmeval.log: Records Configuration Manager Client status evaluation activities and details for components that are required by the Configuration Manager client. • C:\Windows\ccm\logs\CcmEvalTask.log: Records the Configuration Manager Client status evaluation activities that are initiated by the evaluation scheduled task.

This can also be achieved via PowerShell using the commands below:

```
#On WKS0002
Start-Process -Filepath ("gpupdate") -ArgumentList ("/force") -wait
Start-sleep 10

Start-Process -Filepath ("gpupdate") -ArgumentList ("/force") -wait
Start-sleep 10

#get service information
Get-Service -Name BITS

#Option 1
Get-ScheduledTask -TaskName "Configuration Manager Health Evaluation" |
Start-ScheduledTask

#Option 2
Start-Process -Filepath ("c:\windows\ccm\ccmeval.exe") -wait
```

16.5. Monitoring Client Health

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Client Status and click Client Check. Click **client check passed**. **Note:** All links are clickable as well as the graphic
03. Under the Client that passed client check from All Desktop and Server clients, select a WKS0001 and click Client Check details.
04. Confirm that all rules passed evaluation and/or remediation (green button) **Note:** if there is any remediation to be taken and passed, the rule name will be displayed.
05. Click Monitoring
06. Expand Client Status and click Client Check. Click **client check failed**.
07. Under the Client that failed client check from All Desktop and Server clients, select a WKS0002 and click Client Check details.
05. A Remediation Failed result is shown for the Verify BITS service exist with error Access Denied and another for Verfity/Remediate BITS Startup type with error Dependency Failed.

This can also be achieved via PowerShell using the commands below:

```
#On SRV0002
$SiteCode = "001"

$Device = Get-CMDevice -Name "WKS0001"
gwmi -namespace "root\sms\site_$SiteCode" -query "select * from
SMS CH EvalResult where ResourceID = $($Device.ResourceID)" | select
HealthCheckDescription

$Device = Get-CMDevice -Name "WKS0002"
gwmi -namespace "root\sms\site $SiteCode" -query "select * from
SMS_CH_EvalResult where ResourceID = $($Device.ResourceID)" | select
HealthCheckDescription
```

Perform this task on the srv0001 virtual machine logged on as administrator
01. Open Active Directory User and Computers
02. Navigate to classroom.intranet->Classroom->Workstations->Disabled
03. Move WKS0002 to classroom.intranet->Classroom->Workstations-> Enabled

This can also be achieved via PowerShell using the commands below:

```
#On SRV0001
Get-ADComputer "WKS0002" | Move-ADObject -TargetPath
'OU=Enabled,OU=Workstations,OU=Classroom,DC=classroom,DC=intranet'
```

Perform this task on the wks0002 virtual machine logged on as administrator
01. Open command prompt
02. Type gpupdate /force and press Enter and execute it again
03. Start services console
04. Search for Background Intelligent Transfer Service. The service now exists and the Startup type is set to Manual and the Status is not running.
05. On the client, open the Task Scheduler
06. On the task scheduler, expand Task Scheduler (Local) -> Task Scheduler Library -> Microsoft and click Configuration Manager
07. Select Configuration Manager Health Evaluation and click Run

This can also be achieved via PowerShell using the commands below:

```
#On WKS0002
Start-Process -Filepath ("gpupdate") -ArgumentList ("/force") -wait
Start-sleep 10

Start-Process -Filepath ("gpupdate") -ArgumentList ("/force") -wait
Start-sleep 10

#get service information
Get-Service -Name BITS

#Option 1
Get-ScheduledTask -TaskName "Configuration Manager Health Evaluation" |
Start-ScheduledTask

#Option 2
Start-Process -Filepath ("c:\windows\ccm\ccmeval.exe") -wait
```

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Client Status and click Client Check. Click **client check passed**. **Note:** All links are clickable as well as the graphic
03. Under the Client that passed client check from All Desktop and Server clients, select a WKS0002 and click Client Check details.
04. Confirm that all rules passed evaluation and/or remediation (green button) **Note:** if there is any remediation to be taken and passed, the rule name will be displayed.

This can also be achieved via PowerShell using the commands below:

```
#On SRV0002
$SiteCode = "001"

$Device = Get-CMDevice -Name "WKS0002"
gwmi -namespace "root\sms\site_$SiteCode" -query "select * from
SMS_CH_EvalResult where ResourceID = $($Device.ResourceID)" | select
HealthCheckDescription
```

17. Software Metering

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002
More information	Software metering in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/monitor-app-usage-with-software-metering
Description	In this chapter, we will look at how to configure SCCM to collect information about software when they run as well as we will look how we can report based on collected data

17.1. Changing Default Client Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Client Settings
03. Select the default client settings and click Properties
04. Under Default Settings, click Software Metering and change the schedule from 7 days to 1 day

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$ClientSettingsName = "Default Client Agent Settings"
$schedule = New-CMSchedule -RecurCount 1 -RecurInterval Days
Set-CMClientSetting -Name "$ClientSettingsName" -SoftwareMetering -Enable
$True -Schedule $schedule
```

17.2. Updating Default Software Metering Settings and Clearing existing rules

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Software Metering
03. Click Software Metering Properties
04. Under Software Metering Properties, uncheck the Automatically create disabled metering rules from recent usage inventory data and click Ok
05. Select all already create software metering rules and click delete.
06. On the question Are you sure you want to delete these YY items?, click Yes

This can also be achieved via PowerShell using the commands below:

```
#Disable auto-create sw metering rules
Set-CMSoftwareMeteringSetting -AutoCreateDisabledRule $False

#delete all already create sw metering rules
Get-CMSoftwareMeteringRule | Remove-CMSoftwareMeteringRule -Force
```

17.3. Creating Rule

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Software Metering
03. Click Create Software Metering Rule
04. Under Create Software Metering Rule Wizard, fill up with the following: • Name: Notepad • File Name: notepad.exe • Original File Name: NOTEPAD.EXE.MUI • Version: * • Language: - Any – Click Next **Note:** In version, we used * instead of the version to capture any version **Note:** In Language, we used – Any – to capture any language **Note:** You can use the Browse button to have all settings filled up automatically **Note:** All information can be view when you right click the application and select properties, and then go to the details tab
05. Under Summary, click Next
06. Under Completion, click Close

This can also be achieved via PowerShell using the commands below:

```
New-CMSoftwareMeteringRule -ProductName Notepad -FileName notepad.exe -
FileVersion * -OriginalFileName NOTEPAD.EXE.MUI -LanguageId 65535
```

17.4. Starting Validation Software Metering

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under Machine Policy Retrieval & Evaluation Cycle click Ok

Note: Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take a few minutes to complete
04. Open notepad.exe and leave it open for couple of minutes
05. Open the Configuration Manager Properties
06. Change to the Actions Tab, select Software Metering Usage Report Cycle **Note:** Using this option will force the client to send report usage data to the server. By default, this happen every 7 days and can be changed under Client Settings -> Software Metering -> Schedule
07. Under Software Metering Usage Report Cycle click Ok

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
Start-Sleep 60

for($i=1; $i -le 3; $i++){
      Start-Process -Filepath ("notepad.exe")
      start-sleep 60
      Stop-Process -Name notepad
      start-sleep 10
}

$SMSCli = [wmiclass] "root\ccm:SMS Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000031}")

Start-Sleep 60
```

17.5. Summarization Software Metering Data Manually

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Command Prompt as administrator
02. type **"C:\Program Files (x86)\ConfigMgr 2012 Toolkit R2\ServerTools\runmetersumm.exe" CM_001** enter **Note:** This task is managed by the Site Maintenance Tasks Summarize Software Metering File Usage Data and Summarize Software Metering Monthly Usage Data that, by default, runs every day between 00:00 and 05:00

This can also be achieved via PowerShell using the commands below:

```
& "c:\Program Files (x86)\ConfigMgr 2012 Toolkit
R2\ServerTools\runmetersumm.exe" "CM_001"
```

17.6. Monitoring Software Metering Data via Reports

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Under monitoring, expand Reporting and click Reports
03. Search for Software Metering. Select Users that have run a specific metered software program and click Run
04. Once the report is open, fill up the filters and click view report
05. Select Time of day usage summary for a specific metered software program and click run
06. Once the report is open, fill up the filters and click view report

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

#Open Report
Invoke-CMReport -ReportPath "Software Metering/Users that have run a
specific metered software program" -SiteCode "$SiteCode" -SrsServerName
"$servername"
```

17.7. Monitoring Software Metering Data via Collections

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Device Collections
03. Select Device Collections and click Create Device Collection
04. Under Specify details for this collection type Computers that Run Notepad.exe Last 30 days under name and select All Systems under Limiting Collection. Click Next
05. Under Define membership rules for this collection, select Add query and click Query Rule
06. Under Query Rule Properties, type Software Metering Rule Notepad and click Edit Query Statement
07. Under Query Statement Properties, click Show Query Language and type select * from SMS_R_System inner join SMS_MonthlyUsageSummary on SMS_MonthlyUsageSummary.ResourceID = SMS_R_System.ResourceID inner join SMS_MeteredFiles on SMS_MeteredFiles.FileID = SMS_MonthlyUsageSummary.FileID and SMS_MeteredFiles.SecurityKey = "00100037" where DateDiff(dd, SMS_MonthlyUsageSummary.LastUsage, GetDate()) < 30 Click Ok

Note: Change the 00100037 by the Rule ID you want to use
08. Under Create Device Collection Wizard, select Use incremental updates for this collection and click Next **Note:** It is not recommended to have over 250 collections with the Incremental updates option enabled because it might cause evaluation delays when enabled it for many collections.
09. Under confirm the settings, click Next
10. Under The Create Device Collection Wizard completed successfully, click Close
11. Under device collections, the new Collection is still under update status. **Note:** Once the collection is created, there is a process to populate it and it may take a while. In this lab, wait 30 seconds or refresh it couple of times until you see Member Count change to 1
16. Select the collection and click Show Members
17. The collection will be expanded under Devices and all devices that match the query filter will be displayed.

This can also be achieved via PowerShell using the commands below:

```
$swrule = Get-CMSoftwareMeteringRule -ProductName Notepad
$CollUpdate = New-CMSchedule -Start "01/01/2015 9:00 PM" -DayOfWeek
Saturday -RecurCount 1
$NewCol = New-CMDeviceCollection -Name "Computers that Run Notepad.exe Last
30 days" -LimitingCollectionName "All Systems" -RefreshSchedule $CollUpdate
-RefreshType Both
Add-CMDeviceCollectionQueryMembershipRule -CollectionId
$NewCol.CollectionID -RuleName "Software Metering Rule Notepad"  -
QueryExpression "select * from SMS_R_System inner join
SMS_MonthlyUsageSummary on SMS_MonthlyUsageSummary.ResourceID =
SMS_R_System.ResourceID inner join SMS_MeteredFiles on
SMS MeteredFiles.FileID = SMS MonthlyUsageSummary.FileID and
SMS_MeteredFiles.SecurityKey = '$($swrule.SecurityKey)' where DateDiff(dd,
SMS_MonthlyUsageSummary.LastUsage, GetDate()) < 30"
start-sleep 30
Invoke-CMDeviceCollectionUpdate -Name "$($NewCol.Name)"
start-sleep 30
```

18. Site Roles for User Centric Management

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	Plan for and configure application management in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/plan-design/plan-for-and-configure-application-management Introduction to Application Management in Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/understand/introduction-to-application-management Configure Software Center and the Application Catalog (Windows PCs only) https://docs.microsoft.com/en-us/sccm/apps/plan-design/plan-for-and-configure-application-management#configure-software-center-and-the-application-catalog-windows-pcs-only
Description	In this chapter, we will look at the infrastructure required to perform deployment to Users

18.1. Application Catalog WebService Role

18.1.1. Creating Firewall Rules

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Windows Firewall with Advanced Security and click Inbound Rules
02. Click New Rule
03. On New Inbound Rule Wizard, select Port and click Next
04. On Protocol and Ports select TCP and type 80 under specify local ports and click Next
05. On Action, click Next
06. On Profile, click Next
07. On Name, type IIS Application Catalog WebService Point (TCP 80) Inbound and click Finish
08. Click New Rule
09. On New Inbound Rule Wizard, select Port and click Next
10. On Protocol and Ports select TCP and type 443 under specify local ports and click Next
11. On Action, click Next
12. On Profile, click Next
13. On Name, type IIS Application Catalog WebService Point (TCP 443) Inbound and click Finish

This can also be achieved via PowerShell using the commands below:

```
New-NetFirewallRule -DisplayName "IIS Application Catalog WebService (TCP 80) Inbound" -Action Allow -Direction Inbound -LocalPort 80 -Protocol TCP
New-NetFirewallRule -DisplayName "IIS Application Catalog WebService (TCP 443) Inbound" -Action Allow -Direction Inbound -LocalPort 443 -Protocol TCP
```

18.1.2. Install Requirements

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next
06. Under select server roles, select Web Server (IIS)
07. On Add Roles and Features Wizard, click Add Features and click Next
08. Under Select Features, select .NET Framework 3.5 Features -> HTTP Activation
09. On Add Roles and Features Wizard, click Add Features
10. Under Select Features, select .NET Framework 3.5 Features -> non-HTTP Activation and click Next
11. Under Select Features, select .NET Framework 4.6 Features -> WCF Services -> HTTP Activation
12. On Add Roles and Features Wizard, click Add Features
13. Under Web Server Role (IIS), click Next
14. On Select role service, select Management Tools->IIS 6 Management Compatibility->IIS 6 Metabase Compatibility. Click Next
15. Under Confirm installation selections, click Install
16. Once the installation is succeeded. Click Close

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name Web-Server | Install-WindowsFeature
Get-WindowsFeature -Name Web-Metabase | Install-WindowsFeature
Get-WindowsFeature -Name Web-ASP-Net | Install-WindowsFeature
Get-WindowsFeature -Name NET-WCF-HTTP-Activation45 | Install-WindowsFeature
Get-WindowsFeature -Name NET-HTTP-Activation | Install-WindowsFeature
Get-WindowsFeature -Name NET-Non-HTTP-Activ | Install-WindowsFeature
```

18.1.3. Installing Site System Role

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Site System Roles
03. Right click \\SRV0002.classroom.intranet and click Add Site System Roles
04. On Add Site System Roles Wizard, General, click Next
05. Under proxy, click Next
06. Under Specify roles for this server, select Application Catalog web service point and click Next
07. Under Specify setting for the Application Catalog web service Point click Next
08. Under Specify Settings to configure IIS for this Application Catalog WebService point click Next
09. Under confirm the settings, click Next

10. Under You have successfully completed the Add Site System Roles wizard with the following settings click close
11. Click Monitoring
12. Expand System Status and click Component Status
13. Search for SMS_AWEBSVC_CONTROL_MANAGER
14. Right Click SMS_AWEBSVC_CONTROL_MANAGER, Show Messages and click All
15. Under Status Messages: Set Viewing Period, click OK
16. Verify the existence of Message ID 1013, 1014, 1015 and 8102
17. Double click the message id 8102 message to see its details. Once done, click Ok
18. Right Click SMS_PORTALWEB_CONTROL_MANAGER, Show Messages and click All
19. Under Status Messages: Set Viewing Period, click OK
20. Verify the existence of Message ID 1013, 1014, 1015 and 8002
21. Double click the message id 8102 message to see its details. Once done, click Ok
22. Open Internet Explorer and navigate to http://SRV0002.classroom.intranet/CMApplicationCatalogSvc/ApplicationOfferService.svc
23. You can also review the following logs: • C:\ConfigMgr\Logs\SMSAWEBSVCSetup.log: Records the installation activities of the Application Catalog web service.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

if ((Get-CMSiteSystemServer -SiteSystemServerName "$servername") -eq $null)
{ New-CMSiteSystemServer -SiteCode $SiteCode -UseSiteServerAccount -
ServerName $servername }
Add-CMApplicationCatalogWebServicePoint -SiteSystemServerName "$ServerName"
-CommunicationType HTTP -PortNumber 80 -SiteCode $SiteCode

start-sleep 90

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS_AWEBSVC_CONTROL_MANAGER' and stmsg.MessageID = 1013 and stmsg.SiteCode
= '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_AWEBSVC_CONTROL_MANAGER 1013 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_AWEBSVC_CONTROL_MANAGER' and stmsg.MessageID = 1014 and stmsg.SiteCode
= '$SiteCode'"
```

```
    if ($component -ne $null)
    {
        Write-Host "Found SMS_AWEBSVC_CONTROL_MANAGER 1014 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS_AWEBSVC_CONTROL_MANAGER' and stmsg.MessageID = 1015 and stmsg.SiteCode
= '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS AWEBSVC CONTROL MANAGER 1015 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS_AWEBSVC_CONTROL_MANAGER' and stmsg.MessageID = 500 and stmsg.SiteCode
= '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS AWEBSVC CONTROL MANAGER 500 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS AWEBSVC CONTROL MANAGER' and stmsg.MessageID = 8102 and stmsg.SiteCode
= '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS AWEBSVC CONTROL MANAGER 8102 id's"
        break
    } else { Start-Sleep 10 }
}

$web = New-Object -ComObject msxml2.xmlhttp
$url =
"http://$($servername):80/CMApplicationCatalogSvc/ApplicationOfferService.s
vc"
while ($true)
{
    try
    {
        $web.open('GET', $url, $false)
            $web.send()
```

```
            if ($web.status -eq "404") { start-sleep 10 }
            if ($web.status -eq "200")
            {
                        Write-Host "Found CMApplicationCatalogSvc site"
                  break
            }
    } catch {
      #
    }
}
```

18.2. Application Catalog WebSite Role

18.2.1. Creating Firewall Rules

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Windows Firewall with Advanced Security and click Inbound Rules
02. Click New Rule
03. On New Inbound Rule Wizard, select Port and click Next
04. On Protocol and Ports select TCP and type 80 under specify local ports and click Next
05. On Action, click Next
06. On Profile, click Next
07. On Name, type IIS Application Catalog WebSite Point (TCP 80) Inbound and click Finish
08. Click New Rule
09. On New Inbound Rule Wizard, select Port and click Next
10. On Protocol and Ports select TCP and type 443 under specify local ports and click Next
11. On Action, click Next
12. On Profile, click Next
13. On Name, type IIS Application Catalog WebSite Point (TCP 443) Inbound and click Finish

This can also be achieved via PowerShell using the commands below:

```
New-NetFirewallRule -DisplayName "IIS Application Catalog WebSite (TCP 80)
Inbound" -Action Allow -Direction Inbound -LocalPort 80 -Protocol TCP
New-NetFirewallRule -DisplayName "IIS Application Catalog WebSite (TCP 443)
Inbound" -Action Allow -Direction Inbound -LocalPort 443 -Protocol TCP
```

18.2.2. Install Requirements

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next

06. Under select server roles, select Web Server (IIS)
07. On Add Roles and Features Wizard, click Add Features and click Next
08. Under Select Features, select .NET Framework 4.6 Features -> ASP.NET 4.6 and click Next
12. Under Web Server Role (IIS), click Next
10. On Select role service, select Security->Windows Authentication, Application Deployment->ISAP Extensions, Management Tools->IIS 6 Management Compatibility->IIS 6 Metabase Compatibility, Web Server -> Application Development -> ASP.NET 4.6
11. On Add Roles and Features Wizard, click Add Features and click Next
12. Under Confirm installation selections, click Install
13. Once the installation is succeeded. Click Close

This can also be achieved via PowerShell using the commands below:

```
Get-WindowsFeature -Name Web-Server | Install-WindowsFeature
Get-WindowsFeature -Name Web-Metabase | Install-WindowsFeature
Get-WindowsFeature -Name Web-Windows-Auth | Install-WindowsFeature
Get-WindowsFeature -Name Web-ASP-Net | Install-WindowsFeature
```

18.2.3. Installing Site System Role

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Site System Roles
03. Right click \\SRV0002.classroom.intranet and click Add Site System Roles
04. On Add Site System Roles Wizard, General, click Next
05. Under proxy, click Next
06. Under Specify roles for this server, select Application Catalog website Point and click Next
07. Under Specify Settings to configure IIS for this Application Catalog website point, select SRV0002.classroom.intranet under Site System Server and click Next
08. Under Specify Customized Settings for this Application Catalog website point type Training Lab under Organization name and click Next
09. Under confirm the settings, click Next
10. Under You have successfully completed the Add Site System Roles wizard with the following settings click close
11. Click Monitoring
12. Expand System Status and click Component Status
13. Search for SMS_PORTALWEB_CONTROL_MANAGER
14. Right Click SMS_PORTALWEB_CONTROL_MANAGER, Show Messages and click All
15. Under Status Messages: Set Viewing Period, click OK
16. Verify the existence of Message ID 1013, 1014, 1015 and 8002 **Note:** This message can take some time to be generated.
17. Double click the message id 8002 message to see its details. Once done, click Ok
18. Open Internet Explorer and navigate to http://SRV0002.classroom.intranet/CMApplicationCatalog

Note: When connecting from a machine without Silverlight or a supported Browser, a warning message will be displayed

Note: If a Windows Security dialog box appears asking User name and password, you should add the website to the Intranet List on the Internet Explorer so users do not need to type their username/password when accessing the site

19. You can also review the following logs:

- C:\ConfigMgr\Logs\SMSPORTALWEBSetup.Log: Records the installation activities of the Application Catalog website.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "$SiteCode"
$servername = "SRV0002.classroom.intranet"

$servernameNetbios = "SRV0002"
if ((Get-CMSiteSystemServer -SiteSystemServerName "$servername") -eq $null)
{ New-CMSiteSystemServer -SiteCode $SiteCode -UseSiteServerAccount -
ServerName $servername }
Add-CMApplicationCatalogWebsitePoint -ApplicationWebServicePointServerName
"$servername" -CommunicationType HTTP -SiteSystemServerName "$servername" -
ClientConnectionType Intranet -NetBiosName $servernameNetbios -
OrganizationName "Training Lab" -Port 80 -SiteCode $SiteCode

start-sleep 90

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_PORTALWEB_CONTROL_MANAGER' and stmsg.MessageID = 1013 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_PORTALWEB_CONTROL_MANAGER 1013 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_PORTALWEB_CONTROL_MANAGER' and stmsg.MessageID = 1014 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_PORTALWEB_CONTROL_MANAGER 1014 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
```

```
    $component = gwmi -Namespace ("root\sms\site $SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS_PORTALWEB_CONTROL_MANAGER' and stmsg.MessageID = 1015 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS PORTALWEB CONTROL MANAGER 1015 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS PORTALWEB CONTROL MANAGER' and stmsg.MessageID = 500 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS PORTALWEB CONTROL MANAGER 500 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS PORTALWEB CONTROL MANAGER' and stmsg.MessageID = 8002 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_PORTALWEB_CONTROL_MANAGER 8002 id's"
        break
    } else { Start-Sleep 10 }
}

$web = New-Object -ComObject msxml2.xmlhttp
$url = "http://$($servername):80/CMApplicationCatalog"
while ($true)
{
    try
    {
        $web.open('GET', $url, $false, "classroom\sccmadmin", 'Pa$$w0rd')
            $web.send()

            if ($web.status -eq "404") { start-sleep 10 }
            if ($web.status -eq "200")
            {
                        Write-Host "Found CMApplicationCatalog site"
                break
            }
    }
    catch {
      #
    } }
```

18.3. Enable the new SCCM Software Center

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Client Settings
03. Select the default client settings and click Properties **Note:** Changing the Default Client Settings should be avoided as it applies to all clients and there is no way to revert it back. My recommendation is to create new custom client settings with the changes you want applied and deploy those custom settings to a collection with all clients if needed. Of course, there are exception to this rule, like, but not limited to, setting the Company Name, Software Update information.
04. Under Default Settings, click Computer Agent and change Use new Software Center to Yes. Click Ok **Note:** Use new Software Center should already be set to Yes, this is the default setting for SCCM 1606. When performing upgrade from early version, this option should be configured

This can also be achieved via PowerShell using the commands below:

```
$ClientSettingsName = "Default Client Agent Settings"
Set-CMClientSetting -Name "$ClientSettingsName" -ComputerAgent -
UseNewSoftwareCenter $true
```

18.4. Starting validation of the new Software Center

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties **Note:** This option does not need to be configured on a new installation of SCCM 1606 or newer, only when updating from early version
02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under Machine Policy Retrieval & Evaluation Cycle click Ok **Note:** Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Open Windows Explorer and navigate to C:\Windows\CCM. A ClientUX folder should exist
05. Open Software Center under Start -> Microsoft System Center -> Configuration Manager

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
Start-Sleep 60

Start-Process -FilePath  "$($env:Programdata)\Microsoft\Windows\Start
Menu\Programs\Microsoft System Center\Configuration Manager\Software
Center.lnk"
```

19. Primary Users

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0002
More information	Associate users with a destination computer in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/osd/get-started/associate-users-with-a-destination-computer Link users and devices with user device affinity in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/link-users-and-devices-with-user-device-affinity
Description	In this chapter, we will assign a primary user to a device. This can be useful when you want deploy applications only when users connect to their primary machines

19.1. Changing Default Client Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Client Settings
03. Select the default client settings and click Properties
04. Under Default Settings, click Computer Agent. Select Yes under the following: • Add default Application Catalog website to Internet Explorer trusted sites zone • Allow Silverlight applications to run in elevated trust mode **Note:** I personally do not like using the Add Default Application Catalog website to Internet Explorer trusted site zone as this can be easily be added to the Intranet List using GPO. Use this option where GPO is not an option. For more information about using GPO to set the Intranet List, refer to http://www.grouppolicy.biz/2010/03/how-to-use-group-policy-to-configure-internet-explorer-security-zone-sites/
05. Click User and Device Affinity. Select Yes under the following: • Automatically configure user device affinity from usage data • Allow user to define their primary devices Click Ok

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$ClientSettingsName = "Default Client Agent Settings"
```

```
Set-CMClientSetting -ComputerAgentSettings -Name "$ClientSettingsName" -
AddPortalToTrustedSiteList $True -AllowPortalToHaveElevatedTrust $True -
PortalUrl "http://srv0002/CMApplicationCatalog"
Set-CMClientSetting -Name "$ClientSettingsName" -UserDeviceAffinitySettings
-AllowUserAffinity $True -AutoApproveAffinity $True
```

19.2. Manual association by the Administrator

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Devices
03. Select WKS0001 and click Edit Primary Users
04. Search for User01 and click Add. Once done, click Ok
05. Once back to the console, select the WKS0001 machine and click Primary Users under Related Objects
06. A new node will appear under Users called Primary Users for the WKS0001

This can also be achieved via PowerShell using the commands below:

```
Get-CMUserDeviceAffinity -DeviceName "WKS0001"
Add-CMUserAffinityToDevice -DeviceName "WKS0001" -UserName
"CLASSROOM\User01"
Get-CMUserDeviceAffinity -DeviceName "WKS0001"
```

19.3. Manual association by the User

Perform this task on the wks0002 virtual machine logged on as user02
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under Machine Policy Retrieval & Evaluation Cycle click Ok **Note:** Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Open Internet Explorer and navigate to http://SRV0002/CMApplicationCatalog
05. Click My Devices and select “I regularly use this computer to do my work”.

This can also be achieved via PowerShell using the commands below:

```
#On the Client
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
```

```
Start-Sleep 60

$IE=new-object -com internetexplorer.application
$IE.Visible = $true
$IE.navigate("http://SRV0002.classroom.intranet/CMApplicationCatalog")
#once the IE is open Click My Devices and select "I regularly use this
computer to do my work".
```

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Users
03. Search for User02. Once done, select the CLASSROOM\User02 user and click Primary Device under Related Objects
04. A new node will appear under Devices called Primary Devices for the CLASSROOM\User02

This can also be achieved via PowerShell using the commands below:

```
#On the Server
Get-CMUserDeviceAffinity -UserName "CLASSROOM\User02"
```

20. Application Management - Basic

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	Introduction to application management in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/understand/introduction-to-application-management Create applications with System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/create-applications
Description	In this chapter, we will look at all steps required to create an application

20.1. Creating Application

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Click Create Application
04. Under Specify settings for this application, use the following: • Type: Windows Installer (*.msi file) • Location: \\srv0001\TrainingFiles\Source\Chrome for Windows\googlechromestandaloneenterprise64.msi Click Next
05. Under Import Information, click Next
06. Under General Information confirm that the Installation program has been populated with a silent command line and click Next
07. Under Summary, click Next
08. Under The Create Application Wizard completed successfully click Close

This can also be achieved via PowerShell using the commands below:

```
$AppName = "Google Chrome"
New-CMApplication -Name "$AppName"

Add-CMMsiDeploymentType -ApplicationName "$AppName" -ContentLocation
"\\srv0001\TrainingFiles\Source\Chrome for
Windows\googlechromestandaloneenterprise64.msi" -DeploymentTypeName "Google
Chrome - Windows Installer (*.msi file)"
```

20.2. Adding Requirements to Applications

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Applications Management and click Applications
03. Select the Google Chrome application and click Deployment Types tab at the bottom of the console
04. Select the Deployment Type and click Properties
05. Under Google Chrome - Windows Installer (*.msi file) Properties, General tab, change to Requirements tab
06. Under the requirement tab, click Add
07. Under Create Requirement, select the following information: • Category: Device • Condition: Operating System • Rule Type: Value • Operator: One of • Values: All Windows 10 (64-bit) Click Ok twice

This can also be achieved via PowerShell using the commands below:

```
. c:\TrainingFiles\Scripts\Add-CMDeploymentTypeGlobalCondition.ps1

$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"
$AppName = "Google Chrome"
$DTName = "Google Chrome - Windows Installer (*.msi file)"

Add-CMDeploymentTypeGlobalCondition -ApplicationName "$AppName" -
DeploymentTypeName "$DTName" -sdkserver "$dkserver" -sitecode "$sitecode" -
GlobalCondition OperatingSystem -Operator OneOf -Value
"Windows/All_x64_Windows_10_and_higher_Clients"
```

20.3. Return Codes

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Applications Management and click Applications
03. Select the Google Chrome application and click Deployment Types tab at the bottom of the console
04. Select the Deployment Type and click Properties
05. Under **Google Chrome - Windows Installer (*.msi file)** Properties, General tab, change to Return Codes tab
06. Under return codes, note that the list is already populated with the default return codes for the most common situations. Click Ok

Note: If the MSI app fails with an exit code in {4, 5, 8, 13, 14, 39, 51, 53, 54, 55, 59, 64, 65, 67, 70, 71, 85, 86, 87, 112, 128, 170, 267, 999, 1003, 1203, 1219, 1220, 1222, 1231, 1232, 1238, 1265, 1311, 1323, 1326, 1330, 1618, 1622, 2250}, a retry will happen every 2 hours for up to 10 times.

This can also be achieved via PowerShell using the commands below:

```
$AppName = "Google Chrome"
$DTName = "Google Chrome - Windows Installer (*.msi file)"
([xml](Get-CMDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName").SDMPackageXML).AppMgmtDigest.deploymenttype.installer.customdata
.exitcodes.exitcode
```

20.4. Distributing Application Content to Distribution Point Group

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management, Applications
03. Select Google Chrome and click Distribute Content
04. Under Review selected content click Next
05. Under Review the content to distribute, click Next
06. Under Specify the content destination, click Add Distribution Point Group
07. Under Add-Distribution Point Groups, select Training Lab and click Ok. Once back to the Specify the content destination, click Next
08. Under Confirm the settings, click Next
09. Under The distribute content wizard completed successfully, click Close

This can also be achieved via PowerShell using the commands below:

```
Start-CMContentDistribution -ApplicationName "Google Chrome" -
DistributionPointGroupName "Training Lab"
```

20.5. Monitoring Application Content Distribution via Console

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Distribution Status and click Content Status
03. Select Google Chrome and verify the Completion Statistics. You can also view a break down when clicking View Status
04. You can also review the following logs: • C:\ConfigMgr\Logs\DistMgr.log: Records details about package creation, compression, delta replication, and information updates.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"
$AppName = "Google Chrome"

gwmi -Namespace root\sms\site_$SiteCode -ComputerName $servername -Query
"SELECT * FROM SMS ObjectContentExtraInfo" | Where-Object {$ .SoftwareName
-eq $AppName } |select Targeted, NumberErrors, NumberInProgress,
NumberSuccess, NumberUnknown
```

20.6. Monitoring Application Content Distribution via Reports

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Reporting, Reports and click Software Distribution – Content
03. Select Application content distribution status and click Run
04. Once the report opens, click Values, select Google Chrome application, and click View Report

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

#Open Report
Invoke-CMReport -ReportPath "Software Distribution - Content/Application
content distribution status" -SiteCode "$SiteCode" -SrsServerName
"$servername"
```

21. Deploying and monitoring Applications

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002 WKS0004
More information	Deploy applications with System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/deploy-applications Monitor applications from the System Center Configuration Manager console https://docs.microsoft.com/en-us/sccm/apps/deploy-use/monitor-applications-from-the-console
All Systems & All Users collection	Deploying anything to All Systems or All Users is not recommended and it goes against best practices. We are using these 2 collections for demonstration purposes only.
Description	In this chapter, we will look at all steps required to deploy an existing application as well as monitor its deployment status

21.1. Deploying Application

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Select Google Chrome and click Deploy
04. Under Specify general information for this deployment, click Browse (Collection) and select the Collection you want to deploy. Click Next **Note:** You can deploy to Devices as well as Users. In this example, we are using a Device Collection - All Systems
05. Under Specify the content destination, click Next
06. Under specify settings to control how this software is deployed, click Next **Note:** Action can be Install or Uninstall and Purpose can be Available or Required.
07. Under Specify the schedule for this deployment, click Next
08. Under Specify the user experience for the installation of this software on the selected devices, click Next
09. Under specify Configuration Manager and Operations Manager alert options, click Next
10. Under Confirm the settings for this new deployment click Next
11. Under Completion, click Close

This can also be achieved via PowerShell using the commands below:

```
$AppName = "Google Chrome"
$ColName = "All Systems"
Start-CMApplicationDeployment -CollectionName "$ColName" -Name "$AppName" -
DeployAction Install -DeployPurpose Available
```

21.2. Installing available application

Perform this task on the wks0001 virtual machine logged on as user01
01. Open Control Panel and then Configuration Manager.
02. Change to the actions tab. Select Machine Policy Retrieval & Evaluation cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Once the Machine Policy Retrieval & Evaluation cycle message appears, click Ok. **Note:** Depending on the SCCM environment, the machine policy retrieval & evaluation cycle can take few minutes.
04. Click Start, Microsoft System Center, Configuration Manager and click Software Center
05. Under Available Software, select Google Chrome and click Install
06. If needed, click Installation Status tab to follow the installation process
07. Once the installation is finished, click Installed Software to see all installed software **Note:** Repeat the process on the WKS0002. **Note:** You will not be able to install the application on WKS0004 as it does not meet the requirements

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
Start-Sleep 60

$AppName = "Google Chrome"
$app = gwmi -Namespace 'root\CCM\ClientSDK' -Class 'CCM_Application' |
Where-Object { ($_.Name -eq "$($AppName)") -and ($_.InstallState -eq
"NotInstalled") -and ($ .AllowedActions -contains "Install")}

[int]$code = Invoke-WmiMethod -Namespace 'root\CCM\ClientSDK' -Class
'CCM_Application' -Name Install -ArgumentList @(0, $app.Id,
$app.IsMachineTarget, $false, 'High', $app.Revision) | select -
ExpandProperty ReturnValue

if ($code -ne 0) {
      throw "Error invoking Installation of '$($app.Name)' ($code)."
} else {
```

```
        "Successfully invoked Installation of '$($app.Name)'."
}
```

21.3. Monitoring Application Deployment via Console

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Click Deployments
03. Select Google Chrome and click Summarization **Note:** The Application Deployment summarization runs once every 60 minutes by default, this can be changed on Administration -> Site Configuration -> Sites -> <Site> -> Status Summarizers -> Application Deployment Summarizer
04. Click Ok once the Configuration Manager information screen appears
05. After the summarization, under Completion Statistics, view Status. Click View Status for more information
06. Under View Status, Asset Details, you can see which device received the software and Under Requirements Not Met you can see which device did not have the software installed.

This can also be achieved via PowerShell using the commands below:

```
$AppName = "Google Chrome"
$ColName = "All Systems"

Get-CMDeployment -CollectionName "$ColName" -SoftwareName "$AppName" |
Invoke-CMDeploymentSummarization
Start-Sleep 10
Get-CMDeployment -CollectionName "$ColName" -SoftwareName "$AppName" |
select ApplicationName, CollectionName, NumberErrors, NumberInProgress,
NumberOther, NumberSuccess, NumberTargeted, NumberUnknown
```

21.4. Monitoring Application Deployment via Client Logs

Perform this task on the wks0001 virtual machine logged on as user01
01. On the client, you can also review the following client logs: • C:\Windows\ccm\Logs\AppDiscovery.log: Records details about the discovery or detection of applications on client computers. • C:\Windows\ccm\Logs\AppEnforce.log: Records details about enforcement actions (install and uninstall) taken for applications on the client. • C:\Windows\ccm\Logs\ContentTransferManager.log: Schedules the Background Intelligent Transfer Service (BITS) or the Server Message Block (SMB) to download or to access packages. • C:\Windows\ccm\Logs\DataTransferService.Log: Records all BITS communication for policy or package access. • C:\Windows\ccm\Logs\SCCClient_<Domain>@<User>_1.log: Records the activity in Software Center for the specified user on the client computer.

- C:\Windows\ccm\Logs\SCCClient_<Domain>@<User>_2.log: Records the historical activity in Software Center for the specified user on the client computer.
- C:\Windows\ccm\Logs\SCNotify_<Domain>@<User>_1.log: Records the activity for notifying users about software for the specified user.
- C:\Windows\ccm\Logs\SCNotify_<Domain>@<User>_2.log: Records the historical information for notifying users about software for the specified user.

This can also be achieved via PowerShell using the commands below:

```
Start-Process -Filepath ("c:\windows\cmtrace.exe") -ArgumentList
("c:\Windows\ccm\Logs\AppDiscovery.log c:\Windows\ccm\Logs\AppEnforce.log
c:\Windows\ccm\Logs\ContentTransferManager.log
c:\Windows\ccm\Logs\DataTransferService.Log")
```

21.5. Monitoring Application Deployment via Reports

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Reporting, Reports and click Software Distribution – Application Monitoring
03. Select Application Compliance and click Run
04. Under Application Compliance reports, use Google Chrome for Application and All System for collection and click View report **Note:** You can drill down to a more specific report using the links inside the reports

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

#Open Report
Invoke-CMReport -ReportPath "Software Distribution - Application
Monitoring/Application compliance" -SiteCode "$SiteCode" -SrsServerName
"$servername"
```

22. Application Management – App-V 5 Applications

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001
More information	Create App-V virtual environments in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/create-app-v-virtual-environments
Description	In this chapter, we will look at all steps required to create an application that depends on some requirements

22.1. Creating an Application App-V Client 5.0

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Select Application and Click Create Application
04. Under Specify Settings for this application, select Manually specify the application information and click Next
05. Under Specify information about this application type App-V Client 5.0 as Name and click Next
06. Under Specify the Configuration Manager Application Catalog entry click Next
07. Under Configure deployment types and the priority in which they will be applied for this application click Add
08. Under Specify settings for this deployment type, select Script installer as type and click Next
09. Under specify general information for this deployment type App-V client for Windows 8 x64 as Name and click Next
10. Under Specify information about the content to be delivered to target devices, fill up the following information: • Content location: \\srv0001\trainingfiles\source\App-V5 Client • Installation program: "appv_client_setup.exe" /q /ACCEPTEULA • Uninstall program: "appv_client_setup.exe" /UNINSTALL /q Click Next
11. Under Specify how this deployment type is detected, click Add Clause
12. Under detection rule, fill up the following: • Settings Type: Windows Installer • Product Code: {6313DBA3-0CA9-4CD8-93B3-373534146B7B} Click Ok
11. Under Specify how this deployment type is detected, click Add Clause
13. Under detection rule, fill up the following: • Settings Type: Registry • Hive: HKEY_LOCAL_MACHINE

- Key: SOFTWARE\Microsoft\AppV\Client
- Value: Version
- Data Type: String
- This registry setting must satisfy the following rule to indicate the presence of this application
- Operator: Begins with
- Value: 5.2

Click Ok and Once back to the specify how this deployment type is detected, click Next

14. Under specify user experience settings for the application, fill up the following:

- Installation behaviour: Install for system
- Logon requirement: Whether or not a user logged on

Click Next

15. Under Specify installation requirements for this deployment type, click Add

16. Under Create Requirement, select the following information:

- Category: Device
- Condition: Operating System
- Rule Type: Value
- Operator: One of
- Values: All Windows 8.1 (64-bit) and All Windows 10 (64-bit)

Click Ok and once back to the Specify installation requirements for this deployment type, change the connector from And to Or and click Next

17. Under specify software dependencies for this deployment type, click Next

18. Under Confirm the settings for this deployment type, click Next

19. Under The create Deployment Type Wizard completed successfully, click Close

20. Once back to Create Application Wizard, click Next

21. Under confirm the settings for this application, click Next

22. Under The Create Application Wizard completed successfully, click Close

This can also be achieved via PowerShell using the commands below:

```
. c:\TrainingFiles\Scripts\Add-CMDeploymentTypeGlobalCondition.ps1
. c:\TrainingFiles\Scripts\Set-CMDetectionRule.ps1

$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

$AppName = "App-V Client 5.0"
$DTName = "App-V client for Windows 8 x64"
$MSIProductCode = "{6313DBA3-0CA9-4CD8-93B3-373534146B7B}"

New-CMApplication -Name "$AppName"
Add-CMScriptDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -InstallCommand '"appv_client_setup.exe" /q /ACCEPTEULA' -
ProductCode "$MSIProductCode" -ContentLocation
"\\srv0001\trainingfiles\source\App-V5 Client" -LogonRequirementType
WhereOrNotUserLoggedOn -UninstallCommand '"appv client setup.exe"
/UNINSTALL /q'
```

```
Get-CMDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" | Set-CMDeploymentType -MsiOrScriptInstaller -
InstallationBehaviorType InstallForSystem
Add-CMDeploymentTypeGlobalCondition -ApplicationName "$AppName" -
DeploymentTypeName "$DTName" -sdkserver "$dkserver" -sitecode "$sitecode" -
GlobalCondition OperatingSystem -Operator OneOf -Value
"Windows/All x64 Windows 8.1 Client;Windows/All x64 Windows 10 and higher C
lients"

Set-CMDetectionRule -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -CreateRegistryValidation -RegistryKey
"SOFTWARE\Microsoft\AppV\Client" -RegistryKeyValue "Version" -
RegistryKeyValidationValue "5.2" -CreateMSIValidation -MSIProductCode
"$MSIProductCode"
```

22.2. Creating a Virtual Application Robocopy

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Select Application and Click Create Application
04. Under Specify Settings for this application, select Microsoft Application Virtualization 5 as Type and \\srv0001\trainingfiles\Source\Robocopy App-V5\Robocopy.appv as Location and click Next
05. Under View imported information, click Next
06. Under Specify information about this application, click Next
07. Under Confirm the settings for this application, click Next
08. Under The create application wizard complete successfully, click Close
09. Select the Robocopy application and click Deployment Types tab at the bottom of the console
11. Select the Deployment Type and click Properties
12. Under Robocopy – Microsoft Application Virtualization 5, change to Content tab
13. On Content tab, under Select the deployment option to use when a client is connected with a fast (LAN) network boundary, change deployment options to Stream content from distribution point. Change to the Requirements tab
14. Under the requirement tab, click Add
15. Under Create Requirement, select the following information: • Category: Device • Condition: Operating System • Rule Type: Value • Operator: One of • Values: All Windows 8.1 (64-bit) and All Windows 10 (64-bit) Click Ok and change to the Dependencies tab
16. Under Dependencies, click Add
17. Under Add Dependency, type App-V Client as Dependency group name and click Add

18. Under Specify required application Select App-V Client 5.0 and check App-V Client for Windows 8 x64. Click Ok three times

Note: Once done, distribute the application content to DP Group and note that all requirements are also going to be distributed

This can also be achieved via PowerShell using the commands below:

```
. c:\TrainingFiles\Scripts\Add-CMDeploymentTypeGlobalCondition.ps1

$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$ParentAppName = "App-V Client 5.0"
$AppName = "Robocopy"
$DTName = "Robocopy - Microsoft Application Virtualization 5"

New-CMApplication -Name "$AppName"

Add-CMAppv5XDeploymentType -ApplicationName "$AppName" -ContentLocation
"\\srv0001\trainingfiles\Source\Robocopy App-V5\Robocopy.appv" -
DeploymentTypeName "$DTName" -FastNetworkDeploymentMode
DownloadContentForStreaming

Get-CMDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" | New-CMDeploymentTypeDependencyGroup -GroupName "App-V Client" |
Add-CMDeploymentTypeDependency -DeploymentTypeDependency (Get-
CMDeploymentType -ApplicationName "$ParentAppName") -IsAutoInstall $true

Add-CMDeploymentTypeGlobalCondition -ApplicationName "$AppName" -
DeploymentTypeName "$DTName" -sdkserver "$dkserver" -sitecode "$sitecode" -
GlobalCondition OperatingSystem -Operator OneOf -Value
"Windows/All_x64_Windows_8.1_Client;Windows/All_x64_Windows_10_and_higher_C
lients"

Start-CMContentDistribution -ApplicationName "$AppName" -
DistributionPointGroupName "Training Lab"
```

22.3. Visualizing Application Relationship

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Select Robocopy and click View Relationships -> Dependency
04. The Robocopy Dependencies will open and you will be able to see all software dependencies

This can also be achieved via PowerShell using the commands below:

```
. c:\TrainingFiles\Scripts\ShowDependentApplications_v0_9.ps1

$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"
$AppName = "App-V Client 5.0"

ShowDependentApplications -ApplicationName "$AppName" -SiteCode "$SiteCode"
-SiteServer "$servername"
```

23. Deploying and monitoring available Virtual Applications to a User

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002 WKS0004
More information	Deploy applications with System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/deploy-applications Monitor applications from the System Center Configuration Manager console https://docs.microsoft.com/en-us/sccm/apps/deploy-use/monitor-applications-from-the-console Enable the App-V in-box client https://technet.microsoft.com/en-gb/itpro/windows/manage/appv-enable-the-app-v-desktop-client?f=255&MSPPError=-2147217396
All Systems & All Users collection	Deploying anything to All Systems or All Users is not recommended and it goes against best practices. We are using these 2 collections for demonstration purposes only.
Description	In this chapter, we will look at all steps required to deploy an App-V application to a Windows 10 (with Update Anniversary) as well as Windows 8.1 where a App-V client needs to be installed before

23.1. Enable App-V in-box

Perform this task on the WKS0001 virtual machine logged on as user01
01. Open Windows PowerShell (as administrator)
02. Type Enable-Appv and press ENTER
03. Restart the device. **Note:** Repeat the process on WKS0002

This can also be achieved via PowerShell using the commands below:

```
Enable-Appv
start-sleep 5
Start-Process -Filepath ("shutdown") -ArgumentList ("/r /t 0")
```

23.2. Deploying Application

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Select Robocopy and click Deploy
04. Under Specify general information for this deployment, click Browse (Collection) and select the Collection you want to deploy. Click Next **Note:** You can deploy to a Device as well as Users. In this example, we are using the user collection - All Users
05. Under Specify the content destination, click Next
06. Under specify settings to control how this software is deployed, click Next **Note:** Action can be Install or Uninstall and Purpose can be Available or Required.
07. Under Specify the schedule for this deployment, click Next
08. Under Specify the user experience for the installation of this software on the selected devices, click Next
09. Under specify Configuration Manager and Operations Manager alert options, click Next
10. Under Confirm the settings for this new deployment click Next
11. Under completion, click Close

This can also be achieved via PowerShell using the commands below:

```
Start-CMApplicationDeployment -CollectionName "All Users" -Name "Robocopy"
```

23.3. Installing available application

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select User Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under User Policy Retrieval & Evaluation Cycle click Ok
04. Open Software Center under Start -> Microsoft System Center -> Configuration Manager
05. Under Available Software, select Robocopy and click Install
06. If needed, click Installation Status tab to follow the installation process
07. Once the installation is finished, click Installed Software to see all installed software **Note:** Repeat the process on the WKS0002 and WKS0004 **Note:** The App-V 5.0 Client will only be installed on WKS0004

24. Application Management – Advanced

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002 WKS0004
More information	How to revise and supersede applications in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/revise-and-supersede-applications How to create global conditions in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/create-global-conditions Simulate application deployments with System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/deploy-use/simulate-application-deployments
Description	In this chapter, we will look at all steps required to replace an application with a new version; steps required to deploy an application that requires approval from the SCCM administrator; creating custom deployment requirements as well as perform simulated deployment, so we can test the detection method, requirements and dependencies of an application deployment without installing or uninstalling the application.

24.1. Preparing the Environment

Before Starting this Module, you will need to:

1- Create a Firefox 40 Application with the Following Information, Deploy and Install
   - Specific Settings for this application select Manually specify
   - Name: Firefox 40
   - Deployment Type: Script
   - Name: Firefox 40 Installation for Windows 8.1 and 10
   - Content Location: \\srv0001\TrainingFiles\Source\Firefox 40
   - Install program: "Firefox Setup 40.0.exe" -ms
   - Uninstall program: "C:\Program Files (x86)\Mozilla Firefox\uninstall\helper.exe" -ms
   - Detection Rule:
     - Setting Type: File System
     - Type: File
     - File: %ProgramFiles%\Mozilla Firefox
     - File or folder name: firefox.exe
     - This file or folder is associated with a 32-bit application on 64-bit systems checked
     - Property: Version

    - Operator: Greater than or equal to
    - Value: 40.0
- Installation behaviour: Install for a system
- Logon requirement: Whether or not a user is logged on
- Requirements: Operating System, one of, All Windows 10 (64-bit), All Windows 8.1 (64-bit)
- Distribute content to a Distribution Point Group
- Make it available to all users (Available deployment)
- Log on to WKS0001, WKS0002, WKS0004 as user01 and install the software

2- Create a Firefox 49 Application with the Following Information
- Specific Settings for this application select Manually specify
- Name: Firefox 49
- Deployment Type: Script
- Name: Firefox 49 Installation for Windows 8.1 and 10
- Content Location: \\srv0001\TrainingFiles\Source\Firefox 49
- Install program: "Firefox Setup 49.0.1.exe" -ms
- Uninstall program: "C:\Program Files (x86)\Mozilla Firefox\uninstall\helper.exe" -ms
- Detection Rule:
    - Setting Type: File System
    - Type: File
    - File: %ProgramFiles%\Mozilla Firefox
    - File or folder name: firefox.exe
    - This file or folder is associated with a 32-bit application on 64-bit systems checked
    - Property: Version
    - Operator: Greater than or equal to
    - Value: 49.0
- Installation behaviour: Install for a system
- Logon requirement: Whether or not a user is logged on
- Requirements: Operating System, one of, All Windows 10 (64-bit), All Windows 8.1 (64-bit)
- Distribute content to a Distribution Point Group

3- Create a Java 8 Application with the Following Information:
- Specific Settings for this application select Manually specify
- Name: Java8
- Deployment Type: Script
- Name: Java8 for Windows 10
- Content Location: \\srv0001\trainingfiles\Source\Java8
- Install program: "Java8.exe" /s
- Detection Rule:
    - Setting Type: File System
    - Type: folder
    - File: %ProgramFiles%\Java\jre1.8.0_101\bin
    - File or folder name: java.exe

    - This file or folder is associated with a 32-bit application on 64-bit systems checked
    - Property: Version
    - Operator: Greater than or equal to
    - Value: 8.0.1010
  - Installation behaviour: Install for a system
  - Logon requirement: Whether or not a user is logged on
  - Requirements: Operating System, one of, All Windows 10 (64-bit)
  - Distribute content to a Distribution Point Group
4- Create an Acrobat Reader XI Application with the Following Information:
  - Specific Settings for this application select Manually specify
  - Name: Acrobat Reader XI
  - Deployment Type: Script
  - Name: Acrobat Reader XI for Windows 10
  - Content Location: \\srv0001\trainingfiles\Source\AdobeXI
  - Install program: "AdbeRdr11010_en_US.exe" /msi EULA_ACCEPT=YES REMOVE_PREVIOUS=YES /qn
  - Detection Rule:
    - Setting Type: Windows Installer
    - Product Code: {AC76BA86-7AD7-1033-7B44-AB0000000001}
  - Installation behaviour: Install for a system
  - Logon requirement: Whether or not a user is logged on
  - Requirements: Operating System, one of, All Windows 10 (64-it)
  - Distribute content to a Distribution Point Group

This can also be achieved via PowerShell using the commands below:

```
. c:\TrainingFiles\Scripts\Add-CMDeploymentTypeGlobalCondition.ps1
. c:\TrainingFiles\Scripts\Set-CMDetectionRule.ps1

$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

$AppName = "Firefox 40"
$DTName = "Firefox 40 Installation for Windows 8.1 and 10"

New-CMApplication -Name "$AppName"

#we will remove the need for product key later on, it is just easier to add
like that
Add-CMScriptDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -InstallCommand '"Firefox Setup 40.0.exe" -ms' -ProductCode
"{6313DBA3-0CA9-4CD8-93B3-373534146B7B}" -ContentLocation
"\\srv0001\TrainingFiles\Source\Firefox 40" -LogonRequirementType
WhereOrNotUserLoggedOn -UninstallCommand '"C:\Program Files (x86)\Mozilla
Firefox\uninstall\helper.exe" -ms'
Get-CMDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" | Set-CMDeploymentType -MsiOrScriptInstaller -
InstallationBehaviorType InstallForSystem
```

```
Add-CMDeploymentTypeGlobalCondition -ApplicationName "$AppName" -
DeploymentTypeName "$DTName" -sdkserver "$dkserver" -sitecode "$sitecode" -
GlobalCondition OperatingSystem -Operator OneOf -Value
"Windows/All_x64_Windows_10_and_higher_Clients;Windows/All_x64_Windows_8.1_
Client"
Set-CMDetectionRule -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -CreateFileValidation -FolderPath "%ProgramFiles%\Mozilla
Firefox" -FileName "firefox.exe" -IsPath64bit $false  -FileValidationValue
"40.0"

Start-CMContentDistribution -ApplicationName "$AppName" -
DistributionPointGroupName "Training Lab"
Start-CMApplicationDeployment -CollectionName "All Users" -Name "$AppName"

$AppName = "Firefox 49"
$DTName = "Firefox 49 Installation for Windows 8.1 and 10"

New-CMApplication -Name "$AppName"

#we will remove the need for product code later, it is just easier to add
like that
Add-CMScriptDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -InstallCommand '"Firefox Setup 49.0.1.exe" -ms' -ProductCode
"{6313DBA3-0CA9-4CD8-93B3-373534146B7B}" -ContentLocation
"\\srv0001\TrainingFiles\Source\Firefox 49" -LogonRequirementType
WhereOrNotUserLoggedOn -UninstallCommand '"C:\Program Files (x86)\Mozilla
Firefox\uninstall\helper.exe" -ms'
Add-CMDeploymentTypeGlobalCondition -ApplicationName "$AppName" -
DeploymentTypeName "$DTName" -sdkserver "$dkserver" -sitecode "$sitecode" -
GlobalCondition OperatingSystem -Operator OneOf -Value
"Windows/All_x64_Windows_10_and_higher_Clients;Windows/All_x64_Windows_8.1_
Client"
Set-CMDetectionRule -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -CreateFileValidation -FolderPath "%ProgramFiles%\Mozilla
Firefox" -FileName "firefox.exe" -IsPath64bit $false  -FileValidationValue
"49.0"

Get-CMDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" | Set-CMDeploymentType -MsiOrScriptInstaller -
InstallationBehaviorType InstallForSystem
Start-CMContentDistribution -ApplicationName "$AppName" -
DistributionPointGroupName "Training Lab"

$AppName = "Java8"
$DTName = "Java8 for Windows 10"

New-CMApplication -Name "$AppName"

#we will remove the need for product code later, it is just easier to add
like that
Add-CMScriptDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -InstallCommand '"Java8.exe" /s' -ProductCode "{6313DBA3-0CA9-
4CD8-93B3-373534146B7B}" -ContentLocation
"\\srv0001\trainingfiles\Source\Java8" -LogonRequirementType
WhereOrNotUserLoggedOn
```

```
Add-CMDeploymentTypeGlobalCondition -ApplicationName "$AppName" -
DeploymentTypeName "$DTName" -sdkserver "$dkserver" -sitecode "$sitecode" -
GlobalCondition OperatingSystem -Operator OneOf -Value
"Windows/All_x64_Windows_10_and_higher_Clients"
Set-CMDetectionRule -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -CreateFileValidation -FolderPath
"%ProgramFiles%\Java\jre1.8.0_101\bin" -FileName "java.exe" -IsPath64bit
$false  -FileValidationValue "8.0.1010"

Get-CMDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" | Set-CMDeploymentType -MsiOrScriptInstaller -
InstallationBehaviorType InstallForSystem
Start-CMContentDistribution -ApplicationName "$AppName" -
DistributionPointGroupName "Training Lab"

$AppName = "Acrobat Reader XI"
$DTName = "Acrobat Reader XI for Windows 10"

New-CMApplication -Name "$AppName"
Add-CMScriptDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" -InstallCommand '"AdbeRdr11010_en_US.exe" /msi EULA_ACCEPT=YES
REMOVE_PREVIOUS=YES /qn' -ProductCode "{AC76BA86-7AD7-1033-7B44-
AB0000000001}" -ContentLocation "\\srv0001\trainingfiles\Source\AdobeXI" -
LogonRequirementType WhereOrNotUserLoggedOn
Add-CMDeploymentTypeGlobalCondition -ApplicationName "$AppName" -
DeploymentTypeName "$DTName" -sdkserver "$dkserver" -sitecode "$sitecode" -
GlobalCondition OperatingSystem -Operator OneOf -Value
"Windows/All_x64_Windows_10_and_higher_Clients;Windows/All_x64_Windows_8.1_
Client"

Get-CMDeploymentType -ApplicationName "$AppName" -DeploymentTypeName
"$DTName" | Set-CMDeploymentType -MsiOrScriptInstaller -
InstallationBehaviorType InstallForSystem
Start-CMContentDistribution -ApplicationName "$AppName" -
DistributionPointGroupName "Training Lab"
```

24.2. Creating an Application Supersedence

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Select Firefox 49 and click Properties
04. Under Firefox 49 Properties, change to the Supersedence tab
05. Under supersedence tab click Add
06. Under Specify supersedence relationship click Browse
07. Under Choose Application, click Firefox 40 and click Ok
08. Once back to the Specify Supersedence Relationship, change the New Deployment Type to Firefox 49 Install for Windows 8.1 and 10, check Uninstall and click Ok twice

This can also be achieved via PowerShell using the commands below:

```
$OldAppName = "Firefox 40"
$OldDTName = "Firefox 40 Installation for Windows 8.1 and 10"

$NewAppName = "Firefox 49"
$NewDTName = "Firefox 49 Installation for Windows 8.1 and 10"

$OldDeploymentType = Get-CMDeploymentType -ApplicationName "$OldAppName" -
DeploymentTypeName "$OldDTName"
$NewDeploymentType = Get-CMDeploymentType -ApplicationName "$NewAppName" -
DeploymentTypeName "$NewDTName"

Add-CMDeploymentTypeSupersedence -SupersededDeploymentType
$OldDeploymentType -SupersedingDeploymentType $NewDeploymentType -
IsUninstall $true
```

24.3. Creating an Application Supersedence Deployment

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Once back to the select Firefox 49 and click Deploy
02. Under Specify general information for this deployment, select All Users Collection and click Next
03. Under Specify the content destination, click Next
04. Under Specify settings to control how this software is deployed, select the following: • Action: Install • Purpose: Available • Automatically upgrade any superseded version of this application: Checked Click Next
05. Under Specify the schedule for this deployment, click Next
06. Under specify the user experience for the installation of this software on the selected devices, click Next
07. Under Specify Configuration Manager and Operations manager alert options, click Next
08. Under Confirm the settings for this new deployment, click Next
09. Under The Deploy Software Wizard completed successfully, click Close

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$AppName = "Firefox 49"
$CollName = "All Users"

Start-CMApplicationDeployment -CollectionName "$CollName" -Name "$AppName"
$Deployment = Get-WmiObject -Namespace "root\SMS\site_$($SiteCode)" -Class
"SMS_ApplicationAssignment" | Where-Object { $_.ApplicationName -like
"$AppName" -and $_.CollectionName -like "$CollName"}
$Deployment.UpdateSupersedence = "True"
$Deployment.Put()
```

24.4. Installing an Application Supersedence

Perform this task on the wks0001 virtual machine logged on as user02
01. Open Control Panel and then Configuration Manager.
02. Change to the actions tab. Select User Policy Retrieval & Evaluation cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Once the User Policy Retrieval & Evaluation cycle message appears, click Ok. **Note:** Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Click Start, Microsoft System Center, Configuration Manager and click Software Center
05. Under Available Software, for a short period you should see both Firefox 40 and Firefox 49 available. Short after you should see the Installation of Firefox 49 start
06. If needed, click Installation Status tab to follow the installation process. Repeat the process on WKS0002 and WKS0004

24.5. Creating a Global Condition

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Global Conditions
03. Click Create Global Condition
04. Under Specify details about this global condition, use the following: • Name: Computer Model • Device Type: Windows • Condition type: Setting • Setting type: WQL Query • Data Type: String • Namespace: root\cimv2 • Class: Win32_ComputerSystem • Property: Model Click Ok **Note:** Be sure that you typed the information correctly as, once it is used, it cannot be changed

This can also be achieved via PowerShell using the commands below:

```
New-CMGlobalCondition -Class "Win32_ComputerSystem" -DataType "String" -
DeviceType "Windows" -Name "Computer Model" -Property "Model" -Namespace
"root\cimv2"
```

24.6. Using a Custom Global Condition

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Applications Management and click Applications
03. Select the Java8 and click Deployment Types tab at the bottom of the console
04. Select the Deployment Type and click Properties
05. Under Java8 for Windows 10 Properties, change to Requirements tab and click Add
06. Under Create Requirement, create a new requirement with the following information: • Category: Custom • Condition: Computer Model • Rule Type: Value • Operator: Equals • Value: SVD11225CYB Click Ok Twice

24.7. Deploying an Approval Required Application

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Applications Management and click Applications
03. Select Java8 and click Deploy
04. Under Specify general information for this deployment, click Browse (Collection) and select the Collection you want to deploy. Click Next **Note:** You can deploy to a Device as well as Users. In this example, we are using All Users collection
05. Under Specify the content destination, click Next
06. Under specify settings to control how this software is deployed, select Require administrator approval if users request this application and click Next
07. Under Specify the schedule for this deployment, click Next
08. Under Specify the user experience for the installation of this software on the selected devices, click Next
09. Under specify Configuration Manager and Operations Manager alert options, click Next
10. Under Confirm the settings for this new deployment click Next
11. Under Completion, click Close

This can also be achieved via PowerShell using the commands below:

```
Start-CMApplicationDeployment -CollectionName "All Users" -Name "Java8" -
ApprovalRequired $true
```

24.8. Requesting application via Software Center

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select User Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under User Policy Retrieval & Evaluation Cycle click Ok
04. Open Software Center under Start -> Microsoft System Center -> Configuration Manager
05. Under Available Software, select Java8
06. On Java8, type some information on the text box and click Request
07. Once the request is submitted, a message will be displayed

24.9. Approving/Denying application requests

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Approval Requests
03. Right click the visible columns and add Comments
04. Once done, select the application, and click Approve
05. Under Application Request(s) add an Approver's Comments and click OK

This can also be achieved via PowerShell using the commands below:

```
#approve request
Get-CMApprovalRequest -ApplicationName "Java8" -User "CLASSROOM\User01" |
Approve-CMApprovalRequest
#deny request
#Get-CMApprovalRequest -ApplicationName "Java8" -User "CLASSROOM\User01" |
Deny-CMApprovalRequest
```

24.10. Installing an approved application

Perform this task on the wks0001 virtual machine logged on as user01
01. Open Control Panel and then Configuration Manager.
02. Change to the actions tab. Select User Policy Retrieval & Evaluation cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Once the User Policy Retrieval & Evaluation cycle message appears, click Ok.

Note: Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Click Start, Microsoft System Center, Configuration Manager and click Software Center
05. Under Available Software, select Java 8 and click Install **Note:** The installation will not be completed and the Status will be changed to "This software is not applicable to your device"

24.11. Simulate Application Deployment

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Applications Management and click Applications
03. Select Acrobat Reader XI and click Simulate Deployment
04. Under Specify general information for this deployment, click Browse (Collection) and select the Collection you want to deploy and select Install for Action. Click Next **Note:** You can deploy to a Device as well as Users. In this example, we are using All Systems collection
05. Under Confirm the settings for this new Deployment, click Next
06. Under Completion, click Close **Note:** Once done, refresh the machine policy on WKS0001, WKS0002 and WKS0004 and monitor the application deployment using the same steps as we have seen under the Monitoring Application Deployment module.

This can also be achieved via PowerShell using the commands below:

```
Start-CMApplicationDeploymentSimulation -CollectionName "All Systems" -Name
"Acrobat Reader XI"
```

25. Application Management for Linux

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 SRV0003
More information	Create Linux and UNIX server applications with System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/apps/get-started/creating-linux-and-unix-server-applications How to manage clients for Linux and UNIX servers in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/clients/manage/manage-clients-for-linux-and-unix-servers
Description	In this chapter, we will look at all steps required to execute commands on a Linux machine. The command can be to install software or run application. Using Package and Program is also available for Windows machines.

25.1. Creating a Package and Program

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Packages
03. Select Packages and click Create Package
04. Under Specify information about this package, use the following: • Name: Chrome for Linux • This package contains source files: checked • Source folder: \\srv0001\trainingfiles\source\Chrome for Linux Click Next
05. Under choose the program type that you want to create click Next
06. Under Specify information about this standard program, use the following: • Name: Install Chrome for Linux • Command line: rpm -ivh google-chrome-stable_current_x86_64.rpm • Program can run: Whether or not a user is logged on Click Next
07. Under specify the requirements for this standard program, select This program can run only on specified platforms and select CentOS 7 (x64). Click Next
08. Under confirm the settings, click Next
09. Under the Create Package and Program wizard completed successfully, click Close **Note:** Once done, distribute the application content to DP Group

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

$pkg = New-CMPackage -Name "Chrome for Linux" -Path
"\\srv0001\trainingfiles\source\Chrome for Linux"
$prg = New-CMProgram -CommandLine "rpm -ivh google-chrome-
stable current x86 64.rpm" -PackageName "Chrome for Linux" -
StandardProgramName "Install Chrome for Linux" -ProgramRunType
WhetherOrNotUserIsLoggedOn -RunMode RunWithAdministrativeRights

$prgInfo = gwmi SMS Program -namespace root\sms\site $SiteCode -filter
"PackageID='$($pkg.PackageID)' and ProgramName='$($prg.ProgramName)'"
$prgInfo.Get()

$newOS =
([WMIClass]("\\.\root\sms\site $($sitecode):SMS OS Details")).CreateInstanc
e()
$newOS.MaxVersion = "7.99.9999.9999"
$newOS.MinVersion = "7.00.0000.0"
$newOS.Name = "CentOS"
$newOS.Platform = "x64"

$prgInfo.SupportedOperatingSystems = $newOS
if ($prgInfo.ProgramFlags -band 0x08000000) -ne 0)
{
      $prgInfo.ProgramFlags = $prgInfo.ProgramFlags -bxor 0x08000000
}
$prgInfo.Put() | out-null

Start-CMContentDistribution -PackageId $pkg.PackageID -
DistributionPointGroupName "Training Lab"
```

25.2. Deploying Application to a Linux

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Packages
03. Select Chrome for Linux and Click Deploy
04. Under Specify general information for this deployment, click Browse (Collection) and select the CentOS Servers Collection. Click Ok and then Next
05. Under Specify the content destination, click Next
06. Under specify settings to control how this software is deployed, confirm that Purpose is Required and click Next
07. Under Specify the schedule for this deployment, click New
08. Under Assignment Schedule click Assign immediately after this event and select As soon as possible. Click Ok and then next
09. Under User Experience, click Next
10. Under Distribution Points, click Next
11. Under Confirm the settings for this new deployment click Next

12. Under Completion, click Close

This can also be achieved via PowerShell using the commands below:

```
Start-CMPackageDeployment -CollectionName "CentOS Servers" -PackageName
"Chrome for Linux" -ProgramName "Install Chrome for Linux" -StandardProgram
-DeployPurpose Required -ScheduleEvent AsSoonAsPossible -FastNetworkOption
DownloadContentFromDistributionPointAndRunLocally -SlowNetworkOption
DoNotRunProgram
```

25.3. Installing Required Applications

Perform this task on the SRV0003 virtual machine logged on as Administrator
01. On a Linux machine, open a terminal Window
02. log on as root using the command: su
03. Run /opt/microsoft/configmgr/bin/ccmexec -rs policy **Note:** Once done, monitoring application deployment using the same steps as we have seen under the Monitoring Application Deployment module.

25.4. Monitoring Application Deployment via Console

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Click Deployments
03. Select Chrome for Linux (Install Chrome for Linux) and click Summarization **Note:** The Application Deployment summarization runs once every 60 minutes by default, this can be changed on Administration -> Site Configuration -> Sites -> <Site> -> Status Summarizers -> Application Deployment Summarizer
04. Click Ok once the Configuration Manager information screen appears
05. After the summarization, under Completion Statistics, view Status. Click View Status for more information
06. Under View Status, Asset Details, you can see which device received the software and Under Requirements Not Met you can see which device did not have the software installed.

This can also be achieved via PowerShell using the commands below:

```
$DeploymentName = "Chrome for Linux (Install Chrome for Linux)"
Get-CMDeployment -SoftwareName "$DeploymentName" | Invoke-
CMDeploymentSummarization
Start-Sleep 10
Get-CMDeployment -SoftwareName "$DeploymentName" | select ApplicationName,
CollectionName, NumberErrors, NumberInProgress, NumberOther, NumberSuccess,
NumberTargeted, NumberUnknown
#for detailed summary, use:
```

```
#Get-CMPackageDeploymentStatus -Name "Chrome for Linux" | Get-
CMDeploymentStatusDetails | select DeviceName, StatusDescription,
StatusType
```

25.5. Monitoring Package Deployment via Reports

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Reporting, Reports and click Software Distribution – Packages and Program Deployment Status
03. Select Status of a specified package and program deployment and click Run
04. Under Status of a specific package and program deployment reports, fill up the parameters (you may use the Values link) and click View report **Note:** You can drill down to a more specific report using the links inside the reports

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

#Open Report
Invoke-CMReport -ReportPath "Software Distribution - Package and Program
Deployment Status/Status of a specified package and program deployment" -
SiteCode "$SiteCode" -SrsServerName "$servername"
```

26. Software Update

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002 WKS0004
More information	Prerequisites for Software Updates in Configuration Manager https://docs.microsoft.com/en-us/sccm/sum/plan-design/prerequisites-for-software-updates Best Practices for Software Updates in Configuration Manager https://docs.microsoft.com/en-us/sccm/sum/plan-design/software-updates-best-practices Introduction to Software Updates in Configuration Manager https://technet.microsoft.com/en-us/library/gg682168.aspx Configuring Software Updates in Configuration Manager https://technet.microsoft.com/en-us/library/gg712312.aspx Example Scenario for Using Configuration Manager to Deploy and Monitor the Security Software Updates Released Monthly by Microsoft https://technet.microsoft.com/en-us/library/jj134348.aspx
Description	In this chapter, we will look at all steps required to manage Microsoft updates in a SCCM environment, including deployment, monitoring and reporting compliance.

26.1. Installation of Windows Server Update Services

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Server Manager
02. Click Manage and Add Roles and Features
03. Before you begin, click Next
04. Select Role-based or feature-based installation and click Next
05. Select select a server from the server pool and select the server “SRV0002.classroom.intranet” and click Next
06. Under select server roles, select Windows Server Update Services
07. The Add Roles and Features wizard will open to add required requirements, click Add Features and once back to the Select server roles page, click Next
08. Under Select features, click Next
09. Under Windows Server Update Services, click Next
10. Under Select role services, select WSUS Services and SQL Server Connectivity and uncheck WID Connectivity. Click Next

11. Under Content location select, type C:\WSUS under Store updates in the following location and click Next
12. Under DB Instance, type localhost and click Check Connection. Click Next
13. Under Confirm installation selections, click Install
14. Once the installation is succeeded, click Close
15. Click the yellow triangle and then Launch Post-Installation tasks
16. Once done, the yellow triangle will disappear

This can also be achieved via PowerShell using the commands below:

```
#using SQL Server Installed on the same box
Get-WindowsFeature -Name UpdateServices-Services,UpdateServices-DB |
Install-WindowsFeature -IncludeManagementTools
Start-Process -Filepath ('C:\Program Files\Update
Services\Tools\WsusUtil.exe') -ArgumentList ('PostInstall
CONTENT DIR="C:\WSUS" SQL INSTANCE NAME="srv0002"') -wait

#using WID
#Get-WindowsFeature -Name UpdateServices | Uninstall-WindowsFeature #this
does install WID
#& "C:\Program Files\Update Services\Tools\WsusUtil.exe" PostInstall
CONTENT_DIR="C:\WSUS"
```

26.2. Installation of the SCCM Software Update Point

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Site System Roles
03. Right click \\SRV0002.classroom.intranet and click Add Site System Roles
04. On Add Site System Roles Wizard, General, click Next
05. Under proxy, click Next
06. Under Specify roles for this server, select Software Update point and click Next
07. Under Specify software update point settings, select WSUS is configured to use ports 8530 and 8531 for client communications (default settings for WSUS on Windows Server 2012) and click Next
08. Under specify proxy and account settings for the software update point, click Next
09. Under Specify synchronization source settings, click Next
10. Under specify on a schedule and run every 1 days. Click Next the synchronization schedule, select Enable synchronization
11. Under select behaviour for software updates that are superseded, select Immediately expire a superseded software update and Run WSUS cleanup wizard. Click Next
12. Under select the software update classifications that you want to synchronize, leave selected only Critical updates. Click Next
13. Under Select the products that you want to synchronize, unselect all and Click Next **Note:** The updated Products and classifications are going to be added after the initial synchronization

14. Under select the language settings that you want to synchronize, leave select only English (Software Update File and Summary Details). Click Next
15. Under confirm the settings, click Next
16. Under You have successfully completed the Add Site System Roles wizard with the following settings click close
17. Click Monitoring
18. Expand System Status and click Component Status
19. Search for WSUS
20. Right Click SMS_WSUS_CONTROL_MANAGER, Show Messages and click All
21. Under Status Messages: Set Viewing Period, click OK
22. Verify the existence of Message ID 1013, 1014 and 1015
23. Right Click SMS_WSUS_CONFIGURATION_MANAGER, Show Messages and click All
24. Under Status Messages: Set Viewing Period, click OK
25. Verify the existence of Message ID 501 and 500
26. Verify the existence of Message ID 6600 **Note:** If you see the message id 6600, confirm you have selected the correct port during the installation of the Software Update Point
27. Double click any 6600 messages to see its details. Once done, click Ok
28. Right Click SMS_WSUS_WSYNC_MANAGER, Show Messages and click All
24. Under Status Messages: Set Viewing Period, click OK
25. Verify the existence of Message ID 501 and 500
26. Verify the existence of Message ID 6703 **Note:** If the logs reveal that there was a problem during the synchronization, it is normally network issues. **Note:** During the installation of the Software Update Point, this message is normal, however, it should not occur after the installation

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

if ((Get-CMSiteSystemServer -SiteSystemServerName "$servername") -eq $null)
{ New-CMSiteSystemServer -SiteCode $SiteCode -UseSiteServerAccount -
ServerName $servername }

Add-CMSoftwareUpdatePoint -SiteSystemServerName "$ServerName" -
ClientConnectionType "Intranet" -SiteCode $SiteCode -WsusiisPort 8530 -
WsusiissslPort 8531 -WsusSsl $false

start-sleep 90

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsgin.InsStrValue from SMS_StatMsg stmsg inner join SMS_StatMsgInsStrings
stmsgin on stmsg.RecordID = stmsgin.RecordID where stmsg.Component =
```

```
'SMS WSUS CONTROL MANAGER' and stmsg.MessageID = 1013 and
stmsgin.InsStrIndex = 0 and stmsgin.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_WSUS_CONTROL_MANAGER 1013 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_WSUS_CONTROL_MANAGER' and stmsg.MessageID = 1014 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_WSUS_CONTROL_MANAGER 1014 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_WSUS_CONTROL_MANAGER' and stmsg.MessageID = 1015 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_WSUS_CONTROL_MANAGER 1015 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site $SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_WSUS_CONTROL_MANAGER' and stmsg.MessageID = 500 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_WSUS_CONTROL_MANAGER 500 id's"
        break
    } else { Start-Sleep 10 }
}

$component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS WSUS CONTROL MANAGER' and stmsg.MessageID = 6600 and stmsg.SiteCode =
'$SiteCode'"
if ($component -ne $null)
{
    Write-Host "ERROR: Found SMS_WSUS_CONTROL_MANAGER 6600 id's" -
ForegroundColor Red
```

```
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS StatMsg stmsg where stmsg.Component =
'SMS WSUS CONTROL MANAGER' and stmsg.MessageID = 4629 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS WSUS CONTROL MANAGER 4629 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS WSUS SYNC MANAGER' and stmsg.MessageID = 4629 and stmsg.SiteCode =
'$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS WSUS SYNC MANAGER 4629 id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
    $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query "select
stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_WSUS_CONFIGURATION_MANAGER' and stmsg.MessageID = 4629 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_WSUS_CONFIGURATION_MANAGER 4629 id's"
        break
    } else { Start-Sleep 10 }
}

$Languages = @("Chinese (Simplified, China)", "French", "German",
"Japanese", "Russian")
$schedule = New-CMSchedule -RecurCount 1 -RecurInterval Days
#remove all classifications from sync
Set-CMSoftwareUpdatePointComponent -SiteCode $SiteCode -
ImmediatelyExpireSupersedence $True -RemoveUpdateClassification @("Security
Updates", "Service Packs", "Update Rollups") -RemoveLanguageSummaryDetail
$Languages -RemoveLanguageUpdateFile $Languages
start-sleep 5
#Add critical updates only
Set-CMSoftwareUpdatePointComponent -SiteCode $SiteCode -
AddUpdateClassification "Critical Updates"
#remove all products and classifications from sync (there is no option to
remove products from sync)
```

```
$wmiList = Get-WmiObject -Namespace "Root\SMS\Site $SiteCode"
SMS UpdateCategoryInstance -Filter "IsSubscribed = 'True'"
foreach ($wmiItem in $wmiList)
{
      $wmiItem.IsSubscribed = $false
      $wmiItem.Put() | out-null
}
Start-Sleep 5

Set-CMSoftwareUpdatePointComponent -EnableSynchronization $True -Schedule
$Schedule -SiteCode $SiteCode -SynchronizeAction
SynchronizeFromMicrosoftUpdate
```

26.3. Manual Synchronization

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Software Updates, select All Software Updates and click Synchronize Software Updates
03. Once the Run Synchronization question windows appear, click Yes
04. Click Monitoring
05. Click Software Update Point Synchronization Status and wait until the synchronization is completed
06. You can also review the following logs: • C:\ConfigMgr\Logs\wsyncmgr.log: Records details about the software updates synchronization process.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
Sync-CMSoftwareUpdate -FullSync $True
Start-Sleep 30

#6705 = In progress Database
#6704 = In progress (WSUS Server)
#6702 = Done
#6701 = Starting

while ($true)
{
      $return = gwmi -Class SMS SupSyncStatus -Namespace
"root\sms\site_$SiteCode" | select LastSyncErrorCode, LastSyncState
      Write-Output $return
      if ($return.LastSyncState -ne 6702) {start-sleep 10 } else { break }
}
```

26.4. Changing the List of Products

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.

02. Expand Site Configuration and click Sites
03. Right click 001 – Training Lab and click Configure Site Components -> Software Update Point
04. On Software Update Point Component Properties, change to the Products Tab
05. Under Products, select Windows 8.1 and Windows 10 and click Ok.
06. Force a manual Synchronization of the Software Updates
07. Once the synchronization is completed, Click Software Library
08. Expand Software Updates and click All Software Updates to see all updates that have been synchronized

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

Set-CMSoftwareUpdatePointComponent -SiteCode $SiteCode -AddProduct
@('Windows 8.1', 'Windows 10')
Start-Sleep 5
Sync-CMSoftwareUpdate -FullSync $True
Start-sleep 30

while ($true)
{
      $return = gwmi -Class SMS_SupSyncStatus -Namespace
"root\sms\site_$SiteCode" | select LastSyncErrorCode, LastSyncState
      Write-Output $return
      if ($return.LastSyncState -ne 6702) {start-sleep 10 } else { break }
}
```

26.5. Changing Default Client Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Client Settings
03. Select the default client settings and click Properties
04. Under Default Settings, click Software Updates and change the following: • Software update scan schedule: Occurs every 1 day • Software deployment re-evaluation: Occurs every 3 days • When any software update deployment is reached, install all other software update deployment with deadline coming within a specific period of time: Yes • Period of time for which all pending deployments with deadlines in this tie will also be installed: 7 days Once done, click Ok.

This can also be achieved via PowerShell using the commands below:

```
$ScanSchedule = New-CMSchedule -RecurCount 1 -RecurInterval Days
$DeploymentReEvaluation = New-CMSchedule -RecurCount 3 -RecurInterval Days

$ClientSettingsName = "Default Client Agent Settings"
Set-CMClientSetting -Name "$ClientSettingsName" -SoftwareUpdate -
BatchingTimeout 7 -DeploymentEvaluationSchedule $DeploymentReEvaluation -
Enable $True -EnforceMandatory $True -ScanSchedule $ScanSchedule -TimeUnit
Days
```

26.6. SCCM Client Scan

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Once the User Policy Retrieval & Evaluation cycle message appears, click Ok twice. **Note:** Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Open the Configuration Manager Properties
05. Change to the Actions Tab, select Software Updates Scan Cycle and click Run now **Note:** Software Update Scan Cycle action did not exist before as the computer had not received the latest policy
06. Under Software Updates Scan Cycle click Ok
07. On the client, you can also review the following client logs: • C:\Windows\ccm\Logs\ScanAgent.log: Records details about scan requests for software updates, the WSUS location, and related actions. • C:\Windows\ccm\Logs\UpdatesDeployment.log: Records details about deployments on the client, including software update activation, evaluation, and enforcement. Verbose logging shows additional information about the interaction with the client user interface. • C:\Windows\ccm\Logs\UpdatesHandler.log: Records details about software update compliance scanning and about the download and installation of software updates on the client. • C:\Windows\ccm\Logs\UpdatesStore.log: Records details about compliance status for the software updates that were assessed during the compliance scan cycle. • C:\Windows\ccm\Logs\WUAHandler.log: Records details about the Windows Update Agent on the client when it searches for software updates. • C:\Windows\WindowsUpdate.log: Records details about when the Windows Update Agent connects to the WSUS server and retrieves the software updates for compliance assessment and whether there are updates to the agent components.

Note: Repeat the process for the WKS0002 and WKS0004 machines
Note: WindowsUpdate.log is no longer used in Windows 10 as the logs are now generated using ETW (Event Tracing for Windows). For more information, refer to https://blogs.technet.microsoft.com/charlesa_us/2015/08/06/windows-10-windowsupdate-log-and-how-to-view-it-with-powershell-or-tracefmt-exe/

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
Start-Sleep 60

$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000113}")
Start-Sleep 60
```

26.7. Reporting Compliance via Console for All Updates

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Software Updates, select All Software Updates and click Run Summarization
03. Click Ok once the Configuration Manager information screen appears
04. Search for Update for Update for Windows 8.1 for x64-based Systems (KB3173424) and see the compliance via Statistics

This can also be achieved via PowerShell using the commands below:

```
Invoke-CMSoftwareUpdateSummarization
Start-Sleep 10
Get-CMSoftwareUpdate -Name "*KB3173424*" -fast | select
LocalizedDisplayName, NumMissing, NumNotApplicable, NumPresent, NumTotal,
NumUnknown, PercentCompliant
```

26.8. Deploying a Patch

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Software Updates and click All Software Updates
03. Search for Update for Windows 8.1 for x64-based Systems (KB3173424)
04. Select the Update and click Create Software Update Group
05. Under Create Software Update Group type Windows 8x Critical Updates as Name and click Ok
06. Select Software Update Groups
07. Select Windows 8x Critical Updates and click Show Members
08. Confirm that all updates have been added to the list.
09. Select Software Update Group
10. Select Windows 8x Critical Updates and Click Deploy

11. Under Specify general information for this deployment, select Windows 8 Workstations collection and click Next

12. Under Specify deployment settings for this deployment, select:

- Type of Deployment: Available
- Detail level: All messages

click Next

13. Under Configure schedule details for this deployment, click Next

14. Under Specify the user experience for this deployment, click Next

15. Under Specify Software update alert options for this deployment, click Next

16. Under specify download settings for this deployment, select if software updates are not available on preferred distribution point or remote distribution point, download content from Microsoft Updates and click Next

Note: Selecting if software updates are not available on preferred distribution point or remote distribution point download content from Microsoft Updates allows the download of Microsoft updatesfrom Microsoft in case a machine cannot download from a Distribution Point, is useful for mobile users or in the case of a distribution point failure. Use carefully as this will generate extra internet network traffic and this will need to be allowed by your company firewall and may affect 3g/4g network data usage and encur costs

17. Under Specify the package to use the following:

- Create a new deployment package: checked
- Name: Windows 8x Critical Updates
- Package source: \\SRV0001\WSUSDownloadContent\W8xCriticalUpdates

Once done, click Next

18. Under Specify the distribution points or distribution points groups to host the content, click Add Distribution Point Group, under Add-Distribution Point Groups, select Training Lab, and click Ok. Once back, click Next

19. Under specify the source location for software updates that you will download, select Download software updates from the internet and click Next

20. Under Specify the language of the updates, click Next

21. Under Confirm the settings, click Next

22. Under The Deploy Software Updates wizard completed successfully, click Close

23. Navigate to \\SRV0001\WSUSDownloadContent\W8xCriticalUpdates and confirm that the windows8.1-kb3173424-x64.cab has been downloaded

Note: The downloaded file will be under a subfolder of the W8xCriticalUpdates
Note: Once done, confirm that the content has been distributed to the distribution point using the Content Status under Monitoring->Distribution Status

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

$KB = "KB3173424"
```

```
$SupGroup = "Windows 8x Critical Updates"
$DepGroup = "Windows 8x Critical Updates"
$DepPath = "\\SRV0001\WSUSDownloadContent\W8xCriticalUpdates"
$ColName = "Windows 8 Workstations"
New-CMSoftwareUpdateGroup -Name "$SupGroup"
Start-Sleep 5

$UpdateList = Get-CMSoftwareUpdate -Name "*$($KB)*" -fast | Where-Object
{$_.LocalizedDisplayName -like "*x64*"}
Set-CMSoftwareUpdateGroup -Name "$SupGroup" -AddSoftwareUpdate $UpdateList
Start-Sleep 5
(Get-CMSoftwareUpdate -UpdateGroupName "$SupGroup" -
fast).LocalizedDisplayName
Start-Sleep 5

New-CMSoftwareUpdateDeploymentPackage -Name "$DepGroup" -Path "$DepPath"
Start-Sleep 20
Save-CMSoftwareUpdate -DeploymentPackageName "$DepGroup" -
SoftwareUpdateGroupName "$SupGroup"
Start-Sleep 60
Start-CMContentDistribution -DeploymentPackageName "$DepGroup" -
DistributionPointGroupName "Training Lab"
Start-Sleep 10
Start-CMSoftwareUpdateDeployment -CollectionName "$ColName" -
SoftwareUpdateGroupName "$SupGroup" -AcceptEula -DeploymentType Available -
DownloadFromMicrosoftUpdate $True -VerbosityLevel AllMessages
```

26.9. Installing a Patch

Perform this task on the wks0004 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under Machine Policy Retrieval & Evaluation Cycle click Ok **Note:** Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Open Software Center and click Updates
05. Select the Update and click Install
06. Once done, the button will be disabled and the label will change to Uninstall

This can also be achieved via PowerShell using the commands below:

```
#update policies
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
```

```
Start-Sleep 60

$Updates = Get-WmiObject -Namespace "root\ccm\clientSDK" -Class
CCM_SoftwareUpdate | Where-Object { $_.ComplianceState -eq 0 -and
$_.EvaluationState -eq 0}
Invoke-WmiMethod -Class CCM_SoftwareUpdatesManager -Name InstallUpdates -
ArgumentList (,$Updates) -Namespace root\ccm\clientsdk
```

26.10. Reporting Compliance via Console for an Update Group

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Software Updates and click Software Update Groups.
03. Select the Update Group ad click Run Summarization **Note:** The Summarization, by default, occurs once per hour.
04. Click Ok once the Configuration Manager information screen appears
05. After the summarization, under Statistics, review the statistics.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

Invoke-CMSoftwareUpdateSummarization
Start-Sleep 20
Get-CMSoftwareUpdate -IsDeployed $true | select LocalizedDisplayName,
NumMissing, NumNotApplicable, NumPresent, NumTotal, NumUnknown,
PercentCompliant
```

26.11. Reporting Compliance via Reports

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Reporting, Reports and click Software Updates – A Compliance
03. Select Compliance 4 - Updates by vendor month year
04. Under Compliance 4 - Updates by vendor month year report, fill up the parameters (you may use the Values link) as following: • Collection: Windows 8 Workstations • Vendor: Microsoft • Update Class: Critical Updates • Product: Windows 8.1 click View report **Note:** You can drill down to a more specific report using the links inside the reports **Note:** You can order by Approved to easily find approved updates information

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

#Open Report
Invoke-CMReport -ReportPath "Software Updates - A Compliance/Compliance 4 -
Updates by vendor month year" -SiteCode "$SiteCode" -SrsServerName
"$servername"
```

27. Endpoint Protection

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002 WKS0004
More information	Planning for Endpoint Protection in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/protect/plan-design/planning-for-endpoint-protection Configuring Endpoint Protection in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/protect/deploy-use/configure-endpoint-protection Using Configuration Manager Software Updates to Deliver Definition Updates https://docs.microsoft.com/en-us/sccm/protect/deploy-use/endpoint-definitions-configmgr
Description	In this chapter, we will look at all steps required to manage the Microsoft anti-virus (Endpoint Protection) as well as Windows Defender (when running on Windows 10)

27.1. Configuring Software Update Point

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Sites
03. Select 001 – Training Lab, Configure Site Components and click Software Update Point
04. Under Software Update Point Component Properties, click Classifications Tab
05. Under Classifications Tab select Definition Updates. Change to the Products Tab
06. Under Products Tab, select: • Forefront Endpoint Protection 2010 under Forefront products (used for Windows 8.1 and earlier) • Windows Defender under Windows (used for Windows 10 and later). click Ok **Note:** Once it is done, force a manual Synchronization of the Software Update. This is required so the new updates are going to be synchronized.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

Set-CMSoftwareUpdatePointComponent -SiteCode $SiteCode -
ImmediatelyExpireSupersedence $True -AddUpdateClassification "Definition
Updates"
Start-Sleep 5
Set-CMSoftwareUpdatePointComponent -SiteCode $SiteCode -AddProduct
@('Forefront Endpoint Protection 2010', 'Windows Defender')
Start-Sleep 5
Sync-CMSoftwareUpdate -FullSync $True
Start-sleep 30

while ($true)
{
      $return = gwmi -Class SMS SupSyncStatus -Namespace
"root\sms\site_$SiteCode" | select LastSyncErrorCode, LastSyncState
      Write-Output $return
      if ($return.LastSyncState -ne 6702) {start-sleep 10 } else { break }
}
```

27.2. Installing Endpoint Protection Point

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Site System Roles
03. Right click \\SRV0002.classroom.intranet and click Add Site System Roles
04. On Add Site System Roles Wizard, General, click Next
05. Under proxy, click Next
06. Under Specify roles for this server, select Endpoint Protection point.
07. Under Configuration Manager warning message, click Ok. Once back, click Next
08. Under Endpoint Protection License Terms, click by checking this box, I acknowledge that I accept the License Terms and Privacy Statement and click Next
09. Under Specify Microsoft Active Protection Service membership type, select the level of participation that you want and click Next **Note:** More information about the Microsoft Active Protection membership type can be found http://go.microsoft.com/fwlink/?LinkID=626987
10. Under confirm the settings, click Next
11. Under You have successfully completed the Add Site System Roles wizard with the following **sett**ings click close
12. Click Monitoring
13. Expand System Status and click Component Status
14. Search for ENDPOINT
15. Right Click SMS_ENDPOINT_PROTECTION_CONTROL_MANAGER, Show Messages and click All
16. Under Status Messages: Set Viewing Period, click OK
17. Verify the existence of Message ID 1013, 1014, 1015 and 500

18. Once installed, the System Center Endpoint Protection client is also installed with Real-time protection disabled.

Note: Depending on the server performance, it may take a while for the Latest update definition be applied.

19. You can also review the following logs:

- C:\ConfigMgr\Logs\EPSetup.log: Provides information about the installation of the Endpoint Protection site system role.
- C:\ConfigMgr\Logs\EPMgr.log: Monitors the status of the Endpoint Protection site system role.
- C:\ConfigMgr\Logs\EPCtrlMgr.log: Records details about the synchronization of malware threat information from the Endpoint Protection role server into the Configuration Manager database.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

if ((Get-CMSiteSystemServer -SiteSystemServerName "$servername") -eq $null)
{ New-CMSiteSystemServer -SiteCode $SiteCode -UseSiteServerAccount -
ServerName $servername }

Add-CMEndpointProtectionPoint -ProtectionService BasicMembership -
SiteSystemServerName "$ServerName" -SiteCode $SiteCode
start-sleep 90

while ($true)
{
      $component = gwmi -Namespace ("root\sms\site $SiteCode") -query
"select stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_ENDPOINT_PROTECTION_CONTROL_MANAGER' and stmsg.MessageID = 1013 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_ENDPOINT_PROTECTION_CONTROL_MANAGER 1013
id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
      $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query
"select stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS_ENDPOINT_PROTECTION_CONTROL_MANAGER' and stmsg.MessageID = 1014 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_ENDPOINT_PROTECTION_CONTROL_MANAGER 1014
id's"
        break
    } else { Start-Sleep 10 }
}
```

```
while ($true)
{
      $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query
"select stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS ENDPOINT PROTECTION CONTROL MANAGER' and stmsg.MessageID = 1015 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS_ENDPOINT_PROTECTION_CONTROL_MANAGER 1015
id's"
        break
    } else { Start-Sleep 10 }
}

while ($true)
{
      $component = gwmi -Namespace ("root\sms\site_$SiteCode") -query
"select stmsg.* from SMS_StatMsg stmsg where stmsg.Component =
'SMS ENDPOINT PROTECTION CONTROL MANAGER' and stmsg.MessageID = 500 and
stmsg.SiteCode = '$SiteCode'"
    if ($component -ne $null)
    {
        Write-Host "Found SMS ENDPOINT PROTECTION CONTROL MANAGER 500 id's"
        break
    } else { Start-Sleep 10 }
}
```

27.3. Automating the definition updates delivery

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Software Updates, and click Automatic Deployment Rules
03. Select Deployment Rules and click Create Automatic Deployment Rule
04. Under Specify the settings for this automatic deployment rule use the following information: • Name: ADR-Definition Updates • Template: Definition Updates • Collection: All Desktop and Server Clients Click Next
05. Under Specify the settings for this Automatic Deployment Rule select Detail level as All Messages and click Next
06. Under Select the property filters and search criteria, click Next
07. Under Specify the recurring schedule for this rule, click Next
08. Under Configure schedule details for this deployment select Time based on Client local time and click Next
09. Under Specify the user experience for this deployment, click Next
10. Under Specify software update alert options for this deployment, click Next
11. Under Specify the software updates download behaviour for clients on slow boundaries, click Next

12. Under Select Deployment Package for this automatic deployment rule, use the following information:

- Create a new deployment package: selected
- Name: ADR-Definition Updates
- Package Source: \\SRV0001\wsusDownloadContent\ADR-DefUpdates

Click Next

13. Under Specify the distribution points or distribution points groups to host the content, click Add Distribution Point Group, under Add-Distribution Point Groups, select Training Lab, and click Ok. Once back, click Next

14. Under Specify download location for this Automatic Deployment Rule, click Next

15. Under Select Language for which software update files are downloaded, click Next

16. Under Confirm the settings, click Next

17. Under The Create Automatic Deployment Rule Wizard completed successfully, click Close

18. Select the ADR-Definition Updates and click Run Now

19. You can also review the following log:

- C:\ConfigMgr\Logs\ruleengine.log: Records details about automatic deployment rules for the identification, content download, and software update group and deployment creation.

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"
$ADRName = "ADR-Definition Updates"

$pkg = New-CMSoftwareUpdateDeploymentPackage -Name "$ADRName" -Path
\\SRV0001\wsusDownloadContent\ADR-DefUpdates
Start-CMContentDistribution -DeploymentPackageId $pkg.PackageID -
DistributionPointGroupName "Training Lab"
New-CMSoftwareUpdateAutoDeploymentRule -CollectionName "All Desktop and
Server Clients" -Name "$ADRName" -AddToExistingSoftwareUpdateGroup $True -
AllowSoftwareInstallationOutsideMaintenanceWindow $True -AvailableTime 1 -
AvailableTimeUnit Hours -DeadlineImmediately $True -DeploymentPackageName
"$ADRName" -DeployWithoutLicense $True -DownloadFromInternet $True -
DownloadFromMicrosoftUpdate $True -EnabledAfterCreate $True -
GenerateSuccessAlert $True -RunType
RunTheRuleAfterAnySoftwareUpdatePointSynchronization -UserNotification
HideAll -UseUtc $False -VerboseLevel AllMessages -UpdateClassification
@("Definition Updates") -Language @("English") -LanguageSelection
@("English") -Product @("Windows Defender", "Forefront Endpoint Protection
2010")
start-sleep 5
Invoke-CMSoftwareUpdateAutoDeploymentRule -Name "$ADRName"
```

27.4. Changing Default Client Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin

01. Start Configuration Manager Console and Click Administration.

02. Click Client Settings

03. Select the default client settings and click Properties
04. Under Default Settings, click Endpoint Protection. Confirm that the following settings are configure: • Manage Endpoint Protection client on client computers: Set to Yes • Install Endpoint Protection client on client computers: Set to Yes • Disable alternative sources (such as Microsoft Windows Update, Microsoft Windows Server Update Services, or UNC share) for the initial definition updates on client computers: Set to No Click Ok **Note:** Force the machine policy refresh on WKS0001, WKS0002 and WKS0004 for the Endpoint Protection client installation start and wait the definition updates before testing the malware activity
05. You can also review the following client log: • C:\Windows\CCM\Logs\EndpointProtectionAgent.log: Records details about the installation of the Endpoint Protection client and the application of antimalware policy to that client.

This can also be achieved via PowerShell using the commands below:

```
Set-CMClientSetting -EndpointProtection -Name "$ClientSettingsName" -Enable $True -InstallEndpointProtectionClient $true -DisableFirstSignatureUpdate $False
```

27.5. Testing Malware activity

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Once the User Policy Retrieval & Evaluation cycle message appears, click Ok twice. **Note:** Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Navigate to \\srv0001\TrainingFiles\Source\Eicar and open eicar test file.txt file
05. The Anti-virus will detect this a malware file and will deny access. Once it happens, click Ok to the warning message
06. Open Windows Defender and click History
07. Select All detected items and click View details

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
```

```
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
Start-Sleep 60

#open notepad
Start-Process -Filepath ('notepad.exe') -ArgumentList
("\\srv0001\TrainingFiles\Source\Eicar\eicar test file.txt") -wait

#get info about malware
Get-MpThreat
```

27.6. Monitoring Malware Activity via Console

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Click Device Collections
03. Select Windows 10 Workstation collection and click Properties
04. Under Windows 10 Workstation Properties, change to the Alerts tab
05. Under Alerts tab, click View this collection in the Endpoint Protection dashboard and click Ok
06. Click Monitoring
07. Expand Security, Endpoint Protection Status and click System Center Endpoint Protection Status
08. Confirm that there is a malware activity identified. Click Malware Detected **Note:** Running Summarization may be needed
09. The list of identified malwares appears. Select it on the list and click Files Modified **Note:** Running Summarization may be needed
10. Under Files Modified is possible to review list of files that the Endpoint Protection identified

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$dkserver = "SRV0002.classroom.intranet"

$Collection = Get-CMDeviceCollection -Name "Windows 10 Workstations"
Invoke-WmiMethod -path
"\\.\root\sms\site_$($SiteCode):SMS_Collection.CollectionID='$($Collection.
CollectionID)'" -Name UpdateVisibilityInEPDashBoard -ArgumentList @(1)
start-sleep 5
Invoke-CMEndpointProtectionSummarization
start-sleep 10
Get-CMDetectedMalware -CollectionName "$($Collection.Name)"
```

27.7. Monitoring Malware Activity via Reports

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Expand Reporting, Reports and click Endpoint Protection
03. Select Antimalware activity report and click Run

04. Under Antimalware activity report, fill up the parameters (you may use the Values link) and click View report

Note: You can drill down to a more specific report using the links inside the reports

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

#Open Report
Invoke-CMReport -ReportPath "Endpoint Protection/Antimalware activity
report" -SiteCode "$SiteCode" -SrsServerName "$servername"
```

28. Compliance Settings

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002
More information	Get started with compliance settings in System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/compliance/get-started/get-started-with-compliance-settings Common tasks for managing compliance on devices with the System Center Configuration Manager client https://docs.microsoft.com/en-us/sccm/compliance/plan-design/common-tasks-for-managing-compliance-on-devices-with-the-client
Description	In this chapter, we will look

28.1. Changing Default Client Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Client Settings
03. Select the default client settings and click Properties
04. Under Default Settings, click Compliance Settings and then, set the schedule evaluation to 1 day. Click Ok

This can also be achieved via PowerShell using the commands below:

```
$Schedule = New-CMSchedule -RecurCount 1 -RecurInterval Days

$ClientSettingsName = "Default Client Agent Settings"
Set-CMClientSetting -Compliance -Name "$ClientSettingsName" -
EnableComplianceEvaluation $true -Schedule $Schedule
```

28.2. Registry Configuration Items with Auto-remediation

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Asset and Compliance.
02. Expand Compliance Settings and click Configuration Items
03. Select Compliance Settings and click Create Configuration Item
04. Under Specify general information about this configuration item type Internet Explorer Default Start Page under name and under Specify the type of configuration item that you want to create select Windows Desktops and Servers (custom) that is below Settings for devices managed with the Configuration Manager client. Click Next

05. Under specify the version of Windows that will assess this configuration for compliance, keep only All Windows 10 (64-bit) selected and click Next
06. Under Specify settings for this operating system click New
07. Use the following information: • Name: IE Start Page • Settings type: Registry value • Data Type: String • Hive Name: HKEY_CURRENT_USER • Key Name: Software\Microsoft\Internet Explorer\Main • Value Name: Start Page • Select This registry is associated with a 64-bit application **Note:** As this setting validates a 64-bit application and not a 32-bit application in a 64-bit machine (not under Wow6432 registry), selecting the "This registry is associated with a 64-bit application" is required) **Note:** This setting has been configured as a User setting and will be evaluated only when user is logged on
08. Change to Compliance Rules tab and click New
09. On Specify rules to define compliance conditions for this setting, type Start Page Must Exist in the Name, select Existential as Rule Type, select Registry value must exist on client devices and select Critical with Event on Noncompliance severity for reports. click Ok
10. click New
11. On Specify rules to define compliance conditions for this setting, type Start Page Equals http://www.tucandata.com in the Name, on the following values: type http://www.tucandata.com, select Remediate noncompliant rules when supported and Report noncompliance if this setting instance is not found and select Critical with Event on Noncompliance severity for reports. click Ok twice
12. Once back to the specify settings for this operating system, click Next
13. Under specify compliance rules for this operating systems, confirm that 2 rules have been already created and click Next
14. Under the wizard will create an operating system configuration item with the following settings, click Next
15. Under The create Configuration Item Wizard completed successfully, click Close

This can also be achieved via PowerShell using the commands below:

```
. c:\TrainingFiles\Scripts\Add-TDCMComplianceSetting.ps1

New-CMConfigurationItem -Name "Internet Explorer Default Start Page" -
CreationType WindowsOS | Add-TDCMComplianceRegistrySetting -SettingName "IE
Start Page" -RegRootKey CurrentUser -RegKey "Software\Microsoft\Internet
Explorer\Main" -RegKeyValueName "Start Page" -CreateExistentialValidation -
CreateMustExistValidation -CreateValueValidation -ValidationValue
"http://www.tucandata.com" -RemediateNonCompliant -ReportNonCompliant
```

28.3. Application Settings Configuration Items without Auto-remediation

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Asset and Compliance.
02. Expand Compliance Settings and click Configuration Items
03. Select Compliance Settings and click Create Configuration Item
04. Under Specify general information about this configuration item type Validate Mozilla Firefox Version under name and under Specify the type of configuration item that you want to create select Windows Desktops and Servers (custom) and This configuration item contains application settings that are below Settings for devices managed with the Configuration Manager client. Click Next
05. Under specify how this application is detected on client devices select Always assume application is installed and click Next
06. Under Specify settings for this operating system click New
07. Use the following information: • Name: Mozila Firefox CurrentVersion • Settings type: Registry value • Data Type: String • Hive Name: HKEY_CURRENT_USER • Key Name: Software\Mozilla\Mozilla Firefox • Value Name: CurrentVersion • Select This registry is associated with a 64-bit application **Note:** As this setting validates a 64-bit application and not a 32-bit application in a 64-bit machine (not under Wow6432 registry), selecting the “This registry is associated with a 64-bit application” is required) **Note:** This setting has been configured as User setting and will be evaluated only when user is logged
08. Change to Compliance Rules tab and click New
09. On Specify rules to define compliance conditions for this setting, type Mozila Firefox CurrentVersion must exist in the Name, select Existential as Rule Type, select Registry value must exist on client devices and select Critical with Event on Noncompliance severity for reports. click Ok
10. click New
11. On Specify rules to define compliance conditions for this setting, type Mozila Firefox CurrentVersion Equals 49.0.1 (x64 en-GB) in the Name, on the following values: type 49.0.1 (x64 en-GB) and select Report noncompliance if this setting instance is not found and select Critical with Event on Noncompliance severity for reports. click Ok twice
12. Once back to the specify settings for this application, click Next
13. Under specify compliance rules for this application, confirm that 2 rules have been already created and click Next
14. Under specify the client operating system that will assess this configuration item for compliance, keep only All Windows 10 (64-bit) selected and click Next
15. Under the wizard will create an operating system configuration item with the following settings, click Next
16. Under The create Configuration Item Wizard completed successfully, click Close

This can also be achieved via PowerShell using the commands below:

```
. c:\TrainingFiles\Scripts\Add-TDCMComplianceSetting.ps1

New-CMConfigurationItem -Name "Validate Mozilla Firefox Version" -
CreationType WindowsApplication | Add-TDCMComplianceRegistrySetting -
SettingName "Mozila Firefox CurrentVersion" -RegRootKey CurrentUser -RegKey
"Software\Mozilla\Mozilla Firefox" -RegKeyValueName "CurrentVersion" -
CreateExistentialValidation -CreateMustExistValidation -
CreateValueValidation -ValidationValue "49.0.1 (x64 en-GB)" -
ReportNonCompliant
```

28.4. Creating Baselines

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Asset and Compliance.
02. Expand Compliance Settings and click Configuration Baselines
03. Select Compliance Baselines and click Create Configuration Baseline
04. Under Create Configuration Baseline type Workstation Baseline as Name and click Add -> Configuration Items
05. Add All Available configuration items and click OK twice

This can also be achieved via PowerShell using the commands below:

```
New-CMBaseline -Name "Workstation Baseline"

$ci1 = Get-CMConfigurationItem -name "Internet Explorer Default Start Page"
$ci2 = Get-CMConfigurationItem -name "Validate Mozalla Firefox Version"
Set-CMBaseline -Name "Workstation Baseline" -AddOSConfigurationItem
$ci1.CI_ID -AddRequiredConfigurationItem $ci2.CI_ID
```

28.5. Deploying Baselines

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Asset and Compliance.
02. Expand Compliance Settings and click Configuration Baselines
03. Select Workstation Baseline and click Deploy
04. Under Deploy Configuration Baselines, check Remediate noncompliant rules when supported and click Browse
05. Select Windows 10 Workstation and click Ok twice

This can also be achieved via PowerShell using the commands below:

```
$Name = "Workstation Baseline"
$schedule = New-CMSchedule -RecurCount 1 -RecurInterval Days
Start-CMBaselineDeployment -CollectionName "Windows 10 Workstations" -Name
"$Name" -EnableEnforcement $true -Schedule $schedule
```

28.6. Starting Validation Compliance Settings

Perform this task on the wks0001 virtual machine logged on as user01
01. Open and Close Internet Explorer. **Note:** Confirm the Start Page is not set as www.tucandata.com
02. Open the Configuration Manager Properties
02. Change to the Actions Tab, select Machine Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under Machine Policy Retrieval & Evaluation Cycle click Ok **Note:** Depending on the SCCM environment, the user policy retrieval & evaluation cycle can take few minutes
04. Change to Configurations tab. **Note:** click refresh button may be needed
05. Select the Workstation Baseline and click evaluate. **Note:** Repeat the process for the WKS0002 machine

This can also be achieved via PowerShell using the commands below:

```
$SMSCli = [wmiclass] "root\ccm:SMS_Client"
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000021}")
start-sleep 10
$SMSCli.TriggerSchedule("{00000000-0000-0000-0000-000000000022}")
Start-Sleep 60

$Baselines = gwmi -Namespace root\ccm\dcm -Class SMS DesiredConfiguration -
Filter "DisplayName = 'Workstation Baseline'"

$Baselines | % {
    $MC = [WmiClass]"root\ccm\dcm:SMS_DesiredConfiguration"
    $InParams = $mc.psbase.GetMethodParameters("TriggerEvaluation")
    $InParams.IsEnforced = $true
    $InParams.IsMachineTarget = $false
    $InParams.Name = "$($_.Name)"
    $InParams.Version = "$($_.Version)"
    $MC.InvokeMethod("TriggerEvaluation", $InParams, $null)
}
```

28.7. Monitoring Baselines from Client

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to Configurations tab
03. Select the Baseline you want to view the report and click View Report **Note:** You must be member of the local administrators group to visualize a Baseline report from the client
04. You can also review the following logs: • C:\Windows\ccm\logs\CIAgent.log: Records details about the process of remediation and compliance for compliance settings, software updates, and application management. • C:\Windows\ccm\logs\CIDownloader.log: Records details about configuration item definition downloads. • C:\Windows\ccm\logs\CITaskManager.log: Records information about configuration item task scheduling. • C:\Windows\ccm\logs\DCMAgent.log: Records high-level information about the evaluation, conflict reporting, and remediation of configuration items and applications. • C:\Windows\ccm\logs\DCMReporting.log: Records information about reporting policy platform results into state messages for configuration items. • C:\Windows\ccm\logs\DcmWmiProvider.log: Records information about reading configuration item from Windows Management Instrumentation (WMI).

28.8. Monitoring Baselines via SCCM Console

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Click Deployments.
03. Select the Workstation baseline deployment and click Run Summarization
04. Click Ok when the Configuration Manager confirmation message appear.
05. Select the Workstation baseline deployment and click View Status
06. Under compliant see all assets that are compliant with the baseline
07. Change to the Non-Compliant tab to see all assets in that state

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$Deployments = Get-CMDeployment -CollectionName "Windows 10 Workstations" |
where {$ .SoftwareName -eq "Workstation Baseline"}
$ID = $Deployments.AssignmentID
$Deployments | Invoke-CMDeploymentSummarization
start-sleep 10
```

```
gwmi -namespace root\sms\site $SiteCode -class
"SMS_DCMDeploymentNonCompliantAssetDetails" -Filter "AssignmentID = $($ID)
and (RuleSubState = 0)" | select AssetName, RuleName, RuleStateDisplay
```

28.9. Monitoring Baselines via Reports

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Monitoring.
02. Under monitoring, expand Reporting and click Reports
03. Search for Compliance and Settings. Select Summary compliance by configuration baseline and click Run
04. Once the report is open, you can navigate using the links. **Note:** You can drill down to a more specific report using the links inside the reports

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

#Open Report
Invoke-CMReport -ReportPath "Compliance and Settings Management/Summary
compliance by configuration baseline" -SiteCode "$SiteCode" -SrsServerName
"$servername"
```

28.10. Monitoring Baselines via Collections

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Assets and Compliance.
02. Expand Compliance Settings and click Configuration Baselines
03. Select Workstation Baseline and click Deployments
04. Select Windows 10 workstations and click Create New Collection -> Non-compliant
05. On Create Device Collection Wizard, click Next
06. On Membership Rules, select Use incremental updates for this collection and click Next
07. On Summary, click Next
08. On Completion, click Close
09. Click Device Collections
10. Select Workstation Baseline_Windows 10 Workstations_noncompliant Collection and click Show Members **Note:** Once the collection is created, there is a process to populate it and it may take a while. In this lab, wait 30 seconds or refresh it couple of times until you see Member Count increment to 2

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"
```

```
$Deployments = Get-CMDeployment -CollectionName "Windows 10 Workstations" |
where {$_.SoftwareName -eq "Workstation Baseline"}

$CollUpdate = New-CMSchedule -Start "01/01/2015 9:00 PM" -DayOfWeek
Saturday -RecurCount 1
$Collection = New-CMDeviceCollection -Name "Non-Compliant Machines " -
LimitingCollectionName "Windows 10 Workstations" -RefreshSchedule
$CollUpdate -RefreshType Both
Add-CMDeviceCollectionQueryMembershipRule -CollectionId
$Collection.CollectionID -RuleName "Non-Compliant Machines"  -
QueryExpression "select * from  SMS_R_System inner join
SMS_G_System_DCMDeploymentState on
SMS_G_System_DCMDeploymentState.ResourceID = SMS_R_System.ResourceId WHERE
BaselineID = '$($Deployments.ModelName)' AND CollectionID =
'$($Deployments.CollectionID)' AND ComplianceState = 3"
```

29. Role Based Access Control

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	Configure role-based administration https://docs.microsoft.com/en-us/sccm/core/plan-design/security/configure-security#BKMK_ConfigureRBA
Description	In this chapter, we will look

29.1. Creating Security Scope

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Security and click Security Scopes
03. Select Security Scopes and click Create Security Scope
04. Under Create and assign security scope type Application Administrator for Windows 10 Machines as Security scope name and click Ok

This can also be achieved via PowerShell using the commands below:

```
$SecurityScope = "Application Administrator for Windows 10 Machines"
New-CMSecurityScope -Name "$SecurityScope"
```

29.2. Using Security Scope

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Select Firefox 49 and Java8 and click Set Security Scope
04. Under Set Security scope for the selected securable objects, select Application Administrator for Windows 10 Machines and click Ok

This can also be achieved via PowerShell using the commands below:

```
$SecurityScope = "Application Administrator for Windows 10 Machines"

get-cmApplication -name "Firefox 49" | Add-CMObjectSecurityScope -Name
"$SecurityScope"
get-cmApplication -name "Java8" | Add-CMObjectSecurityScope -Name
"$SecurityScope"
```

29.3. Creating an Application Administrator

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Security and click Administrative Users
03. Select Administrative Users and click Add User or Group
04. Under Specify a user or group to add as a Configuration Manager administrative user and use the following: • User or group name: CLASSROOM\Workstation Admins • Assigned security roles: Application Administrator • Only the instances of objects that area assigned to the specified security scope or collections: Selected • Security scopes and collections: Application Administrator for Windows 10 Machines security scope and Windows 10 Workstations Collection. Click Ok

This can also be achieved via PowerShell using the commands below:

```
New-CMAdministrativeUser -Name "CLASSROOM\Workstation Admins" -RoleName
@("Application Administrator") -CollectionName @("Windows 10 Workstations")
-SecurityScopeName @("Application Administrator for Windows 10 Machines")
```

29.4. Testing new Security Rights

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Select the Configuration Manager Console and click Run as different user
02. Log on as workstationadmin
03. Once the console is open, notice that there are few missing nodes
04. Click Devices and notice that only two devices appear
05. Click Device Collections and notice that only Windows 10 Workstation and Workstation Baseline_Windows 10 Workstations_Noncompliant appears **Note:** The Workstation Baseline_Windows 10 Workstations_Noncompliant collection appear because it is limited by the Windows 10 workstations
06. Click Software Library
07. Expand Application Management and click Applications. **Note:** Only application associated with Security scope are visible.

This can also be achieved via PowerShell using the commands below:

```
$username = "CLASSROOM\workstationadmin"
$password = 'Pa$$w0rd' | convertto-securestring -AsPlainText -Force
```

```
$cred = new-object -typename System.Management.Automation.PSCredential -
argumentlist $username, $password

Start-Process
"C:\ConfigMgr\AdminConsole\bin\Microsoft.ConfigurationManagement.exe" -
Credential $Cred
```

30. Cloud Distribution Point (Windows Azure Integration)

Computers used in this Lab	ROUTER01 SRV0001 SRV0002 WKS0001 WKS0002
More information	Use cloud services with System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/understand/use-cloud-services Install cloud-based distribution points in Microsoft Azure for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/servers/deploy/configure/install-cloud-based-distribution-points-in-microsoft-azure
All Systems & All Users collection	Deploying anything to All Systems or All Users is not recommended and it goes against best practices. We are using these 2 collections for demonstration purposes only.
Description	In this chapter, we will look

30.1. Preparing the Environment

Before Starting this Module, you will need to:

1- Create a 7-Zip Application with the Following Information, Deploy and Install
- Specific Settings for this application select Manually specify
- Name: 7-Zip 16
- Deployment Type: Script
- Name: 7-Zip 16 Installation for Windows 8.1 and 10
- Content Location: \\srv0001\Trainingfiles\Source\7-zip
- Install program: "7z1604-x64.exe" /S
- Detection Rule:
  - Type: File: %ProgramFiles%\7-Zip
  - File or folder name: 7zFM.exe
  - This file or folder is associated with a 32-bit application on 64-bit systems not checked
  - Property: Version
  - Operator: Greater than or equal to
  - Value: 16.04
- Installation behaviour: Install for a system
- Logon requirement: Whether or not a user is logged on
- Requirements: Operating System, one of, All Windows 10 (64-bit), All Windows 8.1 (64-bit)

30.2. Windows Azure Calculator

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Internet Explorer and navigate to https://azure.microsoft.com/en-us/pricing/calculator/?scenario=full and use the following information: • Windows Virtual Machines: 1 A0 • Storage (Locally Redundant - LRS): 100GB • Bandwidth: 100GB Confirm the estimated price **Note:** If need access to the data for multiple regions, use Storage (Geo Redundant - GRS) and increase number of virtual machines per localization **Note:** Always calculate the worst scenario for Storage. If you may need 100GB, calculate as 100GB **Note:** Always calculate the worst scenario for bandwidth, for example, if you have 100 clients downloading 100 GB per month, calculate 10,000 GB per month

30.3. Windows Azure Subscription

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Create a Windows Azure Subscription via http://www.windowsazure.com

30.4. Changing Default Client Settings

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Client Settings
03. Select the default client settings and click Properties
04. Under Default Settings, click Cloud Services and change Allow Access to cloud distribution point to Yes. Click Ok

30.5. Windows Azure Self Signed Certificate

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Command Prompt
02. Type \\srv0001\trainingfiles\Source\makecert\makecert.exe -sky exchange -r -n "CN=clouddp.classroom.intranet" -pe -a sha1 -len 2048 -ss My C:\TrainingFiles\clouddpcert.cer
03. Open MMC
04. Click File->Add/Remote Snap-in
05. Under Add or Remove Snap-ins, select Certificates and click Add
06. Under Certificates snap-in select My user account and click Finish
07. Back to the MMC console, expand Certificates – Current User, Personal and click Certificates

08. Select the clouddp.classroom.intranet certificate and click All Tasks->Export
09. Under Welcome to the Certificate Export Wizard, click Next
10. Under Export Private Key, select Yes, export the private key, and click Next
11. Under export file format, click Next
12. Under Security, select password and type a password and click Next
13. Under file to export, type C:\TrainingFiles\clouddpcert.pfx and click Next
14. Under Completing the Certificate Export Wizard, click Finish
15. Once the The export was successful message appears, click Ok

30.6. Upload Certificate to Windows Azure

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Log in to the Windows Azure portal via http://manage.windowsazure.com
02. Click Settings and Manage Certificates
03. Click upload
04. Under Upload a management certificate, browse for C:\TrainingFiles\clouddpcert.cer and click Ok
04. Once the certificate is imported, make a note for the Subscription ID.

30.7. Adding Certificate to Enterprise Trusted store

Perform this task on the SRV0001 virtual machine logged on as sccmadmin
01. Open Group Policy Management
02. Expand Forest: classroom.intranet, Domains, classroom.intranet and click Group Policy Management
03. Right click Default Domain Policy and click Edit
04. On the Group Policy Management Editor, expand Computer Configuration, Policies, Windows Settings, Security Settings, Public Key Policies and click Trusted Root Certification Authorities
05. Right click Trusted Root Certification Authorities and click Import
06. Under Certificate Import Wizard, Welcome to the Certificate Import Wizard, click Next
07. Under File to Import navigate to \\srv0002\c$\TrainingFiles\clouddpcert.cer and click Next
08. Under Certificate Store, click Next
09. Under Completing the Certificate Import Wizard, click Finish
10. On the The import was successful message, click Ok **Note:** Once the Certificate has been imported into the Default Group Policy, perform a gpupdate /force on all WKS0001 and WKS0002 to have this information updated.

30.8. Cloud Distribution Point Creation

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Hierarchy Configuration and click Cloud Services

03. Select Cloud Distribution Point and click Create Cloud Distribution Point
04. Under Specify details for this cloud service, type the Subscription ID and click Browser
05. Under Management Certificate, select C:\TrainingFiles\clouddpcert.pfx (the exported certificate with Private Key) and click Open
06. Under Password type the password used to export the certificate and click Ok. Once back to the specify details for this cloud service, click Next
07. Under Specify additional details for this distribution point, select the Region where the Distribution Point will be located and import the Exported Certificate again. Take a note of the Service Name and click Next
08. Under Configure alerts for this distribution point, click Next
09. Under This wizard will create a new site system cloud service that has the following settings click Next
10. Under The create Cloud Distribution Point Wizard completed successfully, click Close
11. The Cloud Distribution Point provisioning will start. **Note:** The process will take up to 30 minutes and once it is done, the Status Description column will change to Provisioning completed
12. Click Monitoring
13. Expand System Status and click Component Status
14. Search for SMS_CLOUD_SERVICES_MANAGER
15. Right Click SMS_CLOUD_SERVICES_MANAGER, Show Messages and click All
16. Under Status Messages: Set Viewing Period, click OK
17. Verify the existence of Message ID 9405
18. Double click this message to see its details. Once done, click Ok
19. Verify the existence of Message ID 9403
20. Double click this message to see its details. Once done, click Ok
21. Verify the existence of Message ID 9406
22. Double click this message to see its details. Once done, click Ok
23. Verify the existence of Message ID 9407
24. Double click this message to see its details. Once done, click Ok
25. Verify the existence of Message ID 9404
26. Double click this message to see its details. Once done, click Ok
27. Verify the existence of Message ID 9420
28. Double click this message to see its details. Once done, click Ok
29. Verify the existence of Message ID 9408
30. Double click this message to see its details. Once done, click Ok
31. Verify the existence of Message ID 9409
32. Double click this message to see its details. Once done, click Ok
33. Once the status changes to Provisioning completed, Log on the Windows Azure Portal via http://manage.windowsazure.com and click All items. **Note:** There are two items created. The Cloud service and the Storage Account
34. You can also review the following logs: • C:\ConfigMgr\Logs\CloudMgr.log: Records details about the provisioning of content, collecting storage and bandwidth statistics, and administrator initiated actions to stop or start the cloud service that runs a cloud-based distribution point.

30.9. DNS Configuration

Perform this task on the SRV0001 virtual machine logged on as sccmadmin
01. Open DNS Manager
02. Expand SRV0001, Forwarded Lookup Zones and click classroom.intranet
03. Select classroom.intranet and click New Alias (CNAME)
04. Under new resource record type the following: • Alias name: clouddp • Fully Qualified domain name: <Service Name>.cloudapp.net **Note:** Change <Service Name> for the Service Name that appeared while creating a new Cloud Distribution Point

30.10. Distribute Content to Cloud DP

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management, Applications
03. Select 7-Zip 16 and click Distribute Content
04. Under Review selected content click Next
05. Under Review the content to distribute, click Next
06. Under Specify the content destination, click Add Distribution Point
07. Under Add-Distribution Points, select clouddp.classroom.intranet and click Ok. Once back to the Specify the content destination, click Next **Note:** The clouddp.classroom.intranet type is set to Cloud
08. Under Confirm the settings, click Next
09. Under The distribute content wizard completed successfully, click Close **Note:** Once done, monitor the distribution status under the Content Status Monitoring node
10. Log in to the Windows Azure portal via http://manage.windowsazure.com and click the Storage Account
11. Under the Storage, click Containers
12. Under the containers, click the container content-<Site Code><Package ID>
13. Under the content-<Site Code><Package ID> will show all files related and the size of each one

30.11. Deploying an Application

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Software Library.
02. Expand Application Management and click Applications
03. Select 7-Zip 16 and click Deploy

04. Under Specify general information for this deployment, click Browse (Collection) and select the Collection you want to deploy. Click Next **Note:** You can deploy to a Device as well as Users. In this example, we are using a User Collection - All Users
05. Under Specify the content destination, click Next
06. Under specify settings to control how this software is deployed, click Next **Note:** Action can be Install or Uninstall and Purpose can be Available or Required.
07. Under Specify the schedule for this deployment, click Next
08. Under Specify the user experience for the installation of this software on the selected devices, click Next
09. Under specify Configuration Manager and Operations Manager alert options, click Next
10. Under Confirm this setting for this new deployment click Next
11. Under The Deploy Software Wizard completed successfully, click Close

30.12. Installing an Application using Cloud DP

Perform this task on the wks0001 virtual machine logged on as user01
01. Open the Configuration Manager Properties
02. Change to the Actions Tab, select User Policy Retrieval & Evaluation Cycle and click Run now **Note:** Using this option will force the client to connect to the server and update its settings. By default, this happen every 60 minutes and can be changed under Client Settings -> Client Policy -> Client policy polling interval (minutes)
03. Under User Policy Retrieval & Evaluation Cycle click Ok
04. Open Software Center under Start -> Microsoft System Center -> Configuration Manager
05. Under Available Software, select 7-Zip 16 and click Install
06. If needed, click Installation Status tab to follow the installation process
07. You can also review the following logs: • C:\Windows\CCM\LocationServices.log: Records the client activity for locating management points, software update points, and distribution points. **Note:** while looking at this log the Distribution Point log line will start with: Distribution Point='https://clouddp.classrom.intranet

31. Backup via SCCM

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	Backup and recovery for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/protect/understand/backup-and-recovery SQL Server Backup Recommendations for Configuration Manager https://stevethompsonmvp.wordpress.com/2013/06/07/sql-server-backup-recommendations-for-configuration-manager/ The CD.Latest folder for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/core/servers/manage/the-cd.latest-folder
Description	In this chapter, we will look

31.1. Configuring Backup

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Site Configuration and click Servers and Sites
03. Select 001 – Training Lab site and click Site Maintenance
04. Under Site Maintenance, select Backup Site Server and click Edit
05. Under Backup Site Server Properties, check Enable this task and backup destination select \\srv0001\SCCMBackup. Click Ok three times

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

Set-CMSiteMaintenanceTask -SiteCode $SiteCode -MaintenanceTask
"BackupSiteServer" -DaysOfWeek
Sunday,Monday,Tuesday,Wednesday,Thursday,Friday,Saturday -Enabled $true -
BeginTime "02:00" -LatestBeginTime "05:00" -devicename \\srv0001\SCCMBackup
```

31.2. Starting Backup Manually

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Services console
02. Select SMS_SITE_BACKUP and click Start

This can also be achieved via PowerShell using the commands below:

```
Get-Service -Name "SMS_SITE_BACKUP" | Start-Service
```

31.3. Monitoring Backup

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Event Viewer
02. Expand Windows Logs and click Application
03. Verify the existence of the following Event IDs: • 5055 (Site Backup task is starting) • 6829 (SMS Writer is about to stop the ConfigMgr Services as part of the preparation for the Site backup) • 3197 (I/O is frozen on Database) • 3198 (I/O is resumed on Database) • 5056 (Site Backup is starting to copy the files from the snapshot) • 5057 (Site Backup has successfully completed copying the files from the snapshot) • 6833 (Site Backup task has completed successfully)
04. Open Windows Explorer and navigate to \\srv0001\SCCMBackup\001Backup. This folder will contain all files needed to restore a SCCM infrastructure. Note: SCCM will keep only the latest backup under this folder. To have a historical backup, a file called AfterBackup.bat can be created under C:\ConfigMgr\inboxes\smsbkup.box with the content like: xcopy "\\srv0001\SCCMBackup*.*" "\\AnotherServer\Folder\%DATE%\<SITE CODE>Backup*.*" /E /O /C
14. You can also review the following logs: • C:\ConfigMgr\Logs\smsbkp.log: Records details about the site backup activity.

This can also be achieved via PowerShell using the commands below:

```
Get-Eventlog -Newest 100 -LogName Application -Source "SMS Server" -After
(Get-Date).AddMinutes(-60) | where {$_.eventID -in (5055,
6829,3197,3198,5056,5057,6833)} | select EventID, Message, TimeGenerated |
sort-object TimeGenerated -Descending | format-list
```

32. Restore

Computers used in this Lab	ROUTER01 SRV0001 SRV0002
More information	Backup and recovery for System Center Configuration Manager https://docs.microsoft.com/en-us/sccm/protect/understand/backup-and-recovery
Description	In this chapter, we will look

32.1. Stop SCCM Site

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open Command Prompt (Run as Administrator)
02. execute preinst.exe /stopsite from C:\ConfigMgr\bin\x64\00000409
03. Once the command is execute successfully, all site services will be stopped

This can also be achieved via PowerShell using the commands below:

```
Start-Process -Filepath ("C:\ConfigMgr\bin\x64\00000409\preinst.exe") -
ArgumentList ('/stopsite') -wait
Start-Sleep 5
```

32.2. Delete SCCM Database

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Open SQL Server Management Studio
02. Connect to the SRV0002 server
03. Expand Databases
04. Select CM_001 and click Delete
05. Under delete object, make sure delete backup and restore history information for database and close existing connections are checked. Click Ok
06. Confirm the database has been deleted

This can also be achieved via PowerShell using the commands below:

```
$Server="SRV0002.classroom.intranet"
$dbName="CM_001"
[System.Reflection.Assembly]::LoadWithPartialName("Microsoft.SqlServer.SMO"
) | out-null
$SMOserver = New-Object ('Microsoft.SqlServer.Management.Smo.Server') -
argumentlist $Server
$SMOserver.Databases | select Name, Size,DataSpaceUsage, IndexSpaceUsage,
SpaceAva
if ($SMOserver.Databases[$dbName] -ne $null) {
      $smoserver.KillAllProcesses($dbname)
      $smoserver.databases[$dbname].drop()
}
```

32.3. Restore SCCM Site

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Execute splash.hta from \\srv0001\SCCMBackup\001Backup\CD.Latest
02. Under Microsoft System Center Configuration Manager, click Install
03. On Before You Begin, click Next
04. On Available Setup Options, click Recover a site and click Next
05. Under Site Server and Database Recovery Options select Recover the site database using the backup set of the following locations and type \\srv0001\SCCMBackup\001Backup. Click Next
06. Under Site Recover Information, click Next
07. Under Product Key, Click Next
08. Under Product License Terms, select all I accept the license terms and click Next
09. Under Prerequisite Downloads, select Use previously downloaded files and in path type \\srv0001\TrainingFiles\Source\SCCMCB\Redist and click Next
10. Under Site and Installation Settings, click Next
11. Under Database Information (Server Information), click Next
12. Under Database Information (File Location Information), click Next
13. Under Diagnostic and Usage Data, click Next
14. Under Settings summary, review the settings and click Next
15. The prerequisite check will validate the system. Once it is done, click Begin Install
16. Once the installation is completed, click Next. **Note:** The Site Reset takes less than 10 minutes, however, post-recovery tasks can take extra minutes to complete, depending on the size of the infrastructure.
17. Under finished, confirm some of the post-recovery actions that need be done by the administrator, click Close.
18. At the root of C-partition, multiple log files are created that tell the status of the installation: • ConfigMgrPrereq.log: Prerequisites review log • ConfigMgrSetup.log: site server installation log • ConfigMgrSetupWizard.log: installation wizard log

This can also be achieved via PowerShell using the commands below:

```
$inifile = @"
[Identification]
Action=RecoverPrimarySite

CDLatest=1

[Options]
ProductID=EVAL
SiteCode=001
SiteName=Training Lab
SMSInstallDir=c:\ConfigMgr
SDKServer=SRV0002.classroom.intranet
PrerequisiteComp=1
PrerequisitePath=\\srv0001\TrainingFiles\Source\SCCMCB\Redist
AdminConsole=0
JoinCEIP=0

[SQLConfigOptions]
SQLServerName=SRV0002.classroom.intranet
DatabaseName=CM 001
SQLSSBPort=4022
SQLDataFilePath=C:\SQLServer\MSSQL13.MSSQLSERVER\MSSQL\DATA\
SQLLogFilePath=C:\SQLServer\MSSQL13.MSSQLSERVER\MSSQL\DATA\

[CloudConnectorOptions]
CloudConnector=0
CloudConnectorServer=
UseProxy=0
ProxyName=
ProxyPort=

[SystemCenterOptions]
SysCenterId=noBrzNn4z23AIHXl/w4kkO1Cp5bc/pxa6q23N56BIUM=

[HierarchyExpansionOption]

[RecoveryOptions]
ServerRecoveryOptions=4
DatabaseRecoveryOptions=10
BackupLocation=\\srv0001\SCCMBackup\001Backup
"@

$inifile -replace "`n", "`r`n"| Out-File -FilePath
"\\srv0001\TempFiles\restorecmcb.ini"

Start-Process -Filepath
("\\srv0001\SCCMBackup\001Backup\CD.Latest\SMSSETUP\BIN\X64\setup.exe") -
ArgumentList ('/script "\\srv0001\TempFiles\restorecmcb.ini"') -wait
```

32.4. Post-Restore Tasks

32.4.1. Accounts Password Reset

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Expand Security and click Accounts
03. Select CLASSROOM\svc_sccmna and click Properties
04. On CLASSROOM\svc_sccmna Properties, click Set
05. On CLASSROOM\svc_sccmna Windows User Account, type: • Password: Pa$$w0rd • Confirm Password: Pa$$w0rd Click Verify
06. Under verify type \\SRV0002\sms_site for Network Share and click Test Connection
07. Once the connection was successfully verified, click Ok three times **Note:** Repeat the password reset for the CLASSROOM\svc_ssrsea and CLASSROOM\svc_sccmpush

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"

$Secure = 'Pa$$w0rd'| ConvertTo-SecureString -AsPlainText -Force
$account = "CLASSROOM\svc_sccmna"
Set-CMAccount -UserName "$account" -Password $Secure -SiteCode $SiteCode

$account = "CLASSROOM\svc_ssrsea"
Set-CMAccount -UserName "$account" -Password $Secure -SiteCode $SiteCode

$account = "CLASSROOM\svc_sccmpush"
Set-CMAccount -UserName "$account" -Password $Secure -SiteCode $SiteCode
```

32.4.2. Distribution Point Self-Signed Certificate Reset

Perform this task on the SRV0002 virtual machine logged on as sccmadmin
01. Start Configuration Manager Console and Click Administration.
02. Click Distribution Points
03. Select SRV0002.CLASSROOM.INTRANET and click Properties
04. On SRV0002.CLASSROOM.INTRANET Properties, change the create self-signed certificate date/time and click Ok **Note:** Add a minute to the already existing time is enough to generate a new certificate

This can also be achieved via PowerShell using the commands below:

```
$SiteCode = "001"
$servername = "SRV0002.classroom.intranet"

$CurrentExpirationDate = [DateTime]::FromFileTime(((Get-CMDistributionPoint
-SiteSystemServerName "$servername").Props | where {$_.PropertyName -eq
"CertificateExpirationDate" }).Value1)

Set-CMDistributionPoint -SiteSystemServerName "$servername" -
CertificateExpirationTimeUtc
"$($CurrentExpirationDate.AddMinutes(1).ToString())"
```

33. Appendix A – Tools

33.1. DataExplorer

Data Explorer is a data visualization platform which helps enterprises to streamline the access control and management of strategic business information.

By displaying key metrics and indicators in one single screen, the software interface can be tailored and expandable to support particular objectives and needs.

More info at: http://www.tucandata.com

33.2. HealthCheck Toolkit

The Healthcheck tool supports you to analyse the health conditions of the Configuration Manager in an easy and practical manner.

Through the software, users can assess the status of the Configuration Manager's performance, latest updates, disk space, client data and other key indicators.

More info at: http://www.rflsystems.co.uk/software/healthcheck-toolkit/

33.3. CM12Automation

Configuration Manager 2012 Automation is a PowerShell project to help perform the basic implementation of a CM12 infrastructure.

More info at: http://cm12automation.codeplex.com/

33.4. ConfigMgrRegistrationRequest

ConfigMgrRegistrationRequest allows you to simulate a client using SCCM Client SDK.

More info at: https://configmgrregistratio.codeplex.com/

33.5. SCCM Client Center

The tool is designed for IT Professionals to troubleshoot SMS/SCCM Client related issues. The SCCM Client Center provides a quick and easy overview of client settings, including running services and SCCM settings in a good easy to use, user interface.

More info at: https://sourceforge.net/projects/smsclictr/

33.6. Mark Cochrane RegkeytoMof 3.3a

RegKeytoMof is used to quickly create custom Hardware Inventory entries formatted correctly for the sms_def.mof and configuration.mof files, when the target is Registry keys.

More info at: http://www.enhansoft.com/blog/how-to-use-regkeytomof

Download at: http://mnscug.org/images/Sherry/RegKeyToMOFv33a.zip

33.7. System Center Update Publisher 2011

System Center Updates Publisher is an application that enables independent software vendors or line-of-business application developers and IT administrators to import software update catalogs, create and modify software update definitions, export update definitions to catalogs, and publish software updates information to a configured update server. By using Updates Publisher to define software updates and publish them to the update server, administrators can begin detecting and deploying published updates with System Center Configuration Manager to client and server computers in their organization.

More info at: https://www.microsoft.com/en-gb/download/details.aspx?id=11940

33.8. OSD WebPortal

Complementary tool for SCCM that enables a couple of scenarios to simplify the OS staging process.

More info at: https://gallery.technet.microsoft.com/OSD-Webportal-10139926-e13f2d78

33.9. Recast RCT Free 2.4

Recast RCT drastically increases the usability and functionality of SCCM. The extension helps administrators access live and inventoried data and then take immediate action. Recast RCT Enterprise drastically increases the usability and functionality of SCCM. The extension helps administrators access live and inventoried data and then take immediate action. Recast RCT Enterprise drastically increases the usability and functionality of SCCM. The extension helps administrators access live and inventoried data and then take immediate action.

More info at: http://www.nowmicro.com/recast/#target-tab-1

33.10. RuckZuck

Software package manager, a quick way to install and update your Software.

More info at: http://ruckzuck.tools/

33.11. Clean Software Update Groups console extension for ConfigMgr

Console extension for ConfigMgr for cleaning up Software Update Groups automatically using PowerShell.

More info at: https://gallery.technet.microsoft.com/Clean-Software-Update-5ae68ba2

33.12. Cireson Remote Manage app

The Cireson Remote Manage app is the analyst's dream. It is a toolbox of common remote functionalities needed by an analyst. Its integration with Configuration Manager allows an analyst to quickly deploy software, run diagnostic processes, view and manipulate services, processes, and view information – all without end-user interaction! The Cireson Remote Manage app is designed around simple remote administration. It's an all-in-one toolbox, that doesn't interrupt the end user.

More info at: http://cireson.com/apps/remote-manage/

33.13. Reg2CI

Reg2CI is a Command-Line Tool to convert .POL (Policy) or .REG Files (Registry) into System Center Configuration Manager CI's (Configuration Items).

More info at: http://reg2ci.codeplex.com/

33.14. Collection Commander

The tool is designed for IT Professionals to trigger PowerShell Scripts on a list of devices. Collection Commander provides a bunch of PowerShell Scripts to manage and troubleshoot Configuration Manager 2012 Agents. But ConfigMgr is not a requirement. The tool can run as standalone solution.

It's possible to integrate own PS scripts by placing the .ps1 files in the "PSScripts" Directory. That makes it super easy to access your own PowerShell Script repository with a few clicks. The selected PS Script will run against all marked devices (mark the full row !) with multiple Threads so you can run it against hundreds of Devices.

More info at: http://cmcollctr.codeplex.com/

33.15. SQL Server Index and Statistics Maintenance

IndexOptimize is the SQL Server Maintenance Solution's stored procedure for rebuilding and reorganizing indexes and updating statistics. IndexOptimize is supported on SQL Server 2005, SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, and SQL Server 2014.

More info at: https://ola.hallengren.com/sql-server-index-and-statistics-maintenance.html

33.16. PowerShell – SQL Audit Script

Used to baseline SQL Server instance and find common mis-configurations. Document current settings, database files, database properties and much more. Script output is converted to an XLS for analysis.

More info at: https://stevethompsonmvp.wordpress.com/2014/05/19/powershell-sql-audit-script/

33.17. 1E's Free Tools

You can get cool tools, tips, white papers and scripts to simplify the daily life of the ConfigMgr Administrator. These free tools are focused on optimizing processes and resolving common issues in order to help you manage your Software Lifecycle Automation.

More info at: http://www.1e.com/free-tools/

33.18. ConfigMgr Task Sequence Monitor

ConfigMgr Task Sequence Monitor is an application that connects to your System Center Configuration Manager database to display data from task sequence deployments. It can be used to monitor running task sequences, such as OS deployments, or to review the results of historic task sequence deployments. If you are using MDT integration in ConfigMgr, you can view monitoring data both from MDT and ConfigMgr for your ZTI OS deployments.

More info at: https://smsagent.wordpress.com/tools/configmgr-task-sequence-monitor/

33.19. ConfigMgr OSD FrontEnd

ConfigMgr OSD FrontEnd has been developed with the goal to ultimately function in any environment easing some of the pains with operating system deployment that we have today. This version of the software has been created with the purpose of being initiated outside of the task sequence as a prestart command.

More info at: http://www.scconfigmgr.com/2017/01/02/configmgr-osd-frontend-public-preview-available/

33.20. Configuration Manager Web Frontend

Configuration Manager Web Frontend (CMFrontend) is a HTML5 web application designed to provide quick access to information, tools, and deployments without the use of the Configuration Manager console.

More info at: https://scottkeiffer.wordpress.com/2016/09/15/configuration-manager-web-frontend-update-2-0/

33.21. Windows-Nood OSD FrontEnd

The Windows-Nood OSD FrontEnd is a frontend for Configuration Manager which allows Network administrators/techs to use three common OSD scrnarios for UEFI (and Legacy) based hardware, namely Backup, Reinstall and New Computer.

More info at: https://www.windows-noob.com/forums/topic/11864-the-cm12-uefi-bitlocker-frontend-hta-part-1-the-features/

34. Appendix B – Unmissable Sites

Site Address	Comments
https://www.rflsystems.co.uk	3rd Party tools and Consultancy
http://www.thedesktopteam.com	MVP Raphael Perez and MVP David Nudelman
http://blog.colemberg.ch	MVP Mirko Colemberg
http://www.dekeukelaere.com http://www.scug.be/tim	MVP Tim De Keukelaere
http://www.ronnipedersen.com	MVP Ronni Pedersen
http://sccm.biz	MVP Nicolai Henriksen
http://Stevethompsonmvp.wordpress.com	MVP Steve Thompson
https://rzander.azurewebsites.net/	MVP Roger Zander
http://sms-hints-tricks.blogspot.com/	MVP Matthew Hudson
http://faqshop.com/	MVP Cliff Hobbs