

# WINDOWS SERVER 101

## Abstract

Aimed as an introduction to Windows Server, this book provides an overview of administration and management for Windows Server operating systems, without any prerequisite knowledge of Windows Server necessary.

Brian Svidergol  
December 2016

# Introduction

Maybe you work in a call center, for a helpdesk team, as a desktop support technician. Or, maybe you are a database administrator or a network guy. Or a developer. And you want to learn more about server administration, especially with Windows Server. With the proliferation of DevOps (combining server administration skills and development skills), having working knowledge of Windows Server is important. While the public cloud makes it easy to deploy a new server, you must know how to configure it, maintain it, and troubleshoot it. I wrote this book as a first step. You won't become an expert just by reading this book and working through the exercises and labs. But you will be on your way!

# Audience

Anybody that is on a quest to learn about Windows Server or managing servers. If you are already an expert with Windows Server, then this book probably isn't for you (although you might pick up a few things here and there). If you are looking to learn all about the latest features with Windows Server 2016, then this book isn't for you (instead, I opt to cover the most deployed components of Windows Server at the time of this writing). Many of the newest features require foundational knowledge about Windows Server (which this book can give you). If you suddenly find yourself managing Windows Server and you don't have much experience, this book is for you. If you are working in IT and want to move up (or move to) working with servers, then this book is for you.

# About the Author

Brian Svidergol specializes in Microsoft infrastructure and cloud-based solutions built around the Windows operating system, Active Directory, Microsoft Azure, and Office 365. He holds a bunch of Microsoft and industry certifications including Microsoft Certified Trainer (MCT) and Microsoft Certified Solutions Expert (MCSE Server Infrastructure). Brian has spent time working as a consultant, working in startup companies, and working in large enterprise organizations. Brian has authored books on Active Directory, Windows Server, Exchange Server, and virtualization. He also has worked as a Subject Matter Expert and technical reviewer on many Microsoft Official Curriculum courses, Microsoft certification exams, and authored or reviewed related training content.

# Acknowledgements

Brian would like to thank his family Lindsay (wife), Jack (son), and Leah (daughter) for their continued support and love. He would also like to thank Charles Pluta and Bob Clements for their ongoing help with whatever needs to be done. Right at the end of the project, Charles helped finalize the organization, layout, and formatting (which was a big help). Thanks for Santos Martinez for being so patient while I wrap up projects! Thanks to Elias Mereb for agreeing to meet me in Florida even though it isn't close to his house! Finally, thanks to Frances Lefkowitz who was originally the development editor for this book. She was great to work with and helped shape this book in many ways.

---

# Windows Server 101

<b>Chapter 1: Before you Begin .....</b>	<b>16</b>
What is Windows Server?.....	16
Is this book for you?.....	20
How to use this book.....	20
Setting up the lab environment.....	22
Hit the Ground Running!.....	23
<b>Chapter 2: A Peek into Windows .....</b>	<b>24</b>
Client vs. Server.....	25
Hardware vs. Software.....	26
Four core hardware components.....	27
CPU .....	28
Memory.....	28
Storage.....	30
Network interface card (NIC) .....	34
The makeup of the software components .....	34
Fundamental networking concepts for server administrators .....	35
Physical servers vs. virtual servers.....	37
Workgroup vs. domain .....	37
How security fits with Windows Server.....	38
Windows Server in the cloud.....	40
<b>Chapter 3: Introduction to Server Manager .....</b>	<b>42</b>
What you can do with Server Manager .....	42
The dashboard .....	43
Managing multiple servers.....	45
Configuring local server settings.....	48
Computer name.....	48

Domain or Workgroup.....	49
Windows firewall .....	51
Remote management.....	51
Remote Desktop.....	52
Ethernet.....	53
Windows Update.....	54
Windows Server Antimalware.....	55
IE Enhanced Security Configuration.....	56
Controlling Server Manager startup behavior .....	57
Lab .....	58
Add servers to Server Manager .....	58
Create a server group.....	58
Disable IE ESC for administrators.....	58
Configure Server Manager startup behavior.....	58
<b>Chapter 4: Getting started with PowerShell.....</b>	<b>59</b>
How to work with PowerShell? .....	60
Windows PowerShell syntax .....	62
Commands vs. cmdlets .....	64
Stringing commands together.....	64
Symbols in PowerShell.....	65
Common administrative commands to manage a server.....	66
The PowerShell execution policy.....	67
Other commands for managing a server.....	68
Working with Windows hotfixes .....	68
Troubleshooting network connectivity .....	69
Other server management commands.....	70
Working with the pipeline .....	71
Using help in PowerShell.....	74
Lab .....	75
Identify PowerShell command parts.....	75
Work with the execution policy .....	76

Perform common administrative tasks with PowerShell.....	76
Working with the pipeline .....	76
Getting help .....	76
<b>Chapter 5: Adding server roles and features .....</b>	<b>78</b>
Overview of server roles and features.....	78
Adding roles and features in the GUI.....	80
Working with roles and features by using PowerShell .....	82
Removing server roles and features .....	84
Working with source files and Features on Demand .....	85
Lab .....	87
Add a role by using Server Manager.....	87
Remove a role by using Server Manager.....	87
Add the .NET Framework 3.5 Features feature .....	87
Uninstall the source files for a role.....	87
<b>Chapter 6: Networking Fundamentals .....</b>	<b>88</b>
Working with Subnets.....	88
Ideas for on your own.....	95
Fundamentals of Protocols and Ports .....	95
NIC Teaming .....	97
Securing Network Communications .....	101
When to secure network communications.....	103
Methods of securing network communications.....	104
Lab .....	105
Understanding subnets.....	105
Configuring NIC teaming .....	105
Identify the ports.....	105
Secure network communications.....	105
<b>Chapter 7: Managing a DHCP Server Role .....</b>	<b>106</b>
Using the DHCP management console .....	106
Creating DHCP scopes .....	107
Viewing the DHCP configuration .....	112

View the configuration and current client activity for a scope.....	113
Viewing the configuration for the server .....	113
Backing up and restoring DHCP.....	114
DHCP Relay Agents .....	115
Add the Remote Access role and required role services .....	116
Add the DHCP relay agent.....	117
Using PowerShell to manage a DHCP server .....	117
Lab .....	120
Create and configure a new DHCP scope in the DHCP management console.....	120
Create a DHCP reservation in the DHCP management console .....	120
Create a DHCP reservation by using PowerShell .....	120
Backup the DHCP configuration by using PowerShell.....	120
Restore the DHCP configuration by using PowerShell.....	120
<b>Chapter 8: Managing client name resolution .....</b>	<b>121</b>
How names are resolved.....	121
Name resolution methods.....	124
Understanding DNS client cache .....	126
Troubleshooting client name resolution issues.....	129
Developer reports that he can't get to www.tailspintoys.com .....	129
User reports different name resolution than others.....	130
IT Admin reports stale name resolution.....	131
Summary .....	132
Lab .....	133
Preload the DNS client cache .....	133
Use PowerShell to view DNS client cache.....	133
True or false questions.....	133
Fill in the blanks.....	133
<b>Chapter 9: Managing the DNS Server Role.....</b>	<b>134</b>
Using the DNS Management Console .....	134
Working with the DNS Server service .....	139
Working with the server cache .....	141

Understanding resource records.....	142
Root servers and forwarders.....	143
Root servers.....	143
Forwarders.....	144
Conditional forwarders .....	145
All About Scavenging .....	148
How to turn scavenging on.....	150
Lab .....	151
Clear the DNS Server cache .....	151
Create two records for DNS Round Robin .....	152
Confirm the validity of the Root Hints file.....	152
Configure DNS scavenging .....	152
<b>Chapter 10: Working with DNS Zones .....</b>	<b>153</b>
Primary, Secondary, and Stub Zones.....	153
Primary zones .....	153
Secondary zones.....	154
Stub zones .....	157
Standard vs. Active Directory integrated zones .....	161
Standard zone file storage.....	161
Active Directory integrated zone storage.....	162
Secure dynamic updates .....	163
Lab .....	164
Create and modify an Active Directory integrated primary DNS zone .....	164
Create and modify a standard primary DNS zone.....	164
Reconfigure a zone's replication scope.....	165
Update a zone's data by using Notepad.....	165
<b>Chapter 11: Troubleshooting network communications .....</b>	<b>166</b>
Troubleshooting overview .....	166
Troubleshooting network communication with Ping.....	166
Troubleshoot network connectivity with PowerShell .....	168
Testing connectivity to specific ports.....	169

Testing connectivity to specific ports using PowerShell .....	171
Reviewing the IP configuration of a computer.....	173
Reviewing the IP configuration of a computer with PowerShell.....	173
Tracing the network route .....	174
Troubleshooting name resolution.....	175
Troubleshooting name resolution with PowerShell.....	176
Finding out where the email goes.....	178
Troubleshooting listening ports .....	179
Lab .....	181
Use Telnet to validate network connectivity.....	181
Use PowerShell to investigate network connectivity.....	182
Resolve names.....	182
Use Netstat to look at network connections and listeners .....	182
<b>Chapter 12: Active Directory Basic Administration .....</b>	<b>183</b>
Managing your environment remotely.....	183
Configuring remote management.....	184
Customizing your management environment.....	187
Working with Organizational Units .....	190
Creating OUs.....	190
Managing OUs .....	192
Protecting Objects from Accidental Deletion.....	193
Lab .....	195
Perform remote management .....	195
Customize an MMC .....	195
Work with OUs .....	195
Work with protecting objects from accidental deletion.....	196
<b>Chapter 13: Creating user accounts.....</b>	<b>197</b>
Creating a new user account .....	197
Creating user accounts with Active Directory Users and Computers.....	198
Creating user accounts with PowerShell .....	200
Modifying the properties of an account .....	201

Updating user accounts with Active Directory Users and Computers.....	202
Modifying AD DS attributes for user accounts.....	203
Modifying user accounts with PowerShell.....	204
Enabling and disabling an account .....	205
Enabling user accounts.....	205
Disabling user accounts.....	207
Creating a template for user creation.....	208
Lab .....	210
Create a new user account.....	210
Modify existing users.....	210
Enable and disable users.....	210
Work with user account templates.....	210
<b>Chapter 14: Managing user accounts.....</b>	<b>211</b>
Resetting Passwords .....	211
Unlocking User Accounts .....	213
Finding users that are locked out .....	214
Unlocking user accounts with ADUC and ADAC.....	215
Unlocking a user account by using PowerShell.....	215
Managing Stale Accounts .....	216
Finding stale user accounts in ADUC.....	216
Finding stale user accounts by using PowerShell .....	217
What to do with stale users after you find them .....	218
Deleting User Accounts .....	219
Deleting user accounts in ADUC .....	219
Deleting user accounts in PowerShell .....	221
Lab .....	221
Reset a password using PowerShell.....	221
Unlock all locked out users using PowerShell.....	221
Find stale accounts.....	222
Delete user accounts .....	222
<b>Chapter 15: Managing Active Directory Computer Objects .....</b>	<b>223</b>

Prestaging Computer Accounts.....	223
Prestaging a computer account using ADUC and ADAC.....	225
Prestaging a computer account using PowerShell.....	226
Joining a Computer to a Domain.....	227
Join a computer to the domain using the GUI .....	227
Join a computer to the domain using PowerShell .....	228
Computer Secure Channels.....	229
Check the health of secure channels using PowerShell .....	229
Resetting Computer Accounts .....	230
Resetting a computer account using ADUC .....	230
Resetting a computer account using PowerShell .....	231
Managing Stale Computer Objects.....	231
Finding stale computer accounts by using dsquery .....	232
Finding stale computer accounts by using PowerShell.....	233
What to do with stale computer accounts after you find them.....	233
Lab .....	234
Pre-stage 10 new computer accounts .....	234
Join a new computer to the domain.....	234
Reset a computer account.....	235
Check secure channel status.....	235
Find stale computer accounts .....	235
<b>Chapter 16: Managing Groups.....</b>	<b>236</b>
Group Types and Scopes .....	236
Group types.....	237
Group scopes.....	238
Default Groups.....	240
Adding Members to a Group .....	241
Add members to a group using ADUC and ADAC.....	241
Managing group membership with ADAC.....	242
Add members to a group using PowerShell.....	243
Removing Members from a Group .....	245

Lab .....	246
Group Types and Scopes .....	246
Default Groups.....	246
Adding Accounts to a Group.....	246
Removing Accounts from a Group.....	247
<b>Chapter 17: Working with Group Policy .....</b>	<b>248</b>
What is Group Policy? .....	248
Policies vs. Preferences .....	252
Creating GPOs.....	253
Creating GPOs using GPMC.....	253
Creating GPOs using PowerShell .....	254
Linking GPOs.....	254
Linking GPOs in GPMC.....	255
Link a GPO by using PowerShell.....	255
Blocking and Enforcing.....	256
Troubleshooting Group Policy .....	259
How to get around inheritance blocking.....	259
What to check when GPO settings aren't applying as you intended.....	260
How to minimize the need for inheritance blocking.....	261
Lab .....	261
Create and link a GPO by using the GUI .....	261
Create and link a GPO by using PowerShell .....	261
Block policy inheritance on a child OU .....	262
Enforce a GPO link.....	262
Disable the user configuration settings for a GPO.....	262
<b>Chapter 18: Securing Windows Server with Group Policy .....</b>	<b>263</b>
What are security settings? .....	263
Password and Lockout Policies .....	264
Standard password and account lockout policies.....	265
Password settings .....	265
Account lockout settings.....	267

Fine-grained password and account lockout policies.....	268
User Rights Assignment.....	270
Auditing Policies.....	272
Managing the Windows Firewall.....	275
Lab .....	277
Configure a fine-grained password policy .....	277
Configure user rights.....	278
Configure advanced audit policy settings .....	278
Configure the Windows firewall .....	278
<b>Chapter 19: Managing Windows Services.....</b>	<b>279</b>
Using the Services Management Console .....	279
Startup parameters.....	282
Log on for services.....	282
Service recovery options .....	282
Service dependencies.....	282
Starting and Stopping Services.....	283
Starting and stopping services in the management console.....	283
Starting and stopping services with PowerShell .....	284
Service Accounts .....	285
Choosing which account you should use for a service.....	286
Managed Service Accounts .....	288
Group Managed Service Accounts .....	288
Creating Group Managed Service Accounts.....	289
Lab .....	290
Using the Service Management Console.....	290
Starting and Stopping Services.....	290
Service Accounts .....	291
Group Managed Service Accounts .....	291
<b>Chapter 20: Working with Local Storage.....</b>	<b>292</b>
Manage Disks and Volumes.....	292
Managing Disks .....	293

RAID Levels.....	297
Creating a RAID 0 volume.....	298
Storage Spaces.....	301
Managing storage from the command line and Windows PowerShell.....	304
Diskpart.exe.....	304
Windows PowerShell .....	306
Lab .....	307
Initialize and format disks .....	307
Create a software RAID.....	307
Create Storage Spaces .....	307
Manage Storage Pools.....	307
<b>Chapter 21: Managing NTFS Permissions.....</b>	<b>308</b>
What are NTFS Permissions? .....	308
Denying permissions .....	311
Standard and Advanced Permissions.....	311
Ownership and the impact on permissions .....	312
Effective Access.....	314
Encrypting Files and Folders .....	316
Auditing Files and Folders .....	317
Usage scenarios for auditing .....	318
How to enable auditing at the server level .....	320
How to enable auditing at the folder level .....	320
Lab .....	321
Set permissions.....	322
Viewing effective permissions.....	322
Using EFS.....	322
Auditing files and folders.....	322
<b>Chapter 22: Managing Shared Folders and File Services.....</b>	<b>323</b>
Administrative Shares.....	323
Viewing shared folders.....	324
Share Permissions .....	325

Combining NTFS and Share Permissions.....	327
Access-Based Enumeration .....	329
Using Shadow Copies.....	331
The File Server Resource Manager.....	332
Quotas and Templates .....	333
File Screening .....	333
Storage Reports.....	334
File Classification and Management.....	335
Lab .....	336
Administrative Shares.....	336
Shared Folder Permissions .....	336
Combining NTFS and Share Permissions.....	336
Access-Based Enumeration .....	337
Using Shadow Copies.....	337
The File Server Resource Manager.....	337
<b>Chapter 23: Managing Print Services .....</b>	<b>338</b>
Installing Print Drivers .....	338
Installing print drivers using the Print Management console.....	339
Installing print drivers using the command line and PowerShell .....	340
Managing Printer Permissions .....	341
Managing Print Queues.....	343
Printer Pooling .....	345
Publish a printer in AD DS using the Print Management console.....	347
Publish a printer in AD DS using PowerShell.....	348
Lab .....	348
Installing Print Drivers .....	348
Managing Printer Permissions .....	349
Managing Printer Queues.....	349
Printer Pooling .....	349
Publishing Printers in Active Directory.....	349
<b>Chapter 24: What now?.....</b>	<b>350</b>

Ideas for further exploration.....	350
And here are the areas to explore .....	350
Other resources you'll grow to love.....	352

## CHAPTER 1: BEFORE YOU BEGIN

---

Windows Server is a version of Windows built to run applications and services for businesses, schools, and other organizations. This book focuses on how to perform many of the tasks you need to manage Windows Server, such as configuring a server's local settings, adding and managing key server roles, and troubleshooting network communications. The information presented in this book will help application administrators, developers, and other IT workers manage their servers. It is not meant as a guide to Windows Server 2016 for experts on previous versions of Windows Server. It is meant to teach you Windows Server, including the 2016 version and earlier versions, from scratch. For desktop administrators and other IT workers who focus on client operating systems or applications, this book is a gateway to the server side of IT. Based on your background and skill set, some chapters will be review while others present information on new topics. The goal is to provide a broad view of Windows Server and topics that teach you how to manage existing Windows servers. Thus, we won't cover the installation of Windows Server. This book shows some background and theory, but not much. Instead, it is geared toward the practical side, providing information and knowledge that can be applied immediately. It is essential that you work through all the hands-on exercises and labs. It is especially helpful if you work through a lab as soon as you read the preceding chapter.

Certain topics are large enough or go deep enough that URLs for additional material are included. These URLs are optional but may be useful, depending on your level of experience and needs.

This book will get you up to speed quickly, but it skips (or quickly covers) historical information and theory. The style is in contrast with 1,500-page books that dive into detail about every feature and configuration option of Windows Server. In such a huge book, many administrators get bogged down with information that isn't practical for his or her job. By the end of this book you will be an effective Windows Server administrator, able to manage one or more Windows servers, troubleshoot server issues, and secure Windows servers by using Group Policy. But there are other areas to explore and expand upon, depending on how far you want to go. We'll give you some ideas for continuing your Windows Server journey at the end of this book. Have fun!

### What is Windows Server?

Most people have hands-on experience with the Windows operating system, usually with a client such as Windows 7, Windows 8.1, or Windows 10. When the Windows server operating system, such as Windows Server 2012 R2 and Windows Server 2016, is running the graphical installation option, it is like the Windows client operating systems in look and feel. This helps new administrators learn Windows Server (at least the GUI portion). In Figure 1.1, a screen capture shows the desktop and start screen of Windows 10 Enterprise. Below that, in figure 1.2, a screen capture shows the desktop and start screen of Windows Server 2016. Both have a Start menu, a taskbar, and shortcuts to popular applications such as the Microsoft Edge browser. The Start menu in Windows 10 is more

consumer focused while the Start menu in Windows Server 2016 is primarily focused on server administration.

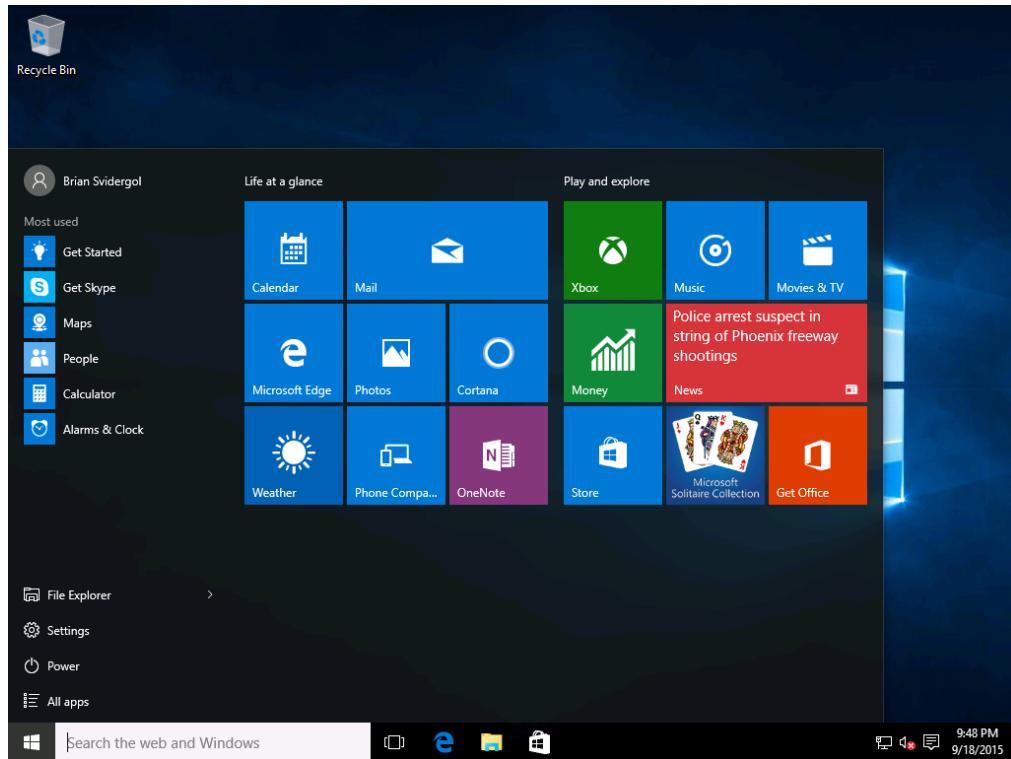


Figure 1.1 Windows 10 Enterprise desktop and Start screen

### Hands-on Exercise

Flip back and forth between the screen capture of the Windows 10 desktop and the Windows Server 2016 desktop to see if you can spot similarities and differences.

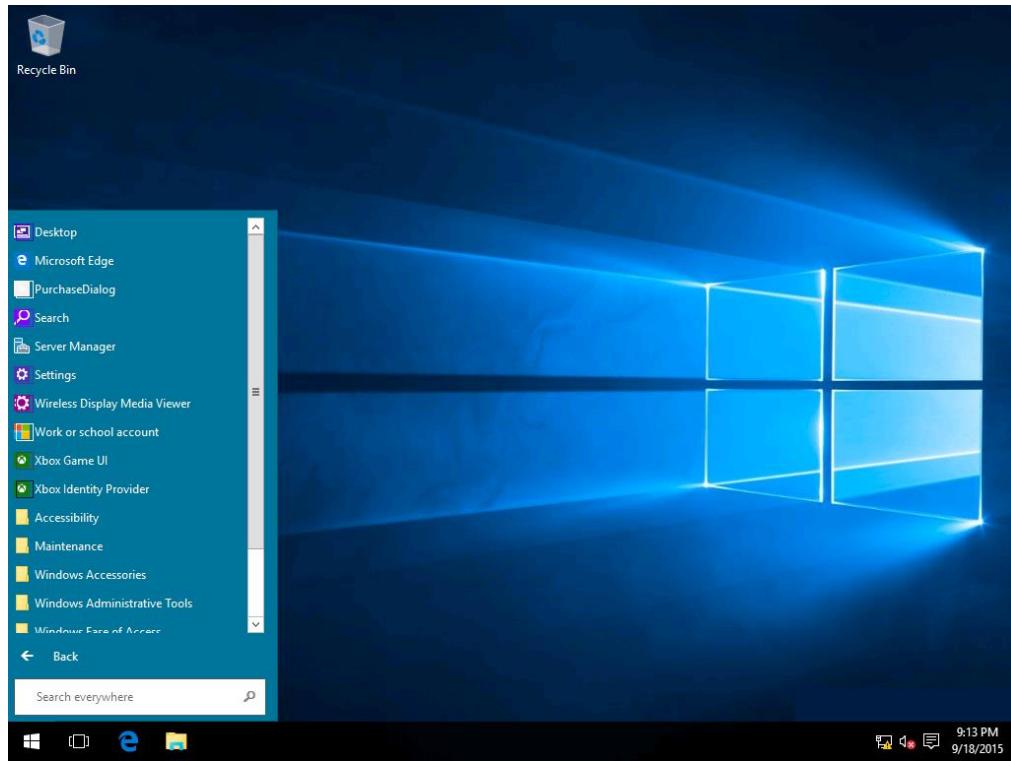


Figure 1.2 Windows Server 2016 desktop and Start screen

One of the key differences between the client and server version of Windows is servers are built to run in server rooms, data centers, and in the cloud (public or private). Before we go any further, let's define these.

- **Server rooms.** Server rooms are dedicated rooms in an office, building, or campus that house IT equipment such as servers, telephony equipment, and network devices. Server rooms are smaller than data centers and are often dedicated to a single customer. The smallest server rooms are like a wiring or telephony closet while the largest server rooms resemble small data centers. Small companies with less than 500 employees often have a server room or an equipment closet.
- **Data centers.** Data centers are large dedicated rooms in an office, building, or campus housing IT equipment such as servers, telephone equipment, and network devices. Data centers are much larger than server rooms. In some cases, data centers take up most of a building. Data centers can be dedicated to a single organization or can be a public data center housing multiple customers. In a public data center, customers are separated by cages, racks, or a similar physical separator. Public data center customers often have access to their physical computing equipment and can install new servers, perform maintenance on existing servers, and decommission servers. Large companies with more than one thousand employees often have a data center. Public data centers are normally located in major metropolitan areas.
- **Public cloud.** The public cloud is a type of data center. The primary difference is the data center can only be reached electronically, such as by remotely managing your IT equipment. Most often, customers of the public cloud do not have access to the associated data centers.

They cannot install, maintain, or decommission physical IT equipment in the public cloud. Instead, everything is virtualized and customers manage their resources by using a web browser or with a remote management tool such as Windows PowerShell. Companies of all sizes use the cloud. Small companies use it so they don't have to operate a server room or backend equipment on their premises. Large companies use the cloud to reduce the footprint of their data centers for cost reduction and to take advantage of the benefits of cloud computing such as increased scalability, reduced time to market, and flexible geographic data center locations.

The table below, Table 1.1, shows the characteristics of server rooms, data centers, and the public cloud.

Table 1.1 Characteristics of server rooms, data centers, and the public cloud

<b>Characteristic</b>	<b>Server room</b>	<b>Data center</b>	<b>Public cloud</b>
Gain physical access	Yes	Yes	No
Maximizes security	Sometimes	Often	Sometimes
Shared with other customers	Rarely	Sometimes	Often
Good for a small branch office	Often	Rarely	Often

Servers, instead of being focused on delivering an optimal end-user experience for applications (such as e-mail and web surfing), are optimized for data center workloads such as authenticating users and computing devices, hosting web sites and databases, and securely storing your data.

By default, Windows Server installs without the graphical user interface many people are accustomed to on client computers. File Explorer and Internet Explorer, two popular applications on Windows client computers, are not part of a default installation of Windows Server. The default installation is named the Server Core installation. When you sign into a Server Core installation of Windows Server, you are greeted by a command prompt and nothing more. At first, this can be overwhelming because the graphical user interface (GUI) is so familiar and easy to use but it is not part of the default installation. Most administrators rely on GUI and use a full installation of Windows Server, which includes the GUI. As administrators get more experience on Windows Server, they often give the Server Core installation another chance because it offers increased security and less downtime for Windows updates.

While the Windows client operating systems are best suited for running applications such as Microsoft Office, the Windows Server operating systems are best suited for running background processes and tasks, often called services. For example, a server that manages printers and print

jobs, called a print server, has specific print server services, such as the Spooler service, running in the background. Another example is an Active Directory domain controller that has two or more dedicated services, one of which authenticates users when they sign in to their computer and another that provides a web services interface to facilitate applications such as PowerShell interacting with Active Directory.

In IT, many workers start in a job supporting end users in a call center, as part of a help desk, or working as a desktop support technician. After mastering the support of end users and their computing devices, some IT workers want to expand their skills to include supporting data centers and servers. This is a natural progression for IT support roles. Developers and application administrators spend more time deploying and managing servers but need to expand their server management skills.

## Is this book for you?

This book is for those with limited or no experience but who want to learn how to manage and maintain Windows Server. When we envisioned the audience for this book, we were thinking about administrators currently working on desktop computers, developers that want to expand their server skills, and IT administrators that primarily work with other technologies such as databases. This book does not cover every facet of Windows Server. Instead, it focuses on the primary skills needed to successfully maintain Windows Server in your environment.

This book focuses on making an immediate impact on your ability to manage a Windows Server, diving right into server administration topics that cover widely used components of Windows Server, such as Windows PowerShell, DHCP, DNS, Active Directory, Group Policy, file services, and print services. If your goal is to pick up Windows Server skills that can be used immediately, this book is for you. If you are a seasoned veteran of Windows Server and are looking for an encyclopedic guide to Windows Server 2016, then this book is not for you.

## How to use this book

You should read this book from cover to cover, and read one chapter per day. Later chapters build on knowledge from earlier chapters, so reading sequentially will result in the best knowledge retention. The book was designed to be read one chapter per day for one month. If you can follow that schedule and have enough time to perform the labs at the end of each chapter, you will be ready to manage Windows servers within a month. After each chapter, you will be ready to manage the components taught. Throughout each chapter, you will see "Hands-on Exercise" callouts. These callouts are quick hands-on exercises you should work on while you read the chapter.

This book has 24 chapters (although the 24<sup>th</sup> chapter is more of a "goodbye and good luck" chapter). The chapters are organized into the following high-level parts:

- **Part 1.** Chapters 1 through 5, covers server fundamentals, provides a look at the primary management tool for Windows Server – Server Manager, looks at managing server roles and features, and explores the foundation of Windows PowerShell. Part 1 introduces the core concepts of Windows Server. These concepts are needed as we move to Part 2 and beyond.

Many of the tasks that we cover will rely on items from Part 1, such as Server manager, roles and features, and Windows PowerShell. For example, in most of the later chapters, we will discuss how to perform a task with Windows PowerShell or have a lab that relies on Windows PowerShell.

- **Part 2.** Chapters 6 through 11, covers networking components of Windows Server including network fundamentals, DHCP, DNS, and troubleshooting network communication issues. The network fundamentals help to establish a core understanding of communications that will be helpful before getting into Active Directory in Part 3. Active Directory relies heavily on DNS name resolution. In addition, as part of managing Active Directory, you need to understand how to troubleshoot basic network communication issues.
- **Part 3.** Chapters 12 through 18, focuses on Active Directory Domain Services (AD DS) tasks, such as working with the AD DS management tools, managing user and computer objects, managing groups, and working with Group Policy. To be a proficient Active Directory administrator, it is expected that you have a solid foundation in basic server management and network communications. While Active Directory is often seen as a core and critical foundational component, it still relies on other components to function properly. Thus, Active Directory content is delivered in part 3 once prerequisite knowledge and tasks are understood.
- **Part 4.** Chapters 19 through 21, covers services, storage, and NTFS permissions. These technologies play an important role in file and print services that are covered in Part 5. Storage plays a big role for Hyper-V. The goal of Part 4 is to bring you up to speed on key information to help understand the information in Part 5.
- **Part 5.** Chapters 22 and 23 cover file services and print services. We'll look at shared folders, securing data on files servers, and managing print servers and print queues.

Beyond the text, screen captures, and diagrams, you will find some parts of the chapter stand out, including:

- **Hands-on Exercise.** The Hands-on Exercise callout is a small sidebar/reader aid that instructs the reader to try something from the text in their lab environment while reading the chapter. These are good ways to help you remember what you read and cement the information into memory. You should try to work through each Hands-on Exercise as you come to them in your reading. Otherwise, perform the exercises before you start the lab.
- **Above and Beyond.** An Above and Beyond is an optional sidebar with helpful information to help understand why things work the way they work. If you are short on time, skip them. You can always circle back at another time.
- **Hands-on labs.** At the end of each chapter, except Chapter 1, is a lab covering technologies from the chapter. The labs are critical elements and I highly advise you work through every lab directly after reading the chapter. While some can read a chunk of pages and retain much of it, most retain more information by reading, hearing, seeing, and by doing. Often, combining those methods maximizes retention. If you come across a confusing topic, the lab can often help clear up that confusion.

- **Ideas for on your own.** I will sometimes insert ideas for you to research or try on your own, outside of the book. The ideas will lead you deeper into a technology. These are optional but very effective at helping you master a technology, especially if you have a good grasp on the concept.
- **Supplementary materials.** Occasionally in the book, I will provide URLs to supplementary materials.

## Setting up the lab environment

To maximize your retention, use a lab environment while reading this book. Use the lab environment to complete the exercises and walk through the examples. While this book is written about Windows Server including Windows Server 2016, most the content is valid for any version of Windows Server back to Windows Server 2008. For your lab environment, you have a couple of options:

- **Use a single physical computer.** With this option, install Windows Server 2016 on the computer, although a previous version will suffice if that is your only option.  
Having more than one computer running Windows Server will provide the most value, especially for some of the chapters that focus on network communication and Active Directory. This is the easiest path, especially if you do not have experience setting up a lab and do not have access to an existing lab. You can do this at home or at work, or use the public cloud such as Microsoft Azure. By using VMs with limited power and power down the VMs when you aren't using them to reduce costs. (I didn't understand this sentence) In such cases, the overall monthly costs are inexpensive. Otherwise, you can use VMs in TechNet Virtual Labs. See <https://technet.microsoft.com/en-us/virtuallabs/default> for a list of labs and look for a lab that provides a Windows Server 2016 VM or a Windows Server 2012 R2 VM.
- **Use a single physical computer, such as a Hyper-V host to host virtual machines (VMs).** You will need at least one VM running Windows Server 2016 to take advantage of the latest features of the operating system. The host computer can run Windows Server 2012 R2 or later versions. By running a Hyper-V host, you can run multiple VMs, which is beneficial for the topics that cover network communication and Active Directory. You can run Windows Server 2012 R2 and Windows Server 2016 to gain experience with both operating systems. You can opt for this path and use a computer at home or at work. If you opt for your work environment, make sure not to use your production environment! From a hardware perspective, I suggest the following components:
  - **64-bit processor with virtualization compatibility.** The CPU vendor is not relevant but a CPU produced within the last 3 years that has virtualization support is needed.
  - **8GB of RAM.** Anything less than 8GB will greatly reduce performance, especially if you run several VMs. More than 8GB will improve performance.

- **100GB of free disk space.** You may be able to get by with less, but it helps to have extra too. Especially if you decide to run several VMs.
- **High-performance disks.** While your lab can get by with slow hard drives (such as 5400 RPM SATA drives), you may be frustrated by the lack of performance and wait for your VMs to react. If possible, opt for SSD disks. Performance will be greatly enhanced.
- **Internet connection.** While you can get by without an internet connection (assuming you have all the software locally), you are better off with an internet connection so you can work through some of the network troubleshooting scenarios, update PowerShell help files, and access embedded links.
- **Hypervisor software.** I recommend Hyper-V. But you can opt for any hypervisor platform that enables you to create Windows Server-based VMs.

Whichever path you choose, it is critical to have access to Windows Server as part of reading this book!

## Hit the Ground Running!

This book focuses on helping administrators become immediately effective. What does that mean? It means that mastery of a chapter provides relevant knowledge and skills that can be applied on the job, immediately! In fact, I urge you to take the information you learn in each chapter and lab and put it to use in your environment.

Being immediately effective with Windows Server means you will be able to get in front of the console or connect remotely to a Windows server and have the knowledge to configure it, maintain it, and troubleshoot it. As you master each chapter, you should notice the information and labs presented are geared specifically to you being immediately effective. Most of the background and theory are pushed to the side in favor of a hands-on approach focusing on the most common administrative tasks with the most common Windows Server technologies.

Enough introduction! Let's dive into the content.

## CHAPTER 2: A PEEK INTO WINDOWS

---

While the focus of this book is Windows Server, we are going to start off by looking at Windows from a big picture perspective, piece by piece. Understanding Windows will help you in your quest to learn how to manage Windows Server. In some respects, Windows client operating systems such as Windows 10 are identical to Windows server operating systems such as Windows Server 2016. In other ways, such as how each operating system optimizes their workloads, client operating systems are different than server operating systems. In this chapter, we will look at how clients and servers differ while also looking at some of the similarities that can help you expand your knowledge if you have existing knowledge on the client side. We'll compare hardware and software and discuss some of the key differences that come into play as a server administrator. I'll discuss the four key hardware components that you will deal with most regularly. I'll introduce some of the key software components too. Then, I'll discuss the fundamental networking concepts that you'll need to be comfortable with in a server administration job, the difference between physical and virtual servers, and the difference between workgroup servers and domain-joined servers (often called member servers). Then, to finish this chapter off, I'll discuss where security and the cloud come into play from a server administration perspective. This chapter is a little bit different than most the chapters because it provides a lot of background info that will be helpful as you move forward whereas other chapters provide information about performing administration. If you already have a thorough understanding of the topics planned for this chapter, consider skipping to the end and testing yourself with the review questions. This chapter is in Part 1 of the book which starts with introductory and background information. Then, Part 1 finishes out by covering Server Manager, PowerShell, and Windows Server roles and features.

As a server administrator, you will occasionally use hardware-based skills such as swapping out computer components and deploying new computers. But even more so, you will use software-based skills such as configuring a Windows service so that it does not start up automatically during boot. While the industry is moving to software-based solutions, especially with all the virtualization technologies, there are still some hardware-based skills that you need to be successful. Virtualization, at a high level, creates software representations of hardware components.

Administrators that learned about operating systems in the early days, before the internet, didn't have to learn much about networking because it wasn't all that common to connect computers together on a network. Today, it is a vastly different story. Just about every computing device is connected to other computing devices, whether locally on a network or over the internet. Part of your job with managing Windows servers is networking and we'll get into some of the fundamentals in this chapter while looking more closely at some of the networking components in later chapters.

By the end of this chapter, you should have a good understanding of the role of a Windows server, understand the primary hardware components of a server, describe networking from a server administrator's perspective, and be comfortable with the topics of security and cloud computing from a server perspective.

## Client vs. Server

When we say the phrase “client computer”, we are referring to any computing device that connects to a server to request resources or services. Resources include things like web pages and shared folders and files while services are things like printing and validating the identity of a user. A computing device could be a desktop computer, a laptop computer, a tablet computer, or even a smartphone. There are several terms to describe a client computer. For example, some administrators refer to them as workstations. Others call them productivity computers. For this book, a client computer will refer to a computing device running a Windows client operating system, unless otherwise noted.

When we say the term “server”, we are referring to a backend computer, often stored in a server room, data center, or in the public cloud, configured to service requests from client computers or other servers. For this book, a server will refer to a computer running Windows Server 2016, unless otherwise noted. While older versions of Windows Server, all the way back to Windows Server 2008, are compatible with much of this book’s information and examples, there are some areas that are only valid for Windows Server 2008 R2 and later (such as some PowerShell commands).

The following table, Table 2.1, outlines the primary differences between client computers and servers.

Table 2.1 Differences between client computers and servers

<b>Characteristic</b>	<b>Applicable to client computers?</b>	<b>Applicable to servers?</b>
Multiple physical CPUs	No, not usually	Yes, often
Error correcting memory (ECC memory) which automatically detects and fixes some data corruption issues	No, rarely	Yes, usually
Optimized for applications	Yes	No
Optimized for background services	No	Yes
Hardware redundancy (if a component fails, a second matching component takes over)	No, not usually	Yes, often

Supports multiple simultaneous users	No	Yes
--------------------------------------	----	-----

One of the most difficult things for new administrators is visualizing abstract networking concepts. However, visualizing is often important to understanding concepts. For new server administrators, it is often helpful to look at where servers reside, which types of tasks they perform, and which type of devices they service. The diagram below, Figure 2-1, shows the relationship between client computing devices and servers. Note that the tasks that are shown in the diagram are just some examples of common tasks.

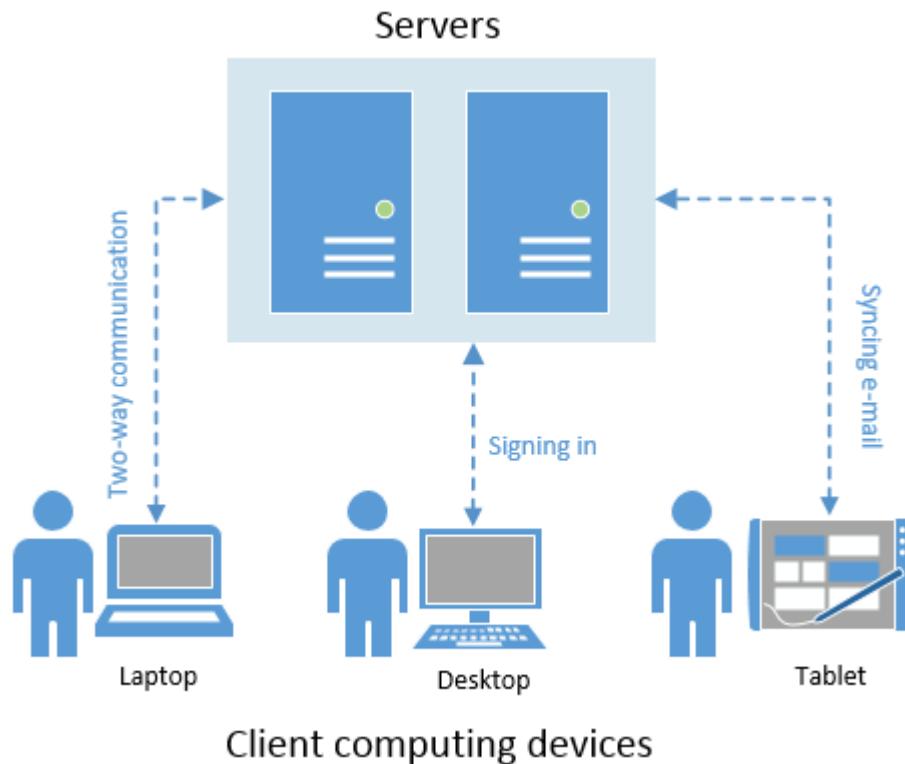


Figure 2.1 Different types of client computing devices connect to servers to perform the routine tasks of signing in and syncing e-mail.

Servers are generally large, powerful computers that are built to be redundant. For example, a server generally has multiple physical CPUs, multiple hard drives, and multiple network interface cards. If one component fails, another can take over seamlessly. At a higher level, if one web server fails completely (such as motherboard failure), a redundant server can continue to provide web services until the failed server is repaired and brought back into service. They also work in concert with other servers. Client computers are generally small personal devices built to maximize the end user experience with hardware flexibility and cutting edge designs.

## Hardware vs. Software

Prior to the explosion of virtualization, hardware skills were skills that most server administrators used regularly. Most organizations relied on physical servers. Thus, administrators

were responsible for maintaining that hardware. It was routine to swap out failed hard drives, add more memory to a server, replace a failed power supply, and update the BIOS and firmware of a server. Today, virtualization has grown in popularity and expanded into almost every area of technology. There are solutions that virtualize servers, virtualize client computers, virtualize networks, and virtualize storage. Because of that, administrators are not working with hardware as often. It isn't uncommon to have administrators that have limited hardware knowledge or skills and are still highly successful in their jobs. However, understanding hardware is still helpful, especially if you are less experienced. Hardware knowledge can often help you understand other aspects of IT, such as software.

Hardware represents all the physical components of a computer, including:

- CPU
- Memory
- Hard drives
- Computer case
- Power supply
- Cables
- Software includes the following items:
  - Operating system
  - Applications
  - Firmware
  - Drivers

Hardware relies on software and software relies on hardware. They work together to provide a working environment for users or a server environment that provides services (web sites and computer backups being two examples) and data (a place to store your team's files being a prime example).

The lines between hardware and software are straightforward. Any computer component that you can pick up with your hands is hardware. Everything else is software! That is until we begin talking about virtualization. This blurs the lines between hardware and software. We will talk a bit more about that process in the upcoming section titled "Physical server vs. virtual servers". For now, I just wanted to introduce the concept at a high level so that you understand the big picture.

## Four core hardware components

In the previous section, we listed some of the traditional hardware components that are part of a computer. Now, we will take a closer look at four core hardware components that you will deal with most often. Besides explaining what each component is, I'll also sometimes call out when a component can be a performance bottleneck. You'll commonly deal with performance

issues so understanding how the components factor into bottlenecks will help you reduce your troubleshooting time.

## CPU

The CPU, or Central Processing Unit, is what many refer to as the brain of the computer. And technically, it is. It processes all the instructions sent by the operating system and hardware components. CPU power has been growing exponentially based on Moore's Law so modern servers often have extra processing power. Thus, CPUs do not often contribute to long-term performance bottlenecks. Instead, administrators deal with short term CPU bottlenecks from time to time. Sometimes, a process or service can consume most or all the available CPU. With virtualization, the CPU sometimes takes a backseat to memory and storage which is what administrators often spend more time managing. While CPUs have taken a backseat, they are still vital to a computer and thus a datacenter.

## ABOVE AND BEYOND

Moore's Law, thought up by Gordon Moore in 1965 and revised by him in 1975, states that the number of transistors in a dense integrated circuit (IC) will double every two years. In simple terms, he suggested that the processing power of CPUs would double every two years. And this theory held up until about 2012.

The semiconductor industry relied on Moore's Law when creating their long-term CPU roadmaps.

## Memory

When we talking about memory, we are talking about random access memory (RAM), the primary memory that your computer uses to temporarily store information for the tasks that it is performing. Memory is a routine cause of performance bottlenecks, especially long term bottlenecks. When a computer runs out of memory or runs with less memory in it than required, it can rely heavily on the pagefile which is a temporary place on a volume that can be used as virtual memory. The pagefile is a file named pagefile.sys that, by default, resides on the system volume. When a computer uses the pagefile heavily, system performance is greatly impacted. This is because a hard drive, even a solid-state drive (SSD) which is the fastest hard drive because it doesn't have any moving components, is a lot slower than RAM. If you've worked on client computers or applications in your job, you've probably run into a situation where memory was low and performance was impacted. It happens on client computers, network devices, and servers!

As a Windows Server administrator, you need to figure out the pagefile settings that you will use for your organization's computers. Otherwise, you may have computers that do not perform optimally or run the risk of running out of disk space on the system volume, which is where the operating system is stored. Generally, it is a good practice to use a non-system volume which is a volume that the operating system files are not stored on. Otherwise, the pagefile shares input/output (I/O) with the operating system. Input/output is data being written or read to or from a hard drive.

## MEMORY FOR VIRTUAL ENVIRONMENTS

With virtualization, instead of thinking only about the amount of RAM a virtual machine has, you should also think about dynamic memory, which is memory that can automatically be expanded or reduced based on the memory configuration. With dynamic memory, you manage more memory settings on a virtual machine than you do with a physical server. This gives you more granular control over the computer's memory. With this control, you can ensure RAM isn't being wasted by being allocated and unused. While each hypervisor platform, which is a hardware and/or software solution for virtualizing servers, uses slightly different technologies and terminology, you should be familiar with the dynamic memory settings for Hyper-V, Microsoft's hypervisor platform. Figure 2.2 shows the settings. Don't worry about memorizing this information. Right now, the goal is to expose you to the high-level concepts. We will dive into the details with deeper explanations as we move to new chapters.

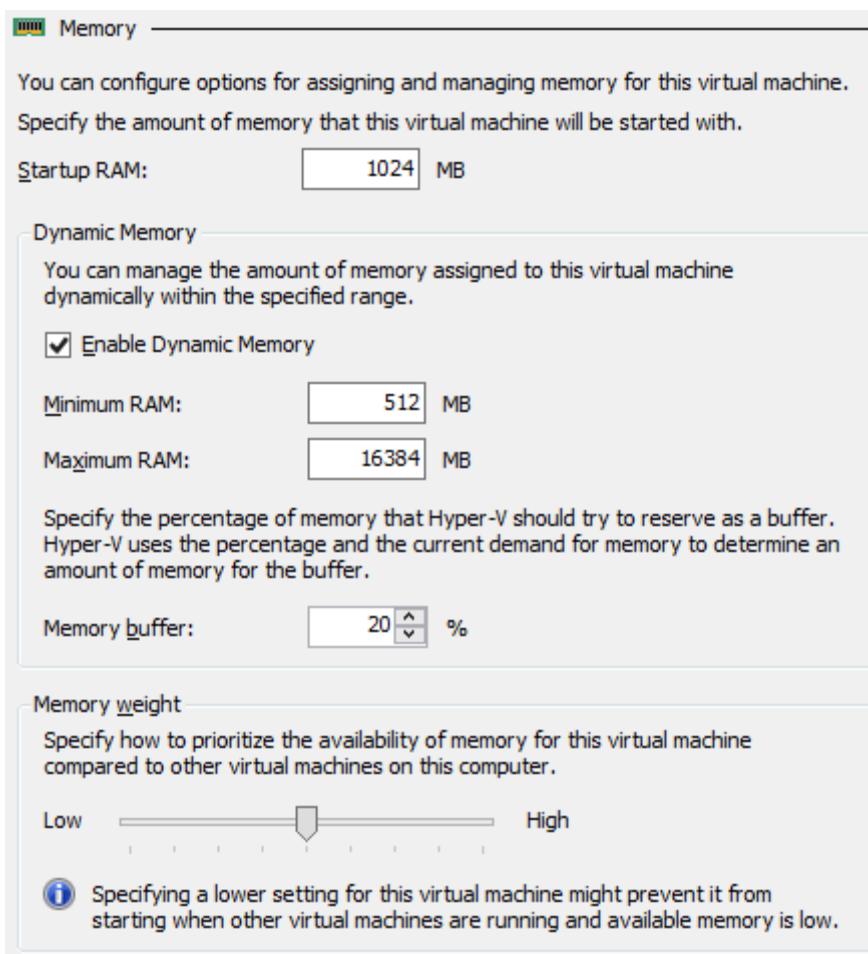


Figure 2.2 Hyper-V's virtual machine memory settings allow you to enable dynamic memory.

- **Startup RAM.** When a VM starts up, it needs a little bit more RAM than it needs while idling. Thus, the amount of startup RAM is often a little bit more than the amount of minimum RAM configured for a VM. For example, if you have a VM that needs 512MB of RAM while idling, it may need 768MB of RAM during startup.

- **Minimum RAM.** The minimum amount of RAM needed is the amount used when a server is sitting idle and performing a minimal workload. It is a good practice to set the minimum RAM to the minimum supported RAM for the operating system (or more).
- **Maximum RAM.** The maximum amount of RAM is the most that you want a server to consume, no matter how much it requires at a given time. If a server requires 32GB of RAM and you set the maximum amount of RAM to 16GB, then the server will be limited to 16GB of RAM. In such a case, the computer will rely on the pagefile and performance will be impacted negatively. On the other hand, the server won't be able to consume an unlimited amount of RAM and impact other VMs running on the same Hyper-V server.
- **Memory buffer.** Hyper-V will reserve the specified amount of RAM to use as a buffer. By default, 20% is reserved. Hyper-V can rely on the buffer while it seeks to claim RAM back from other VMs during a low memory event.
- **Memory weight.** Each VM is configured with a memory weight based on a sliding scale. On the far left of the scale, the memory weight can be set to low. On the far right of the scale, the memory weight can be set to high. VMs that are set with a low memory weight have lower priority to the hypervisor's memory than VMs with a higher weight. If a VM with a high memory weight requires additional RAM, that RAM can be pulled from VMs with a lower weight. Your least critical servers should be weighted with a lower priority while your most critical servers should be weighted with a high priority.

## Storage

Storage is made up of several technologies that are used to store data on a server.

Traditionally, storage meant hard drives. And not just any hard drive but the large and heavy hard drives that you inserted into special hard drive ports. But lately, SSDs are gaining in popularity. Other storage technologies are common too, including storage-area networks (SANs), network-attached storage (NAS), and cloud-based storage.

### SANS

SANs are complex storage environments made up of several components:

- **Disk drives.** The disk drives in a SAN are often the same disk drives you can find in a typical server. The main difference is the quantity of disk drives. A SAN often has anywhere from several disk drives to hundreds of disk drives. Often, the more disk drives that you have in a SAN, the higher the performance of the SAN will be. There are also different types of disk drives in a SAN. You'll find S-ATA disk drives for long-term archival needs, SAS disk drives for typical server workloads, and high-performance SSD disk drives for high-performance workloads such as databases and computational computers. SAN disk drives are stored in disk shelves which are hardware-based enclosures that hold many disk drives.
- **Controllers.** The controllers in a SAN are often appliances running a slimmed down operating system optimized for storage services. They have ports for connecting to switches, storage, and other controllers. It is common to have more than one controller for redundancy.

- **Storage operating system.** The storage operating system is often a slimmed down version of Linux or Windows specifically built to optimize and secure storage services such as backups, sharing, and encryption.
- **Switches.** The switches are often third-party switches. Many vendors, including network device vendors, produce SAN switches to connect the SAN controllers to the servers and/or rest of the network. SAN switches are like network switches but sometimes use different cabling and different network protocols. The industry has been moving toward a converged network where all typical network communication and SAN communications operate on the same network using the same protocols.
- **Adapters.** The adapters are the expansion cards in servers that allow the servers to connect to the SAN switches to gain access to the SAN. The adapters can be host bus adapters (HBAs), iSCSI adapters, or standard network adapters depending on the SAN.

In Figure 2.3 below, you can see how these components work together in a typical SAN.

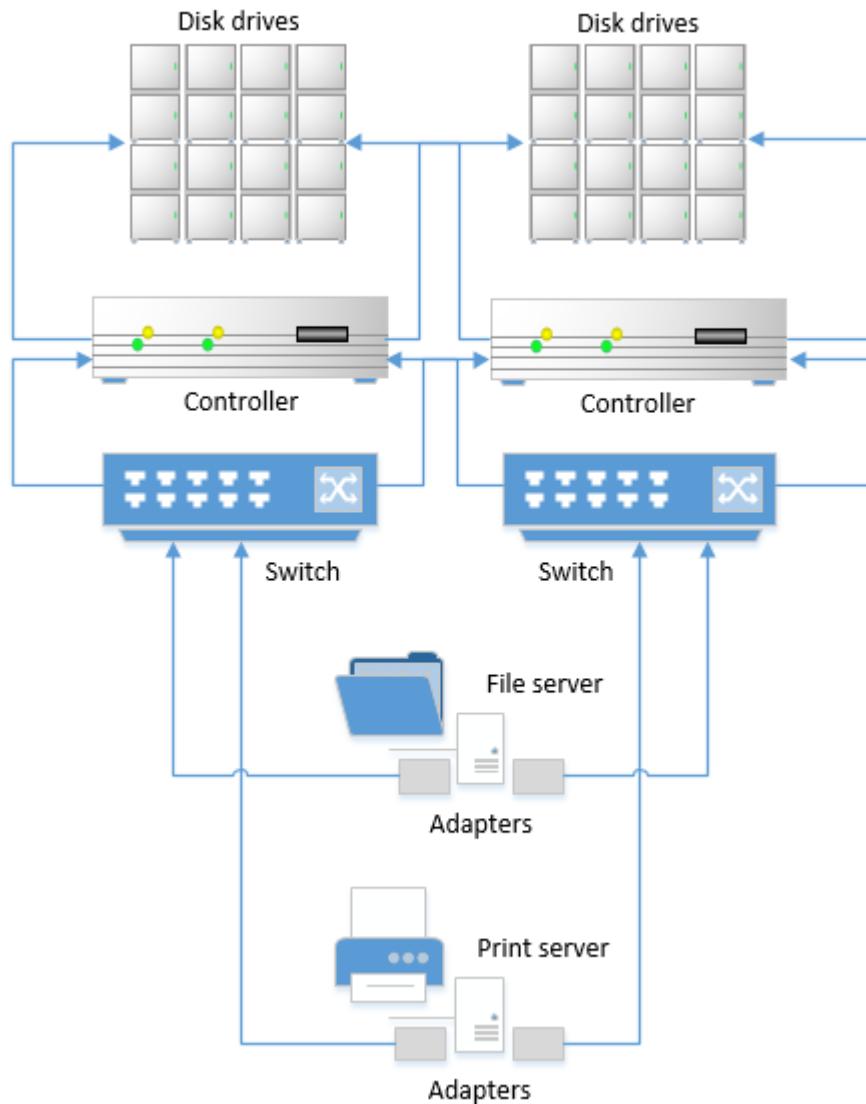


Figure 2.3 A highly available SAN is shown including the disk drives, controllers, switches, servers, and adapters.

Let's walk through the diagram in Figure 2.3:

- **Adapters.** At the bottom of the diagram are a print server and a file server. Each server has two adapters. This provides redundancy. If one of the adapters fails, the servers can still get to the SAN storage through the other adapter. Note that each adapter usually connects to a different switch so that if a switch fails, the servers can still get to the SAN storage through the other switch.
- **Switches.** Just above the adapters are the switches. The switches are the middle men between the SAN controllers and the adapters. The switches each have a connection to each SAN controller. This allows SAN connectivity in the event of a SAN controller failure since the switches have two paths to the SAN storage (1 path through each switch).

- **Controllers.** The SAN controllers are the brains of the SAN. Sometimes called heads or filers, the controllers manage the backend storage and management connectivity from the clients. For redundancy, controllers usually have at least one connection to all the backend storage.
- **Disk drives.** At the top of the diagram are the disk drives. They are often in a special disk enclosure which is called a disk shelf. In many environments, multiple disk shelves are needed to accommodate the demand for disk space. In addition to the physical redundancy of many disk drives, many SAN vendors incorporate proprietary redundancy algorithms to provide enhanced redundancy.

## NAS

NAS is a simple storage technology that provides cross-platform file and folder sharing to servers and/or clients. A NAS provides sharing to Linux and Windows clients, even at the same time. A typical NAS has the following characteristics:

- **Storage.** Each NAS has some type of storage. It can be a directly attached disk shelf or internal disk drives. It can even be storage from a SAN.
- **File system.** Each NAS has a file system. Often, the file system supports multiple file sharing protocols such as server message block (SMB) and network file system (NFS). To clients, a NAS appears as a file server.
- **Operating system.** A NAS is controlled by an operating system. The operating system is often optimized for file sharing and usually does not provide any other services.
- **Cloud-based storage.** Cloud-based storage is the newest storage technology and has grown quickly over the last few years, especially as internet bandwidth pricing has gone down and organizations have more bandwidth. Cloud-based storage can be connected to from the Internet (think OneDrive and Dropbox) or via a private connection to a cloud service provider. For example, Amazon offers Amazon Simple Storage Service (Amazon S3) while Microsoft offers Microsoft Azure's Storage. Both can be consumed from the internet or from a private connection to the cloud service provider's network. Cloud-based storage does not offer high-performance storage for on-premises servers due to the latency of the connection. However, for cloud-based servers, the cloud-based storage is local and can provide as much performance as your workloads require. Companies often use cloud-based storage for sharing data with customers and clients or for off-site backups.

Many companies choose to use more than one of these storage technologies. In fact, at large organizations, it is common to see all the technologies used. Some vendors are even providing a form of SAN, NAS, and cloud-based storage in a single specialized appliance.

Storage plays a big role in the performance and redundancy of a server. For local server storage, with a redundant array of inexpensive disks (RAID) and other redundancy technologies, servers can lose a disk drive, and in some cases, more than one disk drive, and still provide access to data. We will take a close look at local storage in Chapter 20 "Working with local storage".

## Network interface card (NIC)

The network, including the internet, is what drives computing today. Without a network or the internet, a computer seems useless. The NIC is a peripheral or on-board computer chip that enables a computer to connect to a network. In a server, it is common to have two or more NICs to take advantage of load balancing and redundancy capabilities. With the simplest form of load balancing, a client that connects to a load-balanced server will be directed to communicate with a specified NIC while the next client that connects will be directed to the other NIC. This continues with each subsequent connection. With redundant NICs, a server can lose a NIC to hardware failure and still service clients.

From a server administrator's perspective, a NIC is a hardware component that requires configuration, unlike other hardware components that don't such as a CPU. The common configuration tasks for a NIC in a server environment are creating a network team for load balancing or redundancy, configuring the TCP/IP properties of the NIC, and managing the drivers and firmware. While on-premises servers are almost always manually configured with their TCP/IP settings, cloud-based servers most often rely on automation to automatically configure the TCP/IP properties.

## The makeup of the software components

In the Windows Server operating system, there is a myriad of software components. The core software components are the Windows kernel, system processes, services, applications, and Windows subsystems. These components make up the foundation of Windows Server and just about everything that runs on a Windows Server relies on one or more of these components. Administrators work directly with processes, services, and applications daily while the Windows kernel and subsystems are not often directly worked on by administrators.

- **Windows kernel.** The kernel is the foundation of the operating systems. It manages the low-level functions that keep Windows running such as the drivers, hardware abstraction layer (HAL), and virtual memory. The kernel operates in its own protected memory, away from user mode activities.
- **System processes.** The system processes are the lowest level of user mode activities and are processes such as the Local Security Authority Subsystem Service (LSASS) and Winlogon, which handle computer security and authentication such as the sign in process.
- **Services.** Every Windows Server has services which are background processes that provide functionality to the operating system or to other computers and users. Many services are part of Windows Server but third-party applications often add additional services. As an administrator, you control which services get added, which service accounts they run under, and the action (if any) taken if a service fails. Some of the common services that you may be familiar with are the print spooler service and the Remote Desktop Services service. A graphical user interface installation of Windows Server 2016 has 170 built-in services. A default installation, without the GUI, has 106 built-in services.
- **Applications.** Many servers run more than just the Windows Server operating system. They run other applications to enhance existing services or provide functionality that isn't part of

Windows. In a typical corporate environment, every server has a foundational set of applications that are part of a default server build. For example, an anti-virus application is often part of a default server build. Other common applications are security applications (software firewalls, centralized logging, and auditing agents), and backup applications.

- **Windows subsystems.** The subsystems are responsible for dynamic link libraries (DLLs), security and process management, and other low-level operations. These subsystems operate in user mode, separated from the kernel.

## Fundamental networking concepts for server administrators

As part of a job managing Windows servers, you will work directly and indirectly with several networking technologies. Sometimes, these technologies are built into Windows while other times the technologies are not and are managed by a different team. Let's introduce some of these technologies.

- **Subnetting.** While just about everybody in IT is familiar with a subnet mask, there are not a lot of people outside of network administrators that know how to calculate subnet masks based on a given network scenario. As a server administrator, you should be familiar with the most common subnet masks, how to calculate a subnet mask, and what happens when you don't use the correct subnet mask. For example, one of the most commonly used subnet masks is 255.255.255.0 which is used to designate a Class C or /24 network. This knowledge can help you when troubleshooting connectivity issues on your servers. The good news is that you can use a subnet calculator to assist you. Later, in a section titled "Working with subnets", we will look at subnetting in more detail.
- **DHCP.** Few people refer to DHCP as Dynamic Host Configuration Protocol these days. That's because it is widely known by its acronym, even more so than the original long name. It's hard to find a home or business that doesn't have DHCP somewhere on the network. DHCP is found on wireless routers, cable modems, servers, and network appliances. DHCP automates the TCP/IP configuration of a computer so that administrators do not have to initially configure any TCP/IP related settings such as an IP address or DNS servers. Most computers have their TCP/IP settings automatically set by a DHCP server. One common exception is servers. Servers usually have their TCP/IP settings configured manually. While the cloud is proving that DHCP can be functional for many servers (because most providers use DHCP for cloud-based servers), DHCP services will be around for the long haul. As a server administrator, you need to be intimately familiar with DHCP from a server and a client perspective. An entire chapter of this book is dedicated to managing DHCP!
- **DNS.** Domain Name System (DNS) is another widely deployed core networking technology. DNS is used to associate names with IP addresses and IP addresses with names. Instead of having to remember an IP address such as 172.16.44.251, you can use DNS and associate an easy to remember fully qualified domain name (FQDN) such as server1.contoso.com. DNS is a critical infrastructure service because most your services and connectivity rely on DNS. If DNS is down, users can't visit web sites, can't get to their email, and often can't perform any work on their computer. For outsiders, your company web site may not function, any services that you provide to customers may be unreachable, and your customers and

partners may have their email returned to them when trying to send to you. Active Directory Domain Services (AD DS) is wholly dependent on DNS too. Luckily, we have 3 chapters in this book dedicated to DNS.

- **Routing and switching.** As a server administrator, you'll work indirectly with routing and switching. You will often work with a network guy or a network team when you have routing and switching issues. Routing, defined simply, enables computer communication between two computers on different networks. Often, this means two computers at different physical locations. In more complex networks, it can also mean two computers in the same building or same campus but separated by a router. Routing is used to connect disparate networks together. Routing operates at Layer 3 of the Open Systems Interconnection model (OSI model). Switching, performed by specialized network appliances named switches, is used to facilitate network communication between different segments of a local area network (LAN). In the simplest networks, a switch is a way to connect all the computers to the network. As networks grow, segmentation is used to enhance security and maximize performance. Switching mostly operates at Layer 2 of the OSI model. There are some new switching technologies that operate at Layer 3 and Layer 4
- **Load balancing.** Load balancing is a technology that balances communication across multiple endpoints, such as servers. For example, if your company has 6 web servers, load balancing can be used so that the first web site visitor is directed to the first web server, the second web site visitor is directed to the second web server, and so on. Load balancing ensures that communication is evenly distributed across the endpoints to maximize performance. Load balancing operates at Layer 4 of the OSI model. Depending on the configuration used, load balancing can also help maximize redundancy. Many load balancing solutions are dedicated hardware-based appliances. There are also virtualized load balancing solutions and software-based load balancing solutions that run on a typical Windows or Linux server. Load balancing is often deployed and managed by the network team. But, server administrators are sometimes given scoped administrative access to manage their application(s) that are being load balanced.

## ABOVE AND BEYOND

The OSI model outlines the communication methods for computers to help create a standardized communication foundation. The OSI Model was conceptualized in the 1970s and published in 1984 as ISO standard 7498. most of our network communication adheres to the OSI model. There are 7 layers defined: Layer 1 (physical layer), Layer 2 (data link layer), Layer 3 (Network layer), Layer 4 (Transport layer), Layer 5 (Session Layer), Layer 6 (Presentation Layer), Layer 7 (Application Layer). Layers 1 through 4 focus on moving the communication across the network. Layers 5 and 6 focus on how the data is passed back and forth and the format of the data. Layer 7 is focused on application functionality such as authentication and privacy. For additional reading, you can download ISO standard 7498 from [http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269 ISO IEC 7498-1 1994\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269 ISO IEC 7498-1 1994(E).zip).

## Physical servers vs. virtual servers

A physical server is a large computer that you put into a server rack or on a shelf in a server room or data center. A virtual server is a virtual machine (VM) running a server-based operating system on a virtualization platform such as Hyper-V. While a physical server and a virtual server often perform the same tasks, run the same operating system, and run the same applications, there are some fundamental differences between them. The following table, table 2.2, outlines some of the key differences.

Table 2.2 Differences between physical and virtual machines

Characteristic	Physical server	Virtual server
Reduces data center footprint	No	Yes
Reduces cooling costs	No	Yes
Greatly reduces install times	No	Yes
Maximizes hardware utilization	No	Yes
Simple licensing model	Yes	No
Lower cost of ownership	No	Yes

## Workgroup vs. domain

When you install and configure Windows Server, you can choose to configure it as a standalone server. A standalone server is a server that is not joined to an Active Directory Domain Services (AD DS) domain. A standalone server uses local users and groups for authentication and authorization. If you have 25 standalone servers, each server has its own user accounts and security groups. Standalone servers are often referred to as workgroup servers because they are technically joined to a workgroup. A workgroup is a loose collection of computers. When they are on the same network, they can communicate with each other but they do not share authentication, authorization, or any of their other services and features.

Alternatively, and much more common in a business setting, you can choose to join a server to an AD DS domain. Joining a server to a domain enables the server to take advantage of AD DS authentication and authorization. AD DS centralizes authentication and authorization and abstracts the services from the server. A domain-joined computer is commonly referred to as a member server because it is a member of a domain. For example, in a corporate environment, client computers and servers are usually joined to an AD DS domain. But to maximize security, computers in a public lobby such as kiosk computers for public use, are usually not joined to a domain and instead belong to a workgroup.

In many cases, an organization will have most of their servers joined to a domain and a small amount of servers in a workgroup. A typical use of workgroup servers are servers located in the demilitarized zone of a network, also known as the DMZ (or perimeter network). A DMZ is a secure network segment, typically located behind or between firewalls, that houses servers that need to communicate with internet-based clients and services. For example, a public web server often resides in a DMZ.

The following table, table 2.3, outlines some of the key differences between servers in a workgroup and domain-joined servers.

Table 2.3 Differences between workgroup servers and domain-joined servers

<b>Characteristic</b>	<b>Workgroup server</b>	<b>Domain-joined server</b>
Computers can be located on different networks	No	Yes
Computers are centrally managed	No	Yes
Valid for many servers	No	Yes
Viable for niche roles such as a kiosk	Yes	No
Users can have a single user account and sign in to every computer	No	Yes

## How security fits with Windows Server

Security is a pretty big topic these days. The media has picked up and run with big IT security related stories that have educated much of the public about the danger of misconfigured computer networks. As such, it is the job of every person in IT to take security seriously. Security should be a factor in every single IT project.

From a Windows Server perspective, a server administrator has several tools at their disposal. One big one, Group Policy, has a couple of dedicated chapters later in this book (Chapter 16 and Chapter 17). While we won't have an opportunity to dive into every aspect of securing a Windows server in this book, you should allocate some time to familiarize yourself with some of the available tools such as Microsoft's free Security Compliance Manager (see <https://technet.microsoft.com/en-us/library/cc677002.aspx>).

Beyond securing a Windows Server itself, a server administrator must be familiar with many of the other security technologies in place at an organization, such as:

- **Hardware firewalls.** Hardware firewalls are network appliances, usually managed by a network team or a security team, that are configured to permit or deny network traffic based on a set of rules. For example, a firewall might allow all client computers to go to the internet on port 80 (HTTP) while blocking all client computers from connecting to FTP servers on the internet on port 21 (FTP). Most organizations have a firewall between their network and the internet. In larger organizations, firewalls are used to segment data centers, physical offices, and partner networks. The latest generation of firewalls, sometimes called next generation firewalls, move beyond ports and protocols and can understand applications, network locations, and personas.
- **Proxy servers.** A proxy server is used to control the web sites that employees can visit, based on rules and filters. When a proxy server is used at a company, client computers are often configured to use the proxy and cannot get to the internet without the proxy. A browser is configured to send all requests to the proxy server which evaluates the request against the rules and allows the connection or doesn't. For example, some organizations block gambling sites and political sites.
- **De-militarized zones (DMZs).** A DMZ is a network segment that is firewalled off from the internet and sometimes the LAN. It is often used for web servers and other servers that provide services to the internet. A DMZ provides a secure area to provide internet service from, without exposing your internal network to direct network communication from the internet. Figure 2.4 shows a DMZ.

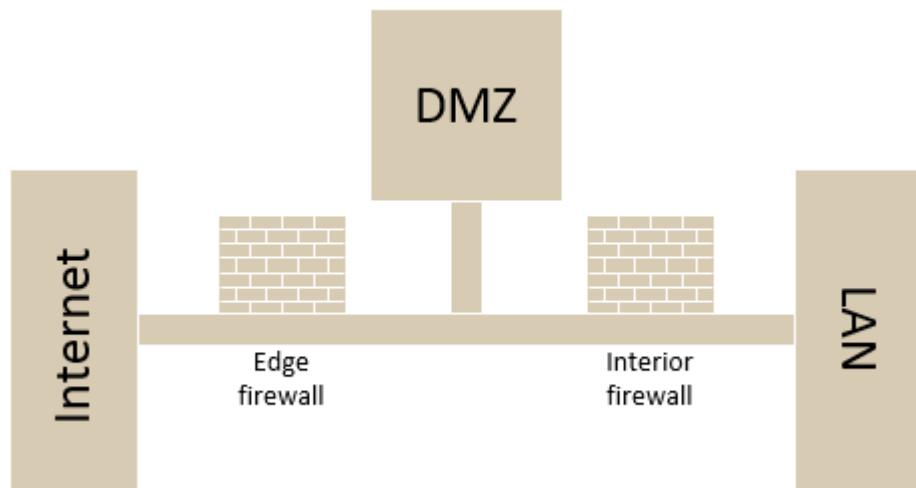


Figure 2.4 A DMZ is shown between an edge firewall and an interior firewall.

- **Virtual LANs (VLANs).** VLANs are used to segment networks from each other, sometimes limiting (or eliminating) communication between VLANs. A VLAN can be used in firewall rules to allow or prevent certain types of communication. For example, if you have a VLAN for your servers, you could prevent client computers from connecting to those servers on specific ports such as 22 (SSH) or 3389 (remote desktop protocol). Instead, you could require

all administrative connections to the servers to originate from an administrative VLAN to maximize security.

- **Intrusion prevention system (IPS).** An IPS is a specialized network appliance that monitors network communication looking for and blocking malicious communication. An IPS is often used with an intrusion detection system.
- **Intrusion detection system (IDS).** An IDS is a specialized network appliance that analyzes network communication and patterns to log and report on malicious communication. It is often used with an IPS. Sometimes, a single appliance can provide IPS and IDS capabilities.

## Windows Server in the cloud

You can run Windows Server at one of your company locations. When you do, it is referred to as running Windows Server on-premises. You can also run Windows Server in the cloud by using a cloud services provider such as Microsoft Azure. When you do, it is referred to as running Windows Server in the cloud. Using the cloud to maximize your company's time to market and reduce development time is a smart move. Many companies are doing it or exploring the opportunity. Often, combining on-premises IT infrastructure with cloud-based services is a good way to go. In such a situation, you can maintain total control of some IT infrastructure on-premises but take advantage of cloud flexibility when you need to quickly ramp up on power and speed. When you have computing resources on-premises and in the cloud, you have a hybrid cloud environment.

While we don't cover cloud-specific tasks in this book, most of the concepts and tasks apply equally to Windows Server whether it is on-premises or in the cloud. There are some small differences in managing cloud-based servers though:

- **You need to have connectivity to the cloud.** Management tools, the command line, and Windows PowerShell easily manage on-premises servers. But cloud-based servers require initial connectivity to the cloud service provider. Often, companies choose to have a permanent network connection to the cloud service provider's network which makes the cloud feel like another branch office location. In other cases, such as those without persistent network connections to the cloud provider's network, you need to make on-demand connections to the provider before you begin using your management tools.
- **There may be higher latency and less bandwidth to your cloud-based servers than your on-premises servers.** With higher latency, some common management tasks are best performed from the cloud to maximize performance and administrative experience. Tasks involving large data copies may not be practical from your location to the cloud (such as copying the Windows Server 2016 .iso file).
- **You don't control some aspects of the computing infrastructure.** For example, you don't have administrative access to the virtualization technologies, storage technologies, or network devices. The way you monitor your cloud-based servers will be different than the way you monitor your on-premises servers. Same with the way you secure your cloud-based servers. Because you only control the software configuration, you should be up to speed on the cloud provider's capabilities, maintenance windows, and security compliance.

In this chapter, we took a brief look under the hood of Windows Server and covered several associated topics. Test your knowledge by filling in the missing word or term in the following sentences.

- \_\_\_\_\_ are backend computers, often stored in a server room, data center, or in the public cloud, configured to service requests from client computers or other servers.
- While hardware skills are not as important today as they were in the past due to virtualization, \_\_\_\_\_ skills are becoming more important partly due to virtualization.
- When administrators do work with hardware, they are often working on one of the four core components: the \_\_\_\_\_, RAM, storage, or \_\_\_\_\_. Each of these components plays a critical role in the performance and stability of a server.
- Although Windows has a tremendous amount of software components, the key components are the Windows \_\_\_\_\_, system processes, services, applications, and Windows subsystems. Just about everything that runs on Windows depends on one or more of the key components.
- The key network technologies that a server admin must know about are \_\_\_\_\_ (dividing networks into management segments), \_\_\_\_\_ (automates the TCP/IP configuration for computers), \_\_\_\_\_ (resolves names to IP addresses and IP address to names), \_\_\_\_\_ (facilitates the communication between computers), and \_\_\_\_\_ (balances client requests evenly across multiple servers).
- \_\_\_\_\_ provide many benefits over physical servers such as lower cost, faster installation times, and a reduced data center footprint. Physical servers offer a simpler licensing model.
- A \_\_\_\_\_ is a small, indirect grouping of computers and is often used in a home or very small business environment. A domain environment is an environment with AD DS, which centralizes user accounts, computer accounts, policies, and management.
- To maximize \_\_\_\_\_, companies are using multiple technologies including firewalls, proxy servers, DMZs, VLANs, and IPS/IDS products. As a server administrator, it is a good practice to be familiar with these technologies so that you can take advantage of their offerings and help protect your servers.
- The \_\_\_\_\_ can minimize your company's time to market and reduce development time. An administrator can often manage cloud-based servers by using the same tools and methods as the administrator uses for on-premises servers.

## CHAPTER 3: INTRODUCTION TO SERVER MANAGER

---

Server Manager will be one of the tools that you rely on for your day-to-day administrative work. It is a tool that you can use to configure a wide range of settings on a server. It is important to know what it is capable of, what the limitations are, and how to work with it for your administrative tasks. This chapter is focused just on Server Manager and some of the settings that it helps you configure. I'll introduce the dashboard which is the first thing you see when you run Server Manager. We'll look at managing multiple servers from Server Manager which will help reduce the administrative time it takes to perform a task on several servers. Then, I'll spend a large amount of time walking through configuring a server with Server Manager. When you manage a server, one of the first tasks that you perform is connecting to the server. This comes in the form of a Remote Desktop Connection session or a connection using a management tool from a remote computer. Server Manager will often be the first thing that you see because, by default, it launches automatically when you first sign into Windows Server over a Remote Desktop Connection or from the console. I'll wrap up by looking at how you can control Server Manager's startup behavior, in case you don't want it to automatically start when you sign into a server.

In the chapter, we'll look at many of the most common tasks that you will perform with Server Manager. At the end of the chapter, you will know how to use Server Manager to perform those tasks on Windows servers, whether local or remote. Let's start by looking at the base functionality of Server Manager.

### What you can do with Server Manager

Many times, you will use Server Manager as a quick "at a glance" view of a server. You'll quickly see if the server is healthy and see which roles are installed on the servers. Beyond that, there are a myriad of administrative tasks that you can perform with Server Manager. The following list represents some of the most common tasks.

- **Configure the local server.** You can configure the server that you are signed into by using Server Manager. This includes the hostname, domain, Windows firewall, Windows Update settings, and many other settings. In section 3.4, we look at the settings that you can configure.
- **Add roles and features.** A server role is a pre-defined feature, or set of features, that provide a specified service. For example, the Web Server (IIS) role provides web site hosting services. A feature is usually smaller or less complex than a role and often just provides a very specific functionality to a server. For example, the Remote Server Administration Tools is a feature and underneath that feature is the individual management tools that you can add to a server.
- **Add other servers to manage.** One way to do this is by managing multiple servers from a single management console such as Server Manager. This saves you time because once you configure Server Manager to manage multiple servers, you don't have to spend time connecting to all the servers that you manage. Instead, they are ready to be managed on demand.

- **Create a server group.** A server group enables you to group servers into single units of management. For example, you could group together all your domain controllers.

Now that you have an idea of Server Manager's capabilities, let's look at the dashboard and how you can use it in your day to day management activities.

## The dashboard

One of your primary tasks as an administrator is gathering information. Sometimes, you are troubleshooting. Other times, you are documenting your environment. You need to be able to find out information about your servers. And you need to be able to find it without taking up too much time. The Server Manager's dashboard is the first view that you see when you launch Server Manager. The dashboard provides an optional welcome screen and displays the health of the currently installed roles and server groups. The optional welcome screen, enabled by default, has a welcome screen that provides shortcuts to add roles and features, add servers to manage, create a server group, and connect the local server to cloud services. Figure 3.1 shows a dashboard without the welcome screen. In the screen capture, notice that there is a server group for domain controllers.

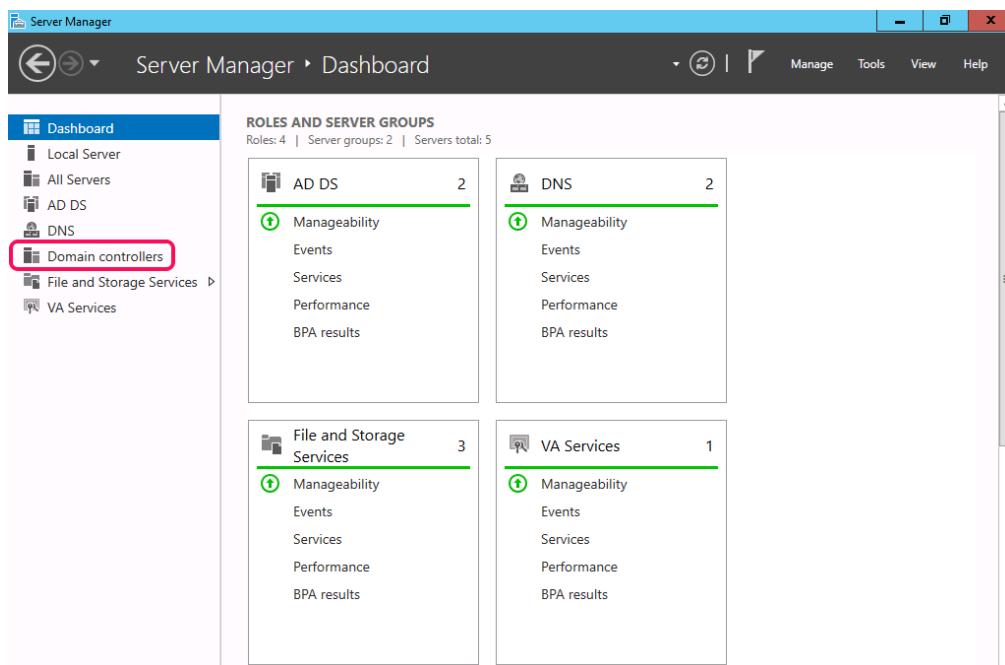


Figure 3.1 A screen capture showing the dashboard view of Server Manager on a Windows Server 2016 server.

The dashboard shows you the following items:

- **Currently installed roles.** If you've installed a role on the local server or any server in one of the server groups, then the role will be displayed in the left pane. And, in the right pane, you will see a management status of the role such as whether it a service is running and whether the role is manageable now. For example, in Figure 3.1, in the left pane is a role named VA Services. That role is installed on one of the domain controllers, not the local server. But it is

manageable from Server Manager on the local server because that domain controller is part of a server group.

- **Server groups.** In our example in Figure 3.1, we have a server group named “Domain controllers” which contains two domain controllers. In the right pane, you see a section for AD DS with the number 2 next to it – this indicates that there are two servers in that group. The green arrow pointing up indicates that the domain controllers are manageable (that means our currently logged on user has valid credentials and access to the servers and the necessary network communication is open).
- **Local server.** The local server and the current overview status is also shown as part of the dashboard. If everything is healthy, it is displayed as shown in Figure 3.1. However, if something isn’t healthy, then the label is shown in red and which category requires attention is shown with a number in a red box next to the troubled area. For example, in Figure 3.2, a service on the local server is not running so a “1” is displayed in a red box next to Services.

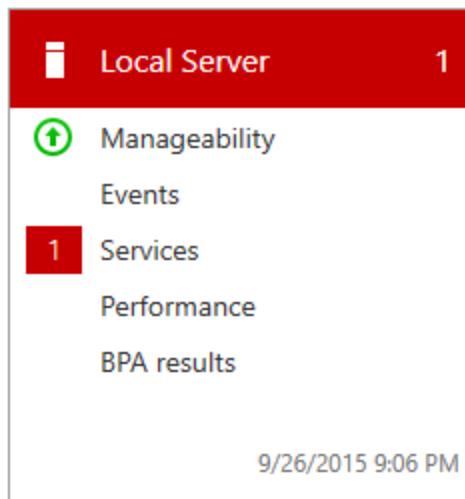


Figure 3.2 A screen capture showing a snippet from the Server Manager dashboard when the local server has a service that is not running.

### Hands-on Exercise

Open Server Manager on a server and look at the dashboard. Compare what you see on your server with what is shown in Figure 3.1 and Figure 3.2.

You can quickly fix minor issues directly from Server Manager’s dashboard. For example, in the case above where we noticed a service isn’t running, you can click Services and bring up the detailed services view of the local server.

From there, you can right-click on the service to bring up a menu to start the service, as shown in Figure 3.3 below.

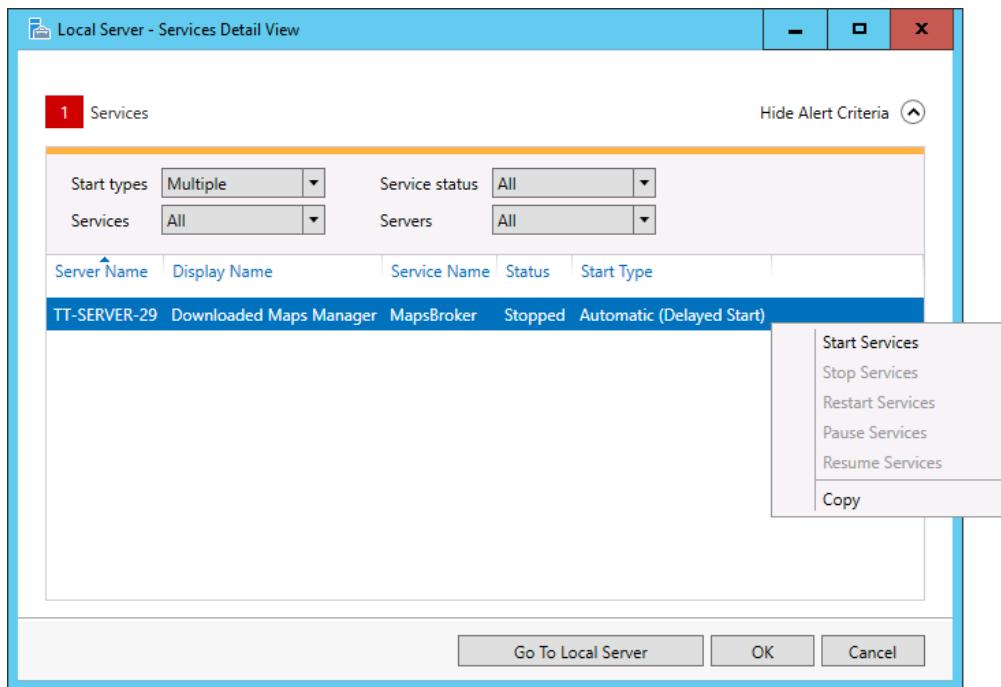


Figure 3.3 A screen capture showing how to start a service using Server Manager's dashboard.

## Managing multiple servers

When you manage one server at a time, you can be effective. But it can be time consuming. As an administrator, you sometimes need a way to configure multiple servers simultaneously. Server Manager is a powerful tool. But it becomes much more powerful when you use it to manage multiple servers. And not only can you manage multiple servers, but you can group them together to create server groups. For example, imagine a scenario where you manage 25 servers. If you add all 25 servers to Server Manager and have only a single server group, the dashboard will be clogged with information. Information overload. Instead, In Server manager, you can create server groups for various role groupings of servers. For example, you can create a server group for domain controllers, a server group for file and print servers, a server group for database servers, and a server group for web servers. Then you can easily manage any of those servers from one Server Manager console (instead of independently connecting to each individual server and running Server Manager on each one).

The first step to managing multiple servers is adding the servers to Server Manager. The Manage menu in Server Manager has a link titled Add Servers and you can use that to browse your domain and find servers. In Figure 3.4, I've searched for all servers that have "tt-" in the front of their name. The search returns 18 computers. In this case, I click to highlight the two utility servers and then click the right arrow to move them to the right pane. Once I click OK, the servers are being managed by Server Manager.

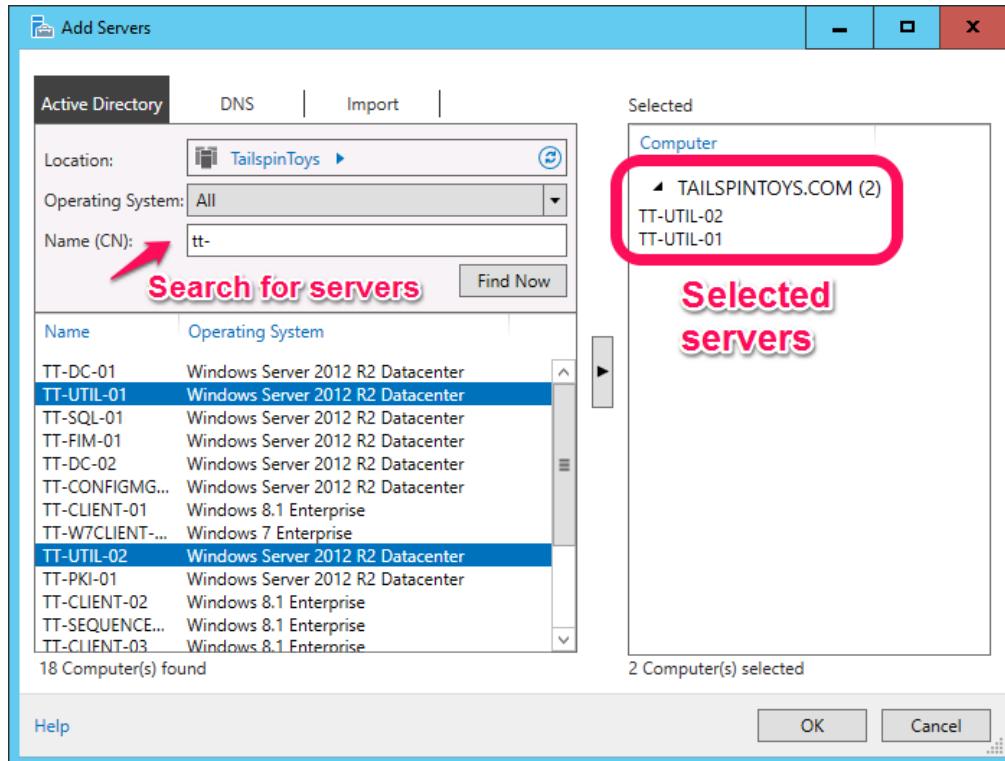


Figure 3.4 A screen capture showing two servers being added for management in Server Manager.

Once added, you can create a server group so that you can view them as a single unit. To create a new server group, click the Manage menu and then click Create Server Group. Then, find the servers in the window, click to highlight them, click the right arrow to move them to the right pane, and then click OK.

In Figure 3.5, two servers have been highlighted and moved to the right pane.

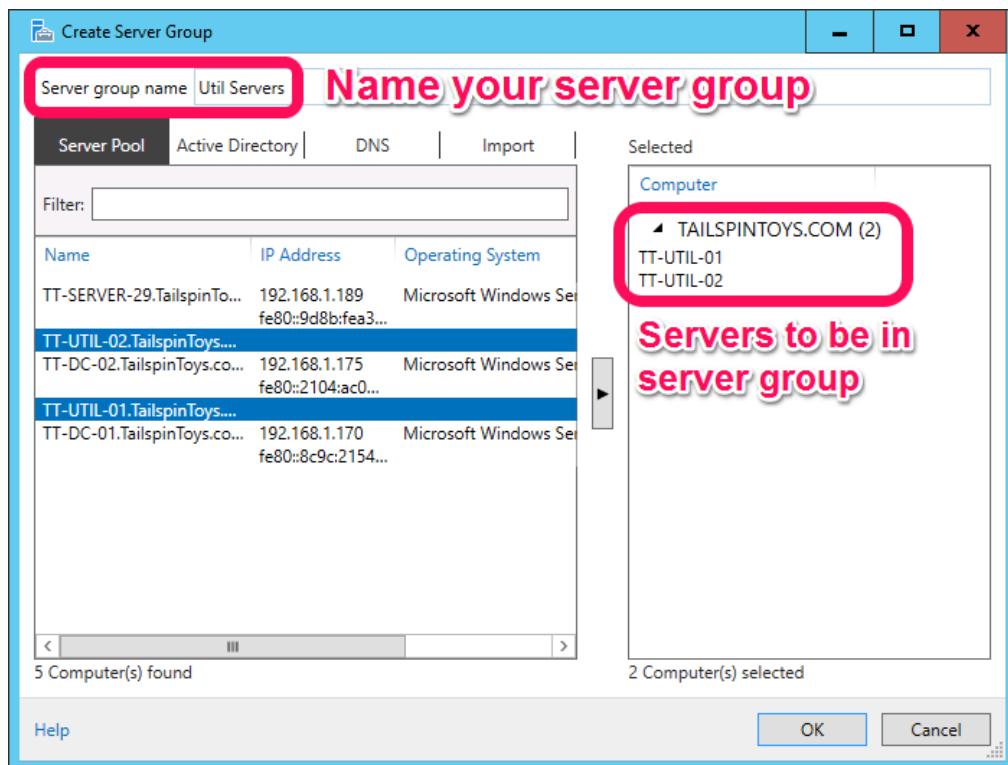


Figure 3.5 A screen capture showing shows two servers being added to a new server group in Server Manager.

### Hands-on Exercise

Add two servers to Server Manager. Then, create a new server group in Server Manager. add the two servers to the group and explore the functionality.

Once you have a server group, you can click it in the left pane and explore some of the management capabilities of the group. Some of the most common tasks to perform with server groups are:

- **View the status of all servers in the group.** In the right pane, you can click one, some, or all the servers in a server group. Thereafter, you can scroll down the right pane to see events, services, Best Practices Analyzer (BPA) scan results, performance data, and roles and features. This is handy. For example, if you noticed a service stopped on a single web server, you can quickly check the status of that service on all other web servers if you grouped them together in a server group. This can help you detect anomalies across similar servers.
- **Open an interactive PowerShell prompt on all servers in the group.** After you highlight servers in a server group, you can right-click them and bring up a large task menu. From there, you can launch a PowerShell prompt across all the highlighted servers. This can be helpful, especially when you need to run a command on each computer.

- **Configure and view performance alerts across all servers in the group.** For example, if you wanted to have an alert generated when the CPU of any domain controllers goes over 95%, you can configure that alert in Server Manager. By grouping similar servers together, such as domain controllers or database servers, you can configure alerts specific for the type of workload the servers are running. For example, database servers often use a large percentage of their RAM so your alert could be configured to generate once memory usage was above 98%. However, web servers don't often use a large percentage of their RAM so you could configure alerts for web servers to be generated at 90%.

## Configuring local server settings

As a server administrator, you will spend much of your time performing configuration tasks. The Local Server settings area of Server Manager gives you access to the same areas as the server groups work area. But, there is one additional configuration item that you get in the Local Server settings which are the properties of the local server. That section enables you to configure several local settings, as described in the subsequent sections. We won't cover all the configuration items in the local server settings but will focus on the core configuration areas. In medium and large organizations, many of these settings are configured automatically as part of a new server build or by using Group Policy. As you go through these settings, you might get a feeling that we are bouncing between many technologies. We are. But, that's because these settings are often configured at the same time when a new server is built or Windows Server is first installed. If you can, follow along in front of a computer so you can click through to the configuration areas while you read.

### Computer name.

Every computer has a name. Officially, this name is called the hostname. Organizations often go with descriptive names that contain identification data. For example, a company with two data centers chose a naming standard that includes the data center location, role, and a designation to indicate whether a server is a virtual server.

Figure 3.6 shows a potential naming standard.



Figure 3.6 A diagram shows two data centers and a server naming standard to identify servers based on location, role, and whether they are virtualized.

To change the name of the local computer by using Server Manager, perform the following steps:

1. Open Server Manager.
2. Click **Local Server** in the left pane.
3. In the right pane, click the current computer name.
4. In the **System Properties** window, click **Change**.
5. In the **Computer name** textbox, type the desired hostname and then click **OK**.
6. In the **Computer Name/Domain Changes** dialog box, click **OK**.
7. In the **System Properties** window, click **Close**.
8. In the **Microsoft Windows** dialog box, click **Restart Now**.

### Hands-on Exercise

Use Server Manager and rename a server. Reboot the server to complete the process and then verify that the name was changed successfully.

### Domain or Workgroup

If you are currently joined to a domain, then the domain name will be shown in the Local Server workspace. If you aren't, then Server Manager will show the name of the workgroup, which is WORKGROUP by default. Before you can join a domain, there are some prerequisites that you must meet:

- **You need to know the FQDN of the domain.** For example, contoso.com or na.tailspintoys.com. Often, the NetBIOS name of the domain (for example, CONTOSO) is sufficient.
- **You need to have permission to join the domain.** Technically, the permission that you need is permission to create computer objects in AD DS. When you join a domain, a computer object is created as part of that process. By default, all users can join up to 10 computers to a domain. But, many organizations change that default setting so that only administrators can join computers to the domain.
- **You need to ensure that the computer can resolve the FQDN of the domain by using DNS.** A quick and easy way to check this is to ping the FQDN of the domain. If the ping shows an IP address as part of the ping results, then you can resolve the name.
- **You need to have administrative permissions on the server that are you configuring to join the domain.** Thus, you must be logged on with an account that is a member of the local Administrators group.

To change the membership of the server's domain, perform the following steps:

1. Open Server Manager.
2. Click **Local Server** in the left pane.
3. In the right pane, click the current domain or workgroup.
4. In the System Properties window, click **Change**.
5. In the **Member of** section, click the **Domain** radio button, type the name of the domain name to join, and then click **OK**.
6. In the Windows Security dialog box, type your domain administrative username and password and then click **OK**.
7. In the **Computer Name/Domain Changes** dialog box indicating that you have joined the domain, click **OK**.
8. In the **Computer Name/Domain Changes** dialog box indicating that you must restart the computer, click **OK**.
9. In the **System Properties** window, click **Close**.
10. In the **Microsoft Windows** dialog box, click **Restart Now**.

### **Hands-on Exercise**

If you have a domain in your test environment, change the membership of a server that is not currently joined to a domain to join your domain. Reboot to complete the process.

## Windows firewall

The Windows firewall is a software-based firewall that inspects network communication and either allow it or deny it based on the firewall configuration. It can also create log files of allowed or denied communication. By default, the Windows firewall in Windows Server (since Windows Server 2008) is enabled and configured to protect the server.

As a server administrator, you will routinely configure the Windows firewall.

To enable or disable the Windows firewall, perform the following steps:

1. Open Server Manager.
2. Click **Local Server** in the left pane.
3. In the right pane, click the status shown next to Windows Firewall.
4. The Windows Firewall Control Panel applet will open. On the left side of the window, click **Turn Windows Firewall on or off**.
5. The **Customize Settings** page will be displayed. To disable the firewall on a profile, click the **Turn off Windows Firewall (not recommended)** radio button in the profile's settings. To enable the firewall on a profile, click the **Turn on Windows Firewall** radio button in the profile's settings. When finished, click **OK**.

## Remote management

Remote management is the management of one computer from a separate and remote computer. Both Windows Server 2012 R2 and Windows Server 2016 are configured to be remotely managed by default. However, many administrators work in an environment with servers running older versions of Windows Server that are not configured for remote management by default or had remote management disabled at some point in the past. To manage a network full of Windows servers, you need to ensure that they can be remotely managed.

Server Manager displays the current state of remote management in the Local Server workspace. It will show as Enabled (remote management currently turned on) or Disabled (remote management currently turned off).

To enable or disable remote management in Server Manager, perform the following steps:

1. Open Server Manager.
2. Click **Local Server** in the left pane.
3. In the right pane, click the status shown next to **Remote management**.
4. The Configure Remote Management window will display a single checkbox which enables remote management. To enable remote management on a server that currently has remote management disabled, click the checkbox to select it.
5. To disable remote management for a server that currently has remote management enabled, click to deselect the remote management checkbox.

6. Click **OK** to save the configuration.

## Remote Desktop

Sometimes, you need to connect to a remote server and configure it as though you were sitting in front of it. Thus, you want the Start menu and all the other functionality you have when you are signed into the console. In such a situation, you can use Remote Desktop, a component built into Windows computers to enable remote connectivity to other Windows computers. Remote Desktop, which is a term that encompasses Remote Desktop Connection (the application you use to connect) and Remote Desktop Protocol (the network communications protocol used by Remote Desktop Connection), is one of the primary administration tools. The other tool is Windows PowerShell which we discuss in Chapter 5 (Getting started with PowerShell).

By default, Windows Server does not accept Remote Desktop connections. Thus, one of your first tasks after installing Windows Server is to turn on and configure Remote Desktop. There are some important things to know about enabling Remote Desktop:

By default, local administrative users (the local Administrator account and any user accounts that are members of the local Administrators group) have permissions to connect to a server by using Remote Desktop. You can configure additional users and groups if desired. Figure 3.7 shows a remote desktop connection to a server at 192.168.1.189.



Figure 3.7 A remote desktop connection to a server at 192.168.1.189.

To enable Remote Desktop, perform the following steps:

1. Open Server Manager.
2. Click **Local Server** in the left pane.
3. In the right pane, click the status shown next to **Remote Desktop**.
4. In the **System Properties** window, under the **Remote Desktop** section, click the **Allow remote connections to this computer** radio button.
5. In the **Remote Desktop Connection** dialog box that indicates that a firewall exception will be enabled, click **OK**.
6. In the **System Properties** window, click **OK**.

### **Hands-on Exercise**

Enable Remote Desktop on a server. Once enabled, use Remote Desktop Connection and try to connect to the server.

### **ABOVE AND BEYOND**

For many years, by default, Remote Desktop created a full Remote Desktop session for all connections to a server. Then, after the connection was established, you were presented with the Windows logon screen just as though you were at the server's console. Starting with Windows Server 2012, Remote Desktop, by default, requires network level authentication. Network level authentication ensures that Windows does not create a desktop session until you are authenticated (entered your username, password, and permissions for the connection are validated). This small change helps improve security by reducing the chance of a denial of service attack on a Windows Server because an unauthorized person without valid credentials cannot establish a full session to the server. It also helps to maximize performance because desktop sessions, which take extra computing resources, are not created until needed, which is after authentication.

### **Ethernet**

Of all the things that you configure on a server, the Ethernet interface (a NIC connected to an Ethernet network) is one of the simplest. Server Manager provides a quick way to access the NIC configuration on a server. When you view the Local Server workspace in Server Manager, you will see Ethernet listed as one of the configuration areas.

When you click it, Server Manager runs the Network Connections Control Panel applet, as shown below in Figure 3.8.

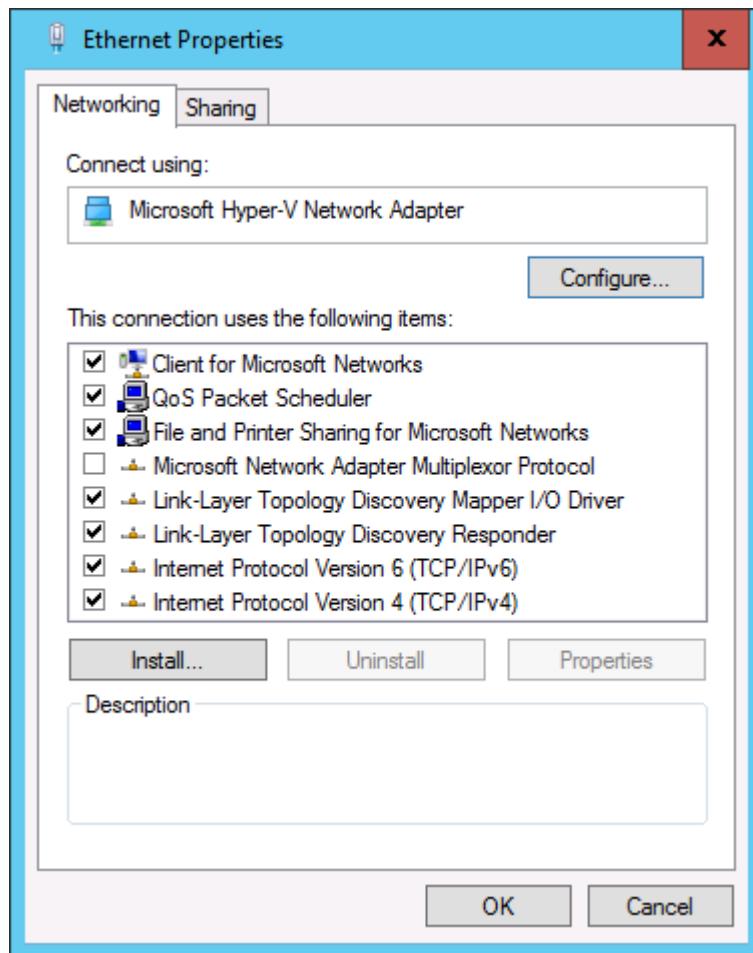


Figure 3.8 A screen capture of the Ethernet configuration window showing configuration items for a NIC.

As a server administrator, you will often be configuring NICs, checking existing settings on NICs, and troubleshooting network connectivity between computers. As you get more advanced in your server administration skills, you may look to other tools to perform NIC configuration tasks such as PowerShell and NETSH.

## Windows Update

One of the most important administrative task, if not the most important, is keeping your computers' software update to date. Vendors, such as Microsoft, routinely release product updates to fix bugs and security vulnerabilities. It is imperative for you to have a plan to evaluate the updates and deploy the updates to all your computers in a timely fashion. Otherwise, you leave your organization exposed.

In smaller organizations, Windows Update is configured on each individual computer. In medium and large sized organizations, Windows Update is usually automated by incorporating Windows Software Update Services (WSUS) or System Center Configuration Manager

(ConfigMgr). Both products provide additional capabilities such as precise reboot windows, the ability to provide self-service to other administrators, and detailed reporting and dashboards.

From a Server Manager perspective, you can click the current Windows Update setting in the Local Server workspace in Server Manager to check for new updates or to change existing settings. Figure 3.9 shows the advanced settings options for Windows Update in Windows Server 2016. Note that these options have changed with Windows Server 2016.

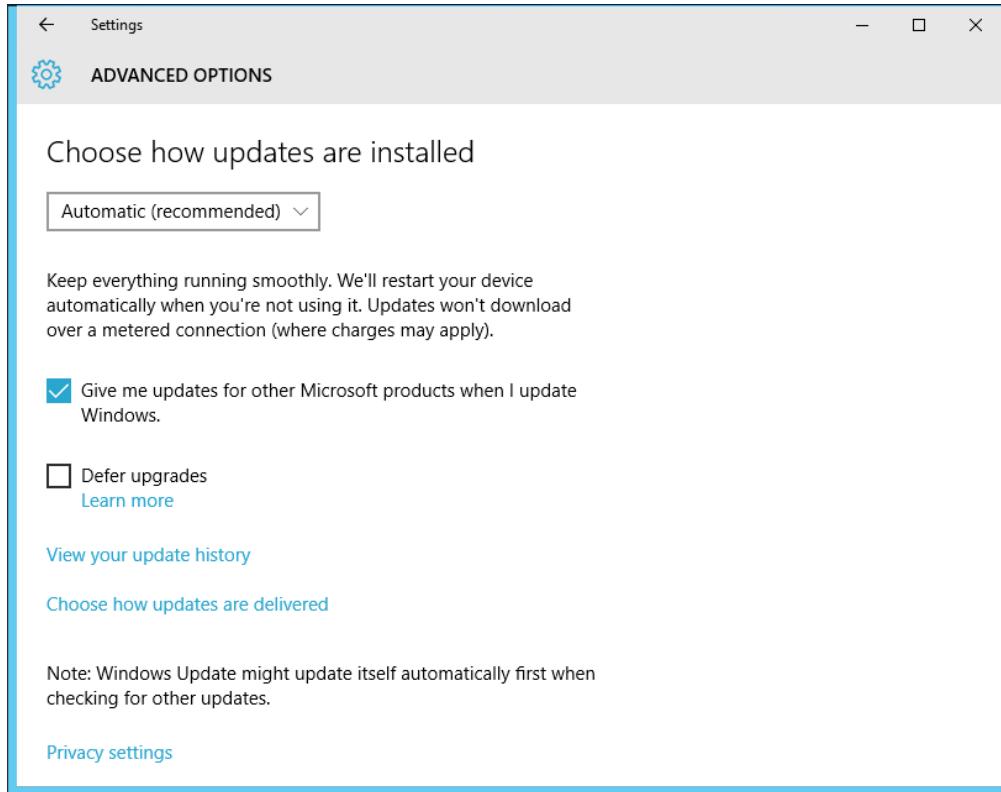


Figure 3.9 Windows Update advanced settings showing a server configured to install updates automatically.

## Windows Server Antimalware

Windows Server Antimalware is the new name for Windows Defender, as of Windows 10 and Windows Server 2016. Windows Server Antimalware is the built-in service that protects your computer against spyware and viruses by providing real-time protection and scheduled malware scans. You can bring up the management console by clicking the status message next to Windows Server Antimalware in the Local Server workspace in Server Manager.

There are a few tasks that you can perform with Windows Server Antimalware:

- **Update the virus and spyware definitions.** Keeping virus and spyware definitions up to date is important. Without the latest definitions, a computer may be vulnerable. While the computer will automatically keep the definitions up to date, if there are network connectivity problems, you may have to manually update the definitions. You can do that by clicking the Update button on the Home screen of the Windows Server Antimalware console.

- **Perform a scan of the computer.** You can run a quick scan, a full scan, or a customer scan. A quick scan looks at the most common areas of infection. A full scan checks just about everything, including every individual file on the computer. A customer scan checks only what you configure it to.
- **View the history of Windows Server Antimalware.** You can see events such as what's been quarantined, which malware has been detected, and which items you've allowed to run on the computer.

### **Hands-on Exercise**

Update the virus and spyware definitions in Windows Server Antimalware. Then, run a full scan of the computer. During the scan, open Task Manager and view the current CPU and memory utilization.

### **IE Enhanced Security Configuration**

IE Enhanced Security Configuration (IE ESC) is a security feature that Microsoft first added to Windows Server for Windows Server 2003. IE ESC provides additional security for the Internet Explorer (IE) browser to help minimize a server's exposure to malware. By default, IE ESC is turned on for administrative users and non-administrative user. You can use Server Manager to turn it off for just administrative users, just non-administrative users, or all users. You should be aware of some of the key impacts to browsing web sites when IE ESC is enabled:

You should be familiar with the following IE ESC impacts, shown in the following table, Table 3.1.

Table 3.1 IE ESC impacts to browser features

Browser feature impacted	IE ESC impact
ActiveX	ActiveX controls are disabled. Add sites to Trusted Sites in IE to fix.
Scripting	Scripting is disabled. Add sites to Trusted Sites in IE to fix.
Intranet sites	Intranet sites are treated like internet sites. To fix, add sites to the Intranet zone in IE.
Downloads	Downloads are disabled. To fix, add sites to Trusted Sites in IE.
Browser add-ons	All browser add-ons are disabled. Turn off IE ESC to fix.

To disable IE ESC, perform the following steps:

1. Open Server Manager.
2. Click **Local Server** in the left pane.
3. In the right pane, click the status shown next to **IE Enhanced Security Configuration**.
4. In the **Internet Explorer Enhanced Security Configuration** window, click the **Off** radio button for administrators and the **Off** radio button for users.
5. Click **OK** to save the IE ESC configuration.

### Hands-on Exercise

Close Internet Explorer. Then, disable IE ESC. After disabling IE ESC, run Internet Explorer. Browse the internet and compare the experience with browsing while IE ESC is enabled.

### Controlling Server Manager startup behavior

By default, Server Manager is shown each time you sign in to a Windows server. Sometimes, this is convenient. Other times, especially after a server has been fully configured, you may not want it to be shown each time you sign in. Instead, you can run it whenever you need it. In

Server Manager, you can perform the following steps to disable Server Manager from starting automatically after sign in:

1. Open Server Manager.
2. Click the **Manage** menu and then click **Server Manager Properties**.
3. In the **Server Manager Properties** window, click the **Do not start Server Manager automatically at logon** option.
4. Click **OK** to save the Server Manager configuration.

In this chapter, we looked at Server Manager's dashboard and showed you how to use it to quickly ascertain the health of your servers. We walked through managing multiple servers which helps you save time performing configuration changes on several servers. We looked at local server settings which had us bouncing around different technologies as part of the initial server configuration. And, we looked at how we can control whether Server Manager starts up after sign in. By now, you probably feel good about your knowledge of Server Manager and its capabilities. But, let's find out by having you try out some server configuration tasks in the lab!

## Lab

This lab is designed to provide hands-on experience configuring a server by using Server Manager.

### Add servers to Server Manager

Add servers to Server Manager by using the following methods:

- Use Active Directory. This is only applicable if the computer is joined to a domain.
- Use the import function. Create a text file containing one server FQDN per line.

### Create a server group

Create a server group named HQ Servers and add two servers to the group.

### Disable IE ESC for administrators

Set IE ESC so that it is disabled for administrators but remains enabled for users.

### Configure Server Manager startup behavior

Configure Server Manager so that it does not start automatically during sign in.

## CHAPTER 4: GETTING STARTED WITH POWERSHELL

---

So far, you've learned about Windows Server including how it stands out from Windows client operating systems. You've learned about the Server Manager tool and many of the tasks you can use it for, such as configuring local server settings. Now, before we get into the day to day management of some of the core Windows roles, features, and services, we are going to talk about Windows PowerShell. The reasons that we are going to talk about it now is because, from here out, we are going to include PowerShell commands throughout the remaining chapters. So, it is important that you are familiar with it before we move on.

Windows PowerShell is one of the primary management technologies built into Windows operating systems. You should think of PowerShell in a couple of ways. One, it is a command shell, just like others such as cmd.exe that have been available since the MS-DOS operating system and like a shell you would find on a Linux computer. Two, it is a scripting language, like other scripting languages such as VBScript and JavaScript. At the shell, you can run commands on demand, just like you would from any command shell. As a scripting language, you can create script files, saved as .ps1 files, and run them from the shell or many other ways such as with a scheduled task. PowerShell is all about IT administration, especially automating administrative tasks (from a shell or in a script), whereas many other scripting languages were built as web (site) technologies or general purpose scripting languages. At its simplest, PowerShell is a command-line console and scripting environment where you use cmdlets or scripts to have a computer perform a task or output information to you. For example, you can run a simple command such as Get-Service to get a list of all the services on the local computer and their status (running, not started, etc.). PowerShell is also able to manage remote computers, which wasn't an easy task with the built-in Windows command prompt. Administrators can use PowerShell to perform basic and routine administrative tasks such as creating a new folder, finding out which programs have been installed, or searching their network for all computers that have less than 4GB of RAM. Administrators can also use PowerShell to create long, complex scripts to automate a function for their organization, such as the onboarding process when a new employee starts (create new user accounts, create an associated mailbox, create a home folder, grant access to the team's web site, etc.).

Microsoft has built many of their applications to be compatible with PowerShell. In fact, many of the applications' GUI management consoles are just front ends for PowerShell, which is executing commands in the background when you perform actions in the GUI management console. Some of the larger applications that have deep support for PowerShell are Exchange, SharePoint, SQL, and Skype for Business. Microsoft now requires that their applications have PowerShell integration. In addition, Microsoft's client and server operating systems since Windows 7 and Windows Server 2008 have PowerShell built in. As a server administrator, you are expected to be familiar with PowerShell and perform some routine administrative tasks with PowerShell. In this chapter, we will show you what PowerShell is all about, what the commands look like, walk through some common management tasks, introduce you to some complex commands, and show you how to find other commands and command parameters. But PowerShell is a big topic. So, we can't cover too much in a single chapter.

## How to work with PowerShell?

Now that you have an idea of what PowerShell is and what kinds of things you can do with it, let's look at the primary methods of working with it. As a shell, you can work with PowerShell two different ways:

- **From the PowerShell console.** The PowerShell console, shown in Figure 4.1, is the standard PowerShell application that resembles a standard command line with a different background color (blue instead of black). Some administrators primarily use the console, especially if most of their work with PowerShell are one-line commands. As an administrator, you need to be able to quickly distinguish between the PowerShell console and a command prompt because PowerShell commands don't run in a command prompt. While most legacy command-line commands run at in a PowerShell console, not all do.

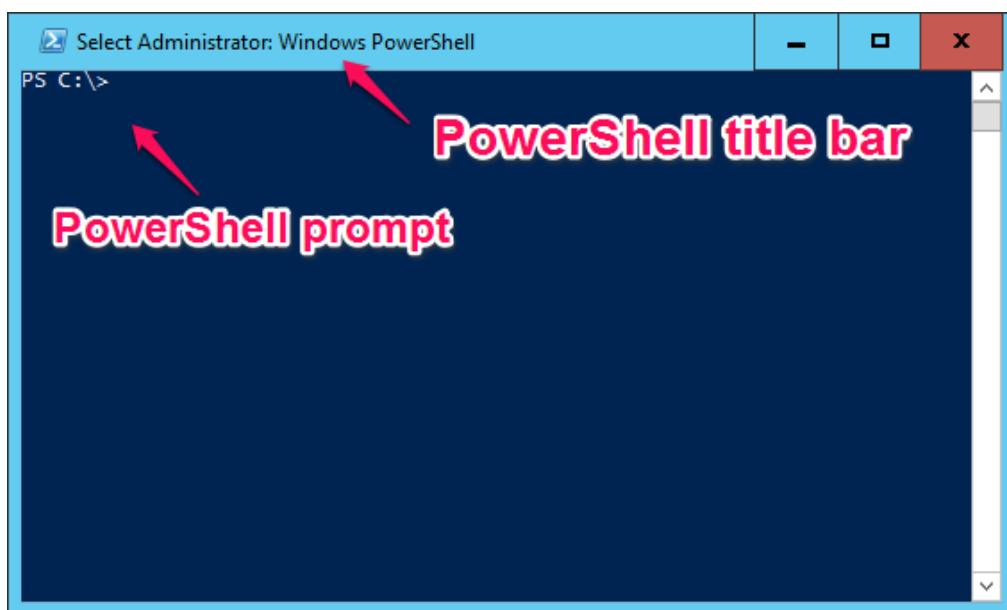


Figure 4.1 The PowerShell console application, which was run as Administrator, is similar in look and feel to the Windows command prompt but has a different background color (blue instead of black).

- **From the Integrated Scripting Environment (ISE) application.** The ISE provides a more advanced environment for running commands and writing scripts. The ISE has automatic command completion, color coding, a list of available commands, and many other features. For example, you can have a script pane at the top of the ISE where you can write multi-lined scripts or commands. Then, on the bottom, you can have the output window where the scripts or commands run. Advanced PowerShell users often prefer the ISE for developing scripts because it offers features for debugging and writing long scripts. While they prefer the console for running interactive commands.

- The PowerShell ISE is shown in Figure 4.2.

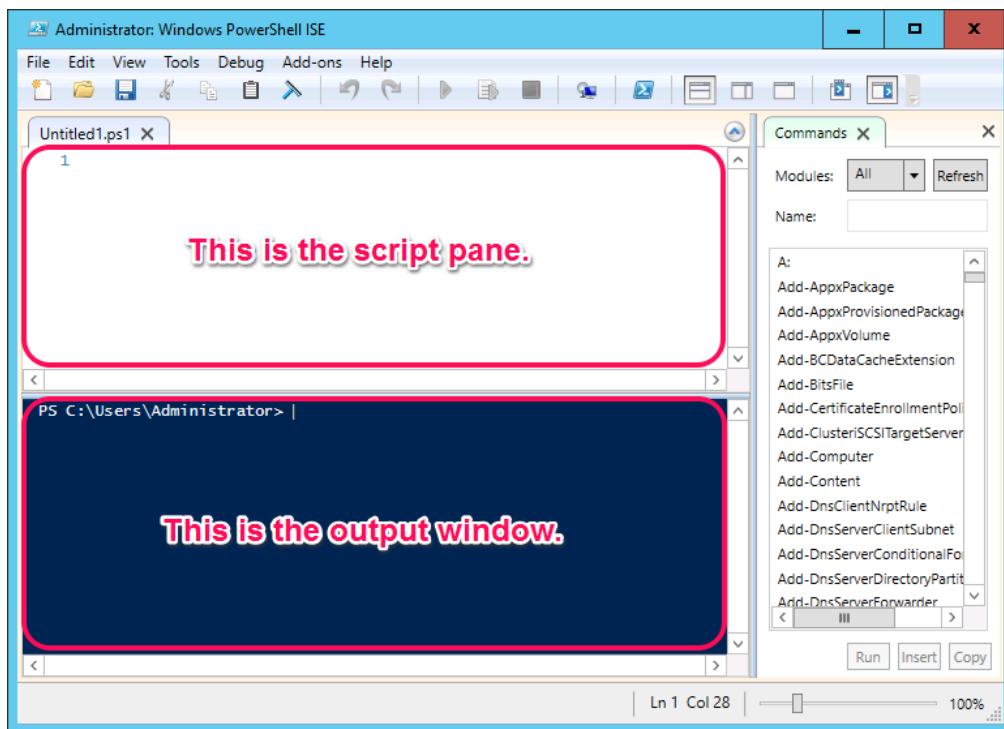


Figure 4.2 The PowerShell ISE application has a script pane where you type commands and an output window where the output of commands and scripts that you run is displayed.

As an administrator, PowerShell is another tool in your toolbox. And, PowerShell is a tool that you are expected to be familiar with and use, even if just at a basic level. Administrators that have advanced PowerShell skills are more valuable to their organization. Why? Because they can automate routine administrative tasks. They can perform complex tasks in a short amount of time. And because of these things, they can spend more time on higher valued activities. For many companies, the value of the administrator is allowing in enabling the company to sell more products or services, bring their product or service to market sooner, have better analytics to support the sales process, and provide better service to existing customers. PowerShell can help administrators do just that.

### Above and Beyond

Back in 2001, Jeffrey Snover, while working at Microsoft, came up with a new methodology for performing administrative work. He called it Monad. His idea was ahead of its time. As he began to evangelize the technology and get others on board, he found that he needed a way to educate others on the details of his technology. So, in 2002, Jeffrey released the *Monad Manifesto*, a document that described his next generation methodology for administrative work. Monad became PowerShell and it grew rapidly until it became a standard for administration across a plethora of products, including the Windows operating system. You can read the *Monad Manifesto* by going to <http://www.jsnover.com/Docs/MonadManifesto.pdf>.

Now that you know what PowerShell is and have an idea of where you can find it and how you can use it, let's dive in and look at how commands are put together. As you go through the next few sections, you should be in front of a computer so that you can try running the example commands found in the rest of the chapter.

## Windows PowerShell syntax

To understand PowerShell syntax, you first must understand what a cmdlet (pronounced 'commandlet') is. A cmdlet is a small self-contained piece of functionality that performs a single task such as fetching data, changing a configuration, or deleting a specific type of object. It is almost always a verb followed by a dash and then followed by a singular noun. By default, PowerShell is case insensitive. However, for readability, we will often use title case or the case displayed in PowerShell's documentation. Get-Service is a cmdlet. The verb is 'Get' and the noun is 'Service'. PowerShell verbs are limited to an approved set of verbs. Some common verbs are Add, Get, New, and Set. You can run the Get-Verb command to see a list of verbs. Or, for a complete list of approved verbs, see [https://technet.microsoft.com/en-us/library/ms714428\(v=vs.85\).aspx](https://technet.microsoft.com/en-us/library/ms714428(v=vs.85).aspx). As you use PowerShell cmdlets, you'll quickly become familiar with them. The verb-noun pairing is a key concept that you should remember as you learn PowerShell. Cmdlets almost always have associated parameters which help identify a target object through filtering, determine which actions should be taken or point to a reference object (such as a text file for importing). For example, while Get-ADUser is a cmdlet to look up an Active Directory user, it doesn't return any data if you don't use any parameters. You can use a parameter to look up a specific user. For example, if you run the following command, you'll get output information about Charles' user account:

```
Get-ADUser -Identity Charles
```

Or, if you want to return all users, you should run the following command:

```
Get-ADUser -Filter *
```

Notice that a parameter is always preceded by a dash. With PowerShell, you'll be using many dashes and symbols. Figure 4.3 shows a command with a verb, noun, parameter, and value.

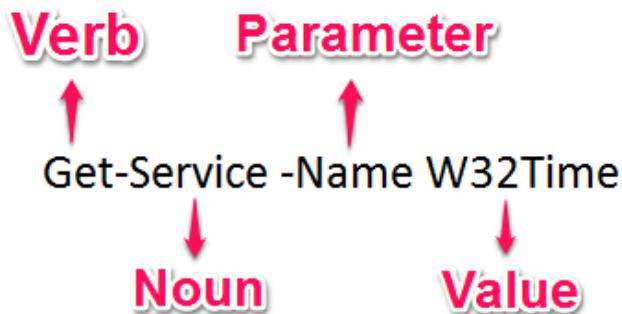


Figure 4.3 A command that shows a verb 'Get', a noun 'Service', a parameter '-Name', and a value 'W32Time'.

Tab completion is a key feature of PowerShell. After you've starting typing a cmdlet name, you can press the Tab key and PowerShell will try to automatically finish the cmdlet name (this works for parameters and other items too). For example, if you type Get-Ser and then press the Tab key, PowerShell will automatically finish the command as Get-Service. This works only if you have typed enough for the completion to be absolute. So, if you type Get-Se and then press the Tab key, PowerShell will automatically complete the command as Get-SecureBootPolicy. If you press the Tab key again, PowerShell will change the command to the next command that starts with Get-Se (which, in this case, is Get-SecureBootUEFI). You can keep pressing the Tab key to cycle through all the commands that start with Get-Se. Not only can you use tab completion for commands, but you can also use it for parameters. For example, if you type Get-Service – (thus a command followed by a space and then a lone dash), you can press Tab and PowerShell will present the first parameter (often –Name). If you keep pressing the Tab key, PowerShell will cycle through all the available parameters. Tab completion is convenient for finding parameter names, remembering command names, and for greatly increasing the speed that you can type complete commands.

### **Hands-on Exercise**

Open the PowerShell prompt (on Windows before Windows Server 2012 and Windows 8 – Start, Run, type PowerShell and then hit the Enter key, on Windows Server 2012 and newer or Windows 8 and newer – click Start, type PowerShell and then hit the Enter key). Then, type Get-Se and then press the Tab key. Press it again and watch how it cycles through to the next matching cmdlet. Continue to press Tab and note how it cycles through and starts back over at the first matching cmdlet. Cycle through the cmdlets until you have Get-Service. Next, type a space and a dash and then press Tab and cycle through the parameters. Cycle through the parameters until you show the –Name parameter. Press the spacebar, type W32Time and then press the Enter key to run the command. If desired, pick another cmdlet by typing a common verb such as Get and a dash and see what else you can find!

PowerShell cmdlets are usually grouped together with other cmdlets that manage a specific technology. A PowerShell module is a grouping of commands such as cmdlets and functions. For example, in the module dedicated to Active Directory (named ActiveDirectory), there are cmdlets for managing Active Directory groups, Active Directory users, and Active Directory domain controllers. A module's functionality is added and removed as a single package. Many of Microsoft's server products, as well as the built-in roles and features, have their own modules. For example, Exchange, SQL, SharePoint, and System Center products all have their own modules with dedicated cmdlets for managing those technologies. Note that these modules are not enabled on a default installation of Windows Server. You need to install the role or feature or the administrative tools for the role or feature to enable the modules.

Figure 4.4 shows a partial listing of cmdlets from the Hyper-V module and the ActiveDirectory module.

Hyper-V module	ActiveDirectory module
Enable-VMEventing Enable-VMIntegrationService Enable-VMMigration <b>Enable-VMRemoteFXPhysicalVideoAdapter</b> Enable-VMReplication Enable-VMResourceMetering Enable-VMSwitchExtension Export-VM Export-VMSnapshot Get-VHD Get-VM Get-VMBios Get-VMComPort Get-VMConnectAccess Get-VMDvdDrive	Remove-ADUser Rename-ADObject Reset-ADServiceAccountPassword Restore-ADObject <b>Revoke-ADAAuthenticationPolicySiloAccess</b> Search-ADAccount <b>Set-ADAccountAuthenticationPolicySilo</b> Set-ADAccountControl Set-ADAccountExpiration Set-ADAccountPassword <b>Set-ADAAuthenticationPolicy</b> <b>Set-ADAAuthenticationPolicySilo</b> Set-ADCentralAccessPolicy Set-ADCentralAccessRule Set-ADClaimTransformLink

Figure 4.4 A partial listing of cmdlets in the Hyper-V module and the ActiveDirectory module is an example of how each module has its own set of commands for managing specific aspects of the operating system or underlying services.

Modules also contain other types of objects such as functions and aliases. Functions, which are often used to supply functionality that is used frequently within a script, are like cmdlets but are written in PowerShell while cmdlets are written in compiled .NET. Aliases are shortened versions of cmdlet names. For example, instead of typing out Format-List, you can type FL (and FL is the alias). Instead of typing out Select-Object, you can type Select (and Select is the alias).

### Commands vs. cmdlets

In its simplest form, PowerShell is a cmdlet followed by a parameter and a value. When you combine a cmdlet and a parameter, it is usually called a ‘command’. When you refer just to the verb-noun pairing, you should call it a cmdlet. PowerShell allows you to string together multiple commands. For example, let’s say that you need to look for all Windows services that are currently running. You can do that with a single line of PowerShell by stringing together two commands, as follows:

```
Get-Service | Sort-Object -Property Status -Descending
```

Don’t worry about understanding the actual command being shown here right now. The idea is to show the structure of stringing commands together.

### Stringing commands together

When you string commands together, you are using what is called the ‘pipeline’. That symbol that you see just before the Sort-Object -Property Status –Descending portion of the command is a pipe. You use a pipe to separate commands. In the example, the objects being returned

from the first command are being piped straight to the second command. Before you try piping a bunch of commands together, be aware that there are limitations and rules that you need to learn about. At this point, just know that many of the commands that look up information can be piped to many of the commands that modify objects. We dive into more about the pipeline and some of the restrictions later in this chapter.

## Symbols in PowerShell

As with many programming and scripting languages, PowerShell has symbols that have special meanings. We've looked a little bit at dashes and pipes. Now, let's look at some of the other symbols and what they are used for in PowerShell. Don't worry if you don't remember any of these immediately. When I learned PowerShell, I printed out a chart with symbols, common parameters, and other key information. That saved me time and helped me learn faster.

Table 4.1 How to use symbols in PowerShell

Symbol	When to use it	Sample command with use
` (back tick/grave accent)	When you have a long command and want to have it continue on the next line.	<code>Get-ADUser -SearchBase 'ou=Boston,dc=contoso,dc=com' -Filter * -Properties -City ` Where {\$_.City -eq 'Philadelphia'}   Select Name,City</code>
\$ (dollar sign)	When you need to declare a variable such as putting all Boston users into a \$Users variable.	<code>\$Users = (Get-ADUser -SearchBase 'ou=Boston,dc=contoso,dc=com' -Filter *)</code>
{ } (curly brackets)	When you need to enclose a block of code. Optional in some situations now, such as using Where-Object with a single comparison.	<code>Get-ADComputer -Filter *   where {\$_.Enabled -ne 'True'}</code>
() (parentheses)	When you want something to execute first or to group expressions.	<code>Get-WmiObject Win32_OperatingSystem -ComputerName (Get-Content .\servers.txt)   Format-List Caption</code>
;(semi-colon)	When you want to run more than one	<code>\$Servers = (Get-Content .\servers.txt) ; \$Servers</code>

	command in a single line.	
% (percent sign)	When you want to use an alias for ForEach-Object (which itself has an alias of foreach). Avoid in scripts to reduce confusion.	1048576,2097512,4194304   % -Process {\$_/1024}
? (question mark)	When you want to use an alias for Where-Object. Avoid in scripts to reduce confusion.	Get-ADComputer -Filter *   ? {\$_.Enabled -ne 'True'}

Table 4.1 PowerShell symbols and their meanings

### Hands-on Exercise

Create a text file named servers.txt. Add one server hostname per line in the text file. Then, open PowerShell and navigate to the folder where servers.txt is located. Then, run the following command:

```
Get-WmiObject Win32_OperatingSystem -ComputerName (Get-Content .\servers.txt) | Format-List Caption
```

Remember that the command inside the parentheses runs first. Now, see if you can come up with any other commands that incorporate parentheses.

If you got anything out of this section, I hope that it is the verb-noun pairing where cmdlet names start with a verb such as 'Get' and end with a noun such as 'Event'. That little tidbit of knowledge will help you remember commands and help you search for commands to see if they exist. In the next section, we will look at some common PowerShell commands and explain how they work.

### Common administrative commands to manage a server

You will rely on some common administrative cmdlets and commands in your day to day administrative work. In this section, I will outline some of the most common commands used by administrators to manage Windows Server. I'll also look a little deeper at some of the details, such as the PowerShell pipeline, that were introduced earlier in this chapter.

## The PowerShell execution policy

The first thing I want to talk about is PowerShell's execution policy because new PowerShell users often run into some confusion when they try to run their first script and it fails. The execution policy determines how PowerShell handles scripts, such as whether it will run them if the scripts aren't digitally signed. By default, Windows Server 2012 R2 and Windows Server 2016 use the RemoteSigned execution policy. This means that PowerShell scripts downloaded from the internet (or a network drive) must be signed by a trusted publisher for you to run them. You will download many scripts that aren't signed so you will want a way to deal with this. Other versions of Windows use the Restricted policy by default. There are six execution policies that you can use, as shown in table 4.2.

Table 4.2 PowerShell's execution policies

Execution policy name	Description
Restricted	Configuration files and scripts do not load or run.
AllSigned	All scripts and configuration files must be digitally signed by a trusted provider, even if they were created on the local server.
RemoteSigned	Scripts and configuration files downloaded from the internet must be digitally signed by a trusted provider. Scripts on the local computer will run.
Unrestricted	All scripts and configuration files can be run without being signed but you get prompted if the source of the files is the internet.
Bypass	All scripts run and all configuration files can be loaded without restriction.
Undefined	The current PowerShell session will not have an execution policy but if a policy is being set by a GPO, then future sessions will adhere to the defined execution policy from the GPO.

There are two primary factors to consider when you decide which execution policy to use. The first factor is security. In a high-security organization, you don't want users downloading (or

accidentally downloading) unsigned PowerShell scripts and then running them on their computers. That's because PowerShell scripts can be powerful. And like any other script, they can be malicious. Even when scripts are signed, you need to trust them based on your level of trust that you have for the signing organization. Always test scripts in nonproduction environments first. The second factor is administrator efficiency. In a fast-paced environment, you don't want to slow your administrators down. If you put too many policies in, too many controls, and too many restrictions, administrators can't get any work done in a timely fashion. Or, they use workarounds that expose your company to risks that you thought you were mitigating with all the policies, controls, and restrictions! It is important to find the right balance and that balance is sometimes different depending on the organization and the type of data that they work with. For example, when you work with highly sensitive data (medical, financial, personally identifiable information), security comes first. For many organizations, the RemoteSigned policy is a good balance of security and getting the job done.

## Other commands for managing a server

There are a myriad of commands available to you. In this section, we'll look at common commands to work with core aspects of a Windows server. We won't be able to touch on most cmdlets though. In Windows Server 2016, there are thousands of cmdlets, functions, and aliases. And that's without adding additional modules! The good news is that you'll be able to work with any of them once you understand the syntax and structure of PowerShell. The first set of commands that we will look at involve Windows hotfixes because you will routinely work with hotfixes on a regular basis as a server administrator. New Microsoft hotfixes are released every month as part of Patch Tuesday and occasionally an out-of-cycle hotfix is released when an urgent security-related bug must be fixed as soon as possible. While this is the current release cycle, there are indications that new release cycle is in the works (one that would be happening more often than monthly). It is important for you to be able to look at which hotfixes have been installed, when they were installed, and who installed the hotfixes.

## Working with Windows hotfixes

The following commands will help you find out which hotfixes have been installed, when they were installed, and who installed them. You will also learn how to filter the output so that only specific hotfix types (such as security updates) are shown.

### Get-Hotfix

This command shows you the Windows hotfixes installed on the server. Very straight forward and simple. But we're going to build on it and show you how it can get a bit more interesting. What we do next you can do with other commands too – filter the output to match the criteria that you are interested in.

```
Get-Hotfix | where InstalledOn -eq '10/18/2015 12:00:00 AM'
```

This command shows you the hotfixes that were installed on 10/18/2015. Note that the time is always shown as 12:00:00 AM so only the date is important. But let's see what else we can do.

```
Get-Hotfix -Description 'Security Update' | where {$_.InstalledOn -eq  
'10/18/2015 12:00:00 AM' -and $_.InstalledBy -eq 'Server22\Administrator'}  
| select HotfixID,InstalledOn,InstalledBy
```

This command shows you hotfixes classified as security updates that were installed on 10/18/2015 by the Administrator account on Server22. The output just shows the hotfix ID, the date of installation, and the account name that installed the hotfixes. The select HotfixID,InstalledOn,InstalledBy command at the end of the pipeline is useful to display only the output that you want to see.

### **Hands-on Exercise**

Look at the hotfixes that are installed on your server by running the following command:

```
Get-Hotfix | select HotfixID,InstalledOn,InstalledBy
```

Now, let's look at a few commands for testing and troubleshooting network connectivity.

### **Troubleshooting network connectivity**

The following commands will help you troubleshoot network connectivity issues.

```
Test-NetConnection -ComputerName TT-UTIL-02
```

This command sends a ping to a computer named TT-UTIL-02 and reports back whether the ping succeeded.

```
Test-NetConnection -ComputerName TT-UTIL-02 -Port 443
```

This command sends a ping to a computer named TT-UTIL-02 and attempts to connect on TCP port 443(HTTPS). This is useful when you want to know if a service is listening on a port and you want to test connectivity to that port to see if a firewall is blocking the communication.

```
$servers = Get-Content .\servers.txt  
Foreach ($server in $servers)  
{  
    Test-NetConnection -ComputerName $server | FT ComputerName,PingSucceeded  
}
```

This small script is a bit more complex. First, you need to have a text file named servers.txt that has a list of server hostnames, one on each line. Then, you can use a for each loop to run the command to have a ping test sent to all the servers and return whether the ping succeeds. If you omit the FT ComputerName,PingSucceeded command, then you get back more verbose output.

```
Resolve-DnsName -Name TT-UTIL-01
```

This command tries to resolve the name TT-UTIL-01 to an IP address by using the DNS servers configured on the computer where you are running the command from.

```
Resolve-DnsName -Name TT-UTIL-01 -CacheOnly
```

This command tries to resolve the name TT-UTIL-01 to an IP address by using the local DNS cache on the computer where you are running the command from.

### **Hands-on Exercise**

Create a text file named servers.txt. Add one hostname per line in the text file. Make sure you have at least one valid hostname and one invalid hostname so that you can see the behavior of the command when there is connectivity and when there isn't connectivity. Then, open the PowerShell prompt, navigate to the folder where servers.txt is, and run the following command:

```
$Servers = (Get-Content .\servers.txt) ; foreach ($Server in $Servers){Test-NetConnection -ComputerName $Server | FT ComputerName,PingSucceeded}
```

Compare the output of the valid hostname and the invalid hostname.

Finally, let's look at a few commands that you may find useful for managing servers.

### **Other server management commands**

The following commands show you a way to look at all Windows services, a way to find .txt files, and a way to view all the PowerShell commands that you've run in the current PowerShell prompt.

#### **Get-History**

This command shows you the list of commands that you have run. This makes it easy for you to see the commands that you ran earlier. But better than that, you can rerun commands easily. For example, let's say that you are troubleshooting network connectivity to a server named Server1. Connectivity fails. Now, after fixing Server1, you want to test connectivity again. Instead of typing the command again, you can find the command in the history by running Get-History. Next to the command is the ID number. Let's say the ID is 45. To rerun the connectivity command, you can run the Invoke-History 45 command. By using an alias, you can do this more easily. Run the R 45 command where R is an alias for Invoke-History.

```
Get-CimInstance -ClassName win32_service | where {$__.StartMode -eq 'Auto' -and $__.State -eq 'Stopped'}
```

This command lists all the Windows services that start up automatically but are currently stopped. It uses the Get-CimInstance cmdlet and specifies the Win32\_Service class. Note that some services start on demand, as needed and thus can't be started manually (or, they can start but will immediately stop). With Windows Management Instrumentation (WMI), there are a bunch of different classes, each pertaining to a different part of Windows software and

hardware. In many cases, you can use Get\_CimInstance to query information that isn't available in native PowerShell.

```
Get-ChildItem C:\Windows -Include *.txt -Recurse | Select Fullname
```

This command finds all .txt files in the C:\Windows folder and all subfolders under C:\Windows. It formats the output as a table, only shows the file name, and automatically sizes the column. You can quickly count how many files it finds by slightly changing the command to:

```
Get-ChildItem C:\Windows -Include *.txt -Recurse.Count
```

### **Hands-on Exercise**

Try using the Count property with other commands such as Get-WmiObject and Get-History. You should find that it is very useful to count things when you are managing a server. Another method to count is using the Measure-Object cmdlet. For example, count the number of services on a computer by running the following command:

```
Get-Service | Measure-Object
```

Counting users, counting computers, counting files, and counting objects that meet certain criteria are all counting tasks that you'll find useful!

In this section, we've looked at some common cmdlets and commands. This will help you get comfortable with the PowerShell syntax and some common cmdlets. The best way to retain this information and expand upon it is to put it into use. While you will get to do that in the upcoming lab exercises, you should also run some commands on your own time, experimenting with command variations along the way. Before I send you off to experiment, I want to quickly introduce the –WhatIf parameter. The –WhatIf parameter can be appended to any command to test the command without making changes on a computer. This enables you to validate that the command is taking the action you think it will take but without taking the action. For example, let's say that you are about to run the following command to delete all .txt files from the C:\Windows folder and all subfolders:

```
(Get-Childitem C:\Windows\*.txt -Recurse | Remove-Item)
```

Before you do, you could run the following command to see a list of files that would be removed:

```
Get-Childitem C:\Windows\*.txt -Recurse | Remove-Item -WhatIf
```

It is a good practice to use the –WhatIf parameter often, especially when you are experimenting or working on critical production servers.

### **Working with the pipeline**

The pipeline has been around for a long time as part of other command-line languages. In fact, some of you may have used a pipe before while working with the Windows command

prompt. For example, if you run a command that returns a lot of output, you can pipe the command to another command to view the output as pages (instead of a large amount of output scrolling by real fast). For example, you can run the following command to get a single page of output for each key press:

```
ipconfig /all | more
```

With PowerShell, the same piping concept applies. And it is even more powerful. You connect two or more commands together into a single line and you have a pipeline. Using the pipeline enables you to shorten and simplify commands. Because you pass objects (the output) from one command to the next, you don't have to get or directly refer to the objects as part of the second command (they are passed automatically from the first command as part of the pipeline). For example, imagine that you want to find all AD users that are locked out and then unlock them. With a pipeline, you could do that by running the following command:

```
Search-ADAccount -LockedOut -UsersOnly | Unlock-ADAccount
```

Without a pipeline, you must store the results from the command in a variable so that you can use the variable to refer to the locked-out users in the second command. Thus, the first command you would run is:

```
$Accounts = Search-ADAccount -LockedOut
```

That command stores all locked out users in the \$Accounts variable. Then you would run:

```
foreach ($Account in $Accounts) {Unlock-ADAccount -Identity $User}
```

Notice how much more complexity is involved when you don't use pipelining? Now, let's define some of the key terms:

- **Pipeline**. When you link together two or more commands with a pipe symbol, it is called a pipeline.
- **Piping**. The act of stringing together two or more commands into a pipeline.
- **Pipeline element**. Each complete command in a pipeline is named a pipeline element.
- **Pipe**. The symbol used to separate pipeline elements.

Let's look at the terms in an actual command to help visualize things. Figure 4.5 shows a command with the elements defined.

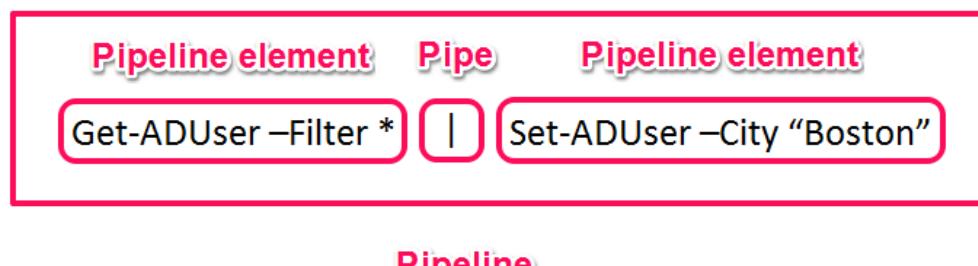


Figure 4.5 A PowerShell command showing the elements defined.

Let's look at a few examples of pipelines. Then, I'll describe them in detail.

```
Get-Process | Sort-Object CPU -Descending
```

This pipeline is simple with just two commands. We get all the processes with Get-Process and then pipe the output to Sort-Object. We sort by CPU, although we could choose to sort by any property. If we don't specify –Descending then the output shows the processes that are using the least amount of CPU first. Instead, we use the –Descending parameter which shows the processes that are using the most amount of CPU first.

```
Get-EventLog -LogName Application -EntryType 'Error' | select  
MachineName,EventID,Message,TimeGenerated | ConvertTo-Html | Out-File  
Errors.htm
```

In this command, we use 4 pipes. At first glance, it may look complex or confusing. But, when we look at the individual pipeline elements (commands), we see a simple series of commands. First, we get all the events in the Application event log. We filter the output using the –EntryType parameter so that we are only passing error events to the next pipeline element. Then, we use select (Select-Object) to reduce the data being returned. If we don't do that, there will be several columns that don't provide much value (for example, the index number) and that makes the output tougher to read. We pipe that to the ConvertTo-Html command which takes the object data and converts it to HTML. Finally, we write the output to a file with the Out-File command. If we don't use the Out-File command at the end, PowerShell automatically assumes that we want to output to the default location which is the console where we are working.

```
Search-ADAccount -LockedOut -UsersOnly | Unlock-ADAccount
```

This pipeline combines two commands. First, we search for locked out users by using Search-ADAccount. Then we pipe the objects found to Unlock-ADAccount. The result? All locked-out users are unlocked. This example, showing a command to find objects followed by a command to act on the objects, is very common.

## Hands-on Exercise

Try running the following command:

```
Get-EventLog -LogName Application | where EntryType -eq 'Error' |  
select MachineName,EventID,Message,TimeGenerated | ConvertTo-Html |  
Out-File Errors.htm
```

Then try changing Out-File Errors.htm to Out-Default. Try removing individual pipeline elements and see how that changes the data or view.

Now that you have a basic understanding of PowerShell, some common commands, and pipelines, it is a good time to show you how to obtain help. Not only will you find it useful for the upcoming lab exercises, but you'll find it useful in your day-to-day work too!

## Using help in PowerShell

When you first begin working with PowerShell, you will need to reference PowerShell's built-in help feature often. In fact, even when you are an experienced PowerShell user, you will still look to the help feature from time to time. Before you begin using the built-in help feature, you need to download the help files to the server. You can do this by running the following command from an elevated PowerShell console:

```
Update-Help -Force
```

This command will download all the initial help files. Or, if you already downloaded help files previously, then the command will update the help files with the latest versions. Once downloaded, you can run the Get-Help command to find specific help. Here are some common examples for getting help:

```
Get-Help Get-Process
```

This command shows you a basic overview of the Get-Process functionality. While this is often useful for seeing what a command does, it lacks the detail needed in many situations.

```
Get-Help Get-Process -Examples
```

This command gets help for the Get-Process cmdlet and shows examples of the cmdlet in common commands. The examples are a big help because they often give you context to the help text. Many times, you will find that one of the examples is the actual command that you need to use!

```
Get-Help Get-Process -Full
```

This command shows you the complete help information available for the Get-Process cmdlet. It includes a full description and all the examples. This is helpful for more complex commands or when you are running a command with a lot of parameters.

Besides getting help for a specific cmdlet, you can find other PowerShell help information too. Let's look at a couple of ways that you can find a cmdlet. Sometimes, you need to see if a cmdlet exists, other times you need to figure out what the actual name of the cmdlet is, and other times you may need to figure out which parameters you can use with a cmdlet.

```
Get-Command *eventlog*
```

This command finds all commands that contain the string 'eventlog'. If you run it, you'll see cmdlets such as Clear-EventLog and Get-EventLog. You can substitute any word or string in place of 'eventlog' such as 'module' or any other verb or noun.

```
Get-Command *Get-* -Module ActiveDirectory
```

This command finds all cmdlets that start with 'Get-' in the module for Active Directory. You could opt not to specify a module, although that will return many results.

```
Get-Module -ListAvailable
```

This command lists all the modules currently available and installed on the server. This is helpful when searching for commands because you can narrow your search to a specific module.

```
Get-Service | Get-Member
```

This command shows you all the properties that you can work with for the cmdlet. In this example, you can quickly see that Get-Service will return the Status property which tells you if the service is currently running.

```
Get-Help about_Pipelines
```

This command shows you a large amount of help information pertaining to pipelines. There are quite a few topics on various aspects of PowerShell. For example, if you run the Get-Help \*about\_\* command, you will see all of them. If you ever get stuck trying to put together a command or script, look at these to see if you can find a solution in the help files.

### **Hands-on Exercise**

In a PowerShell prompt, run the following command:

```
Get-Service | Get-Member
```

In the output, look at the available properties. Then, take a property and get all the services and that property. For example, to list all services and their dependent services, run the following command:

```
Get-Service | select Name,DependentServices
```

Next, try using Get-Help to get help information. For example, run the following commands:

```
Get-Help Get-Process -Examples
```

```
Get-Help about_Pipelines
```

There are a couple of topics that we barely touched on or didn't discuss in this chapter. After you have a good grasp of what we covered in this chapter - PowerShell syntax, basic commands, and the pipeline - you should use the help system to look closely at PowerShell remoting (getting help for the topic: about\_remoting). Now, let's have you run through some PowerShell lab exercises to see everything in action.

## **Lab**

This lab is designed to have you perform some of the tasks that we discussed in this chapter. If you haven't already worked through the Hands-on Exercises in the chapter, you should go through them now, before you start the lab tasks.

### **Identify PowerShell command parts**

Identify the parts of the following PowerShell commands shown in Figure 4.6 and 4.7:

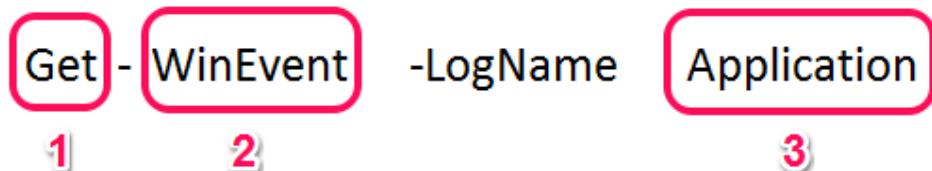


Figure 4.6 A PowerShell command with parts of the command graphically segmented.

- 1.
- 2.
- 3.



Figure 4.7 A PowerShell command with parts of the command graphically segmented.

- 1.
- 2.
- 3.
- 4.

## Work with the execution policy

- Get the current remote execution policy and then set the policy so that there are not any restrictions.

## Perform common administrative tasks with PowerShell

Perform the following tasks by using PowerShell:

- Install the SMTP Server Tools feature.
- Count the number of features currently installed.

## Working with the pipeline

Perform the following task by using PowerShell:

- Get the Application event log, but only list the error events. Hint: first get all events from the Application event log and then find out the property name for finding just errors. Then, you can pipe the command to a filter specifying just errors.

## Getting help

Perform the following tasks by using PowerShell:

- Find all the commands that start with 'Set' in the Microsoft.PowerShell.Management module.
- Find all aliases across all modules.

## CHAPTER 5: ADDING SERVER ROLES AND FEATURES

---

Windows Server, in a default installation, has basic features and functionality such as network connectivity, a storage subsystem, and a file system. But, in most situations, servers are deployed to provide specific services. For example, a company may want to have a web server, a file server, or a server to encrypt Microsoft Office documents. In these examples, a Windows Server role is required. A role is a specific set of files and operating system configurations to provide a service. Like a role, a feature is a specific set of files to provide additional functionality to Windows Server. Think of roles as larger and more complex than features. Think of features as small bits of functionality, such as a network connectivity tool. As an administrator, you will spend a great deal of time adding, removing, and managing roles and features.

In this chapter, we will walk through roles and features with a focus on adding them, removing them, and working with the source files. Working with roles and features will be one of your more common administrative tasks. The focus of the chapter is how to add and remove roles. The functionality that the features and roles provide is not part of this chapter. Instead, we dive into that information later in the book when we dedicate chapters to specific roles and features. So, stay focused on the actions of adding and removing and don't worry about the roles and features themselves! The good news is that you'll find it straight forward, with only a couple of things to watch out for. This chapter wraps up Part 1 of the book. The next chapter, Chapter 6, is the start of Part 2 which focuses on network technologies. At the end of this chapter, we'll test your skills working with roles and features in the lab.

### Overview of server roles and features

There are a multitude of roles and features available in Windows Server. And, with each new version of Windows Server, new roles and features are added. While you don't need to know what every role and feature does, you should be familiar with the most often used roles and features because you will be working with them on a routine basis. Some roles have additional roles underneath them. The roles underneath other roles are called role services and they often complement a role by providing add-on functionality. The following table, table 5.1, describes the functionality of some of the most common roles and features.

Table 5.1 Common roles and features and their functionality

<b>Role/feature name</b>	<b>Role or feature</b>	<b>Role/feature description</b>
Active Directory Domain Services	Role	LDAP compliant directory that contains user, computer, and policy information for a company.

DHCP Server	Role	Automatically assigns TCP/IP settings to computers.
DNS Server	Role	Provides a name resolution service so that common names can be used to locate resources on a network.
Windows Server Update Services	Role	Provides a Windows Update service on the internal network so that computers can get updates without having to go to the internet.
BitLocker Drive Encryption	Feature	Provides disk encryption to hard drives.
Remote Server Administration Tools	Feature	Provides management tools for roles and features so that you can manage servers from any computer.
Windows Server Backup	Feature	Provides a backup application to back up the computer.

Most server roles and features are independent of each other. However, some roles have dependencies that require other roles. For example, to add the Active Directory Rights Management Services role, you need to add the Web Server (IIS) role. The Active Directory Domain Services role requires DNS so you can use the DNS Server role or a third-party DNS solution.

Most roles also have an applicable feature that you can add to manage the role. For example, when you add the DHCP Server role, you should install the DHCP Server Tools feature. You should do this because it enables you to manage the DHCP server locally with a management console. Optionally, if you plan to manage the DHCP server from a different computer, you could add only the role and not the feature (and instead add the feature on another computer).

Figure 5.1 shows the Add Roles and Features Wizard with a partial list of server roles that can be installed on a Windows Server.

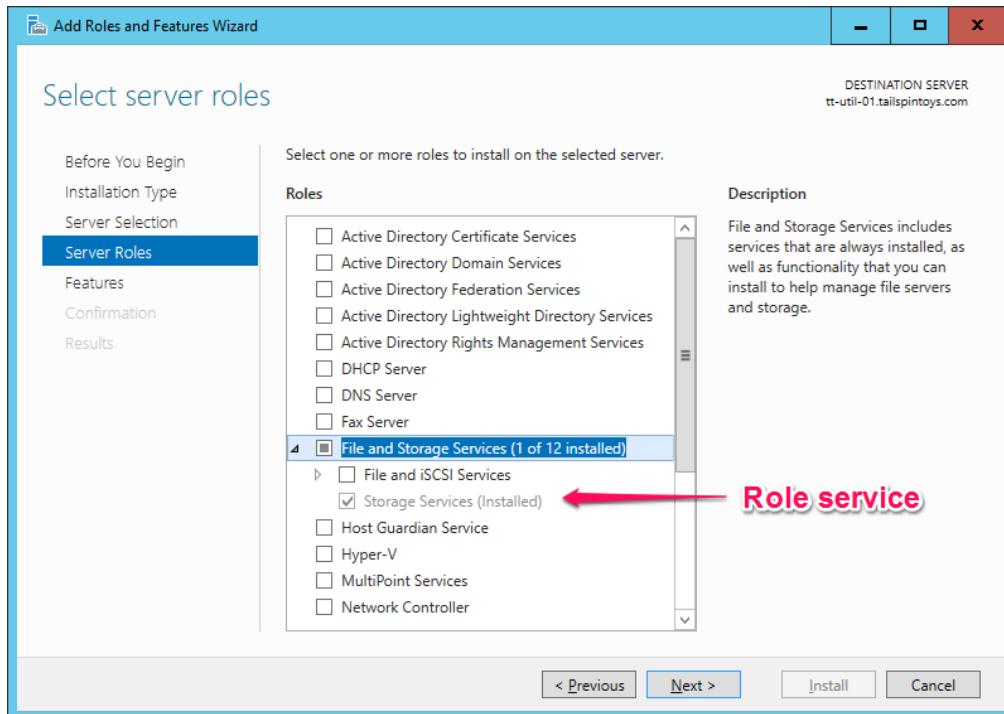


Figure 5.1 A screen capture displaying some of the roles that can be installed.

### Hands-on Exercise

Start the process of adding a role or feature. When you get to the Server Roles page, highlight the roles and read the description on the right side. Thereafter, move to the Features page and look through the descriptions of the features. Once finished, cancel the wizard.

Now that you understand roles and features, let's look at the process of adding them to a server.

### Adding roles and features in the GUI

You can use Server Manager or Windows PowerShell to add Roles and features can be added by using Server Manager or by using Windows PowerShell. We'll look at the PowerShell methods in the next section. In Server Manager, the Manage menu presents the option to add a role or feature. The You use the Add Roles and Features Wizard is used to walk through the installation of a role or feature, which follows the following high level steps:

- Choose the installation type.** You can choose role-based or feature-based (which is what we are covering in this chapter). Or, you can choose a Remote Desktop Service

installation, which is used to install services needed for a Virtual Desktop Infrastructure (VDI) which we will not discuss in this book.

2. **Then, you choose which server that you want to install the role or feature on.** If you've added multiple servers to Server Manager, you can select them. Or, you can just accept the default which is the local server.
3. **Next, you choose the role(s).** You can add one or more roles at a time. Some roles may prompt you to add additional features and management tools.
4. **Then, you choose the feature(s).** Like roles, you can add multiple features at a time if desired.
5. **If the role is a complex role (large, involving a lot of services), you may get prompted for additional setup information.** For simple roles and for most features, you will not get prompted for additional information.
6. **Finish the installation.** You finish the installation by clicking Install.

When a role that requires other roles or features is selected, it will automatically prompt you to add the additional roles or features. For example, when you install the DHCP Server role, a dialog box will prompt you to install the management tools for DHCP. Figure 5.2 displays the Add Roles and Features Wizard dialog box.

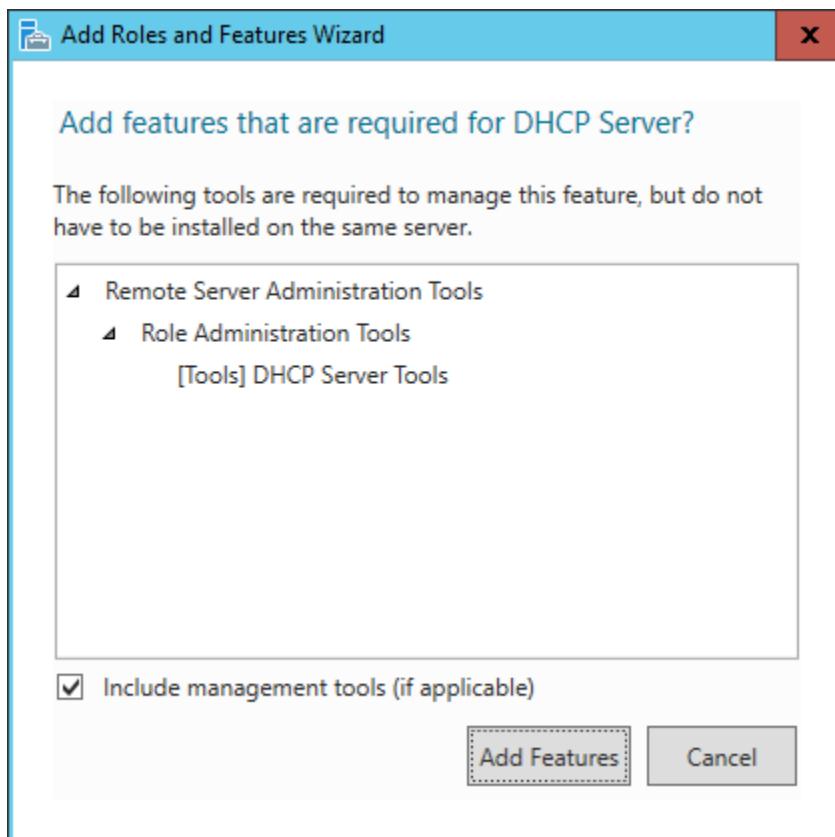


Figure 5.2 A screen capture that displays adding the DHCP Server Tools feature that must be used to manage a DHCP Server.

Usually, management tools are not required to be installed on the same server that a role is being installed on. However, it is often a good practice to have the management tools installed locally in case there are issues preventing you from managing the server remotely.

### **ABOVE AND BEYOND**

To maximize security, you should perform remote management from another computer whenever possible. Some organizations create administrative zones where highly secure computers are used to perform all management tasks. This increases security because remote administration often reduces the need to browse the internet for downloads or updates from a server, administrative computers can be configured for additional security and auditing, and because you may not have to install management software or third-party management software on the server. Remember, the more software that you add to a server, the more software that you must update with bug fixes and security fixes. Finally, by moving management off the server, you can more easily use the Server Core installation of Windows Server, which eliminates most graphical functions and Internet Explorer to further enhance security.

### **HANDS-ON EXERCISE**

Add the Windows Server Update Services role to a server. Pay close attention to dependencies and options as you go. Refer to high-level steps showing how to add roles and features from earlier in this section.

Now that you have a good feel for working with roles and features in the GUI, let's look at working with roles and features by using PowerShell.

### **Working with roles and features by using PowerShell**

In the previous section, we looked at working with roles and features in the GUI. You learned how to add them, remove them, and perform other related tasks. But most of those tasks used Server Manager. Now, let's look at how to work with roles and features in PowerShell.

You can add a role or feature by using the `Install-WindowsFeature` cmdlet. Let's look at a few examples and then break them down. I chose these examples because they represent common commands that you will use when working with roles and features. The goal isn't to learn about the role or feature here but to learn about the PowerShell syntax and structure and to make sure you are comfortable working with roles and features by using PowerShell.

```
Install-WindowsFeature -Name Print-Services -IncludeAllSubFeature -  
IncludeManagementTools -Restart
```

This command installs the Print Services role, all role services, the management tools, and restarts the server if a restart is required as part of the role installation. This command has a lot of parameters. The `-Name` parameter is used to name the role that you want to install. The `-`

IncludeAllSubFeature parameter ensures that all sub features for the role are installed. In this case, the Print and Document Services role (that is the friendly name of the role) has a Print Server role, a Distributed Scan Server role, an Internet Printing role, and an LDP Service role underneath it. The –IncludeManagementTools parameter ensures that the management console(s) that are needed to manage the role and role services are installed with the role. The –Restart parameter will restart the computer after the feature or role is installed if required. Some roles and features require a reboot to complete the installation although many of them don't.

### **Hands-on Exercise**

Run Windows PowerShell as Administrator. Then, at the PowerShell prompt, install the Telnet Client feature by running the following command:

```
Install-WindowsFeature -Name Telnet-Client
```

One thing that you might notice with some features is that the source files aren't always available on the computer. Look at the following command:

```
Install-WindowsFeature -Name Net-Framework-Features -  
IncludeAllSubfeature -Source D:\Sources\SxS
```

The command is like the previous command. However, in this example, we are installing a feature (.NET Framework 3.5 Features) that doesn't have all the source files on the computer by default. Thus, we must use the –Source parameter and specify the path to the source files. If you ever run into an error trying to install a feature, look carefully at the error message because it could indicate that the source files weren't found.

```
Get-WindowsFeature | where {$_ .FeatureType -eq 'Feature' -and $_ .Installed  
-eq '$true'} | select DisplayName, InstallState
```

This command retrieves a list of all features that are currently installed and then displays the display name and the installation state. The \$\_ represents the object that is in the pipeline (in this case, the role/feature from the Get-WindowsFeature command). The select command at the end of the pipeline filters the output to the properties specified (in this case, DisplayName and InstallState).

Figure 5.3 shows the output of this command on a Windows Server 2016 server.

DisplayName	InstallState
.NET Framework 3.5 Features	Installed
.NET Framework 3.5 (includes .NET 2.0 and 3.0)	Installed
HTTP Activation	Installed
Non-HTTP Activation	Installed
.NET Framework 4.6 Features	Installed
.NET Framework 4.6	Installed
ASP.NET 4.6	Installed
WCF Services	Installed
TCP Port Sharing	Installed
Ink and Handwriting Services	Installed
Media Foundation	Installed
SMB 1.0/CIFS File Sharing Support	Installed
User Interfaces and Infrastructure	Installed
Graphical Management Tools and Infrastructure	Installed
Desktop Experience	Installed
Server Graphical Shell	Installed
Windows PowerShell	Installed
Windows PowerShell 5.0	Installed
Windows PowerShell 2.0 Engine	Installed
Windows PowerShell ISE	Installed
Windows Process Activation Service	Installed
Process Model	Installed
.NET Environment 3.5	Installed
Configuration APIs	Installed
Windows Server Antimalware Features	Installed
Windows Server Antimalware	Installed
GUI for Windows Server Antimalware	Installed
WoW64 Support	Installed

Figure 5.3 The output of a PowerShell command to find installed features

### Hands-on Exercise

Experiment with filtering and the Get-WindowsFeature cmdlet. First, run the Get-WindowsFeature command. Note that it shows all roles and features and whether they are installed. Next, run the following command:

```
Get-WindowsFeature | where {$__.FeatureType -eq 'Role' -and $_.Installed -eq $true}
```

This command finds just roles that are currently installed. Finally, try changing the command slightly and compare the output. For example, change \$true to \$false and run the command again.

By now, you should have a good understanding of roles and features and know how to add a role and feature to a server. Next, let's look at how you remove roles and features. The process is quite similar.

### Removing server roles and features

Removing a role or feature from a server is like adding one. To remove a role or feature by using Server Manager, use the Manage menu. The Remove Roles and Features wizard will be

displayed. Remove the checkmark next to the role or feature that you want to remove and then click Remove. Figure 5.4 shows the Remove Roles and Features wizard with the DHCP Server role being removed.

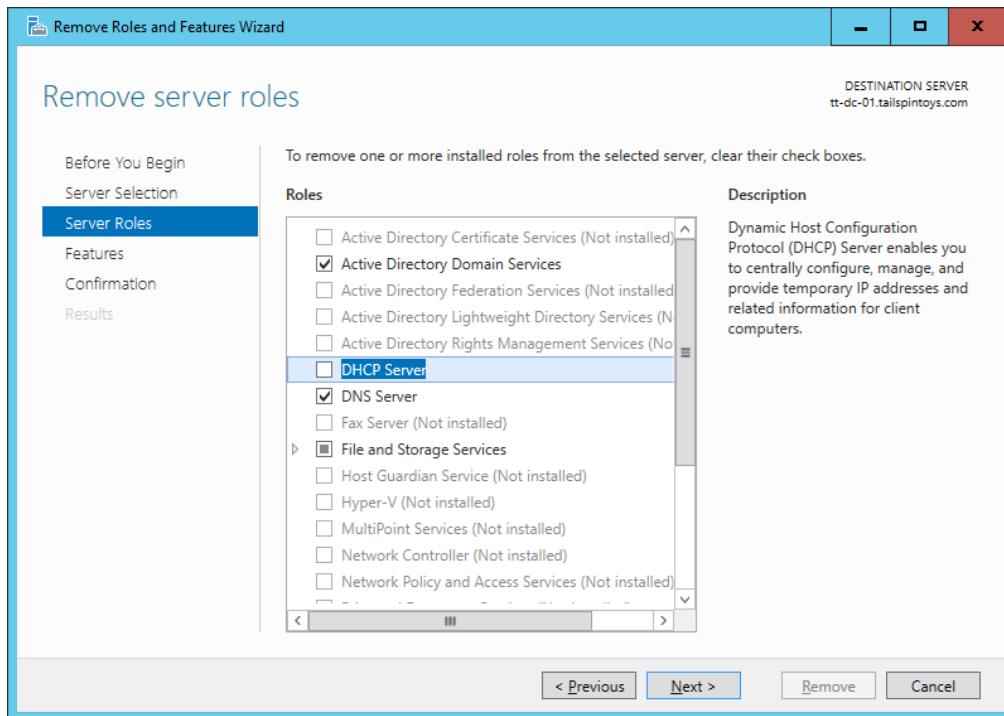


Figure 5.4 A screen capture shows the Remove Roles and Features wizard, with the DHCP Server role being removed from the server.

### Hands-on Exercise

Use Server Manager to remove the Windows Server Update Services role that you installed earlier as part of a Hands-on Exercise.

At this point, you should be comfortable with adding and removing roles and features. Now, let's look at a situation that deviates from the typical installation process.

### Working with source files and Features on Demand

Back in the old days, with Windows Server 2003 and earlier, when you wanted to add components to Windows Server, you had to insert the installation disk. But first you had to find it! Or, you sometimes could manually copy copied the necessary files to a server or a network share. But now, by default, the source files that you need to install most roles and features are located on a default installation of Windows Server. But notice how I said most. There is at least one feature that cannot be installed without specifying the source files, which will be the installation disk or a network file share. In Figure 5.5, a screen capture shows the warning message displayed when you try to add the .NET Framework 3.5 Features feature (for which

source files are not available by default). Don't worry about knowing what the .NET Framework 3.5 Features feature provides. For right now, we are strictly focused on understanding how to specify source files when you need to.

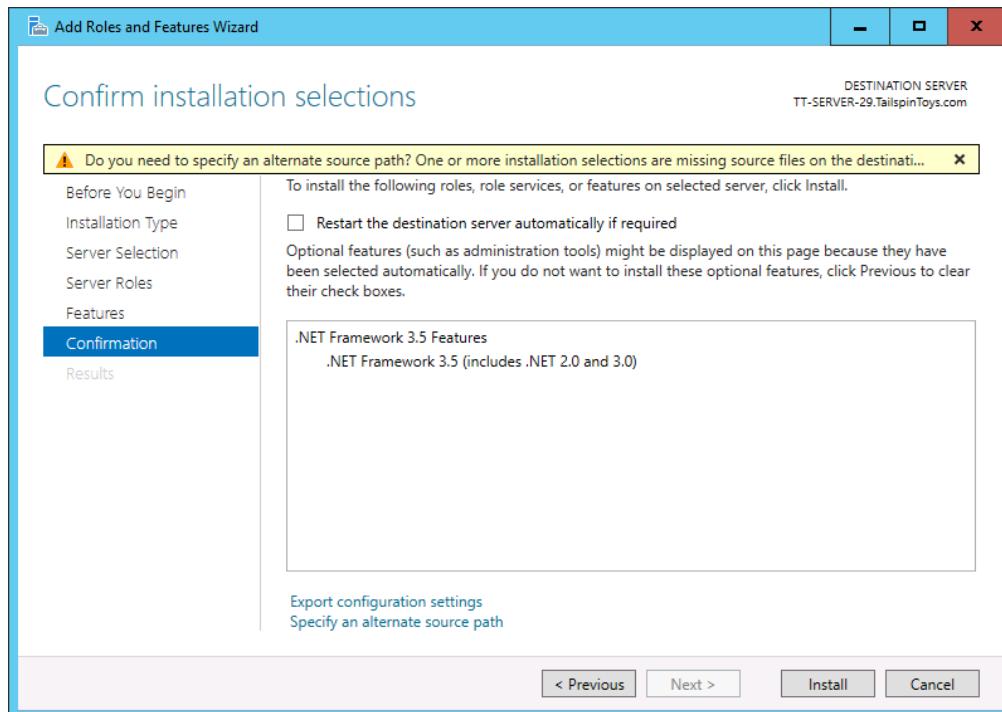


Figure 5.5 A screen capture shows the warning message that is displayed when you try to add a role or feature and the source files are not on the local computer.

You can insert the Windows Server installation disk, then click the small link titled "Specify an alternate source path" to point to the folder that has the source files. The folder is the Sources\SxS folder. Alternatively, you can copy that entire folder to a network file share or the local server so that you can reuse the source files from other servers later, if needed.

What if you don't want to store all those source files on every server? Features on Demand is a feature of Windows Server that allows you to remove the source files for the roles and features! This saves space on the hard drive. To save disk space, some administrators choose to remove the source files on servers and centralize the source files on a file server. In such a situation, administrators can point to the file server when they need to install roles and features. Today, most servers have very large hard drives and disk space isn't a huge concern. So, saving a few gigabytes of space may not be worth the work of removing all the source files.

To uninstall the source files for a role, you can use the Uninstall-WindowsFeature cmdlet. For example, to uninstall the Telnet Client source files, run the following command:

```
Uninstall-WindowsFeature -Name Telnet-Client -Remove
```

In this chapter, we described what roles and features are and what purpose they serve. We showed you how to add a role and feature and how to remove a role and feature. We also looked at how the source files come into play and what you can do if a role or feature can't be installed without using the source files. This concludes Part 1 of the book which focused on the

fundamentals of Windows Server. In the next part, Part 2, we look at network concepts from Chapter 6 through Chapter 11. But before that, let's have you try some roles and features tasks in the lab.

## Lab

This lab is designed to allow you to perform some of the tasks that we discussed in this chapter. If you haven't already performed the Hands-on Exercises throughout the chapter, you should go through them now before you begin the lab exercises.

### Add a role by using Server Manager

Add the following role. Ensure that you use Server Manager and make sure you install the management tools.

- Add the DNS Server role

### Remove a role by using Server Manager

Remove the DNS Server role. Don't forget to remove the management tool too.

### Add the .NET Framework 3.5 Features feature

Add the .NET Framework 3.5 Features feature. Ensure that you use Server Manager and make sure to point to a valid location for the necessary source files.

### Uninstall the source files for a role

Uninstall the Remote Access role source files by using Windows PowerShell.

## CHAPTER 6: NETWORKING FUNDAMENTALS

---

As a server administrator, you will often work with network technologies. You will connect servers to networks such as when you buy a new server or move a server from one place to another. You will enable and troubleshoot services that communicate over a network such as email. You will configure highly available communication so that a server can have a NIC fail without impacting network communication. And, you will secure network communication to keep your company's data safe.

This chapter is the beginning of Part 2 of the book – the networking topics! Part 2 is made up of 6 chapters around network concepts such as DHCP and DNS. In this chapter, we will focus on the networking fundamentals that will get you familiar with the core networking technologies so that you can perform routine administrative tasks. First, we will look at subnets, which are defined portions of a network. Then, we will talk about protocols (a set of rules to enable computers to communicate) and ports (a set of numbers defined for specific services that dictate how computers communicate). In the third section, we will walk through teaming NICs together to provide highly available network connectivity. Then, we will look at securing network communication so that sensitive information isn't sent between computers without protection. To wrap up, we will run through some networking fundamentals tasks in the lab. Be advised that this chapter is one of the tougher chapters for many readers. That's okay. If you haven't been exposed to networking, many of the concepts in this chapter may be difficult initially. After you finish this chapter, the remaining networking chapters will help solidify many of the concepts that we discuss here.

### Working with Subnets

For this chapter and all networking chapters, we are going to focus on IP version 4 (IPv4) networks, which are what most computers use to communicate with each other today. While IPv6 is the future of IP networks, we will only introduce parts of it in this book because so few organizations are using it on a large scale right now. And remember, this book is designed to give you information that you can use today, not sometime in the future! When you have a network and you divide it up into smaller subnetworks (subnets), you are subnetting (the verb for creating subnets from a larger network). The definition of "network" is a little bit confusing. In a large company with several locations, the "company network" is the entire network, including all locations. But each location has its own network which is made up of just the computers and network devices at that location. For the purposes of our discussion, a network is a group of connected computing devices. Any computing devices that are beyond a router are on another network.

Figure 6.1 shows two networks that have been divided into 4 subnets each.

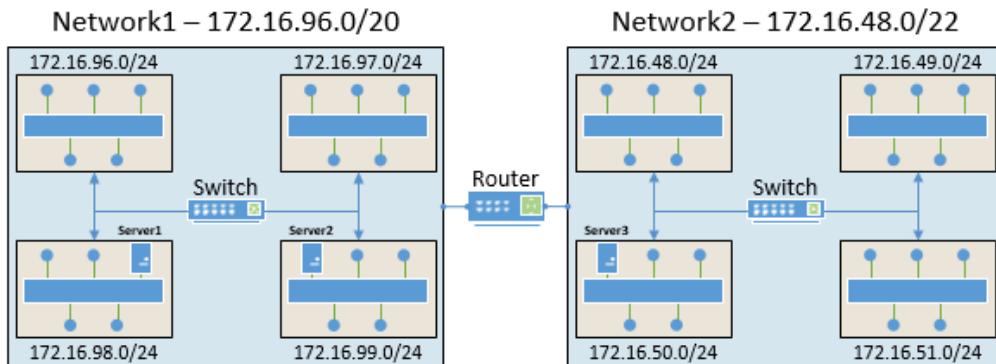


Figure 6.1 A network named Network1 has been divided into 4 subnets, Subnet1, Subnet2, Subnet3, and Subnet4.

Each computing device on a network has a subnet mask which helps identify the subnet that the computer is part of. For example, a computer named Server1 has an IP address of 172.16.98.25 and a subnet mask of 255.255.255.0 (remember, a /24 uses a subnet mask of 255.255.255.0). Server1 is on the 172.16.98.0/24 subnet in Network1. While Server2 has the same subnet mask as Server1, it is on the 172.16.99.0/24 subnet (still part of Network1). In the simplest of networks, such as those with hubs and not switches, a subnet mask is the primary method of identifying and segmenting computers. But in larger networks, it is common to find networks that have many subnets that all use the same subnet mask. You can refer to subnets by using Classless Inter-Domain Routing (CIDR) notation, which is a way to write out a network's identification. In our example, the subnet is 192.168.100.0/24. Don't worry about learning the ins and outs of CIDR notation at this point. Instead, you just need to know that it exists and what it means.

For IPv6, breaking large networks into smaller networks is a little harder than IPv4. This is because IPv6 is 128-bit addressing scheme while IPv4 is a 32-bit addressing scheme. Often, you will be assigned a block of IP addresses from an internet service provider (ISP). Today, it is a common practice to get a /48 prefix from the ISP for IPv6. You can then break that down into smaller subnets based on your needs. Subnet calculators function for IPv4 and IPv6. But, because IPv6 is 128-bit addressing scheme, it is tough to talk small subnets. The current accepted practice today is to subnet without regard for IP address conservation. It is quite common for organizations to use /96 IPv6 subnets for all their networks, even though each /96 can contain over 4 billion hosts. So, for now, you won't do nearly as much subnetting in IPv6 as you will in IPv4. Instead, you'll get a block of IP addresses from your provider and then use /96 subnets everywhere.

### Hands-on Exercise

Open a command prompt and run the following command:

```
ipconfig /all
```

Scroll up to find your IP address and your subnet mask. Then, write out your subnet in CIDR notation. For example, if your IP address is 192.168.1.133 and your subnet mask is 255.255.255.0, then your subnet is 192.168.1.0/24.

In most networks, network switches are used to connect computing devices together. Most network switches are managed switches which means that they can be configured based on your needs (such as dividing a network up into subnets). Hubs, on the other hand, are not configurable and don't offer nearly as many features and security that most companies need today. In most switched networks, every subnet is associated with a virtual LAN (VLAN). A VLAN is a software-defined network segment which is usually configured on a network switch. For this chapter, a subnet and a VLAN are interchangeable. When we say subnet, we also infer that there is an associated VLAN and if we say VLAN, we also mean that there is an associated subnet. As a server administrator, you won't often deal with creating or troubleshooting VLANs because a dedicated network guy (or team) typically handles them. Instead, you need to know how to configure your servers to be on the right subnet and how to identify subnets. And occasionally, you need to know how to break up a larger network into smaller networks. For example, you may need to create a small network for a failover cluster or while creating a development environment in Hyper-V.

In Figure 6.2, a company's office building is shown with the subnets that they've created.

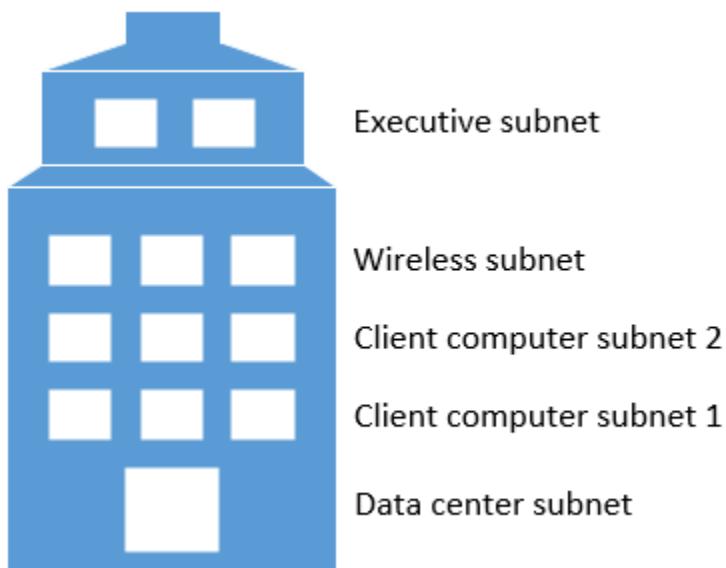


Figure 6.2 A company's office building with subnets defined for different floors.

Now that you have an idea of what a subnet is and how they are commonly used, let's talk about why you break up networks into subnets.



The benefits of subnets are:

- **Minimize the impact of broadcast storms or similar issues.** When computers communicate on a network, sometimes they don't specify a specific computer to talk to and instead broadcast their communication to all computers. Such communication is called a broadcast. For example, when a computer first starts up, it uses broadcast communication to find a DHCP server and obtain an IP address. When there is a very large amount of broadcast traffic on a network, it is called a broadcast storm. When you have a single network, a broadcast storm can render the network unusable. So how does subnetting help during a broadcast storm? When you have multiple subnets, a broadcast storm is limited to the subnet where the storm began while all other subnets are not impacted. Flip back to Figure 6.1 to visualize the scenario. Imagine a broadcast storm on the 172.16.98.0/24 subnet. While that subnet would be impacted, the other 3 subnets on Network1 would not be impacted.
- **Improve the security of your network.** When you have a single network, you are limited in your options to use security policies to limit or block some types of communication across the network. However, when you use subnets, you can apply security policies for network communication that goes between subnets. This is because traffic between subnets goes through a gateway (often a router) whereas communication in a single network is often direct or through a switch. Let's look at a real-world example. Let's say that I have a VLAN for all servers and a VLAN for all client computers. To enhance security, I create a VLAN for IT administrators and I deploy some highly secure administrative computers to the VLAN. Now, I can block administrative network communication (such as RDP) between the client computer VLAN and the server VLAN because all administration will originate from the VLAN for IT administrators. Blocking unnecessary communication between subnets can help to minimize the spread of viruses and malware which both often try to infect every computer that they can connect to.
- **Improve the performance of your network.** Imagine a scenario where you have a single network. Your regular day-to-day network communication (such as users signing into their computers) uses the network. And your servers use the network to back up data. And your telephones use the network for voice traffic. During the time of backups, your network becomes congested and users report poor phone quality. You can minimize the chances of this happening by using subnets. Using multiple subnets allows you to apply policies to the communication. You can configure your network so that IP telephony traffic is prioritized over all other traffic. You can configure backup traffic to have less priority. When there is congestion, IP telephony traffic is prioritized and thus performs better.

When you need to figure out the subnet mask or the CIDR notation for a network, such as when you are preparing to open a new office location or when your network grows and needs additional IP addresses, you can use a subnet table.

Table 6.1 shows common subnet information.

Table 6.1 Subnet notations and subnet masks

<b>Subnet notation</b>	<b>Subnet mask</b>	<b>Subnet notation</b>	<b>Decimal form</b>
/8	255.0.0.0	/14	255.252.0.0
/9	255.128.0.0	/15	255.254.0.0
/10	255.192.0.0	/16	255.255.0.0
/11	255.224.0.0	/17	255.255.128.0
/12	255.240.0.0	/18	255.255.192.0
/13	255.248.0.0	/19	255.255.224.0
/20	255.255.240.0	/26	255.255.255.192
/21	255.255.248.0	/27	255.255.255.224
/22	255.255.252.0	/28	255.255.255.240
/23	255.255.254.0	/29	255.255.255.248
/24	255.255.255.0	/30	255.255.255.252
/25	255.255.255.128	/31	255.255.255.254

Beyond figuring out the subnet notation and the subnet mask, you also should know how many hosts can be on a specific subnet, at least for often used subnets. Otherwise, you run the risk of using a subnet that is too small and some computers won't be able to get on the network.

Table 6.2 shows how many hosts fit on a given subnet.

Table 6.2 Number of hosts that are supported on a given subnet for IPv4

<b>Subnet notation</b>	<b>Subnet mask</b>	<b>Number of hosts supported</b>
/8	255.0.0.0	16,777,214
/16	255.255.0.0	65,534
/20	255.255.240.0	4,094
/22	255.255.252.0	1,022
/24	255.255.255.0	254
/28	255.255.255.240	14

Table 6.3 Number of hosts that are supported on a given subnet for IPv6

<b>Subnet notation</b>	<b>Notes</b>	<b>Number of hosts supported</b>
/48	Can contain 65,536 /64s	1,208,925,819,614,629,174,706,176
/64	Can contain 4,294,967,296 /96s	18,446,744,073,709,551,616
/96	Can contain 65,536 /112s	4,294,967,296
/112		65,536

IPv6 is the future of IP communications. The primary reason for this is that IPv4 has officially ran out of IP addresses. That means that all the IPv4 blocks of addresses have been assigned to ISPs and other large organizations. As more and more devices are connected to the internet, the last IPv4 blocks will be used up. A large part of this is the explosion of devices that connect to the internet such as smartphones, smart watches, smart TVs, smart refrigerators, and even smart lightbulbs. The exhaustion of the IPv4 address space has been well known for many years. And that knowledge spurred on the uptake of IPv6, starting with some major internet companies such as Google. IPv4 is a 32-bit addressing technology that is limited to 4 billion IP addresses. IPv6 is a 128-bit addressing technology that is limited to 340,282,366,920,938,463,463,374,607,431,768,211,456 IP addresses. And no, that's not a typo. For those of us that aren't familiar with so such huge numbers, the 340 represents 340 undecillion. In other words, IPv6 won't run out of IP addresses until we expand our footprint to hundreds of other galaxies and bump up our population and device counts quite a bit!

## Ideas for on your own

You can explore subnets in greater detail by visiting the following sites:

- <https://support.microsoft.com/en-us/kb/164015>
- <https://technet.microsoft.com/en-us/library/cc958832.aspx>
- <http://www.cisco.com/c/en/us/support/docs/ip/routing-information-protocol-rip/13788-3.html>

Now that you have a familiarity with subnets, subnet masks, CIDR notation, and the reasons why you use subnets, let's explore some of the intricacies of network communication.

## Fundamentals of Protocols and Ports

A protocol is a set of rules governing how computers communicate with each other. TCP/IP is the primary protocol for computer communications today. But as part of TCP/IP, there are application layer protocols. For example, to view a web page, your browser uses Hypertext Transfer Protocol (HTTP). You can transfer files from one computer to another using File Transfer Protocol (FTP). Application layer protocols dictate how computers communicate for a given service.

In addition to the application layer protocols, there are network ports that also play a role in how computers communicate. Network services, such as HTTP, FTP, and RDP, have assigned port numbers. This helps computers send the network communication to the right service and application. In Figure 6.3, a diagram shows a user request a web page and an RDP connection to a server. When the server receives those requests, it sends the communication to the application that hosts the corresponding service on the destination port. Thus, when a request comes in for a web page, that request uses HTTP and port 80. The server sees the request and sends the request to Internet Information Server which is the application that hosts the web site.

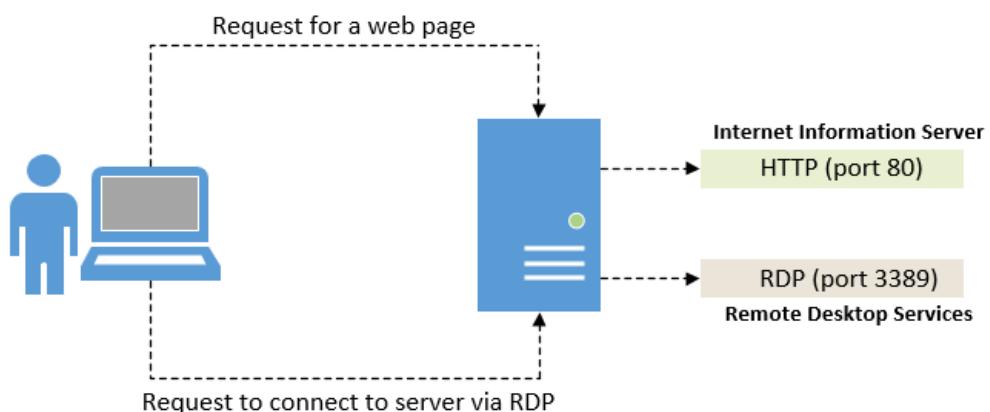


Figure 6.3 A diagram showing two requests for services and the routing of those requests to the correct application

As a server administrator, you will need to know how to troubleshoot network connectivity issues. We will look closely at troubleshooting network connectivity in Chapter 11. Part of the troubleshooting will involve finding out which protocols are being used and which ports are

being used. Why? Because you can use that information to check firewall configurations to ensure that firewalls are configured to allow that type of communication. You can also capture network traffic (think of a cable TV box that can record TV shows) for later analysis. Finally, you can check the server configuration to ensure that applications are configured to use the expected protocol and port.

Network services and their associated ports are officially assigned by the Internet Assigned Numbers Authority (IANA). These assigned ports are known as the "well-known ports". Applications generally use well-known ports. However, applications can use other ports that aren't assigned. In such cases, the application vendor often documents the ports used by the application. To see the list of well-known services and their ports, visit <http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>. Besides the assigned ports, there is a range of ports from 49152 to 65535 that are unassigned and available for use. You can use those ports for your own uses such as if you wanted to change the port that an application uses. Remember that the Windows firewall blocks most ports by default so if you decide to use a port for something, you may need to configure the Windows firewall to allow communication on that port. Sometimes you'll need to change a port because multiple applications want to use the same port. You don't need to memorize all the ports. But, it is helpful to memorize some of the common services and ports because it saves you time when you are configuring and troubleshooting services. In table 6.2, some of the most common services and ports are shown.

Table 6.4 Common services and port numbers

<b>Service</b>	<b>Port number</b>
File Transfer Protocol (FTP)	20 (data) and 21 (transfer)
HTTP	80
HTTPS	443
RDP	3389
Simple Mail Transfer Protocol (SMTP)	25
Domain Name System (DNS)	53
SSH	22

### **Hands-on Exercise**

1. Open a command prompt and run the following command:

```
netstat
```

2. That shows current connections and their ports. Next, run this command:

```
netstat -a
```

3. That shows all connections and listening ports. You can use PowerShell to enhance the view and bring in filtering capabilities. Open a PowerShell console and run the following command:

```
netstat -a | Out-GridView
```

4. In the results window, click the **Add criteria** button, click the **String** checkbox, click **Add**, and then type *listen* in the textbox. Note how the results change to only show listening ports.

You should now know that computers communicate with each other based on protocols (rules) and ports, with most of the common services relying on assigned ports. You should also know that when a server receives a request over the network, it sends that request onto the application based on the service, protocol, and port. We will build on this information in Chapter 11 when we look at how to troubleshoot network communication. For now, let's move on with more network fundamentals by looking at how you can use 2 NICs to increase performance and availability.

## NIC Teaming

In Windows Server, NIC teaming is a method of binding two or more NICs together for high availability or load balancing. You want to use NIC teaming to ensure that your server can communicate on the network if one of the NICs fails. Or, if you want to increase the performance of network communication, especially when a server will be handling a lot of network traffic. When you create a NIC team, there are two primary configurations for inbound traffic (sometimes called "teaming modes"):

- **Switch-independent teaming.** Switch-independent teaming is most commonly an active/standby configuration where two NICs are connected to two separate switches. The active connection will be used for all ingress (inbound) and egress (outbound) traffic, and the standby connection will only be used if the active connection fails. This mode doesn't require any configuration on the network switches (instead, just a server configuration).
- **Switch-dependent teaming.** Switch-dependent teaming, as the name implies, is dependent on network switch configurations because it requires additional configuration to be made on the switches that are connected to the host. This mode is used when you have servers that have large inbound and outbound network utilization. The reason is that all the NICs are connected to the same switch and the switch treats the NICs like 1 big NIC, aggregating the traffic.

Figure 6.4 shows the initial configuration of a NIC team being created in Windows Server 2016. In the configuration, 1 NIC (labeled "Ethernet") will be active and 1 NIC (labeled "Ethernet 2") will be the standby NIC.

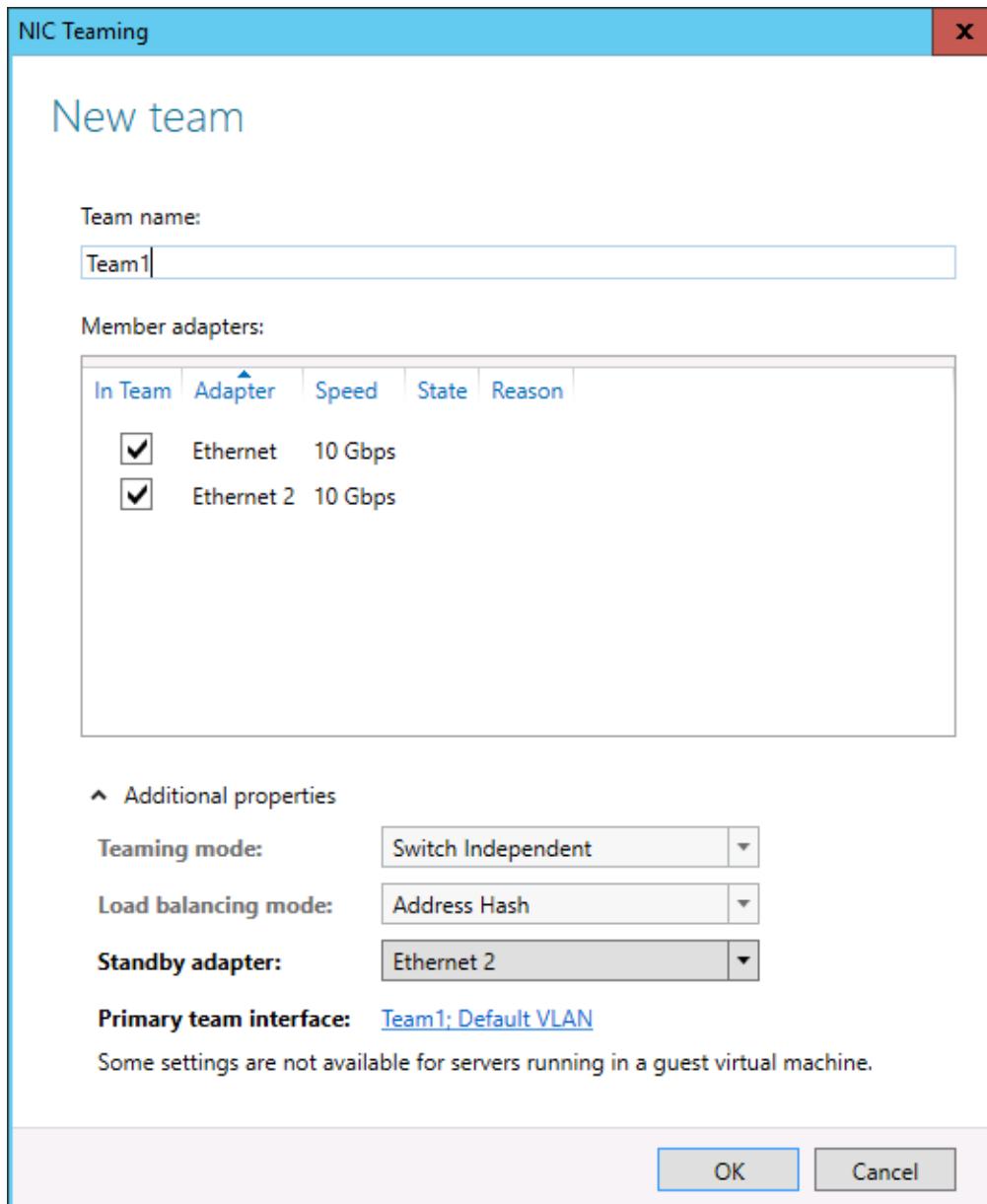


Figure 6.4 A screen capture showing the configuration of a new NIC team configured for high availability

Before you create a NIC team in a Hyper-V VM, you need to configure the NICs to enable them to be added to a team. Perform the following steps to do that.

1. Open Hyper-V Manager on the Hyper-V server.
2. In the left pane, click to highlight the Hyper-V server.

3. In the right pane, right-click the VM that you will create a team on and then click **Settings**.
4. In the settings window, expand each Network Adapter in the left pane.
5. For each NIC, click the **Advanced Features** section in the left pane.
6. For each NIC, in the right pane, scroll down to NIC Teaming section. Then, click the checkbox next to the **Enable this network adapter to be part of a team in the guest operating system** option.
7. Click **OK** to save the configuration.

To create a new NIC team on a server that has two NICs, perform the following steps:

1. Open Server Manager.
2. Click **Local Server** in the left pane.
3. In the right pane, click the status shown next to NIC Teaming.
4. In the **NIC Teaming** window, under Teams, click the **Tasks** dropdown menu and then click **New Team**.
5. In the textbox labeled **Team name**, type Team1.
6. In the list of NICs, click the checkbox next to both NICs and then click **OK**.

Now that you know how inbound traffic to the server is handled, let's talk about outbound traffic (where network communication originates from the server). Outbound traffic can be distributed in different ways. You can load balance network traffic across NICs but this requires reassembling the packets since they typically arrive out of order. This adds latency to the connection process, which means that the network performance can be reduced. Alternatively, if you keep the packets that are associated with a connection on the same NIC, it will improve performance because packets usually arrive in the proper order and do not have to be reassembled.

When you set up the load balancing mode, you must choose a distribution algorithm which dictates how load balancing functions across the NICs. You can choose three different distribution algorithms (sometimes called "load balancing modes"). Often, the mode that you choose will be based on the service that you are load balancing. For example, when you use NIC teaming in a VM, you must use Address Hashing and a switch-independent configuration. This is because Hyper-V VMs do not support any other modes.

- **Address hashing.** Address hashing creates a hash based on the address of the packet and then packets get assigned to a NIC based on the hash. When you configure address hashing to use TCP ports, it creates the most distributed path for traffic but you can't use it in some situations, such as when IPsec is in use. The following components can be configured as inputs to the hashing value, to control traffic:
  - Source and destination TCP ports
  - Source and destination IP addresses

- Source and destination MAC addresses
- **Dynamic.** Dynamic NIC teaming distributes outbound traffic based on the TCP port and IP address and rebalances traffic on the NICs in real time. This mode is new since Windows Server 2012 R2. For inbound traffic, traffic is balanced based on the active port number. You should use this mode for all your load balancing needs unless you have a specific reason not to (such as the example I mentioned earlier about using NIC teaming inside of VMs).
- **Hyper-V switch port.** This mode is only valid for Hyper-V servers. This mode distributes traffic based on the Hyper-V switch port or the destination MAC address (the MAC address of the VM that will receive the network traffic).

Up to 32 NICs can be used to create a team when using Windows Server on a physical host. When using a VM in Hyper-V, only two NICs are supported per team. Additionally, creating a NIC team that is comprised of other NIC teams is not supported.

Besides using the graphical tools to manage NIC teaming, you can also use PowerShell. The New-LbfoTeam cmdlet enables you to create a new NIC team based on the adapters that are defined on the server. Additionally, the Set-NetLbfoTeam, Set-NetLbfoTeamMember, and Set-LbfoTeamNic cmdlets can be used to manage an existing team or NICs. Let's look at a couple of examples.

By using the `-Confirm` parameter, you can avoid having PowerShell prompt you to confirm that you want to perform the action. The following command creates a new team named Team1 from the adapters named Ethernet and Ethernet 2.

```
New-NetLbfoTeam -Name Team1 -TeamMembers 'Ethernet','Ethernet 2' -  
Confirm:$false
```

By default, all adapters are active. The following command sets Ethernet 2 to be a standby adapter.

```
Set-NetLbfoTeamMember -Name 'Ethernet 2' -AdministrativeMode Standby
```

The Get-NetLbfoTeamMember cmdlet shows you the team members and their current configuration and status. This cmdlet also enables you to verify that Ethernet 2 is in standby mode.

## Hands-on Exercise

First, if you are trying this in a VM, enable the NICs to be added to a team by following the steps earlier in this section. Next, find the name of your NICs by running the Get-NetAdapter command. Then, use the New-NetLbfoTeam cmdlet to create a new team. Use the example provided earlier in this section to formulate the needed command. When finished, set one of the NICs to be in standby mode and verify the configuration.

You should now understand that NIC teams are created to improve performance or provide high availability. And that the behavior of the NIC team is based on the chosen configuration. When all NICs are active, higher performance is achievable. When all NICs are not active, you do not get a performance advantage but you do achieve high availability. For all NIC teams, high availability is one of the benefits. You should also know the prerequisites for creating a NIC team and the high-level steps required to create a NIC team. In the final chapter section, we will look at securing network communication, whether that communication involves a NIC team or standalone NICs.

## **Securing Network Communications**

When computers communicate over a network, part or all the communication is often sent in plain text. For example, if you open an email application and retrieve your email, your username and password may be sent in plain text to the server. If another computer on the network is configured to capture all network communication, then your username and password would be captured. The danger of such a scenario is that you wouldn't have any indication that a computer is capturing all the network communication because it would be happening after the communication left your computer. Figure 6.5 shows three points of potential interception – the client computer (your computer), an unauthorized computer connected to the same switch, and the server (the computer that you are trying to communicate with).

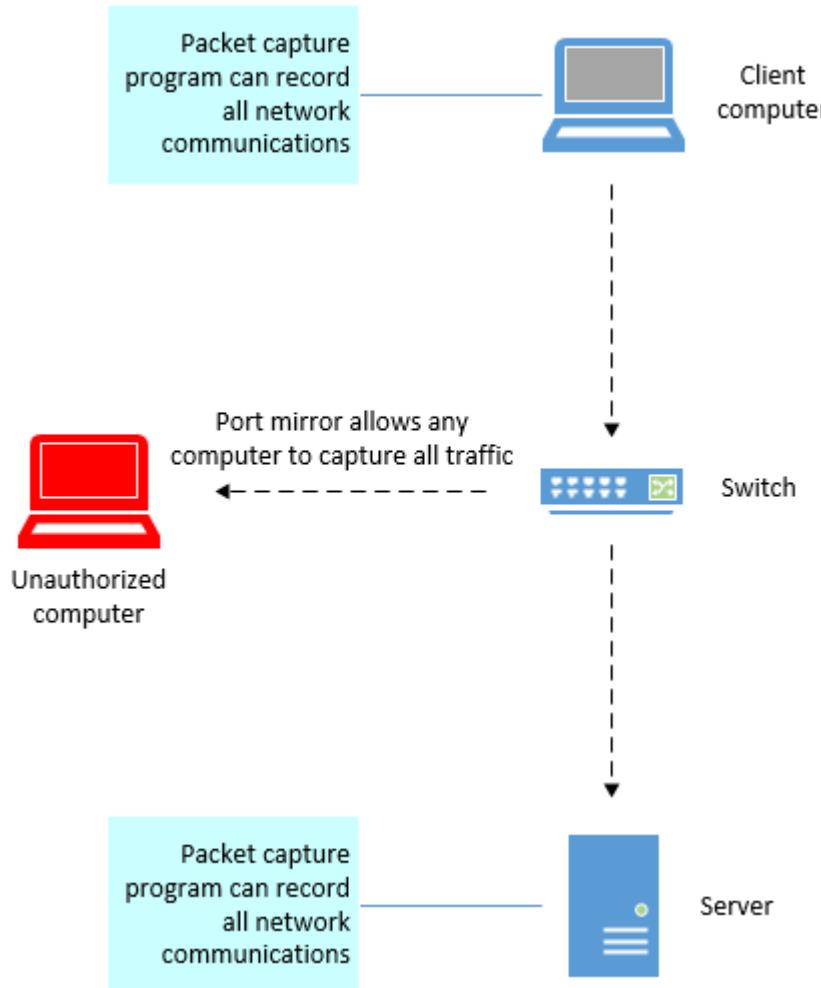


Figure 6.5 A diagram showing the potential points of data interception – a client computer (the source), an unauthorized computer connected to the same switch, and the server (the intended destination).

Besides this scenario, many other scenarios exist that could result in your personal information (identification information, credit card information, or health information) being exposed.

To minimize the risk of data being exposed, researchers created Secure Sockets Layer (SSL) and Transport Layer Security (TLS) in the mid to late 1990s. SSL is often an acronym that is used when discussing securing network communications (even if TLS is being used). With secure network communication, even if communication is captured, it isn't readable because it is encrypted. As of 2015, SSL 3.0 is deprecated and most secure network communication uses TLS. The evolution of TLS is ongoing and standards are rapidly changing.

Both SSL and TLS rely on digital certificates to validate the identity of a server. For example, when you connect to a web site securely, a digital certificate identifies the web site (often by the domain name) and your browser checks the server's certificate to ensure that it is valid for the web site and not expired or revoked. In high security environments, certificates can be used to

validate the identity of a server and a client. When you use certificates, you get two primary benefits – identification of the computers and encrypted data. Figure 6.6 shows the high-level flow of a certificate check.

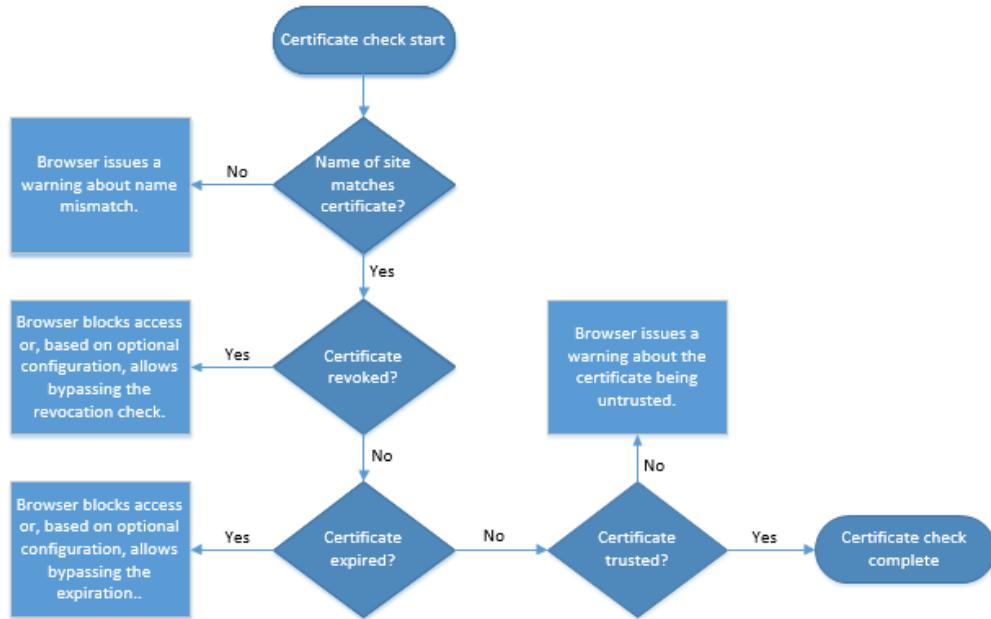


Figure 6.6 A process flow showing the various checks performed to validate the validity of a certificate.

As a server administrator, you will be responsible for figuring out when to secure network communications and the best way to secure network communications. Next, we will look at some common use cases for securing network communication.

### When to secure network communications

Several years ago, it was common to only secure the most sensitive network communications such as those involving the transmittal of credit card data or other sensitive data. But after high profile security incidents, leaks of personal and sensitive information, and the discovery that governments worldwide are participating in some forms of data capture, securing network communications has taken on a real urgency. You should opt for secure network communication wherever you can such as whenever it is possible with the hardware and software that you use. Large organizations such as Google and Microsoft have opted to secure most network communications to their services including their search engines, even when the data seems innocuous. This is a good thing and it is likely that we will see a continuing expansion of securing network communications by other organizations. But remember, securing network communication is one part of an overall data security approach. You also need to secure data after it has been communicated such as data on file servers and in databases. We explore securing such data with data encryption in Chapter 23. For now, let's discuss some of the methods of securing network communications.

## Methods of securing network communications

To maximize security in an organization, multiple teams will need to work together. As a server administrator, you will partner with a network administrator and application administrators to secure network communications from end to end. The network administrator will secure network communications between sites, between your organization and the internet, and may provide secure remote access (such as a VPN or a remote access web portal) to employees. Remember, security is everywhere – every piece of hardware and every piece of software has some tie-in with security. I'll show a few typical tasks throughout this section but it is important to think about security in every single project that you work on. Some of the common security relate tasks of a server administrator are providing a public key infrastructure (PKI) to issue and manage digital certificates, configuring Internet Information Server (IIS) to require SSL on web sites, and using IPsec to encrypt network communications between computers. In general, you will be responsible for configuring the server side for secure communications. The server security provides a base foundation for applications. Then, application administrators often have some application level security that they can apply to network communications to further enhance security. For example, they may be able to configure their application to encrypt data that is replicated to other servers. Follow these principles to maximize security:

- **Secure all network communication when possible.** In other words, don't secure only the communication that you think needs it. Secure all communication.
- **Opt for the highest level of security possible.** For example, when you use digital certificates, you can choose the bit-length used. The more bits, the higher the security. While the industry standard for digital certificates today is 2048-bits, you can opt for 4096-bit certificates for additional security. Another example is with the configuration of a server. You can opt to provide secure communication if requested or require secure communication (thus denying insecure communication). You should opt to require security when possible.
- **Partner with the network and application teams so that everybody is operating under a common strategy – securing all network communication.** Otherwise, you may end up with secure communication in one portion of your network and a lack of secure communication elsewhere.

In this chapter, you learned that subnets are segmented portions of a larger network and that you use them to enhance security and performance. You learned that protocols are rules for computers that want to communicate over a network. You learned that network ports play a role in communications and that many services have assigned and known ports. You gained insight into NIC teaming and understand the process to create a NIC team. And, you became familiar with securing network communications and know that you should opt to secure communications whenever possible. Now, let's test your knowledge and retention in these areas in the lab.

## Lab

This lab is designed to validate your retention of information from this chapter and perform some of the tasks that we discussed in this chapter.

### Understanding subnets

Fill in the missing information:

- A /24 subnet can have up to \_\_\_\_\_ hosts.
- The subnet mask for a /8 subnet is \_\_\_\_\_.
- The use of /8 and /24 are examples of \_\_\_\_\_ notation.

### Configuring NIC teaming

Create a NIC team by using two NICs. Use the second adapter as a standby adapter.

### Identify the ports

Identify the port numbers used by the following services:

- Remote Desktop Protocol (RDP)
- Simple Mail Transfer Protocol (SMTP)

### Secure network communications

You are the server administrator for a hospital. The hospital has a server that stores patient health data. Nurses use tablet computers to record patient health information during patient visits. The health information is sent back to the server's database. The management team has asked you to evaluate the existing security and enhance it, if possible. You investigate and find that the server has a digital certificate and all network communication is secured by using the digital certificate. What should you do (Choose all that apply.)?

- a. Add a digital certificate to the database.
- b. Add a digital certificate to the client.
- c. Increase the bit-length of the digital certificate(s).
- d. Move the digital certificate from the server to the client.
- e. Move the digital certificate from the server to the database.

You are configuring an application for secure communication. The application offers the following options. Which option should you use?

- a. Use SSL to secure communication.
- b. Use TLS to secure communication.
- c. Use SSL and TLS to secure communication.
- d. Use digital certificates on the client computers.

## CHAPTER 7: MANAGING A DHCP SERVER ROLE

---

Continuing with our network theme in Part 2, we are going to look at managing a Windows DHCP server in this chapter. DHCP is in use across virtually all organizations. Thus, as an administrator, you need to know how to maintain a DHCP server to make sure that your company's computers are connected to the company network. DHCP is a network protocol that was first defined in 1993 in Request for Comments (RFC) 1531. Many vendors (operating system vendors, hardware vendors such as wireless router vendors, and software vendors) have created DHCP software and services that adhere to the DHCP standard. DHCP is used to automatically issue IP addresses and network configuration information to computing devices (laptops, smartphones, and other devices that connect to a network). Without DHCP, IT admins would have to manually configure a lot of computing devices so that they could participate in a network. DHCP automates everything and managing DHCP isn't burdensome.

As a server administrator, your job is to ensure that DHCP is running in a healthy state and successfully issuing IP address and network configuration information. When issues arise, your job is to troubleshoot and/or configure the DHCP server by using the built-in tools in Windows Server. While DHCP is often thought of as a service for client computers, it can also be used for servers too (although this isn't common for on-premises environments today).

In this chapter, we are going to first look at the primary DHCP management tool – the DHCP management console. After we get familiar with that, we will look at how relay agents enable DHCP to work across subnets, and we will explore PowerShell solutions to manage a DHCP server. Then, we will have you run through some DHCP management tasks in the lab.

### Using the DHCP management console

Imagine that it is your first day on the job. And a trouble ticket comes in that requests all the printer IP addresses on the 4<sup>th</sup> floor. What do you do? Open the DHCP management console! The DHCP management console is an Microsoft Management Console (MMC)-based interface which serves as your primary tool for managing DHCP in your environment. It provides a graphical view of your configuration and servers. For environments with multiple DHCP servers, the console can be a single view for administering for all Windows-based DHCP servers. In addition to the console, you can also manage DHCP with the dhcpcmd.exe command line tool or with PowerShell cmdlets. For this section, we are going to focus on the DHCP management console which is the most common tool you'll use to perform day-to-day administrative tasks. In the DHCP management console, you can perform the following tasks:

- **Create DHCP scopes and view current leases.** You can create new scopes such as when you add a new wireless network or open a new office. You can view current leases to figure out which IP addresses computers have and see if your scopes have enough free IP addresses for future clients.
- **Retrieve IP addresses and MAC addresses of current leases.** You can look at the details of current leases and see MAC addresses. Because a MAC address is unique to every computing device, you can use a MAC address to verify the identity of a DHCP client.

- **Back up and restore the DHCP database and configurations.** As an administrator, you should always back up your servers and data. In the event of a catastrophe, you want to be able to easily restore services for your company.
- **Create scopes for IPv4 and IPv6.** The DHCP management console enables you to create scopes for IPv4 and IPv6. Using a single tool for all administrative tasks reduces the administrative overhead of switching between tools often.
- **Stop and start the DHCP service.** You'll stop and start the DHCP service to troubleshoot a service that isn't responding or after a configuration change that requires the service to be restarted.
- **Create DHCP policies to customize network configurations based on specified conditions.** Occasionally, you will create DHCP policies to target IP settings to a group of computers. For example, you may want to have all computers on the guest wireless network use different DNS servers than your corporate computers.
- **Configure DHCP failover.** DHCP failover ensures that DHCP remains running if a DHCP server fails.

Let's walk through some tasks in the console now.

### **Hands-on Exercise**

Open the DHCP management console. Expand all the containers in the left pane. Right-click the server name and look at the properties of the server. Then right click one of the DHCP scopes and examine the scope's properties.

### **Creating DHCP scopes**

A scope is a consecutive range of IP addresses that a DHCP server can assign to computers on a subnet. At first glance, you may wonder why you wouldn't just use an entire subnet for a scope. The reason is that you should reserve a portion of a subnet for statically configured devices (routers, switches, firewalls, and sometimes printers) as well as leave a portion of the IP addresses available for reservations. A DHCP server can't issue IP addresses without having at least one scope. One of the first tasks that you perform after a DHCP server is deployed is creating a scope. Let's go through the process, step-by-step.

1. Run the DHCP management console, expand the DHCP server name, expand IPv4, then right-click **IPv4** to bring up the context menu. Next, click **New Scope**.

Figure 7.1 shows the IPv4 context menu.

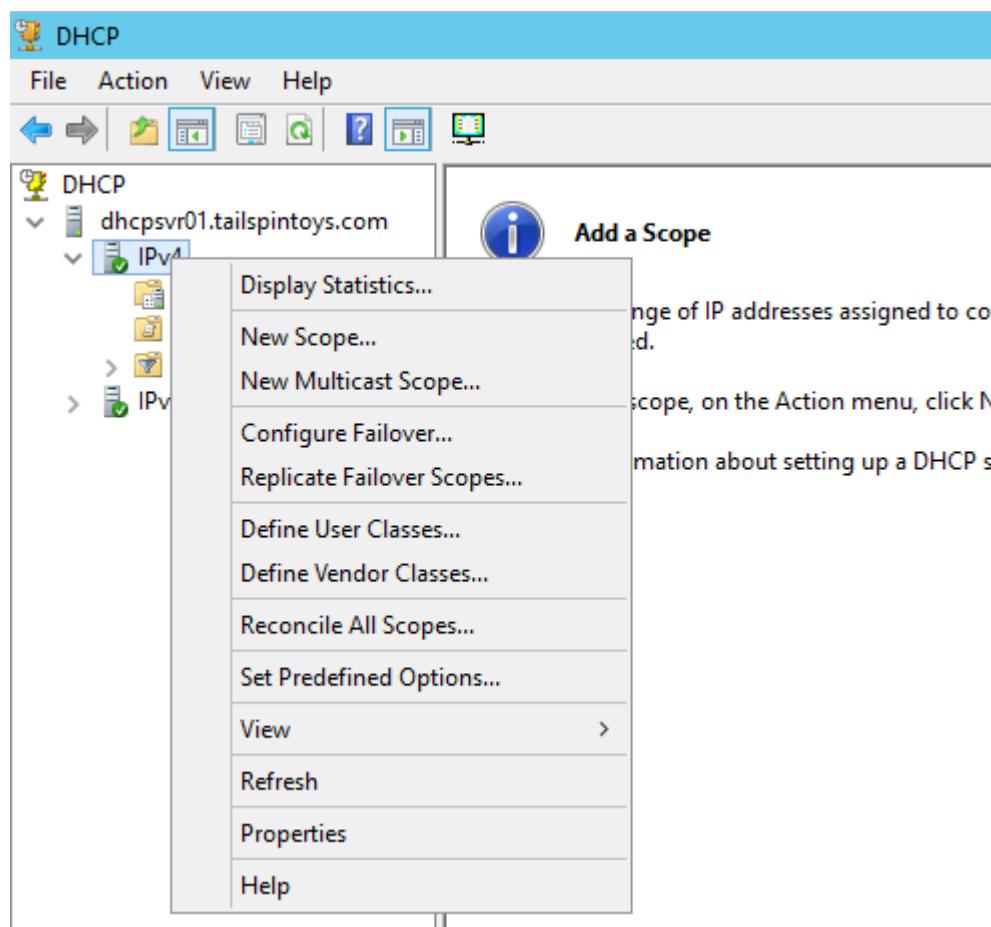


Figure 7.1 The IPv4 context menu provides an array of actions including an option to create a new scope.

2. On the **Welcome to the New Scope Wizard** page, click **Next**.
3. On the **Scope Name** page, type a name and description for the new scope and then click **Next**.
4. On the **IP Address Range** page, shown in Figure 7.2, enter the starting and ending IP address. In this example, imagine that you have the 192.168.1.0/24 subnet and you want to allocate 100 IP addresses for DHCP. After you enter the IP address range, the wizard will specify a subnet mask. Since it is a /24 subnet, the subnet mask is 255.255.255.0. Click **Next**.

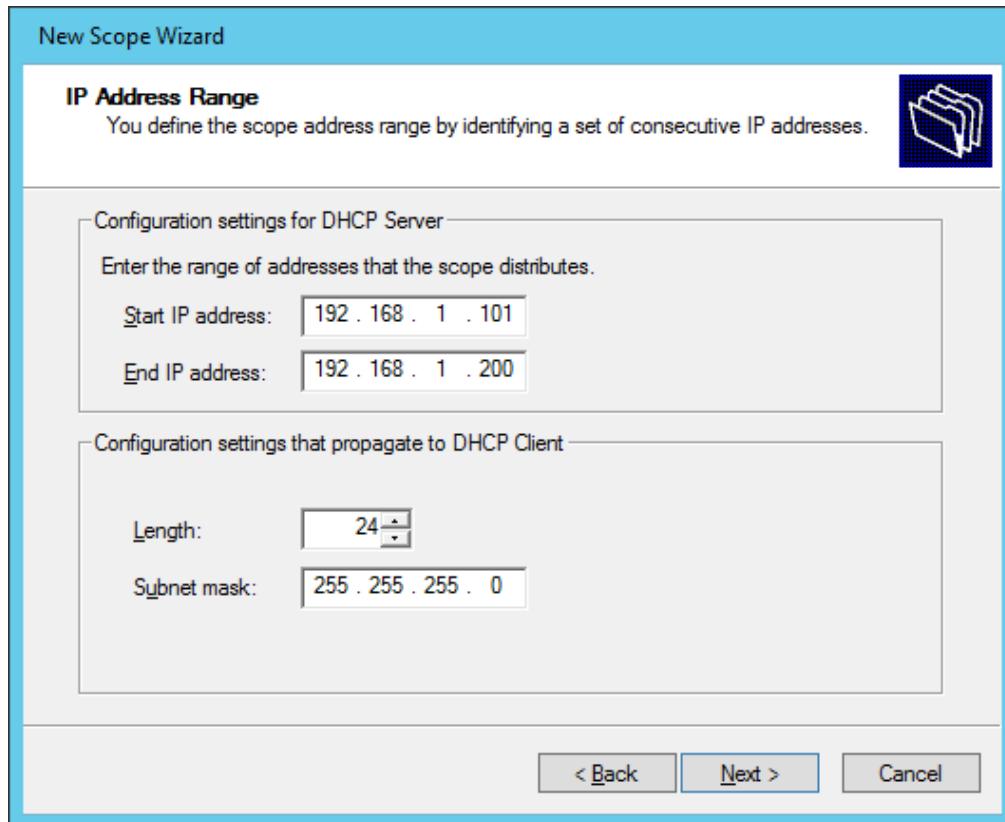


Figure 7.2 The IP Address Range page is where you specify the IP address range for DHCP. Consider using a starting and ending IP address that leaves you some unused IP addresses for devices that do not use DHCP.

5. The **Add Exclusions and Delay** page enables you to specify IP addresses that will not be issued by the server. Click **Next**.

### Above and Beyond

In many organizations, you will add exclusions for printers or other devices that are manually configured with network settings. If you don't exclude IP addresses that are in the DHCP scope's issuing range, then the DHCP server may issue an IP address that is already being used on another device (manually configured) and that can create an IP conflict. During an IP conflict, one or both computers configured with the same IP address may have issues communicating on the network. Notice the option to delay the offer of an IP address? That is used when you have more than one DHCP server and you want to ensure that one of them is the primary server that issues most (or all) IP address while the secondary server stands by in case the primary server fails. If the primary server fails, the secondary server will automatically issue IP addresses. This works because DHCP requests are broadcast to the entire subnet. The first DHCP server to respond 'wins' and issues an IP address to the computer requesting DHCP. By using a delay, you can greatly influence which server responds. For this walkthrough, we will leave all the settings on this page default.

6. The **Lease Duration** page enables you to specify how long an IP address is leased by a computer. By default, the lease duration is set to 8 days. For most wired networks, 8 days is a good setting. If you reduce the duration too much, it can increase network traffic and DHCP server load without providing any benefit. However, for wireless networks or other networks that have temporary clients, you should reduce the lease duration drastically. For most wireless networks, 1 day is a good lease duration because wireless clients tend to be active on the network for less than a day and there are many wireless computing devices coming and going. If lease times are too long on wireless networks, it can cause a shortage of IP addresses. Imagine a coffee shop. The DHCP server would issue an IP address to every customer that used the wireless network. After an hour or so, many customers leave but their IP addresses are unusable for other customers until the lease duration ends. If the lease duration is 8 days and the coffee shop is serving many customers, it won't take long before the server runs out of IP addresses to issue. Click **Next**.
7. The **DHCP Options** page enables you to configure additional options such as the default gateway and DNS servers. Most scopes that you create will be configured to issue a default gateway and DNS servers because those settings are required to have full network functionality. Keep the default setting which indicates that you want to configure the options now and then click **Next**.
8. On the **Router (Default Gateway)** page, shown in Figure 7.3, type the IP address of the gateway and then click **Add**. In our example, the gateway is 192.168.1.1. Then, click **Next**.

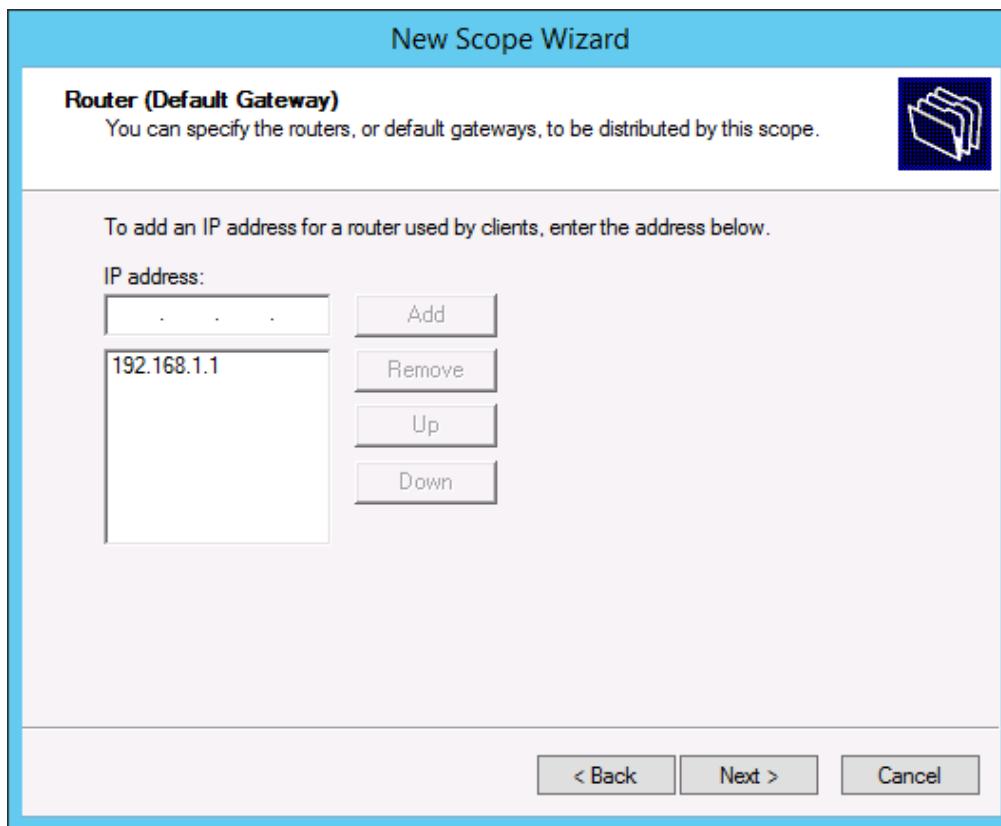


Figure 7.3 The Router (Default Gateway) page enables you to issue the default gateway to DHCP clients.

9. On the **Domain Name and DNS Servers** page, the parent domain will automatically be filled in with the domain of the DHCP server. The DNS server IP addresses will automatically be filled in with the DNS servers that the DHCP server is using in its IP configuration. If the domain name and IP addresses for the DNS servers are the ones that you intend to use for this scope, then click **Next**. Otherwise, update the information and then click **Next**.
10. On the **WINS Servers** page, leave the settings default (thus, servers not specified) and then click **Next**. WINS is a deprecated technology that many companies are retiring and moving away from. However, if you still rely on WINS in your network, you can add them on this page.
11. On the **Activate Scope** page, ensure that the default setting to enable the scope is selected and then click **Next**. If you don't activate the scope now, you can't use it until you activate it.
12. On the **Completing the New Scope Wizard** page, click **Finish** to complete the scope creation process.

### Hands-on Exercise

Open the DHCP management console and walk through the New Scope Wizard to view the available options. Then, open the properties of an existing scope and notice how the settings from the wizard are shown after a scope is created.

Before a domain-joined DHCP server can issue IP addresses, it must be authorized. This is a one-time process. To authorize your DHCP server, perform the following steps:

1. From the DHCP management console, click **DHCP** in the left pane.
2. Click the **Action** menu and then click **Manage Authorized servers**.
3. In the **Manage Authorized Servers** window, click the **Authorize** button.
4. In the **Authorize DHCP Server** window, in the Name or IP address textbox, type the fully qualified hostname of your DHCP server and then click **OK**.
5. In the **Confirm Authorization** window, click **OK**.
6. In the **Manage Authorized Servers** window, click **Close**.

You should now be familiar with creating a new scope (use the New Scope Wizard in the DHCP management console), know what the scope options are used for (configuring additional IP settings such as a gateway and DNS servers), and know what you need before you start creating a new scope (IP address information, gateway IP address, and DNS server IP addresses).

And, you now have a valid DHCP scope that is ready to issue IP addresses to your computers! Let's now look at how you can view the information in your scope and look at some of the advanced configuration options.

### Viewing the DHCP configuration

After you have a functioning DHCP scope, you need to be familiar with working with the scope, especially viewing the configuration and activity. Viewing the scope will help you prevent outages related to running out of IP addresses as well as help you track down helpful troubleshooting information. First, let's take a quick peek at a screen capture of the DHCP management console so you can see what we are going to talk about. In Figure 7.4, a portion of the DHCP management console shows a single scope and configuration items (shown as modified folder icons).

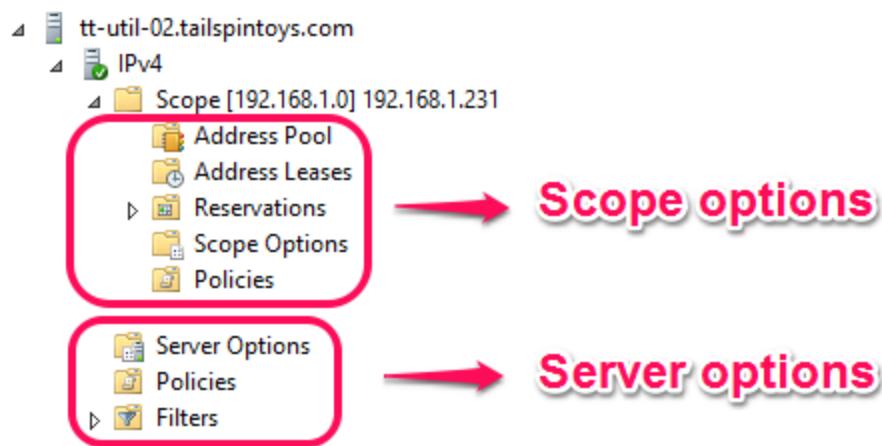


Figure 7.4 The DHCP management console has sections to configure scope-specific settings and server-specific settings.

Scope options take precedence over server options. This means that if you configure 192.168.1.1 as the gateway in the scope options and 10.1.1.1 as the gateway in the server options, 192.168.1.1 will be the gateway. It is a good practice to only configure settings in the scope options or in the server options. Avoid configuring the same settings in both because it complicates the troubleshooting process. Now, let's look in detail at the configuration items.

### Hands-on Exercise

Open the DHCP management console. Expand a scope. Right-click Scope Options under a scope and then click Configure Options. Glance at the available options and then close the window. Next, right-click the Server Options container in the left pane and then click Configure Options. Glance at the available options. Notice anything different from the scope options?

## View the configuration and current client activity for a scope

The following configuration items are specific to each scope. If you have only a few scopes (such as in a small network), configuring individual scopes doesn't take much time. However, if you have many scopes (such as in a large network), individually configuring them can be time consuming.

- **View the address pool.** Expand the scope and click Address Pool in the left pane. This will show the starting and ending IP addresses for the scope. If an exclusion has been specified for the scope, you will see that in this section too.
- **View the DHCP leases.** Expand the scope and click Address Leases in the left pane. This shows you all DHCP leases and reservations, as well as additional information about the leases or reservations including the IP address, the FQDN of the client, and the lease expiration date and time.
- **View the reservations.** Expand the scope and click Reservations in the left pane. This shows you the reservations. You can click the individual reservations in the left pane and view the detailed information such as the DNS servers and default gateway associated with the reservation.
- **View the scope options.** Expand the scope and click Scope Options in the left pane. This displays the options, if any, configured on the scope. As previously mentioned, most scopes will be configured with a default gateway and DNS servers as a minimum (unless those options are configured at the server level).

### Hands-on Exercise

Open the DHCP management console. View the scope options for one of the existing scopes and see which options are being used. Compare those options with other scopes and see if you can find any scopes that are using options that other scopes aren't.

## Viewing the configuration for the server

The following configuration items are specific to each server and apply to all scopes on the server. If you only have a few scopes, there isn't a big benefit to using server options instead of scope options. However, if you have many scopes, you should try to use server options where possible because it will reduce the administrative overhead during the initial configuration and during maintenance.

- **View the server options.** In the left pane, click **Server Options**. In the right pane, you will see the server options, if any have been configured. You can configure additional IP information such as a gateway or DNS servers using the server options. During troubleshooting, it can be helpful to view the server options to see if you are currently using them and to see if they conflict with scope options.

- **View the filters.** In the left pane, click **Filters** to expand it, then click **Allow** or **Deny** to view filters. Filters are used to enhance DHCP security by preventing unknown computing devices from receiving a DHCP lease from the DHCP server. There are a couple of scenarios where you would want to do this. First, imagine that your company decided that personally owned tablet computers shouldn't use the corporate wireless network. By obtaining the MAC address of one of the units (one of each tablet model), you can add a filter to deny DHCP service to them. You could also opt to only issue IP address to known clients. This is called whitelisting. Whitelisting is rarely used due to the administrative overhead of managing the whitelist.

### **Hands-on Exercise**

Open the DHCP management console. If necessary, expand the server and expand IPv4. Expand Filters. Right-click on **Deny** and then click **New Filter**. Enter a 12-digit random number as the MAC address. For the description, type *Compromised laptop* and then click **Add**. Click **Close**. Click **Deny** under Filters and verify that the entry you added is in the Deny list. That computer cannot obtain DHCP services because of the deny filter you just added.

### **Backing up and restoring DHCP**

Once you configure your DHCP server and scopes, you should back up your DHCP configuration. This will save you a lot of time if the server is lost and you must start over with a new server. By having a backup, you won't have to reconfigure everything manually. Perform the following steps to back up DHCP:

- In the DHCP management console, right-click the server and then click **Backup**.
- A window will open and prompt you for the location to save the backup files. Accept the default location at %SYSTEMROOT%\System32\dhcp\backup by clicking **OK**.
- The backup process will run for several seconds on a DHCP server with only a single scope or for several minutes on a DHCP server with many scopes and activity.

The backup process will complete without any indication other than the hourglass mouse icon reverting to the standard mouse icon. Browse to the backup location and verify that the backup files are in place. In the root, there should be a configuration file (DhcpCfg). In a folder named 'new', there should be two database files (dhcp.mdb and dhcp.pat) and transaction log files (\*.log). When you back up DHCP, the following information is saved:

- All scopes, including superscopes and multicast scopes
- Reservations
- Leases
- All options, including server options, scope options, reservation options, and class options
- All registry keys and other configuration settings (for example, audit log settings and folder location settings)

Remember though, a backup is only useful if you know how to restore it and test the backups by trying to restore them occasionally.

### Hands-on Exercise

Using a test DHCP server, create a new DHCP reservation. To create the reservation, expand an existing DHCP scope, right-click **Reservations**, and click **New Reservation**. In the New Reservation window, type *Reservation1* for the name, type any available IP address for the IP address, and type a random 12-digit number for the MAC address (for example, try 005599887744). Click **Add** and then click **Close**. Next, perform the backup process shown above. After the backup completes, delete the DHCP reservation. Then, restore DHCP from the backup. Once complete, verify that the DHCP reservation has been recreated.

By now, you should have a good understanding of the DHCP management console and know that you can use it to create and manage your DHCP server and scopes. You should also be familiar with the backup process which can save you time during a recovery situation. Later, we'll look at some of the management tasks that you can perform outside of the DHCP management console by using PowerShell. But first, a quick diversion to explain what a DHCP relay agent is and when you need to use one. This will help you understand how DHCP works across networks and what you can do to ensure functionality in such a situation.

## DHCP Relay Agents

You are the administrator for your company. You deploy a DHCP server and a scope. All the clients on your floor are functional and people are happy. Suddenly, you get a call from the floor above yours. Their computers aren't working. And it is because their computers are not getting an IP address from the DHCP server. You investigate and find out that the floors are on different subnets. How can you enable that floor to use the new DHCP server? By using a DHCP relay agent! But let's examine why. DHCP requests are sent by clients using a broadcast message. A broadcast message is a message that isn't sent to a specific computer or destination. Instead, it is sent to the entire subnet so all computers on the subnet can see the message. DHCP servers listen for DHCP broadcast messages on their subnet and respond to DHCP requests. The trouble begins when you have a single DHCP server that needs to service more than one subnet. By default, switches do not let broadcast messages pass from one VLAN subnet to another. So, you need a way to get the DHCP requests from all the subnets to the DHCP server. That is where a DHCP relay agent comes in. Its job is to listen for DHCP broadcasts on its subnet and forward those requests over unicast (communication to a specified destination) to the DHCP server, which it is configured to know about. If you need DHCP services on a subnet without a DHCP server then a DHCP relay agent is required. For example, imagine a company that has offices on two floors of a high-rise building. On each floor, they have computers. Each floor has its own subnet and only one floor has a DHCP server. Figure 7.5 shows such a network with two subnets. A DHCP server is on the 192.168.1.0/24 subnet but needs to service its own subnet and the

10.10.50.0/24 subnet. To make this work, a DHCP relay agent is added on the 10.10.50.0/24 subnet. It is a good idea to talk with your network team about the options available before you move forward.

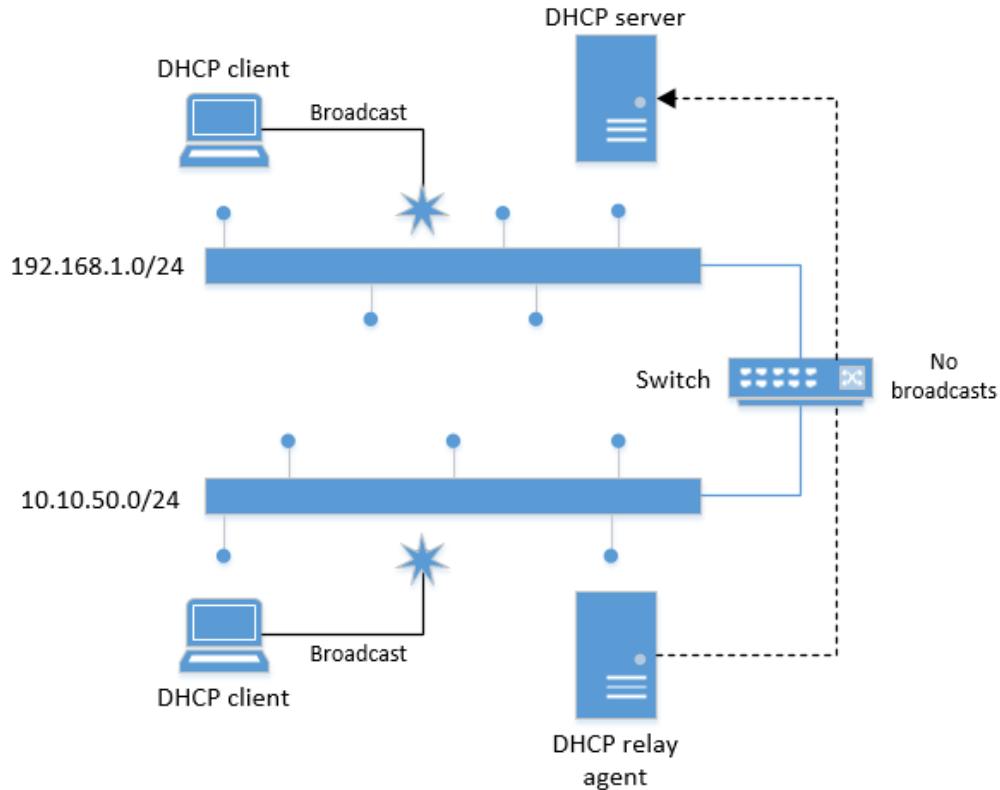


Figure 7.5 A DHCP server services two subnets by using a DHCP relay agent to forward requests.

Now that you understand the DHCP relay agent's job, let's talk about how you can add a DHCP relay agent. Windows Server has a built-in DHCP relay agent that you can use. Using the built-in functionality of Windows Server is often advantageous compared to purchasing a third-party solution and having to install and support it. In Windows Server, to add a DHCP relay agent, you must install the Remote Access role along with the DirectAccess and VPN (RAS) role service and the Routing role service. Once installed, you need to go through the initial configuration wizard to select the services that you want to enable. By selecting a custom configuration, you can add just the LAN routing service which provides routing protocols such as the DHCP relay agent. The configuration thereafter is quite simple. You just add the IP address of the DHCP server! We show you the step-by-step instructions now.

### Add the Remote Access role and required role services

Add the Remote Access role. During the installation, add the DirectAccess and VPN (RAS) role service and the Routing role service. When prompted, add any needed features to support the installation. Otherwise, accept all defaults.

1. Open Server Manager. Click **Tools** and then click **Remote Access Management**.

2. In the Remote Access Management Console, click **DirectAccess and VPN** in the left pane. In the middle pane, click the **Run the Getting Started Wizard** link.
3. In the **Configure Remote Access** window, click **Deploy both DirectAccess and VPN (recommended)**.
4. If you are prompted with any prerequisite warnings, click **Next** to ignore them. Otherwise, add any required prerequisites and then continue by clicking **Next**.
5. For the network topology of the server, accept the default setting, type the FQDN of the server in the textbox, and then click **Next**.
6. Click **Finish** to save the settings.
7. In the confirmation window, click **Close**.

### Add the DHCP relay agent

1. In the Remote Access Management Console, click the **Open RRAS Management** in the right pane.
2. In the **Routing and Remote Access** window, expand the server in the left pane.
3. In the left pane, expand IPv4.
4. In the left pane, under IPv4, right-click **General** and then click **New Routing Protocol**.
5. In the **New Protocol** window, click **DHCP Relay Agent** and then click **OK**.
6. In the left pane, right-click **DHCP Relay Agent** and the click **Properties**.
7. In the **Properties** window, type the IP address for the DHCP relay agent, click **Add**, and then click **OK**.

In addition to using Windows Server as a DHCP relay agent, you can also use a network switch or router. In fact, in most enterprise environments, it is common to use an existing switch or router to handle all DHCP relay services. This is because the devices are commonly connected to all the subnets which means you don't have to add a DHCP relay agent on every subnet.

### Using PowerShell to manage a DHCP server

By now, you should be comfortable in the DHCP management console and be familiar with how DHCP works. In this section, we want to expand your knowledge by introducing PowerShell methods to manage a DHCP server. When you need to perform administrative tasks repetitively or against many objects, PowerShell will save you time and improve consistency.

There is a PowerShell module for managing a DHCP server. Its name is `DhcpServer`. The module is automatically installed and imported on computers that have the DHCP Server Tools role service installed. You can also install the module on other computers by adding the Remote Server Administration Tools feature with the DHCP Server Tools subfeature. Once you have the module installed, you can view all the available cmdlets, functions, and aliases by running the following command:

```
Get-Command -Module DhcpServer
```

### Hands-on Exercise

Run the following command, and look at what is available in the DhcpServer module.

```
Get-Command -Module DhcpServer
```

There is a lot of stuff for you to explore! Let's look at some common commands. Note that the commands need to be run at an elevated PowerShell prompt. In this first command, we are going to create a DHCP scope by using PowerShell. To create a DHCP scope named NYC Floor 33 that issues IPs between 10.10.50.26 and 10.10.50.254 with a subnet mask of 255.255.255.0, run the following PowerShell command:

```
Add-DhcpServerv4Scope -Name 'NYC Floor 33' -StartRange 10.10.50.26  
-EndRange 10.10.50.254 -SubnetMask 255.255.255.0
```

From earlier in this chapter, you know that a DHCP scope isn't going to provide much value if the scope or the server isn't issuing a gateway and DNS servers. But our command to create a scope doesn't address either of those. That's because the Add-DhcpServerv4Scope cmdlet doesn't have a way to set scope options or server options. Instead, to set scope options, you must use the Set-DhcpServerv4OptionValue cmdlet. In the example below, we add a default gateway to the existing scope. Notice that the ScopeID is the network identification (take the starting IP address and change the last octet to 0).

```
Set-DhcpServerv4OptionValue -ScopeID '10.10.50.0' -Router 10.10.50.1
```

Now that you know how to create a scope and add options to a scope, let's look at a few commands for reviewing scope configuration and usage as well as server activity.

To view all your DHCP scopes and their high-level settings such as the IP range and whether they are active, run the following command:

```
Get-DhcpServerv4Scope | FL
```

If you have many scopes, this command will return too much information. To reduce the output to just a single scope such as the 10.10.50.0 scope, you can run the following command:

```
Get-DhcpServerv4Scope -ScopeId 10.10.50.0 | FL
```

You can use this command for many reasons such as checking to see if you have an existing scope for a specific range of IP addresses.

Get-DhcpServerv4ScopeStatistics is a cmdlet that will show you a list of all DHCP scopes, the number of free IP addresses in the scope, how many leases there are, and other information

about the scope. You can use this command to check if you are running low on available IP addresses in a scope.

To view DHCP scopes that are at least 80% used, run the following command:

```
Get-DhcpServerv4ScopeStatistics | where PercentageInUse -gt 80
```

Most scopes that are 80% used don't have a lot of available IP addresses left. This information is helpful because you can take proactive actions such as increasing the IP range before the scope runs out of IP addresses.

To view all the current leases and reservations for a specified scope, run the following command:

```
Get-DhcpServerv4Lease -ScopeId 10.10.50.0
```

This is helpful when you want to see if a reservation is active or you want to see which hostname has leased a specific IP address.

Get-DhcpServerv4Statistics is a cmdlet that will show you some high-level information about the DHCP server such as how many scopes it has, the total number of IP addresses in use, and the percentage of IPs that are available. This is another way to find out if you have enough available IP addresses in your scopes.

Besides viewing the existing configuration, you can also use PowerShell to configure the server and scopes as well as perform backup and restore operations. Let's look at a few of these commands.

To add a new reservation for the training room printer with a MAC address of d8-c6-be-e0-86-59 for the IP of 192.168.1.120 and, run the following command:

```
Add-DhcpServerv4Reservation -ScopeId 192.168.1.0 -IPAddress 192.168.1.120  
-ClientId d8c6bee08659 -Description 'Training Room printer'
```

To back up the local DHCP server to the default backup location, run the following command:

```
Backup-DhcpServer -Path C:\Windows\system32\dhcp\backup
```

To restore the local DHCP server from the backup stored in the default location, run the following command:

```
Restore-DhcpServer -Path C:\Windows\system32\dhcp\backup
```

Now that you know how to perform a few administrative tasks by using PowerShell, you should explore some of the other cmdlets in the DhcpServer module to expand your knowledge.

In this chapter, you learned how to use the DHCP management console to perform a variety of administrative tasks. You learned how to create scopes to support the addition of new networks such as a branch office and view server activity to see information such as if scopes are running out of IP addresses. You learned that a DHCP relay agent can help you provide DHCP services to subnets even if those subnets don't have their own DHCP server. And you learned some common PowerShell commands for tasks such as creating a DHCP reservation and backing up the DHCP server. Let's jump into the DHCP lab and test your skills.

## Lab

This lab is designed to provide hands-on experience configuring DHCP services via the DHCP management console and PowerShell interfaces. If you haven't already performed the Hands-on Exercises throughout the chapter, perform them now before you start the lab.

### Create and configure a new DHCP scope in the DHCP management console

1. Use the following information for the new scope:
  - Network ID: 10.10.50.0/24
  - Name: Training Room 3
  - IP range for leasing: 10.10.50.51 to 10.10.50.254
  - Default gateway: 10.10.50.1
  - DNS servers: 10.10.50.10 and 10.10.50.20

### Create a DHCP reservation in the DHCP management console

Use the following information for the new reservation:

- Reservation name: Training room printer
- IP address: 10.10.50.40
- MAC address: f1-5d-3b-00-3c-81

### Create a DHCP reservation by using PowerShell

Use the following information for the new reservation:

- Reservation name: Training room laptop
- IP address: 10.10.50.41
- MAC address: f3-4d-3c-00-4c-82

### Backup the DHCP configuration by using PowerShell

Use the default location to back up the DHCP configuration.

### Restore the DHCP configuration by using PowerShell

Use the default location to restore from.

## CHAPTER 8: MANAGING CLIENT NAME RESOLUTION

---

Name resolution, which is the mapping of computer names to IP address, is a critical function to a business. When you go to a web page such as www.contoso.com, a process to resolve www.contoso.com to an IP address runs in the background. Ultimately, your computer uses an IP address to communicate over the network. Without name resolution, most computer and internet functionality doesn't work. For example, without DNS resolution your e-mail application can't send or receive email and your connection to a bank's web site doesn't work. For example, if your email application tries to download email from mail.contoso.com and your computer cannot find the IP address for mail.contoso.com, then the email cannot be downloaded. Name resolution is primarily handled through 3 methods – DNS, Windows Internet Naming Service (WINS), and local resolution (HOSTS and LMHOSTS files that manually map IP addresses to names). WINS, while once popular, is now considered a legacy technology because it relies on broadcast network communication which doesn't function on the internet, especially with most computing devices being connected to the internet these days. Local name resolution is inefficient because it is tough to manage since it is decentralized (all computing devices maintain their own name resolution through mapping files which has individual computer names and their associated IP addresses in a text file). DNS is the standard and is widely used as the only name resolution service on a network.

In this chapter, we are going to focus on how name resolution works on clients. Clients represent client computers, servers, or any other computing devices that use DNS to resolve names. We'll look at the name resolution process in detail, show you how client caching works, and walk through several name resolution troubleshooting scenarios. Some of the information presented in this chapter is complex. If you haven't spent much time exploring name resolution in detail, the complex parts of this chapter may be initially difficult. However, we have 3 chapters covering DNS and name resolution which will help you get comfortable with how things work. By the end of the chapter, you should have enough information and know enough techniques to manage and troubleshoot name resolution for your client computers. We'll test you at the end of the chapter in a lab.

### How names are resolved

The name resolution process is complex because there are a lot of steps involved. Windows supports many types of name resolution and it sometimes must try all the types of resolution to resolve a hostname. DNS, WINS, and NetBIOS are types of name resolution. This chapter will focus mostly on DNS because it is the dominant name resolution service used today. Before we get into the details, there are some key terms that you should know:

- **Fully qualified domain name (FQDN).** A hostname appended with a domain name. For example, let's say that we have a computer named Server1. And Server1 is joined to the contoso.com domain. The fully qualified domain name of Server1 is server1.contoso.com.

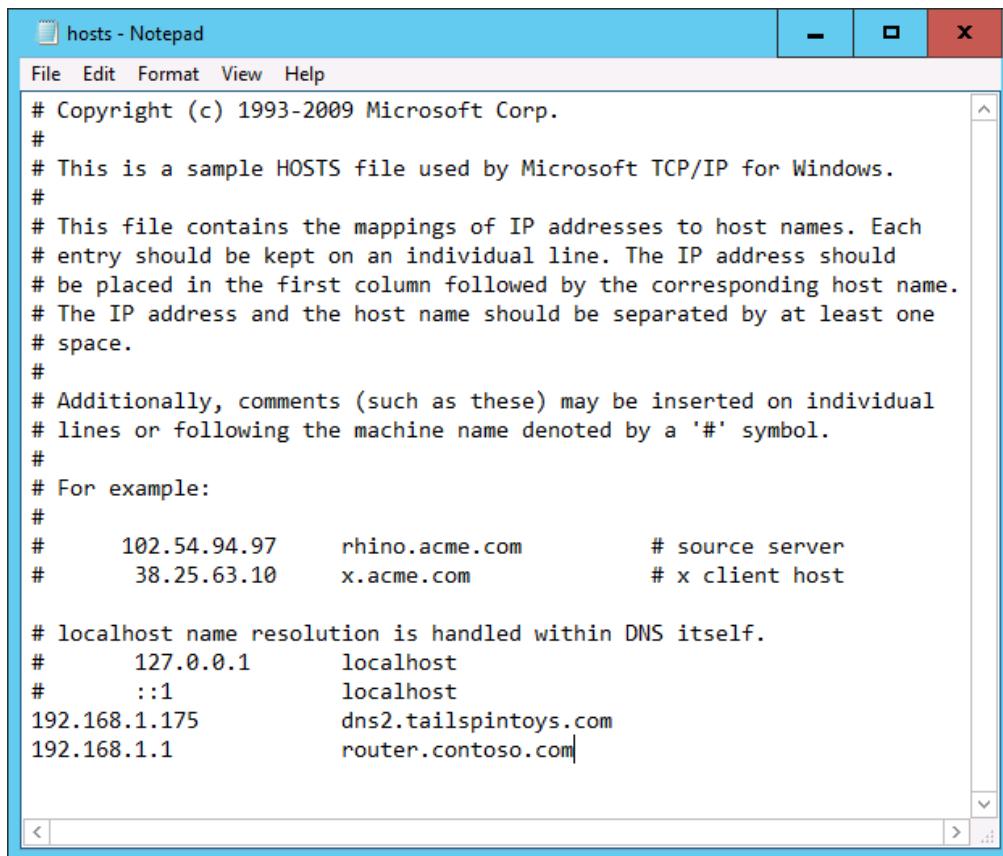
## Hands-on Exercise

Open a PowerShell console. Run the following two commands to find your computer's FQDN:

```
$FQDN=(Get-WmiObject win32_computersystem).DNSHostName+"."+(Get-WmiObject win32_computersystem).Domain  
$FQDN
```

- **Authoritative DNS server.** Each domain, whether public or for internal use only, has at least one DNS server that is responsible for helping DNS clients and servers resolve names for that domain. This server (or servers) are known as the authoritative DNS servers for the domain. The authoritative DNS servers are registered with domain registry for the top-level domain (such as .com or .net, based on the top-level domain of the domain). Because the authoritative DNS servers are registered, any other DNS server or client can find them by querying the root DNS servers or by using querying for the servers from a domain name registrar (companies that sell domain names).
- **Root DNS server.** The DNS servers, which are listed in the root hints on a DNS server, that are responsible for maintaining a list of all authoritative DNS servers for all the domains on the internet. There are 13 root DNS servers spread across multiple geographic regions worldwide.
- **HOSTS file.** A text file on every Windows computer that can be used to resolve hostnames to IP addresses. By default, the HOSTS file does not have any active hostname entries so it is only used if you add hostnames and IP addresses to the file. It is rarely used today because DNS is the name resolution standard and HOSTS files are tedious to maintain and share. However, there are two exceptions. Sometimes, malware will use the HOSTS file to maliciously redirect users to the wrong web site (imagine your bank's web site resolving to the wrong IP address). And occasionally, an IT administrator will use the HOSTS file to block a user from going to a specific web site address (for example, the IT administrator will put the domain name in the HOSTS file and use an internal IP address that isn't valid – effectively blocking the user's access by name).

- Figure 8.1 show the HOSTS file on a server. Two entries, one for dns2.tailspintoys.com and one for router.contoso.com, have been added.



```

hosts - Notepad
File Edit Format View Help
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97    rhino.acme.com        # source server
#      38.25.63.10     x.acme.com            # x client host

# localhost name resolution is handled within DNS itself.
#      127.0.0.1        localhost
#      ::1              localhost
192.168.1.175      dns2.tailspintoys.com
192.168.1.1         router.contoso.com

```

Figure 8.1 A screen capture of a HOSTS file that has 2 entries added at the bottom of the file.

### Hands-on Exercise

Validate the functionality of the HOSTS file. First, ping [www.microsoft.com](http://www.microsoft.com) from the command prompt. Then, open an elevated command prompt, navigate to C:\Windows\System32\Drivers\Etc, and type Notepad HOSTS. When Notepad opens the HOSTS file, add a line for 192.168.100.100 with the fully qualified name being [www.microsoft.com](http://www.microsoft.com). Refer to Figure 8.1 or the HOSTS file help text for syntax. Then, save the HOSTS file. Next, ping [www.microsoft.com](http://www.microsoft.com) and compare the IP address that Windows uses. It should be 192.168.100.100. Once finished, revert the HOSTS file back to its original configuration and save it.

- **NetBIOS (Network Basic Input/Output System).** A legacy network communications protocol used in small local area networks (mostly prior to the internet). NetBIOS mostly operates over TCP/IP now and is still used from time to time for name resolution or file and printer sharing.

- **WINS (Windows Internet Name Service)**. A legacy name resolution service for NetBIOS computer names. WINS is like DNS but used for local area networks while DNS is used for local area networks and the internet. Many organizations still have WINS servers although DNS is the primary name resolution. WINS servers are sometimes required by old applications. Often, administrators aren't sure if anything on their network still uses WINS so the servers end up staying around for longer than they are probably needed.

### Hands-on Exercise

Open the Control Panel and then open Network and Sharing Center. In the left pane, click **Change adapter settings**. In the Network Connections window, right-click the active Ethernet NIC and then click **Properties**. Scroll down to Internet Protocol Version 4 (TCP/IPv4). Double-click it. Click the **Advanced** button. Then, click the **WINS** tab. See if there are any IP addresses listed in the top of the window. If you see some, your computer is using WINS. If you don't see any IP addresses, your computer isn't using WINS servers.

## Name resolution methods

Now that we've defined some of the key terms that we'll use in this section, let's look at the primary methods of name resolution that Windows uses.

- **DNS**. This is, by far, the most common type of name resolution. Thus, when you are troubleshooting name resolution, you should start with DNS. Most computers are configured with two DNS servers (helpful in case one of them isn't responding). Those servers are used as part of the name resolution process. Sometimes, they'll know the answer, especially when they are authoritative for a domain name. A DNS server is authoritative for a domain name when it is registered as the authoritative name server for the domain name. In such cases, the internet's root DNS servers will have a record of the authoritative server which helps other servers on the internet during name resolution for that domain. When a DNS server cannot resolve a fully qualified domain name, it will query a forwarder (another DNS server that is sometimes configured to help a DNS server with all queries that it cannot answer) or it will ask the root servers for the authoritative DNS servers for the domain and then query the authoritative servers.
- **DNS client cache**. On each Windows computer, there is a name resolution cache. You'll use this cache when you troubleshoot and fix name resolution issues. It is made up of the HOSTS file entries (preloaded on startup) and recently resolved names. For example, if you visit [www.microsoft.com](http://www.microsoft.com), then your DNS cache will have an entry for [www.microsoft.com](http://www.microsoft.com) and the IP address it resolved to. This happens automatically. Client cache reduces DNS server load and network traffic because clients often don't need to query a DNS server since the client cache has entries for recently resolved names. We will talk about client cache in more detail in the next section.

- **NetBIOS name resolution.** If Windows tries other name resolution methods but can't resolve a hostname, it will, as a last resort, try to resolve the name by using NetBIOS. In such cases, a broadcast is sent on the subnet asking for name resolution. The name resolution method isn't in wide use today.

The name resolution process is the same on Windows client operating systems as it is on Windows server operating systems. Windows uses the following name resolution order:

1. Checks to see if the hostname is its own hostname.
2. Checks to see if the client cache has the IP address for the hostname. Note that HOSTS file entries are also checked now since they are pre-loaded into the client cache.
3. Checks to see if DNS can resolve the name.
4. Checks to see if NetBIOS name resolution can resolve the name.

Let's look at a visual of the name resolution process. In Figure 8.2, a decision tree walks through the process of a client attempting to resolve www.contoso.com. The process starts at the middle top at the step "Resolve www.contoso.com" and ends when name resolution is achieved or when name resolution is not possible.

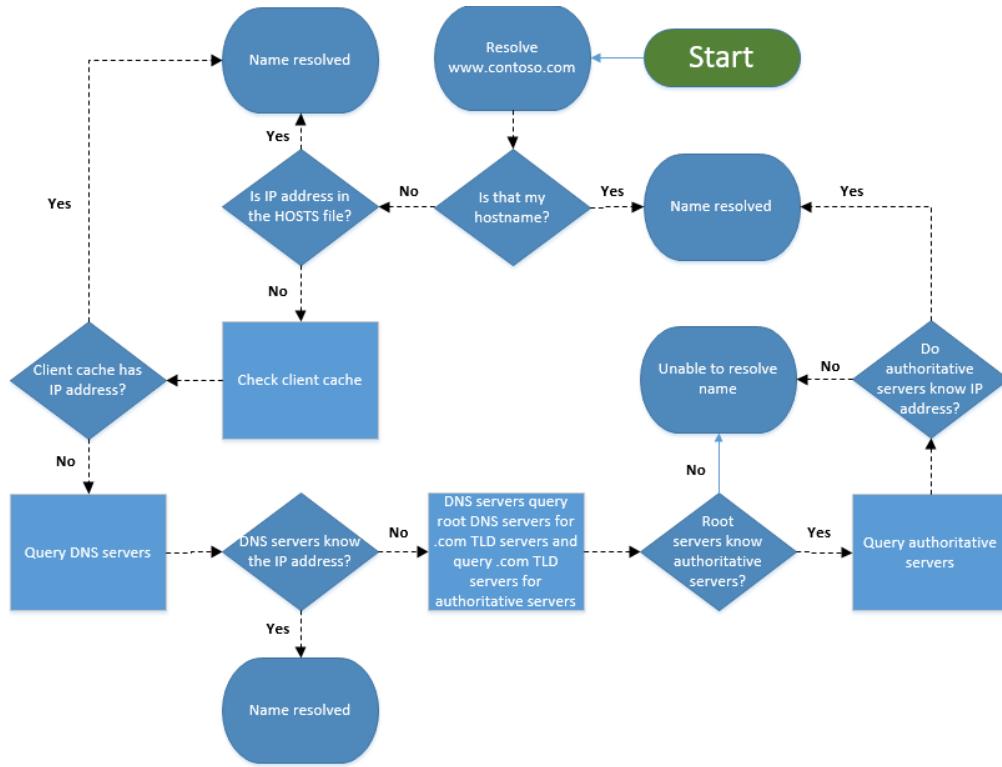


Figure 8.2 A decision tree shows the process a client goes through to resolve www.contoso.com. The flow starts at "Resolve www.contoso.com".

Let's walk through the decision tree shown in Figure 8.2. Start at the "Start" icon and follow the arrows based on the answers.

1. A user opens her browser and tries to go to www.contoso.com. This is depicted by the "Resolve www.contoso.com" action at the top of the diagram.
2. Her computer checks to see if www.contoso.com is the name of her computer. If it is, the name is resolved and her computer attempts to connect to itself on port 80.
3. Her computer checks if www.contoso.com is in the HOSTS file. If it is, the name is resolved and her computer attempts to connect to itself on port 80.
4. Her computer checks to see if www.contoso.com is in the client cache. If it is, the IP address associated with it is used to attempt a connection on port 80.
5. Her computer queries a DNS server from the list of DNS servers configured on her NIC. If her DNS servers have www.contoso.com in the cache or are authoritative for contoso.com, then the name is resolved and a connection is attempted for the IP on port 80.
6. Her computer queries the root servers to find the authoritative servers for the .com TLD and subsequently those DNS servers attempt to find the authoritative servers for contoso.com. If they do not find the authoritative servers, name resolution fails.
7. Her DNS server queries one of the authoritative DNS servers for contoso.com asking for the IP address of www.contoso.com. If the servers do not know, name resolution fails. If the servers know, they return the IP address for www.contoso.com and her computer attempts to connect to the IP on port 80.

With the information in this section, you should now be comfortable describing the name resolution process. You should know that names are resolved in a specific order with the client cache being checked before DNS and DNS servers being checked before NetBIOS name resolution is attempted. You should also be able to identify scenarios when name resolution isn't likely to happen such as when there aren't any authoritative DNS servers for a domain. Now that you have been introduced to the client cache, let's take a closer look at some of the important aspects of the cache. This will help prepare you for troubleshooting name resolution issues and save you time during the troubleshooting process.

## Understanding DNS client cache

Another administrator tells you that he is having trouble connecting to an internal file server. You try and it works. You guys compare information. It turns out that the IP address that his computer has for the file server is different than the one your computer has. This means that the communication for the other administrator is going to the wrong place! If you have a good working knowledge of how the DNS client cache works and what data is stored in it, you will be prepared to troubleshoot and resolve most name resolution issues. In such a situation, you should Let's start by looking at the data in the client cache because this is usually the first thing you'll do when troubleshooting a problem quick and easy check. Sometimes, just clearing the cache resolves problems like this one. But before you clear the cache, you need to look at it to

be sure the problem is with the cache! The following table, table 8.1, shows a partial client DNS cache (a few entries from the cache with a subset of properties). The Entry column is the name that a client resolved. The record name is the name returned by the DNS server. The type specifies the type of DNS record. There are multiple types and we will look closely at those in the next chapter titled “Managing the DNS server role”. The Data column has the data from the DNS entry and will usually contain an IP address, especially for Address (A) records. The TTL is the time to live for the DNS entry in the cache. The time is in seconds. Thus, 86,400 seconds is 1 day. Once the TTL time has elapsed, the entry in the client cache is removed.

Table 8.1 A table showing DNS client cache entries

Entry	Record name	Type	Data	TTL
175.1.168.192.in-addr.arpa	175.1.168.192.in-addr.arpa	PTR	Tailspintoys.com	86400
sls.update.microsoft.com	sls.update.microsoft.com	A	157.56.77.138	166
tt-util-01	tt-util-01.tailspintoys.com	A	192.168.1.231	783
www.windowssearch.com	www.windowssearch.com	A	204.79.197.200	1417

OK, so you have a high-level understanding of what’s in the cache and why you need to look at it, but how can you view the cache on your computer? You can look at the client cache using a couple of different methods:

- **PowerShell.** Run the Get-DnsClientCache command. It will display the entire cache with several columns. You might remember that you can find out which other information is available from a cmdlet from the “Getting started with PowerShell” chapter. To see the properties available for the cache, run this command:

```
Get-DnsClientCache | Get-Member
```

To display the DNS client cache with all the available properties and output it in a list, run this command:

```
Get-DnsClientCache | select *
```

To show the first entry in the cache with all the available properties, run this command:

```
Get-DnsClientCache | select -First 1 | FL *
```

Figure 8.3 shows a single entry from the DNS cache with all the properties displayed.

Property	Data
TTL	: 1167
Caption	:
Description	:
ElementName	:
InstanceID	:
Data	: 192.168.254.200
DataLength	: 4
Entry	: tt-util-02
Name	: tt-util-02.TailspinToys.com
Section	: 1
Status	: 0
TimeToLive	: 1167
Type	: 1
PSComputerName	:
CimClass	: ROOT/StandardCimv2:MSFT_DNSClientCache
CimInstanceProperties	: {Caption, Description, ElementName, InstanceID...}
CimSystemProperties	: Microsoft.Management.Infrastructure.CimSystemProperties

Figure 8.3 A screen capture of a single entry in the client cache with all the properties displayed.

- **Command prompt.** To view the entire cache in a list, run the following command:

```
ipconfig /displaydns
```

### Hands-on Exercise

Open the command prompt and run this command:

```
ipconfig /displaydns
```

Examine the entries in the client cache. Next, ping [www.microsoft.com](http://www.microsoft.com) [www.yahoo.com](http://www.yahoo.com). Then, run the following command"

```
ipconfig /displaydns
```

You should see new entries for [www.microsoft.com](http://www.microsoft.com) and [www.yahoo.com](http://www.yahoo.com). Next, try the same thing with PowerShell. First, look at the entries in the client cache by running the Get-DnsClientCache cmdlet. Then, ping [www.bing.com](http://www.bing.com) and [www.contoso.com](http://www.contoso.com). Finally, check the cache again by running the Get-DnsClientCache cmdlet. You should see entries for [www.bing.com](http://www.bing.com) and [www.contoso.com](http://www.contoso.com).

So how does the cache get populated? There is a service, named DNS Client, that is responsible for populating the client cache when names are resolved. By default, the service starts automatically and remains running in the background. If the service is stopped, then the cache is emptied and new name resolutions are not added to the cache. Be aware that the DNS Client service will automatically start if it is stopped and you resolve some DNS names. To get it to stay stopped, you need to disable the service and then stop it.

## Hands-on Exercise

Press the **Windows+R** key combination. In the Run window, type *services.msc* and then press the **Enter** key. Scroll down to the DNS Client service and double-click it. Click the **Startup type** dropdown menu and then click **Disabled**. Click **Apply**. Then click the **Stop** button. Click **OK**. Open a command prompt. Ping [www.microsoft.com](http://www.microsoft.com) and [www.yahoo.com](http://www.yahoo.com) commands. Then, run the following command:

```
ipconfig /displaydns
```

How is what you see different than what you see when you view the cache while the DNS Client service is running?

Now you know how to view the client cache, view type of data stored in the cache, and how new entries are added to the cache. Let's see how that information comes into play during troubleshooting.

## Troubleshooting client name resolution issues

In my day-to-day work, it is common for me to troubleshoot name resolution issues daily! That's a testament to how common DNS name resolution problems are. To effectively troubleshoot name resolution issues, you need to have a good grasp of how Windows resolves names as well as know about the local cache. Since we covered those in the previous sections, we can now talk about troubleshooting and walk through the troubleshooting steps based on a given situation. But before we begin, let's cover an important point about troubleshooting. And this applies to name resolution issues as well as many other computing issues. Troubleshooting often boils down to comparing a non-working computer with a working computer. Many times, fixing an issue boils down to updating the configuration of the non-working computer to match the configuration of the working computer. Walking through troubleshooting scenarios helps you prepare for real-world troubleshooting situations. It also helps you remember key details about how name resolution works.

### Developer reports that he can't get to [www.tailspintoys.com](http://www.tailspintoys.com)

It is a typical Monday morning. You have a bunch of unread email messages and upcoming meetings. Then, the phone rings and a developer reports that when he tries to go to [www.tailspintoys.com](http://www.tailspintoys.com), he receives an error stating "The page cannot be displayed". He needs to get to that site to test a recent web site change. What should you do? Let's walk through the steps.

1. First, we need to find out if the name is resolving. To do this, open a command prompt on the developer's computer. Ping [www.tailspintoys.com](http://www.tailspintoys.com). The ping command works and shows that the ping went to 192.168.200.99. In this step, we've made an important discovery. Name resolution for [www.tailspintoys.com](http://www.tailspintoys.com) is working from the developer's computer.

2. Now, let's find out if www.tailspintoys.com resolves to the same IP address on another computer. Open a command prompt on your computer and attempt to ping www.tailspintoys.com. Does the ping command show the same IP address as the developer's computer? If so, it may point to a problem upstream from the client computers (such as on the DNS server). If not, it may point to an issue on the developer's computer. For this example, let's imagine that your computer resolves www.tailspintoys.com to 192.168.200.44.
3. Compare the DNS servers that the developer's computer is using and your computer is using to see if they are the same or different. Ensure that the right DNS servers are being used. Let's say that both computers are configured to use the same DNS servers and they are the correct servers to use.
4. Now, you need to figure out why the developer's computer is resolving www.tailspintoys.com to 192.168.200.99 while your computer is resolving it to 192.168.200.44. And, this is throwing you for a loop since both computers are using the same DNS servers. Now is a good time to review name resolution methods: DNS, client cache, and the HOSTS file. This will help us narrow down the culprits. There are two likely culprits in this case. DNS is not a culprit because the developer's computer and your computer are using the same DNS servers. One culprit could be that the developer's computer has a bad cache entry for www.tailspintoys.com. In that case, you can clear the cache by running the ipconfig /flushdns command from a command prompt (or run the Clear-DnsClientCache command from a PowerShell prompt) and then try again. The other culprit is an invalid entry in the HOSTS file on the developer's computer. If the developer has an entry in the HOSTS file for www.tailspintoys.com, then DNS is bypassed and the cache is used (since HOSTS entries are preloaded into the cache). Remove the entry from the HOSTS file, clear the client cache, and then the developer should be able to get to the web site.

### User reports different name resolution than others

A user named Evan reports that he can't sign into Skype for Business, which runs in the company's data center. You immediately suspect DNS, since that is the primary name resolution service used at your company. You check to see if the user's computer resolves the Skype for Business server. It does and the Skype for Business server resolves to 172.16.90.95. But the user cannot sign in. Upon further investigation, you notice that all the other computers in the office resolve the Skype for Business server to 10.10.50.11 and those users can sign into Skype for Business. At this point, we only know that we have one computer that is resolving the server to a different IP address than other computers. So, our approach is going to be to track down the source of the bad IP address. Let's walk through the troubleshooting steps from there.

1. First, let's find out if the bad name resolution is a temporary situation such as a bad cached entry. Clear the DNS client cache on Evan's computer. Then, resolve the Skype for Business server name. In this case, it comes up as 172.16.90.95. Thus, this step eliminates the cache (and thus the HOSTS file) as the source of the problem.

2. Check the DNS servers listed in the TCP/IP configuration on Evan's computer. You check them and find two servers: 172.16.90.10 and 172.16.90.11.
3. Now, let's see if the other computers are using the same DNS servers as Evan's computer is. Check the DNS servers listed in the TCP/IP configuration on other office computers. You check a few and find two servers: 10.10.50.2 and 10.10.50.3. These are different servers than what Evan's computer has.
4. Next, we need to see if we can track down why Evan's computer is getting different DNS servers than all the other computers. We'll look at the DHCP servers to ascertain that. Run the ipconfig /all on Evan's computer and a few other office computers. Compare the DHCP server listed for each. You check and Evan's computer shows the DHCP server being at 172.16.90.1 while the other office computers show a DHCP server at 10.10.50.1. This seems promising – two different DHCP servers. Now, let's find out why.
5. Now we know that different DHCP servers are being used, we should find out why. We know from our previous chapters that DHCP operates per subnet because broadcast traffic (which is what DHCP uses) is usually limited to each subnet. This suggests that Evan's computer is on a different subnet or network. Check the wireless network or wired network that Evan's computer is using. Compare that with what the other office computers are using. You check and find out that Evan's computer is plugged into a different colored network port in the wall. For the other office computers, that port is used for their development computers.
6. Since we found a configuration difference between Evan's computer and the other computer, we need to fix that so that Evan's computer is configured just like the other computers. Unplug Evan's computer from the current network port and plug it into the other network port, ensuring that it matches the location and color of the ports that the other office computers are using.

In this, a user reported a name resolution problem. And that's what it was. But it wasn't due to name resolution not working correctly. Instead, the problem was that Evan's computer was plugged into the wrong network port and ended up using DNS servers for the development network, which happens to have many of the same hostnames and services (for testing). This situation shows you the value in using a working computer while you troubleshoot a computer that isn't working. You can compare the behavior and compare the settings in hopes of finding a difference.

### **IT Admin reports stale name resolution**

An IT admin is performing a web site migration. The web site for [www.alpineskihouse.com](http://www.alpineskihouse.com) is moving from Server1 to Server2. Just before he migrates the web site from Server1 (192.168.200.50), he pings [www.alpineskihouse.com](http://www.alpineskihouse.com) and sees 192.168.200.50 as the IP address. After the web site moves to Server2 (192.168.100.99), he updates the DNS server and changes the entry for [www.alpineskihouse.com](http://www.alpineskihouse.com) from 192.168.200.50 to 192.168.100.99. He does this because the IP address of [www.alpineskihouse.com](http://www.alpineskihouse.com) must match the IP address configured on

the server (and Server2 has the 192.168.100.99 IP address). He tries to go to the web site on his computer but it fails. Then, he boots up another laptop that he has. When he goes to the web site from that laptop, it works. At this point, the information that we have points to a name resolution issue. This is because a computer that just started up correctly resolved the name. We need to figure out why the IT admin's computer is not seeing the new IP address. Let's walk through the troubleshooting steps.

1. We know from the information earlier in this chapter that name resolution comes from various sources such as a DNS server, a HOSTS file, NetBIOS, and the DNS client cache. We need to figure out if any of these sources are the problem with the IT admin's computer. Since the IT admin updated the DNS server, we already know that it has the new IP address for [www.alpineskihouse.com](http://www.alpineskihouse.com). Next, check the HOSTS file on the IT admin's computer (since a HOSTS file could create this problem). You check it but don't find any entries for [www.alpineskihouse.com](http://www.alpineskihouse.com).
2. Since we know the DNS server has the right info and we aren't using a HOSTS file, we need to think about where else name resolution comes from. After pondering it for a minute, you remember that the client cache can also hold name resolution information. You run the `Get-DnsClientCache` cmdlet from the PowerShell prompt on the IT admin's computer. You find an entry for [www.alpineskihouse.com](http://www.alpineskihouse.com) in the client cache. It points to 192.168.200.50 (the old IP address).
3. Because the IT admin's computer has a cached entry for [www.alpineskihouse.com](http://www.alpineskihouse.com) pointing to the old IP address, we need to clear the cache to fix the problem. You run the `Clear-DnsClientCache` cmdlet from the PowerShell prompt on the IT admin's computer. Then, you try to go to [www.alpineskihouse.com](http://www.alpineskihouse.com) and it works correctly.

How could you have checked to see how long the entry would've stayed in the client cache if you hadn't manually cleared the cache? You could've looked at the TTL value in the cache to see how many more seconds the entry would've remained in the client cache.

## Summary

In this chapter, we showed you how names are resolved, including a decision tree incorporating the HOSTS file, DNS client cache, DNS servers, root DNS servers, and authoritative DNS servers. Understanding how names are resolved is one of the keys to quickly troubleshooting and fixing name resolution issues. We looked at the DNS client cache that exists on all Windows computers and showed you what data is stored there and how to view it. By knowing how to view the cache, you can look at it and see if it is the root cause of a problem. We also looked at how the DNS Client service is responsible for adding entries to the cache and that the cache is nonfunctional if the service isn't running. Now, if entries aren't being populated in the cache, you'll know to start the service to fix it. Then, we walked through a few troubleshooting scenarios where we learned to compare different name resolution methods, compare settings on a working computer with a nonworking computer, and check the client cache and HOSTS files as sources of name resolution trouble. You should now be ready to test your name resolution skills in the lab!

## Lab

This lab is designed to validate your name resolution knowledge that you learned in this chapter as well as test your skills in modifying the HOSTS file, checking the DNS client cache, and using PowerShell to view all the properties of a DNS cache entry.

### Preload the DNS client cache

Perform the following tasks:

- Modify the HOSTS file by adding an entry for www.contoso.com with an IP address of 192.168.100.100.
- Check the DNS client cache and verify that the cache has an entry for www.contoso.com.

### Use PowerShell to view DNS client cache

Use PowerShell to view the DNS client cache. Then, choose one entry from the output and use PowerShell to view all the available properties for that entry.

### True or false questions

1. An IT admin reports that the DNS client cache is not being populated on a computer. This is because the DNS Client service is stopped. True or False?
2. The first check that a computer performs during name resolution is checking the DNS client cache. True or False?
3. All DNS servers know the authoritative DNS servers for all domains. True or False?

### Fill in the blanks

\_\_\_\_\_ servers know the authoritative DNS servers for all domains.

During name resolution of an internet domain, the \_\_\_\_\_ servers are queried.

The \_\_\_\_\_ file is pre-loaded into the DNS client cache.

## CHAPTER 9: MANAGING THE DNS SERVER ROLE

---

As we learned in the previous chapter, DNS is a vital component to a network. It is often referred to as a foundation service, like the network. Without DNS, many network technologies don't function correctly. DNS is based on a client and server model where the client asks the server to resolve fully qualified hostnames and the server performs the DNS resolution. In the previous chapter, we covered the client name resolution portion of DNS. In this chapter, we switch to the server side to show you some common DNS server management tasks. We are just over halfway finished with Part 2 which focuses on network technologies. After this chapter, there is one more chapter on DNS zones and then a troubleshooting chapter. Then, we switch gears to Active Directory Domain Services.

In this chapter, we focus on the DNS Manager console which is the tool you'll regularly use to configure and troubleshoot DNS servers. We will take a close look at DNS resource records so that you have a firm understanding of when to use the different records. We'll look at root servers which provide name resolution for domains outside of your organization. Then, we'll talk about forwarders which allow you to dictate which DNS servers resolve names. Our last section is on DNS scavenging which helps keep your DNS environment clean and tidy to help ease troubleshooting and reduce problems with name resolution. By the end of the chapter, you should know enough to effectively manage DNS in your environment as well as understand the important features that DNS provides. You will test your knowledge at the end of the chapter with lab exercises.

### Using the DNS Management Console

The DNS Manager tool is an MMC-based tool which you will use to manage DNS in your environment. The tool is installed with the DNS Server role or as part of the Remote Server Administration Tools feature (the DNS Server Tools subfeature). You can use it to manage one or all your DNS servers. You can perform management tasks such as creating new zones, creating resource records, managing root hints and forwarders, configuring DNS scavenging, and performing a bunch of other tasks. We'll look at tasks these topics in the sections below.

Let's start with looking at the DNS Manager console at a high level so you can understand the layout and information displayed. You can run the DNS Manager console from the Tools menu in Server Manager. Or, you can run it from the Run menu or command prompt by running `dnsmgmt.msc`. The DNS Manager is shown in Figure 9.1 below.

In Figure 9.1, there are two servers – TT-DC-01 and TT-DC-02 – that are being managed by. The primary zone, tailspintoys.com, is shown. In the right pane, some of the DNS records from the zone are displayed.

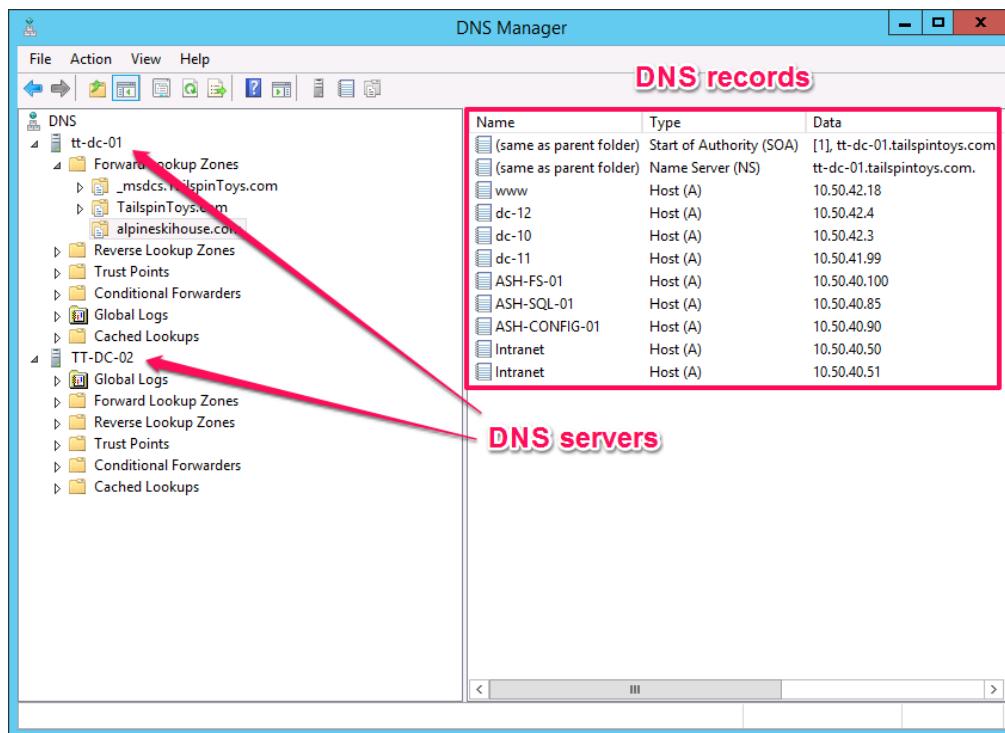


Figure 9.1 DNS Manager is shown managing TT-DC-01 and TT-DC-02 and DNS records from the tailspintoys.com zone are displayed in the right pane.

As you probably noticed from Figure 9.1, DNS Manager shows quite a bit of configuration items and information. If this is your first time looking at the tool, it may look overwhelming. But, as we go through the details, you'll find that it isn't overly complex and your base DNS knowledge from the "Managing Client Name Resolution" chapter will help you gain a quick understanding.

Let's look at the server configuration items first. The server configuration items are accessible from the server properties window. To get to this window, right-click a server in the left pane of DNS Manager (for example, right-click tt-dc-01 as shown in Figure 9.1), click **Properties**, and a window with 8 tabs will be displayed.

### Hands-on Exercise

Open DNS Manager and get to the properties window of a DNS server. From there, click on the 8 tabs as you read through the descriptions of the tabs in the section.

Let's walk through the tabs so you can understand the configuration options available at the DNS server level.

- **The Interfaces tab.** You can configure specific IP addresses to answer inbound DNS queries on this tab. It is common to leave this screen on its default setting where it listens on any of the server's IP addresses (mainly because this requires the least amount of administrative effort). However, if the server is behind a firewall, it is often advantageous to specify the listening IP address because the firewall will be configured to allow DNS communication to that IP address. If you don't specify the listening IP address, then the firewall must be configured for all the IP addresses on the server (which means you must reconfigure the firewall every time you add, change, or remove IP addresses).
- **The Forwarders tab.** Shown in Figure 9.2, the Forwarders tab lists any configured DNS forwarders and whether the server will use root hints if configured forwarders aren't available. Forwarders are DNS servers that are configured to accept queries from other DNS servers and help resolve those queries (often by using the root servers). Forwarders are sometimes used if the DNS server doesn't host a DNS zone for a query. For example, if a client queries for www.alpineskihouse and the DNS server doesn't know about it, the DNS server can use a forwarder to help resolve the name. Forwarders are often used to resolve names outside of your organizations. By using forwarders, you don't have to use your internal DNS servers for internet-based name resolution (instead, the forwarders, often from the perimeter network, handle that). Root hints are the root DNS servers. They are used to find authoritative DNS servers for a domain that isn't hosted by the DNS server. If a query comes in for www.wingtiptoys.com and the DNS server doesn't know about that domain, by default, it will use the root servers to go find the authoritative DNS servers for that domain. You can just use forwarders or root servers and accomplish the same thing. Your setup will be partly based on your network environment and your security posture. We will discuss DNS forwarders and conditional forwarding in more detail later in this chapter.

The Forwarders tab for a DNS servers shows configured forwarders.

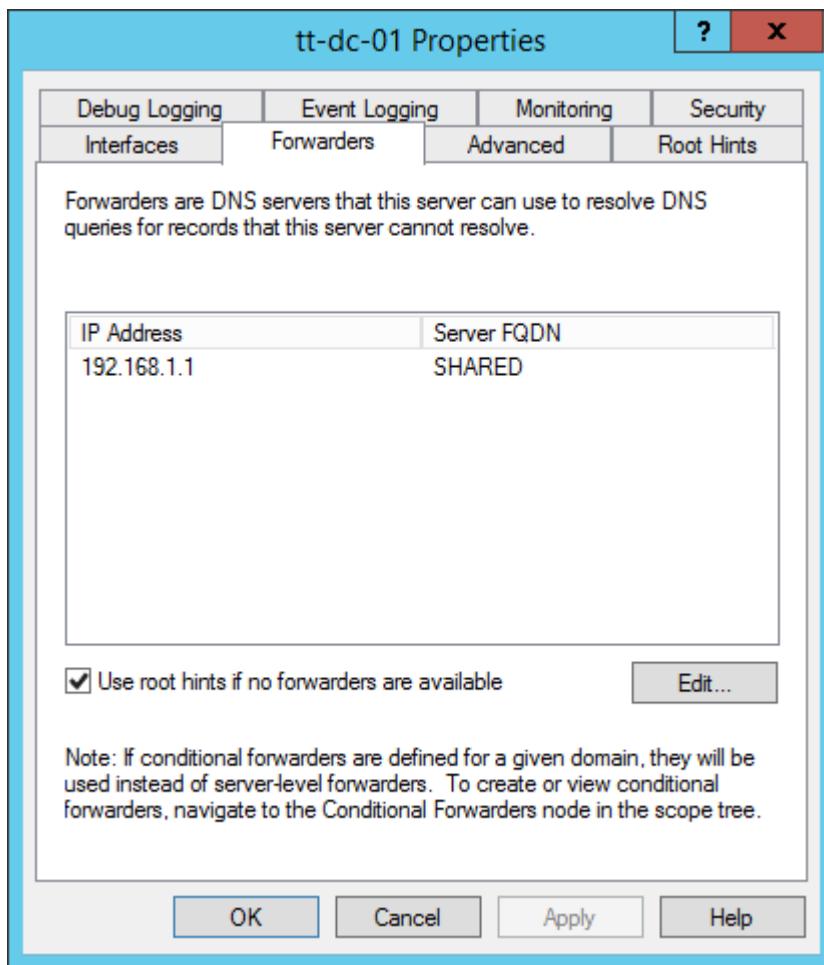


Figure 9.2 The Forwarders tab for the DNS server properties shows a single forwarder configured and root hints are configured if that forwarder isn't available.

- **The Advanced tab.** This tab provides options for advanced (and often, less used) DNS server settings. We won't cover many of the settings because some of them you will either never use or rarely encounter. You can disable and enable the options on this tab by clicking the checkbox next to the option name. The only setting that we want to cover now on this tab is round robin, which is enabled by default (we cover scavenging later in this chapter). It provides functionality like load balancing when a query comes in for an FQDN that has multiple records. For example, let's say that we have 3 web servers that host www.tailspintoys.com. Server1's IP is 172.16.200.51, Server2's IP is 172.16.200.52, and Server3's IP is 172.16.200.53. We want each server to be able to respond to www.tailspintoys.com. So, we add 3 A records for www.tailspintoys.com. 1 A record for each IP address. When clients request www.tailspintoys.com, the first client gets the IP addresses in one order. The next client gets them in a different order. And the order continues to change and loop as more requests come in. Figure 9.3 shows two clients both requesting the IP address for www.tailspintoys.com. The DNS server is configured for round robin and

provides slightly different answers to the clients. Thus, Client1 will use the first IP in the query answer – 172.16.200.51 while Client2 will use the first IP in that query answer – 172.16.200.52. This provides a form of load balancing.

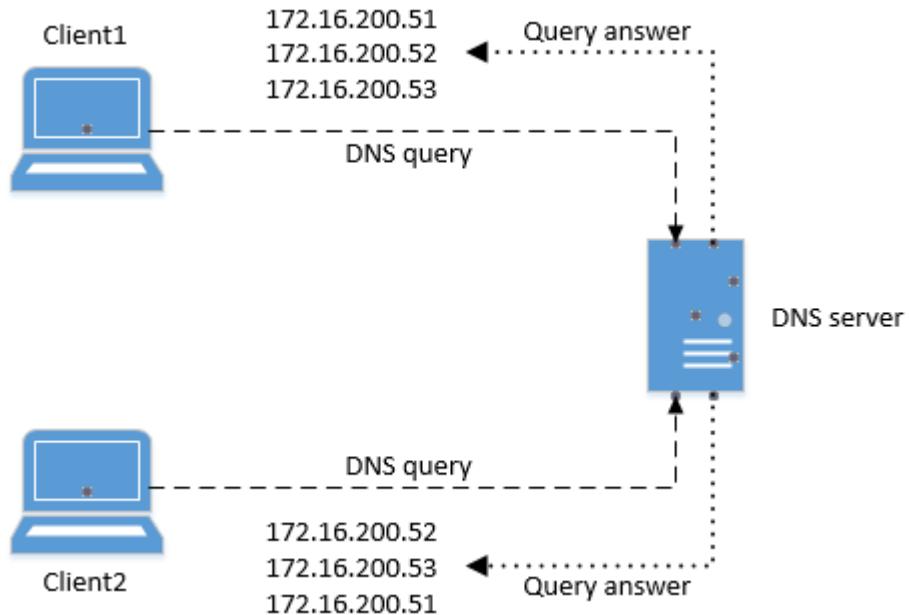


Figure 9.3 Two clients request the same FQDN from a DNS server with round robin enabled.

- **The Root Hints tab.** This tab shows the root name servers which the DNS server uses when it needs to find the authoritative servers for domains. You will occasionally modify these records as the root name servers change. For example, the IP address of a server could change. Or, the name of one of the root servers could change.
- **The Debug Logging, Event Logging, Monitoring, and Security tabs.** These tabs provide settings that you'll only use occasionally, mostly during troubleshooting. The Debug Logging tab contains all the options to record DNS packets sent to and from the server. This is helpful to see the details of client requests and server answers. By default, debug logging is disabled. The Event Logging tab, by default, is configured to log all events to the event log. You'll look at these when the DNS server is experiencing a problem. The Monitoring tab provides a simple way to test your server's DNS functionality. You can test by clicking the Test Now button. This is a good test to validate your DNS server is functioning properly. The security tab provides a method for you to control who can administer the DNS server. As with all technologies that you manage, you should configure DNS so that administrators have the minimum permissions necessary to perform their assigned job tasks.

## Hands-on Exercise

Open the DNS Manager console. Right-click on the server name in the left pane and then click Properties. Click the Monitoring tab. By default, a recursive query test is selected. Click Test Now and view the test results at the bottom of the window. Next, deselect the recursive query and click the simple query test type. Click Test Now to run it. Finally, click the option to perform automating testing every 30 seconds. Click Apply. Wait about a minute and validate that tests are automatically running and test results are updating. After you've verified the automated testing, disable it.

Now that we've seen the console at a high level, let's perform some common DNS server administrative tasks.

## Working with the DNS Server service

You get a monitoring alert that email is down. Not good. You investigate and realize that you can't resolve the email server's fully qualified hostname. After a little more digging, you find out that you can't resolve any names by using your primary DNS server. But you can resolve names by using a different DNS server. In a situation like this when all DNS name resolution fails from a single server, you should look at the DNS Server service. The DNS server's functionality is provided by the DNS Server service. If it isn't running, then the server doesn't respond to DNS queries. Additionally, if the service isn't running, clients can't register their name with the DNS server (this process is called dynamic registration and we cover it in the next chapter titled "Working with DNS zones"). To fix problems with the DNS Server service, you can use DNS Manager. With DNS Manager, you can start, stop, and restart the DNS Server service. Like any other Windows service, you will occasionally need to stop or restart the DNS Server service to troubleshoot a problem or temporarily take the service offline for maintenance.

Figure 9.4 shows the menu where you can control the service.

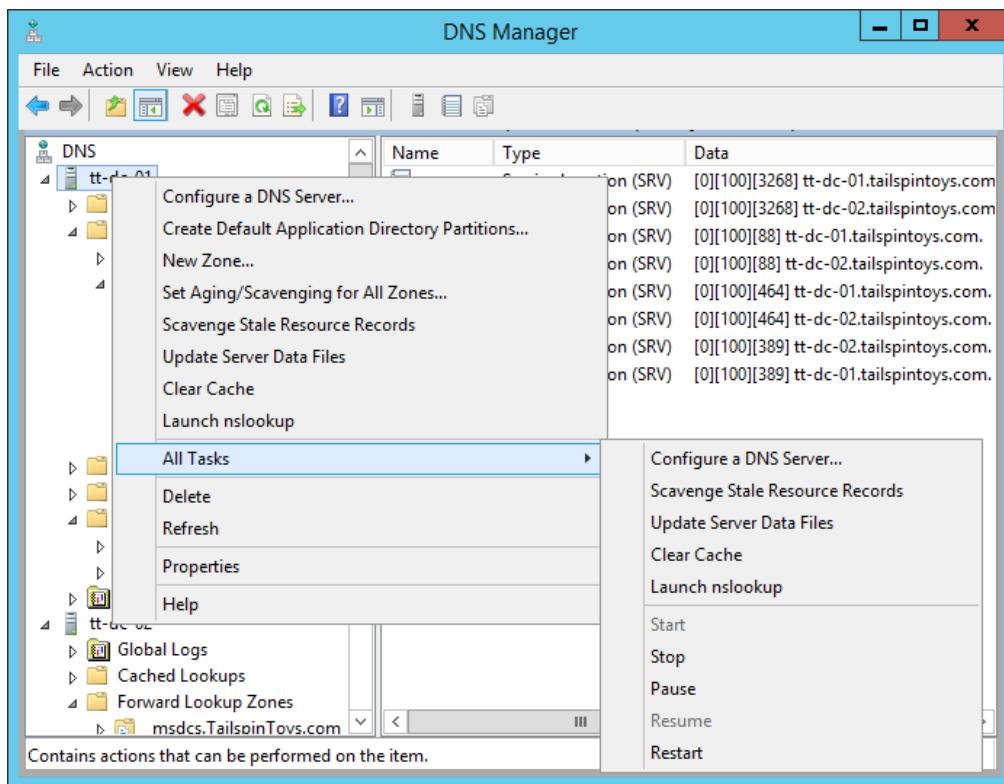


Figure 9.4 The DNS Manager enables you to start, stop, and restart the DNS Server service.

You can also work with the DNS Server service by using PowerShell. To stop the DNS Server service, run the `Stop-Service "DNS Server"` command. To start the service, run the following command:

```
Start-Service "DNS Server"
```

### Hands-on Exercise

Let's see what happens when the DNS Server service is stopped. On the DNS server, Run the PowerShell console as Administrator. Then, open a command prompt. At the command prompt, type `nslookup` and then press the Enter key. At the `nslookup` prompt (shown as ">"), type `www.microsoft.com` and then press the Enter key. Notice that the name resolves. Switch to the PowerShell console. Run the following command:

```
Stop-Service "DNS Server"
```

Switch back to the command prompt. Type `www.bing.com` at the `nslookup` prompt and then press the Enter key. Notice that the name doesn't resolve. That's because the DNS Server service is stopped. Start the DNS Server service and try to resolve `www.bing.com` again.

## Working with the server cache

By default, DNS Manager doesn't display all DNS related items in the console. One key item not displayed by default is cached lookups (lookups that the DNS server already performed and can use to respond to clients until the TTL passes). As an administrator, you will routinely work with cached lookups, such as when you are trying to troubleshoot a problem.

Now let's configure DNS Manager to display the advanced view where cached lookups will be shown. In the toolbar, click **View** and then click **Advanced**. You should now see another object in the left pane named Cached Lookups. Expand Cached Lookups and you will see all the answers for DNS queries that the server helped answer (if you don't see any, use the Nslookup tool to perform some queries and populate the cache). For example, imagine a DNS server that hosts one zone for tailspintoys.com. Clients query for www.contoso.com and www.alpineskihouse.com. In the Cached Lookups container, you will find an entry for www.contoso.com and www.alpineskihouse (note that the entries for .com domains are in the "com" container and only shown with their domain name folder). Those entries will remain in the cache until the TTL time has passed.

When troubleshooting a DNS issue, you may need to occasionally purge the DNS server's cache. This could be necessary if a DNS client is receiving an invalid or outdated response to a query. It is not recommended to purge the entire server cache unless you are troubleshooting an issue and need to. DNS performance is improved using caching, so by deleting the cache, there will be a small performance hit from a DNS query response time until the cache is rebuilt. To minimize the impact of clearing the cache, you can individually clear cached records while leaving most the cache intact.

You can also use PowerShell to view the server's cache. To display the current entries in the cache, without viewing the root hints, run the following command:

```
Show-DnsServerCache | where TimeToLive -ne "00:00:00"
```

### Hands-on Exercise

Enable the advanced view in the DNS Manager console. Then, look at the existing cached records. If you don't see any, open a command prompt and run several ping commands to different domains. Then, right-click the Cached Lookups folder in the DNS Manager and refresh the view. Now view the cached records. Notice that you can delete individually cached records. You can also view the TTL for each record.

Now that you've learned how to navigate through the DNS Manager console, view advanced configuration properties, and manage the DNS Server service, let's dig into DNS resource records and learn what the different record types are used for. The whole reason for a DNS server service is resource records – without them, a DNS Server would be mostly useless.

## Understanding resource records

DNS resource records are the individual records that map names and IP addresses. In your network, users will routinely use the names in their day-to-day work on their computing devices for tasks like visiting a web site, logging into an application, and checking their email. Resource records are what the DNS server uses to provide answers to client queries. For example, if a client queries for [www.contoso.com](http://www.contoso.com), the DNS server will look in the contoso.com zone for a resource record for the host named "www". If it finds one, and the resource record type matches the request, then it sends the IP address to the client and the client can proceed to the web site. There are many types of DNS resource records and some have very niche use cases. The table below, table 9.1, highlights the most common types of resource records and their corresponding uses.

Table 9.1 Common DNS resource records defined

<b>Resource record type</b>	<b>Common use</b>
Start of Authority (SOA)	This record contains information about a zone, such as which DNS servers are authoritative for the zone and the administrative contact information.
Name Server (NS)	The primary purpose for the NS record type is to list all DNS servers that are authoritative for a zone.
Address (A)	The A record is the most used record. Every host on a network usually has an associated A record mapping its name to its IP address. The A record is used for IPv4. For IPv6, the Address record is AAAA.
Pointer (PTR)	A PTR record maps an IP address to an FQDN. PTR records are stored in reverse lookup zones. A PTR record provides the opposite functionality of an A record.
Canonical name (CNAME)	The CNAME record is also referred to as an alias record because it creates a synonymous name for an FQDN. Think of a CNAME like a nickname.
Mail Exchanger (MX)	An MX record provides information about the mail server for a domain. These records are most often added on external, public facing DNS servers so that other DNS servers on the internet can locate the mail server.
Service (SRV)	Whereas MX records are used specifically for mail routing, SRV records provide similar functionality for other services and applications (such as Active Directory). You can create an

	SRV record that details a specific protocol or port for clients to connect to an application.
--	---

Table 9.1 shows only the most common and useful DNS resource records. You can look at some of the less used resource record types by reading about them at <https://technet.microsoft.com/en-us/library/cc958958.aspx>.

Now you should have a better understanding of DNS resource records and how they are used. Next, let's learn what happens when a DNS server isn't authoritative for a domain or doesn't have a local resource record to provide an answer to a DNS client.

## Root servers and forwarders

Root servers and forwarders help a DNS server track down the authoritative servers for domains not hosted by the server itself. When a DNS client sends a query to a DNS server for an FQDN, the servers has a few options:

- Respond to the client with information from its cache.
- Respond to the client with information from its zone file, if it has one for the domain being requested.
- Use a forwarder, if one is configured.
- Use the root DNS servers, if they are configured to be used, to find the authoritative DNS servers for the FQDN's domain.

Now that you understand the high-level options that a DNS server can use when helping to resolve a name, let's look at how root servers work.

## Root servers

A text file containing the root server names and IP addresses is copied to a server during the installation of Windows Server. This file is used to populate the root hints for a DNS server. Sometimes the root servers change – one is added, one is removed, or an IP address of a root server changes. In such cases, you need to update your DNS server with the latest root server information. You can find the latest root server information on the Internet Assigned Numbers Authority (IANA) website at <https://www.iana.org/domains/root/files>.

Next, let's talk about how the root servers work. The root servers do not answer DNS queries like a standard DNS server. They do not provide answers to DNS queries for resource records. Instead, they provide referrals to other DNS servers that can help resolve an FQDN. For example, if you try to resolve [www.tailspintoys.com](http://www.tailspintoys.com) and your DNS server doesn't know the answer, the DNS server will contact a root server. The root server will refer the DNS server to another DNS server authoritative for the .com top-level domain (TLD). The server authoritative for the .com TLD will refer the DNS server to the authoritative server for [tailspintoys.com](http://tailspintoys.com). Finally, the DNS server will query the authoritative server for [www.tailspintoys.com](http://www.tailspintoys.com). The authoritative server will send the answer. Then, the DNS server will provide the answer to the client. This process is

shown in Figure 9.5 (however, there is a forwarder in the diagram which doesn't change the process but just adds one extra step).

### **Hands-on Exercise**

Open DNS Manager. Right-click a DNS server and click Properties. Click the Root Hints tab. Look through the list of name servers listed. These are the root servers. Open a command prompt. Run the nslookup command. At the nslookup prompt, run the server 198.41.0.4 command (this tells nslookup to use 198.41.0.4 (one of the root server) for lookups). Next, try to resolve www.microsoft.com. Notice how you don't get an IP address for www.microsoft.com? That's because the root servers provide the authoritative servers for the TLDs only.

While root servers help resolve names by providing the authoritative servers for TLDs, forwarders are helpful in another way – by performing many of the name resolution tasks on a requestor's behalf. Let's look at a forwarder's job in detail now.

### **Forwarders**

Now, let's walk through how the job of a forwarder. A DNS server can be configured to only resolve FQDNs for domains that it is authoritative for, resolve all FQDNs by using the root servers when necessary, or resolve all FQDNs that are not hosted locally by using forwarding servers. Some administrators use public DNS servers, such as Google's DNS public DNS servers at 8.8.8.8 and 8.8.4.4, as forwarders. A forwarding server's job is to answer queries from other DNS servers. The forwarding server performs all the backend work (contacting the root server, contacting an authoritative server for the .com TLD, and then contacting the authoritative servers for the domain). Once the forwarder server has the answer, it sends it back to the original DNS server and the original DNS server sends the answer to the client. Figure 9.5 shows the high-level name resolution process. In the diagram, Client1 requests name resolution for an FQDN that its DNS server does not know about. The DNS server is configured with a forwarder. The forwarder does not know the answer either so it queries a root server. The root server responds with a referral to a server authoritative for the .com TLD. The server authoritative for the .com TLD responds with a referral to the authoritative server for tailspintoys.com. Then the authoritative server responds to the original DNS server with the IP address for www.tailspintoys.com. The original DNS server sends the answer (IP address) to Client1.

The name resolution process is shown below in Figure 9.5.

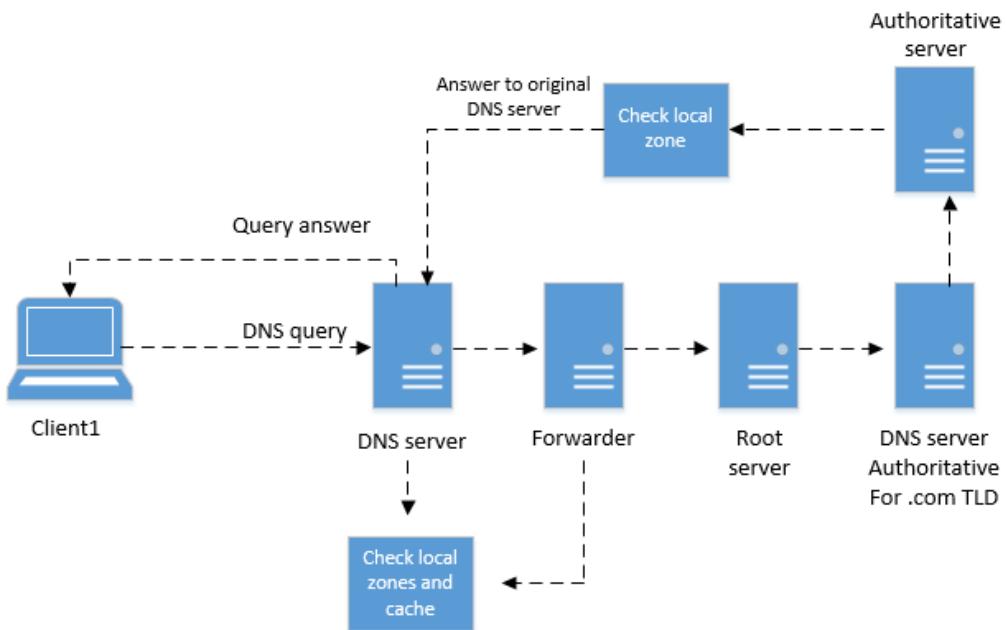


Figure 9.5 The name resolution process is shown for a situation where Client1 attempts to resolve a FQDN that its DNS server does not know about.

### Hands-on Exercise

Open DNS Manager. Right-click a DNS server and then click Properties. Click the Forwarders tab. This is where you can add forwarders. Notice the option at the bottom to use root hints if forwarders aren't available. If you use that option, root servers will be used if your configured forwarders aren't responding. If you do not use that option and you use forwarders, then DNS will not function if your forwarders are not responding. It is a good practice to use the root hints if your forwarders are not available.

Now that you have a clear picture about forwarders and how they handle much of the backend work, let's look at a special kind of forward named a conditional forwarder.

### Conditional forwarders

You work for a toy company. Your company just bought another toy company. Each company has their own network. The plan is to connect the networks together so that employees at both companies can collaborate and share data. But when you try to resolve the names of the other company's internal servers, name resolution fails. Same thing happens when the administrator at the other company tries to resolve your internal server names. This is expected because name resolution is going to the internet. And most companies don't publish their internal resources to their public-facing DNS servers. You need a way to ensure that name

resolution for each company is handled by internal DNS servers. But how can you do that? You can use conditional forwarders! In addition to forwarders, DNS servers offer conditional forwarders. With conditional forwarders, you can configure a DNS server to send queries for a specific domain to a specific DNS server for resolution. For example, you could add a conditional forwarder for contoso.com and forward all DNS requests for that domain to a specific IP address. You can use several conditional forwarders, each with a different domain and different forwarding IP address.

Conditional forwarding is often used when two companies are merging. When merging, it is common that both companies need to connect their internal networks together. Initially, each company still has their own Active Directory environments and network and server infrastructure. Conditional forwarding enables connectivity between the companies' internal networks. Without it, in most cases, FQDNs would resolve to the internet or not resolve at all. For example, consider a situation where Contoso Ltd. and Tailspin Toys have merged. A computer named Client1 needs to contact a Contoso domain controller on Contoso's internal network. Without forwarding or conditional forwarding, when Client1 attempts to resolve the FQDN, the authoritative DNS server for tailspintoys.com would be contacted. That server is in the perimeter network and doesn't have information about the internal network or know about the internal DNS server. Thus, it responds by telling the DNS client that it cannot resolve the name. In cases where a name resolves to a public IP address on the internet (via the authoritative DNS server) and a private IP on the internal network (via an internal DNS server), Client1 will only get the public IP address when attempting to resolve a name (this situation is common in web server scenarios where a company has an internet-based public web site and uses the same URL internally to get to the intranet).

Figure 9.6 displays the name resolution process. We walk through the steps after the diagram.

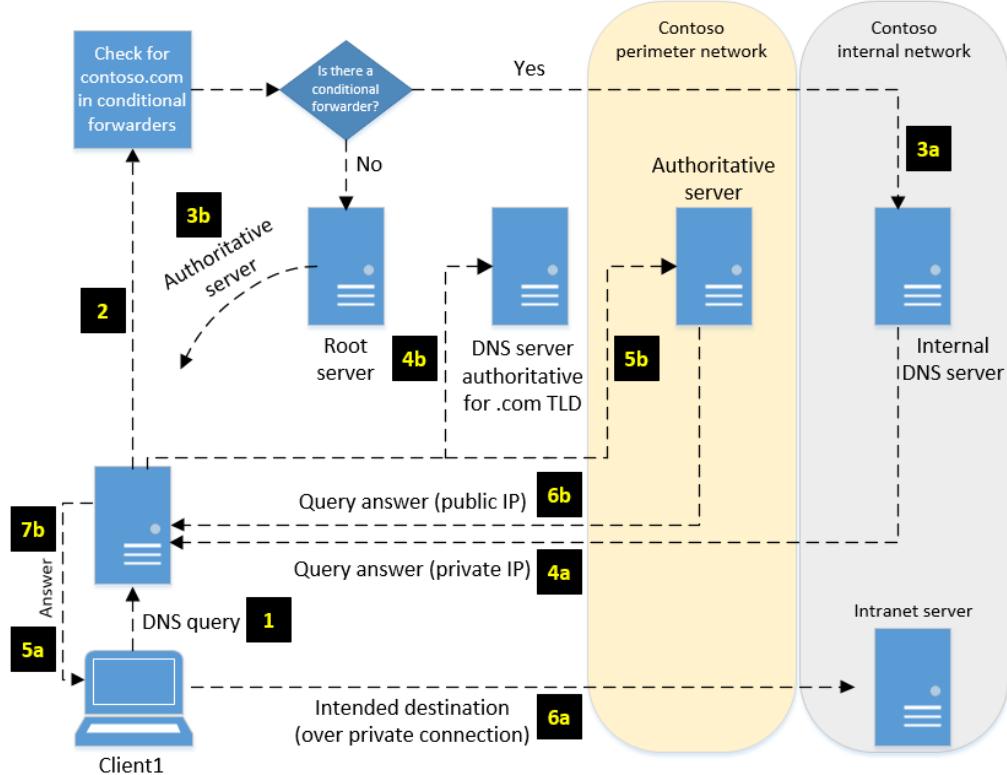


Figure 9.6 Client1 tries to resolve the FQDN of a contoso.com domain controller and gets back different answers based on whether there is a conditional forwarder for contoso.com.

Let's walk through the high-level steps depicted in Figure 9.6 to understand how conditional forwarders can solve name resolution issues. Tailspin Toys and Contoso have connected their networks together and each company has internal connectivity to the other company.

1. A user on a computer named Client1, part of the tailspintoy.com domain, attempts to visit Contoso's intranet server to read company information. Client1 sends a DNS query for www.contoso.com to its configured DNS server which is on Tailspin Toys' internal network. This is labeled as Step #1.
2. The DNS server doesn't host a zone for contoso.com. After checking its cache and not finding an entry for www.contoso.com, the DNS server checks to see if there is a conditional forwarder for contoso.com (Step #2). If there is, the next step is Step #3a. If there isn't a conditional forwarder for contoso.com, skip ahead to Step #3b.
3. Step #3a. The DNS server queries the IP address configured for the contoso.com conditional forwarder, which happens to be an internal DNS server on Contoso's internal network. The internal DNS server provides the internal IP address of www.contoso.com (Step #4a), the DNS server sends the IP address to the client (Step #5a) and Client1 can get to the intranet (Step #6a).

4. Step #3b. Because there isn't a conditional forwarder for contoso.com, the DNS server begins the standard internet name resolution process. First, it contacts a root server. The root server refers the DNS server to a DNS server authoritative for the .com TLD.
5. The DNS server contacts the DNS server authoritative for the .com TLD (Step #4b) and that server refers the DNS server to the authoritative DNS server for contoso.com.
6. The DNS server queries the authoritative DNS server for contoso.com (Step #5b), which resides in Contoso's perimeter network, for www.contoso.com. The authoritative server checks its zones and finds one A record for www.contoso.com. It sends the IP address (which is a public IP address) to the DNS server (Step #6b) and the DNS server sends the IP address to Client1 (Step #7b).
7. User1 sees the public web site for Contoso, not the intranet site (which is not depicted in the diagram).

We've covered quite a bit of content in this section on root servers, forwarding, and conditional forwarding. Let's quickly review before we jump into a new topic. Root servers play a key role in name resolution. Without them, resolving names on the internet wouldn't be possible. As an administrator, you won't often work with the root servers – they'll just be on the root hints tab of a DNS server and need occasional updating. Forwarding and conditional forwarding are great tools for you to use to control which DNS servers are involved in the name resolution process. This is especially helpful when two companies initially merge but still have their own computing infrastructure because conditional forwarders enable internal name resolution across the companies.

Beyond ensuring proper name resolution in your environment, you also need to be thinking about how to keep your DNS database (which contains all your computer hostnames and their IP addresses) updated with the latest information. Computers are often rebuilt, replaced, and retired. And you need to ensure that DNS servers are updated to reflect the latest information. You can use a technology named scavenging to keep your DNS database updated and we are going to look at scavenging in detail now!

## All About Scavenging

You may already be familiar with a process where Windows computers are automatically added to DNS. This can happen in a couple of different ways. One way is a DHCP server that is configured to update DNS whenever it leases an IP address to a computer. Another way is through a computer which is configured to automatically notify the DNS server its IP address and hostname on a regular basis. Automatic updating of a DNS server with hostname and IP address information is known as dynamic DNS (often referred to as "dynamic updates"). For example, let's say that a user named Charles starts his computer named Computer1. His computer requests an IP address from the DHCP server. The DHCP issues 10.50.70.22. At the time of the IP address lease, the DHCP contacts the DNS server and requests that a new DNS record is added for Computer1 with an IP address of 10.50.70.22. We look at the dynamic update process in more detail next chapter. For now, we are going to focus on scavenging so we don't need to worry about how dynamic updates work just yet.

With dynamic updates, it is quite common that resource records aren't removed from the DNS server and that can lead to a large amount of unused or invalid resource records in DNS. Common causes of DNS entries not being removed are computers that have hardware failures and are replaced, computers that are powered off, or computers that are shut down while not being connected to the network. Uncommonly, you can also configure non-dynamic updates (statically entered DNS records) to be scavenged. But because it isn't common, we won't cover the details in this chapter. You can also configure dynamic records so that they are not enabled for scavenging (although this is also uncommon). DNS scavenging is a technology that scans your DNS zones for outdated DNS records and removes them. How does it know when a DNS record is outdated (or "stale")? First, every DNS record has a timestamp which indicates the date and time that the record was initially created or subsequently refreshed. Second, every record is configured for scavenging or not. Third, when DNS scavenging is configured, you set two values:

- **No-refresh interval.** The no-refresh interval is the amount of time that must elapse before a DNS record can be refreshed (refreshing occurs when a computer notifies a DNS server that its hostname and IP address remain "as is" which essentially tells the DNS server "I'm still here, don't scavenge my DNS record"). By default, the no-refresh interval is set to 7 days.
- **Refresh interval.** The refresh interval is the amount of time that must elapse before a DNS record can be scavenged AFTER the no-refresh interval has elapsed. By default, this is set to 7 days. So, by default, scavenging can begin 14 days after a record was created or refreshed. But a record will only be scavenged if it hasn't been refreshed before the 14 days has elapsed.

When scavenging runs, it looks at the DNS record timestamp and adds the no-refresh interval time and the refresh interval time. Then, it compares the current time on the DNS server to see if the current time is greater than the DNS record timestamp after the no-refresh interval and the refresh interval was added to it. If it is, then the record is scavenged. If it isn't, then the record isn't scavenged. For example, let's say a DNS record timestamp is 11/17/15 at 12pm, the no-refresh interval is 7 days, and the refresh interval is 7 days. Then, let's say today is 12/2/15 at 3pm. Based on the comparison, the current time (12/2/15 at 3pm) is greater than the DNS record timestamp (11/17/15 at 12pm) after the no-refresh interval (7 days) and the refresh interval (7 days) are added to it. In this example, the record would be eligible for scavenging after 12pm on 12/1/15.

Figure 9.7 shows a decision tree of the scavenging process.

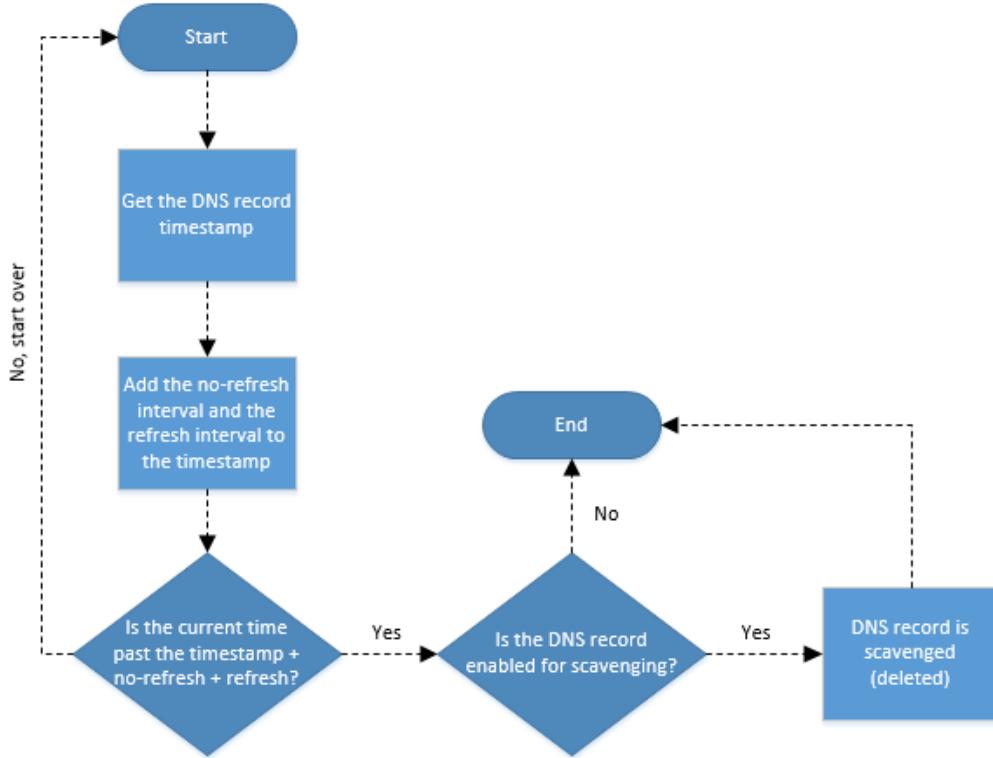


Figure 9.7 The DNS scavenging process examines DNS records and scavenges them if necessary.

So, what happens if scavenging isn't enabled? Often, end users don't notice anything because it usually doesn't directly impact them. But IT teams often notice because it impedes the functionality of some software. If two or three different DNS entries exist for a computer named Server1, management software (configuration management, anti-virus, or similar) may not get the right IP address back when querying DNS for Server1. The management software then attempts to contact the wrong IP address and doesn't get a response (or gets the wrong response). This leads to inaccurate reporting, out of date computers (management software can't push updates), and trouble directly managing hosts. The difficult part is this is that sometimes the correct IP address is received from the DNS server and sometimes the wrong IP address is received. That makes troubleshooting difficult. Think back to the earlier discussion about DNS round robin. That's enabled by default so whenever there are multiple DNS entries for the same hostname, the DNS server will respond to DNS queries like a load balancer where the first host to query gets one answer, the second host gets another, and so on.

### How to turn scavenging on

DNS scavenging must be enabled both at the server level and zone level to function. To enable scavenging at the server level, perform the following steps.

1. Open the DNS Manager console and right-click on the DNS server name and click **Properties**.

2. Click on the Advanced tab and click the **Enable automatic scavenging of stale records** checkbox.
3. Click **OK** to save the change.

Now that server-level scavenging is enabled, you need to enable it on the zone. Right-click on the forward lookup zone and click **Properties**. In the properties window, click the **Aging** button. Click the **Scavenge stale resource records** checkbox. Click **OK** to save the change. At this point, scavenging will begin functioning although you won't see much progress for a couple of weeks. To initiate an on-demand scavenging process, you can now right-click on the DNS server name and click **Scavenge stale resource records**. In the DNS window that asks if you want to scavenge all stale resource records, click **Yes** to initiate the process.

### **Hands-on Exercise**

Open DNS Manager. Expand a DNS server and expand the Forward Lookup Zones container. Click a domain in the left pane. In the right pane, right-click some resources records and view the properties of the records. See if any of the records are set to be scavenged if they come stale. The "Delete this record when it becomes stale" option would have a checkmark next to it if a record was configured to be scavenged. If you don't have scavenging enabled, you probably won't find any. After this chapter's lab, loop back to this Hands-on Exercise and see if any of the records are set to be scavenged then.

In this chapter, we showed you the DNS Manager console and its features and functionality. We discussed the most common DNS resource records used today as well as expanded our knowledge of root servers, forwarders, and conditional forwarders. Lastly, we discussed DNS scavenging and why it is important for an IT administrator to maintain clean and valid DNS domains and zones. Now, you can effectively manage and maintain a DNS server, DNS zones, and DNS resource records while ensuring that your DNS database stays free from some stale records. Test your knowledge by completing the lab!

## **Lab**

This lab is designed to test your knowledge of the DNS server cache, the round robin process, updating root servers, and configuring scavenging. If you haven't completed the Hands-on Exercises in this chapter, perform those now before you start on the lab exercises.

### **Clear the DNS Server cache**

Perform the following tasks:

1. Enable Advanced view mode in the DNS Manager console.
2. Clear the DNS server cache.
3. Confirm that the cache is empty.

## Create two records for DNS Round Robin

Create a DNS test zone and new records for round robin, as follows:

- Create a new forward lookup zone for contoso.com.
- In the contoso.com zone, create an A record for mail.contoso.com with IP address 192.168.100.50.
- Create a second A record with the same hostname with IP address 192.168.100.51.
- Query the FQDN (multiple times and/or from multiple computers) to validate that the DNS server is using round robin.

## Confirm the validity of the Root Hints file

- View the root servers configured for the DNS server.
- Determine if any updates are necessary.
- If updates are required, perform the updates and save the changes.

## Configure DNS scavenging

- Enable DNS scavenging at the server level.
- Enable DNS scavenging for the contoso.com zone.
- Initiate a manual scavenging process.
- Confirm the scavenging process ran successfully.

## CHAPTER 10: WORKING WITH DNS ZONES

---

In the last chapter, you learned how to manage the DNS server role and use the DNS Manager console to perform common administrative tasks such as managing services, clearing the DNS server cache, and viewing log files. Now we are going to look beyond resource records, root hints, forwarders, and scavenging to learn about the various types of DNS zones. It is important to understand the difference between primary, secondary, and stub zones because each zone is designed for different scenarios. You need to use the right zone so that DNS works efficiently! Additionally, you need to know how standard zones differ from Active Directory integrated zones so that you can choose the one that best aligns with your environment.

In this chapter, after we look at the zone types, we will transition to secure dynamic updates. Secure dynamic updating is an optional feature of a zone that allows you to mandate that all dynamic updates to zone records occur securely. Implementing good security is crucial in your environment so it is important to be thinking about security whenever you add or configure a technology. By the end of this chapter, you should know enough to effectively determine which type of DNS zone you should create based on your environment and how to secure your DNS zones by configuring secure dynamic updates.

### Primary, Secondary, and Stub Zones

To add domain names to a DNS server, you must add what's called a DNS zone. That's because domains are stored in a zone. Each DNS zone is responsible for name resolution for a single domain. For example, to resolve names for the contoso.com domain, you would create a single zone for contoso.com and add resource records. Before you create a new DNS zone, the first choice you must make is which type of zone you want to create. There are three types of zones, each with its own traits. Let's look at each zone type so that you know when to use each zone.

#### Primary zones

The primary zone type is the most commonly used zone type. A primary zone is a zone that the DNS server is authoritative for. In other words, the DNS server is the known source of information about the zone as well as the zone records. Imagine that you are starting a new company named Alpine Ski House. You acquire the alpineskihouse.com domain name from a registrar. Now, you want to use the domain for your company's email and web site. One of the first orders of business is to add alpineskihouse.com to your DNS server(s). In this situation, you would use a primary zone for alpineskihouse.com. This is because your server will be responsible for managing the zone as well as the corresponding resource records.

To create a primary zone by using the DNS Manager console, perform the following steps:

1. Launch the **DNS Manager** console, right-click your DNS server name and then click **New Zone**.
2. On the **Welcome to the New Zone Wizard** page, click **Next**.

3. On the **Zone Type** page, make sure the **Primary zone** option is selected, deselect the **Store the zone in Active Directory (available only if DNS server is a writeable domain controller)** option, and then click **Next**.
4. On the **Forward or Reverse Lookup Zone** page, make sure that the **Forward Lookup zone** option is selected and then click **Next**.
5. On the **Zone Name** page, type the name of the zone that you want to create. For this example, type alpineskihouse.com and then click **Next**. The DNS server will be authoritative for alpineskihouse.com after you add it as a zone.
6. On the **Zone File** page, the wizard will specify a zone file with the name of the domain and use a .dns filename extension. For example, if your domain name is alpineskihouse.com, the default zone file name would be alpineskihouse.com.dns. Click **Next** to accept the default name.
7. On the **Dynamic Update** page, make sure that the **Do not allow dynamic updates** option is select and then click **Next**. With a zone stored as a file, you can't require secure dynamic updates. It is a good practice to disable dynamic updates in this situation because it improves DNS security by minimizing the chances of unauthorized DNS record updates.
8. On the **Completing the New Zone Wizard** page, click **Finish** to complete the zone creation.

When finished, you should see the new zone in the left pane.

### **Hands-on Exercise**

Open DNS Manager and create a new primary zone named alpineskihouse.com. Store the zone in a text file and disable dynamic updates.

Now, let's look at secondary zones, learn why you need to create one, and learn how to perform zone transfers.

### **Secondary zones**

Now you have a primary zone file stored locally on a DNS server. But what happens if that DNS server crashes or cannot communicate on the network? Name resolution fails (so nobody can resolve your domain name, can't get to your web site, and can't send you email). To prevent downtime, you need a second DNS server to host your DNS zone. DNS clients, as a good practice, are configured to use two DNS servers. If one fails, the second server can answer queries. But you need your DNS zones on both servers! When you configure a second DNS server to host a DNS zone that is already hosted on another DNS server, you use a secondary zone. A secondary zone is a copy of a primary zone. Importantly, the zone is a read-only zone. That means that all changes to the domain must be performed on the primary zone. Thereafter,

changes replicate from the primary DNS server to the secondary DNS server. Changes replicate based on one of the following actions:

- **The zone's refresh interval expires.** The default refresh interval is 15 minutes. Thus, by default, a zone replicates from a primary to a secondary at least every 15 minutes.
- **The zone is configured to notify secondary servers about changes.** You can set an optional zone setting so that the primary DNS server notifies the secondary DNS server when the zone changes. You should opt to set this setting to reduce the time it takes for the secondary DNS zone to receive zone updates.
- **The DNS Server service is started or restarted on the secondary server.** When the DNS Server service starts, a zone transfer is initiated and the secondary zone is updated.
- **You initiate a manual zone transfer in DNS Manager.** You can right-click a secondary zone in DNS Manager and initiate an immediate zone transfer. This is handy if you are testing zone replication between two servers or if you need to have a zone updated immediately but aren't using the notify feature.

When you view the properties of a DNS record in a secondary zone, the record's properties are grayed out and not editable, as shown in Figure 10.1.

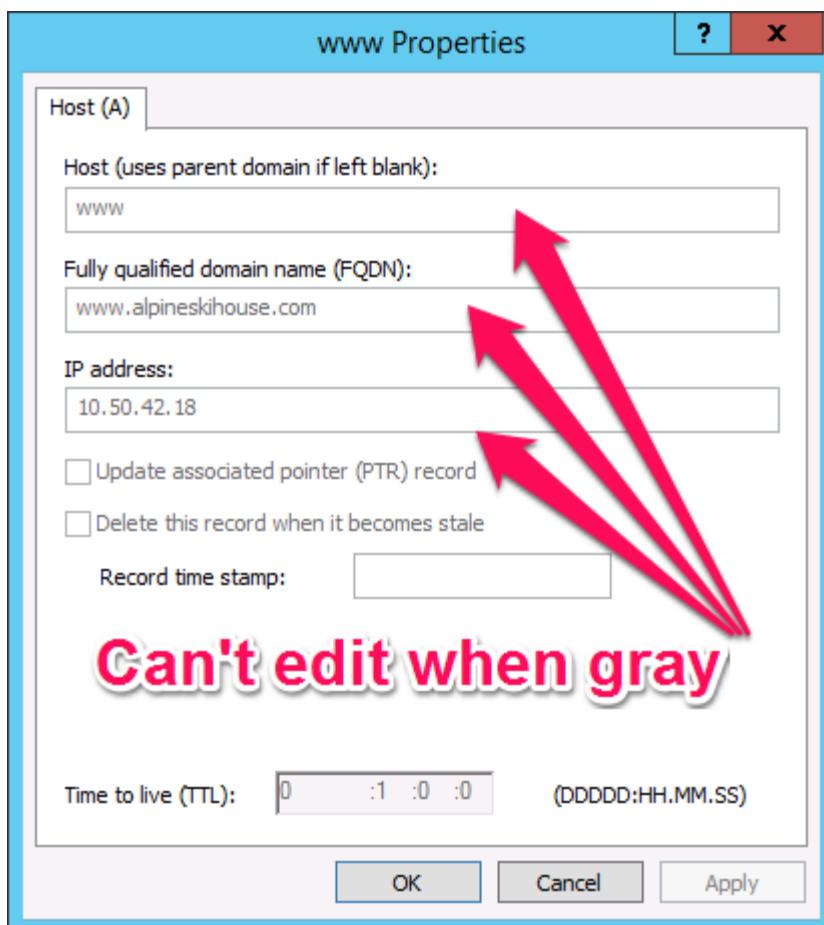


Figure 10.1 A screen capture showing the properties of DNS record in a secondary zone.

## Above and Beyond

A secondary zone cannot be stored in Active Directory. This is because zones stored in Active Directory use Active Directory replication for zone updates while secondary zones only support standard zone transfers between a primary zone and a secondary zone. Because of these different replication mechanisms, secondary zones must be stored as text files on the secondary DNS server.

Now, we are going to create a secondary zone. To perform some of these and other upcoming exercises, you need at least two servers – the primary server that will host a primary zone and a second server that will host the secondary zone (and point to the primary server as part of the secondary zone configuration). You also need to modify the alpineskihouse.com zone that you just created in the last section so that it will accept requests from a secondary server. Perform the following steps to modify the zone to permit zone transfers.

Modify the zone to permit zone transfers

1. Open DNS Manager.
2. Right-click the primary zone that you want to modify and then click **Properties**.
3. Click the **Zone Transfers** tab.
4. Click the **Allow zone transfers** option and then click the **To any server** option. Click **OK** to save the changes.

To create a secondary zone

1. Launch the DNS Manager console, expand your server, right-click Forward Lookup Zones, and then click **New Zone**.
2. On the **Welcome to the New Zone Wizard** page, click **Next**.
3. On the **Zone Type** page, click **Secondary zone** and then click **Next**.
4. On the **Zone Name** page, type the name of the secondary zone that you want to create. For this exercise, type alpineskihouse.com and then click **Next**.
5. On the **Master DNS Servers** page, type the IP address of the primary DNS server in the textbox and then press the **Enter** key. In this example, we use 192.168.1.170 since that is the IP address of the primary server hosting alpineskihouse.com. Click **Next**. If you see any issues or errors adding the IP address of the primary server, double check that the server is configured to allow zone transfer to the secondary server and that the two servers can communicate over the network.
6. On the **Completing the New Zone Wizard** page, click **Finish** to complete the creation of the secondary zone.

You should now see alpineskihouse.com in the left pane of DNS Manager.

### **Hands-on Exercise**

Open DNS Manager and create a new secondary zone named `alpineskihouse.com`. Make sure to use the IP address of the primary DNS server as the master DNS server for the zone.

Now that you have a primary zone and a secondary zone, let's look at how you can initiate an on-demand zone transfer of the zone. You'll use on-demand zone transfers when you first create a secondary zone or when you need to test zone transfers. To initiate a zone transfer, right-click the secondary zone and then click Transfer from Master.

### **Hands-on Exercise**

Add a new Address (A) record to your primary DNS zone. Then, from your secondary DNS server, initiate a zone transfer of the secondary zone. Then, right-click the secondary zone and refresh the zone. After the refresh, check to see if the new Address (A) record is displayed in the secondary zone. Note that this process may take several seconds.

By now, you should be comfortable with primary zones and secondary zones. You know when they are used, how to create them, and how to manage them. But there is a third zone type that we haven't discussed yet! Let's see what the third zone type does now.

## **Stub zones**

In the last chapter, you learned about forwarding and conditional forwarding. You learned that you can configure a DNS server to forward (always or sometimes) to another DNS server for name resolution. Now, we are going to introduce the third type of DNS zone, the stub zone, which performs a similar function to forwarding. A stub zone is a zone that is used to ensure that name resolution for a configured domain uses the name servers defined for that domain. Without a stub zone, name resolution might fail or resolve incorrectly. In an upcoming diagram, Figure 10.4, we show how a stub zone can be used to handle name resolution in a merger scenario. Like a secondary zone, a stub zone is a copy of a primary zone. But instead of containing all the resource records, a stub zone only contains the start of authority (SOA) record, name server (NS) records, and address (A) records for name servers.

Below, in Figure 10.2, a stub zone named contoso.com is shown. Notice that it only contains an SOA record, NS records, and A records for the name servers.

Name	Type	Data
(same as parent folder)	Start of Authority (SOA)	[1087], c-dc-01.contoso.com., hostmaster.contoso.com.
(same as parent folder)	Name Server (NS)	c-dc-01.contoso.com.
(same as parent folder)	Name Server (NS)	192.168.1.31.
(same as parent folder)	Name Server (NS)	c-dc-02.contoso.com.
c-dc-01	Host (A)	192.168.1.190
c-dc-02	Host (A)	192.168.1.195

Figure 10.2 A screen capture showing the resource records in a stub zone.

Let's walk through how a stub zone works. Imagine a scenario where two companies are merging. Contoso Ltd. is merging with Alpine Ski House. They start by connecting their LANs together so that there is network communication over a private connection. Next, Contoso administrators add a stub zone for alpineskihouse.com to their DNS server. Figure 10.3 shows the networks along with the name resolution steps.

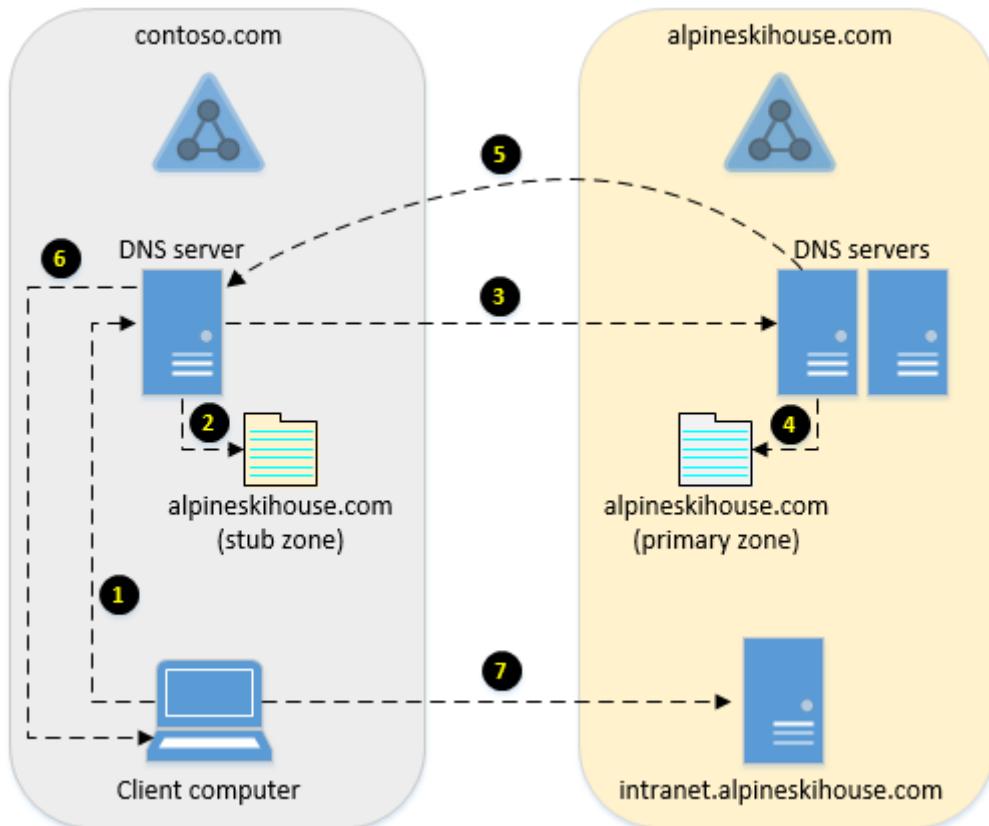


Figure 10.3 A diagram showing the use of a stub zone to provide internal name resolution across companies.

In Figure 10.3, a client computer in contoso.com wants to resolve the name intranet.alpineskihouse.com.

Let's walk through the steps of that name resolution which uses a stub zone.

1. The client computer sends a DNS query to the contoso.com DNS server asking for the IP address of intranet.alpineskihouse.com.
2. The DNS server checks its cache but doesn't see an entry. Then, it checks its configuration to see if it has a zone for alpineskihouse.com. It finds a stub zone.
3. It uses the NS records in the stub zone to contact one of the name servers listed in the alpineskihouse.com domain. It sends a query to that name server to resolve intranet.alpineskihouse.com.
4. The DNS server in the alpineskihouse.com domain checks its cache and configuration and finds that it hosts a primary zone for the domain.
5. It responds to the contoso.com DNS server with the IP address of intranet.alpineskihouse.com.
6. The contoso.com DNS server sends the IP address of intranet.alpineskihouse.com to the client computer.
7. The client computer uses the IP address to communicate with the server hosting intranet.alpineskihouse.com.

Now that we've walked through how a stub zone works, you might be wondering what the difference between stub zones and forwarding is. Conditional forwarding and stub zones both ensure that authoritative name servers handle name resolution for a given domain. But, with conditional forwarding, the servers that handle the name resolution are statically defined when you configure the forwarding. In other words, you manually type the IP addresses for the servers. By contrast, the servers that handle the name resolution for a stub zone are dynamically managed based on the name servers added to the stub zone. Let's walk through two examples.

- **Conditional forwarding.** You configure conditional forwarding for contoso.com with forwarding IP address of 192.168.1.31 (DNS server #1) and 192.168.1.32 (DNS server #2). Everything is functional. 6 months later, the IP address of server #2 changes to 10.15.5.50. Now, your conditional forwarding is partially broken. Conditional forwarding also requires updating when name servers are added for a zone or removed from a zone. If the DNS servers that you use for conditional forwarding are going to stay as is for the foreseeable future, then you should use conditional forwarding.
- **Stub zones.** You configure a stub zone for contoso.com with master DNS servers at 192.168.1.190 (DNS server #1) and 192.168.1.195 (DNS server #2). Everything is functional. 6 months later, the IP address of server #2 changes to 10.15.5.50. 15 minutes later, the stub zone is dynamically updated with the new IP address. Everything remains fully functional. Stub zones also automatically handle new name servers added to a zone and name servers that are decommissioned. If the DNS servers that handle name resolution are going to routinely change, then you should use stub zones instead of conditional forwarding. This lessens the administrative overhead for you because you won't have to perform configuration tasks each time a DNS server changes.

Let's step through the process of creating a stub zone in DNS Manager:

1. Launch the DNS Manager console, right-click your DNS server name and then click **New Zone**.
2. On the **Welcome to the New Zone Wizard** page, click **Next**.
3. On the **Zone Type** page, click **Stub zone**, deselect the **Store the zone in Active Directory (available only if DNS server is a writeable domain controller)** option, and then click **Next**.
4. On the **Zone Name** page, type the name of the stub zone that you want to create. For this exercise, type contoso.com and then click **Next**.
5. On the **Zone File** page, click **Next** to accept the default file name for the stub zone.
6. On the **Master DNS Servers** page, type the IP address of the primary DNS server in the textbox and then press the **Enter** key. In this example, we use 192.168.1.190 since that is the IP address of the primary server hosting contoso.com. Click **Next**.
7. On the **Completing the New Zone Wizard** page, click **Finish** to complete the creation of the stub zone.

When you view your zones in DNS Manager, each zone is labeled with the type of zone it is. In Figure 10.4, there are three primary zones and one stub zone.

Name	Type	Status
_msdcs.TailspinToys.com	Active Directory-Integrated Primary	Running
TailspinToys.com	Active Directory-Integrated Primary	Running
alpineskihouse.com	Active Directory-Integrated Primary	Running
contoso.com	Active Directory-Integrated Stub	Running

Figure 10.4 A screen capture showing zones in DNS Manager.

### Hands-on Exercise

Open DNS Manager and create a new stub zone named contoso.com. Make sure to use the IP address of the primary DNS server as the master DNS server for the zone.

So far, we've looked at all the zones types and when you would use them. But there's one key detail that we avoided during the first part of this chapter. And that's the zone storage options – where you store the zone data and why you use one storage location over another. We're going to talk about zone storage now because introducing it in this section would've taken away from the zone type information. While the technologies are intertwined, it often helps to learn about them separately and then put all the pieces together thereafter! In the next section, we will discuss standard zone storage (in text-based files) and Active Directory integrated storage (where zone data is stored in the AD DS database).

## Standard vs. Active Directory integrated zones

When you were creating zones earlier in this chapter, you probably noticed an option to store your zones in text-based files or in Active Directory. In most cases, such as when you create a new primary zone, you can choose which storage to use. In other cases, such as when you create secondary zones, you will not have a choice because the zone type is limited to only a single type of storage (in the case of secondary zones, they only support text-based file storage). Let's look at the characteristics of each DNS zone storage type so you'll know when to use each of them.

### Standard zone file storage

Standard zone file storage uses text-based files to store all the zone data on the local DNS server where the zone was created. There are some pros to this storage:

- **Compatibility.** Standard zone file storage is most compatible with other DNS servers (such as those running on Linux or on an appliance). Thus, you could have a primary DNS server running Windows Server 2016 and a secondary server running Linux (or vice-versa).
- **Simplicity.** Text-based files are very easy to work with. You can read the zone data by opening the file in a text editor such as Notepad. You can copy and back up the zone data by just copying the zone files to a backup location.
- **Flexibility.** You can update zone data by modifying the zone files directly. While this isn't a common task, and not a good first choice, it is a "nice to have" benefit. Changes to a file can be easily monitored. You can audit a DNS zone file to see when and if anybody reads, copies, or modifies the file. Scripts can perform management tasks such as reading from the file or copying the file to a backup location.

But, in addition to the pros, there are also some cons:

- **Replication is master/slave.** With standard zone files, you have a master server (the authoritative DNS server that hosts the zone) and a slave server (the secondary DNS server). Replication occurs directly between those servers. All updates occur on the master server and are replicated to secondary servers. If you have many DNS servers, managing the replication can become tedious because you must individually add and remove DNS servers from the replication configuration on all primary DNS servers that are replicating. You may also have a single point of failure such as if one primary server is replicating with one secondary server and one of the servers becomes unreachable on the network.
- **You cannot force dynamic updates to be secure.** When computers automatically update their DNS A records, such as when they obtain an IP address from a DHCP server, they can perform the DNS update securely. That secure update process is known as "secure dynamic updates". When you use standard zone file storage, you cannot configure a zone to only allow secure dynamic updates like you can when a zone is integrated with Active Directory. Thus, you may have to turn off dynamic updates on the zone which removes a critical automated DNS update mechanism.

## Hands-on Exercise

Open a zone file in Notepad and review the contents of the file. Compare the contents of the file with the zone data as displayed in DNS Manager. If you do not have a zone stored in a file, create one so that you can look at a zone file in Notepad.

Let's now look at how standard zone file storage compares to Active Directory-integrated zone storage.

### Active Directory integrated zone storage

When you integrate zone storage with Active Directory, you store zone data inside the AD DS database. Beyond this being a Microsoft recommended practice for DNS zones for AD DS, there are some pros to this storage:

- **Multimaster replication.** Instead of a single primary DNS server replicating individually to secondary DNS servers, Active Directory integrated zones use multimaster replication, with replication being controlled by Active Directory. This means that any of the DNS servers replicates with any of the other DNS servers. In addition, any DNS server can perform updates on a zone. This helps spread out the load on DNS servers and reduces the management overhead of configuring and maintaining zone replication. For example, imagine that you have 10 office locations. And you have two DNS servers in each office. When client computers register new DNS records, they can register their records with their local office DNS servers. With multimaster replication, client computers would have to register their records on the primary DNS server at another office. That is inefficient.
- **Force secure dynamic updates.** To force computers to use secure dynamic updates, you must use Active Directory-integrated zones. Forcing secure dynamic updates is a good practice because it minimizes the chances of an unauthorized computer from updating a DNS record.
- **Use additional security on a zone.** When you integrate zones in Active Directory, you can use granular security on the zones and resource records. You can grant a user the permission to update a set of resource records (whereas with a standard zone file, you can't grant such permissions without effectively making the person an administrator of the DNS server). It is a good practice to assign administrators and users the minimum permissions necessary to perform a needed task or their job. And using security on a zone is one way to adhere to the good practice.

There aren't many cons when you store zones in Active Directory. But, one thing to know is that all your DNS servers need to be domain controllers. With standard zone file storage, a DNS server can be a member server or even a standalone server. While storing zones in Active Directory should be your first choice, there are situations where standard zone file storage will make more sense. For example, if you have a perimeter network and need to provide DNS services there, you may not have Active Directory available there to store the zones (since Active Directory is often not available in a perimeter network).

When you store a zone in Active Directory, you have four choices for how you want the zone data to be replicated:

- **To all DNS servers running on domain controllers in the forest.** For this replication option, all DNS servers running in the forest will be replicated to. If most cases where you have a multi-domain forest, you won't want to replicate your zone data across the entire forest because it often isn't needed outside of the domain.
- **To all DNS servers running on domain controllers in the domain.** This is the most common replication choice because it keeps replication to a single domain which is often how name resolution is configured. For name resolution outside of a domain, stub zones and forwarders are often used so there often isn't a need to replicate zones across domains.
- **To all domain controllers in this domain.** This is a compatibility setting for legacy Windows 2000 servers. This is like the domain setting but only useful if you have legacy Windows 2000 DCs.
- **To all domain controllers in the scope of this directory partition.** A directory partition, in this case, is a custom storage location in the AD DS database. To use this option, you must create a directory partition. If you do, you get more granular control over which DCs participate in the replication of DNS data. We won't cover this in more detail in this book because it is outside the scope of regular DNS management tasks.

So far, we've covered all the zone types and the different places that you can store the zone data. You also know how to create and manage zones. But, to round out this chapter of DNS, we need to look at one additional area that we've just glossed over up to this point. And that's secure dynamic updates.

## Secure dynamic updates

There are two primary ways to put data (such as resource records) into DNS zones. One way is to create and update resource records with a tool (such as DNS Manager) or via a script (such as a PowerShell script). The other way is to configure your environment so that computers automatically create and update their own records in DNS. This automatic process is named dynamic updates. And it is most often used for computers that are configured to obtain their IP address from a DHCP server. For most organizations, this process is used for client computing devices such as laptops, desktops, and mobile devices. Servers rarely use DHCP because IP address information is manually set. Thus, servers that are manually set usually do not take advantage of dynamic updates. You can use secure dynamic updates, which ensure that the dynamic update is requested by an authorized computer. Or, you can use nonsecure dynamic updates. Nonsecure dynamic updates are a significant security risk because updates to the DNS record can be requested by unauthorized computers. Thus, when you think about using dynamic updates, only think about secure dynamic updates! When you use dynamic updates, you remove the routine work of creating and removing a large chunk of your DNS records. But not all of them. That's because you'll still need to add and manage static DNS records. Static DNS records are manually managed, often by DNS administrators. Static records are often used with statically configured servers (servers that do not use DHCP to obtain their IP address).

Remember, secure dynamic updates are configurable for zones that are integrated with Active Directory. To configure a zone to use secure dynamic updates, perform the following steps.

1. Open DNS Manager.
2. Expand your DNS server in the left pane and then expand the Forward Lookup Zones container.
3. Right-click the zone that you want to configure and then click **Properties**.
4. In the zone properties window, on the **General** tab, click the **Dynamic updates** dropdown menu, click **Secure only**, and then click **OK**.

### **Hands-on Exercise**

Open DNS Manager and configure an Active Directory integrated zone for secure dynamic updates.

In this chapter, we finished out the topic of name resolution and DNS. That makes 3 straight chapters talking about it! As you can imagine, name resolution is a critical component of your network and that's why we wanted to spend ample time on it. From here out, we won't dive into name resolution much although we do cover a little bit of troubleshooting information about name resolution in the next chapter.

For now, let's test your knowledge of creating and managing DNS zones by having you begin the lab exercises.

## **Lab**

This lab is designed to validate your retention of information from this chapter and perform some DNS server administrative tasks. If you haven't already completed the Hands-on Exercises in this chapter, do that now and then come back to perform the lab exercises.

### **Create and modify an Active Directory integrated primary DNS zone**

Perform the following tasks:

- Create a primary zone named adatum.com and ensure that it is Active Directory integrated.
- Disable dynamic updates

### **Create and modify a standard primary DNS zone**

Create a DNS zone based on the following requirements:

- The domain name is wingtiptoys.com.
- The DNS zone must be stored as a text file on the server.

- The zone must allow zone transfers but only to a server with the IP address of 10.50.99.40.

### **Reconfigure a zone's replication scope**

Modify the adatum.com zone that you created at the beginning of the lab, as follows:

- Change the zone replication scope to replicate to all DCs in the forest.

### **Update a zone's data by using Notepad**

Modify the wingtiptoys.com zone by performing the following actions:

- Use DNS Manager to add a new A record for a host named SERVER20 with an IP address of 192.168.10.20.
- Open the zone file using Notepad.
- Add a new A record for a host named SERVER21 with an IP address of 192.168.10.21. Save the file and exit Notepad.
- Reload the zone file in DNS Manager.
- Refresh the zone in DNS Manager. Validate that you see the new entry.

# CHAPTER 11: TROUBLESHOOTING NETWORK COMMUNICATIONS

---

For the last 5 chapters, we've been looking at network-related topics. While you learned how to manage many aspects of networking, we still have one more crucial area to talk about: troubleshooting. With troubleshooting, you'll often need to use new tools and techniques that you don't use in your day-to-day administrative work. We'll learn several of those tools in this chapter as we tackle the most common problems that pop up when troubleshooting network communication problems.

## Troubleshooting overview

Why can't one computer communicate with another? What if your computer can communicate with a web server but you can't get to a web site? What if one user can resolve a computer's hostname but another user can't? How do you begin to solve these types of problems? When you are troubleshooting network connectivity, there is a good order to use to track down the source of the problem. For example, if a web site isn't responding, part of your troubleshooting is figuring out whether the entire web server is down or whether a single web site is offline. The order that you troubleshoot network connectivity plays a big part in how much time it takes you to find the source of the problem. In most network connectivity troubleshooting scenarios, you should use the following high-level troubleshooting order:

1. Check to see if name resolution is functional.
2. Check to see if the source server can communicate with other destination servers.
3. Check to see if the destination server is communicating on the network.
4. Check to see if the destination service is listening for connections on the service's port.
5. Check the destination server locally for active and listening network connections.
6. Monitor and/or record network communications and look at the packet level to find authentication issues or other issues.

In the following chapter sections, we will look at these troubleshooting steps in detail. While the order above is a good starting point, sometimes you might change the order around based on the information you get from each step. Let's walk through some common scenarios in the next sections, starting with Ping.

## Troubleshooting network communication with Ping

As an administrator, you will often find yourself in a situation where you are unable to connect to a remote computer. It could be that you can't connect to a remote computer by using Remote Desktop Connection. Or, it could be that you can't connect to a web site. You need a tool to diagnose these common network problems. PING is an acronym that stands for Packet InterNet Groper. But few administrators know that because it is commonly referred to just as "ping". First introduced in 1983, it is one of the first network troubleshooting tools ever

written for IP networks, and is still a good tool for finding unknown network connectivity issues. The reason I say “unknown” is that there are other tools that we’ll show later in this chapter that you will use in very specific situations. Ping will often be the first tool you’ll use because it helps you narrow down where a problem is. Based on the information you get back from ping, you may turn to other tools thereafter. Ping is a command-line tool that is mostly platform agnostic, meaning that it is available across a wide variety of operating systems including all modern versions of Windows.

Use cases for the Ping tool

- **Find out if a remote host is active on a network.** For example, you may want to find out if a VM is powered on and connected to the network. Or, you may want to find out if one server has network connectivity to another server. Some common examples of day-to-day use include:
  - **A web site doesn’t open in the browser.** You can ping the web site’s FQDN to see if it is reachable on the network.
  - **Your email application reports that it can’t get your email.** You can ping the email server’s FQDN to see if it is reachable on the network.
  - **Another administrator is deploying a new server at a branch office and you want to see if it is connected yet.** You can ping the server’s FQDN or IP address and see if it is reachable on the network.
- **Perform basic name resolution of an FQDN or IP address.** When you ping an FQDN, it returns information for the ping command along with the IP address it pinged. Thus, you can use it to test basic name resolution. For more advanced name resolution, you should use nslookup or Resolve-DNSName which we discuss later in this chapter.
- **Check for packet loss or degraded connectivity to a remote host.** A default ping command takes a few seconds to complete. However, there is an option to continuously ping a remote host. This is useful if you are troubleshooting intermittent connectivity issues or checking to watch a server go offline and come back online during a reboot.

Ping operates by using the Internet Control Message Protocol (ICMP) protocol. When one host pings another host, it sends 4 ICMP echo requests. When the other host receives the echo requests, it sends 4 ICMP echo replies. The output of a ping command tells you how many of the echo replies were received back and how long it took to receive them. This helps you figure out whether you have intermittent connectivity issues (such as a computer sometimes responding and sometimes not responding, packet loss (where a connection is unstable and some packets are lost in communication), or delayed responses (where a ping request goes through but takes much longer than it should).

Let’s look at some common Ping commands:

What if you need to find out if your computer can communicate with another computer or resolve another computer’s hostname? You can use a basic ping of an FQDN to do that.

```
ping server33.contoso.com
```

This command is the simplest ping command because you only need to use the command and a hostname without any parameters. If the remote computer is on the network and configured to respond to ICMP echo requests, then you will get back 4 echo replies and the approximate round trip time (how long it takes for the packet to be sent from the requestor, go to the remote host, and return to the requestor). If a computer is configured to block or ignore ICMP, you will not get back any echo replies and the computer will appear to be unreachable. A firewall, such as the built-in Windows Firewall, can be used to block or allow ICMP.

What if you wanted to reboot a remote computer and use ping to see exactly when it comes back up? In such a situation, you can use a continuous ping command to remote computer:

```
ping -t server33.contoso.com
```

This command You can stop the ping by pressing CTRL+C. This command is useful if you want to look at packet loss or changes in network connectivity of a few minutes of time.

The next example forces the continuous ping process to use IPv4:

```
ping -t -4 server33.contoso.com
```

Without the -4 parameter, Windows will use IPv6 by default (there are many factors that go into whether IPv6 will be used, but we won't go into those here as they aren't relevant).

If you are troubleshooting intermittent connectivity issues or troubleshooting a high latency network connection, you can use a larger buffer size with Ping to help reproduce issues. This next example sends a larger buffer size that we have specified as 1000 bytes:

```
ping -l 1000 server33.contoso.com
```

This is sometimes useful to see if any changes occur in performance or reliability as the size of the packets increase.

What about a situation where you have an IP address but don't know the server's hostname? You can use Ping for that too! The following example is a ping of an IP address, but also includes the -a parameter. This parameter will also attempt to use reverse DNS, to identify the FQDN of the IP address, if it is available.

```
ping -a 192.168.1.110
```

### **Hands-on Exercise**

Open a command prompt. Then, run ping commands to computers on your network or on the internet. Use the -t, -4, and -l parameters to ping FQDNs.

Then, use the -a parameter to ping an IP address.

### **Troubleshoot network connectivity with PowerShell**

You can use a PowerShell cmdlet that provides similar functionality to the Ping command. The cmdlet is Test-Connection. It uses echo request and echo replies while also reporting the

round-trip time. In addition, it offers some enhanced functionality such as the ability to be scheduled as a job and the ability to specify a source for the test.

The following Test-Connection command illustrates some of its core functionality:

```
Test-Connection -ComputerName server33.contoso.com
```

You can use this command to send ICMP echo requests to server33.contoso.com, like what you would do with the ping command.

### **Hands-on Exercise**

Open a command prompt. Then, run the following command:

```
Test-Connection -ComputerName <FQDN>
```

Replace <FQDN> with a valid FQDN on your network. Try the command for a computer that is running and then try the command for an FQDN that doesn't resolve.

The next command example sends ICMP echo requests to server33.contoso.com from a computer named TT-UTIL-02. You can also specify multiple source computers by separating them by commas. Run this command:

```
Test-Connection -ComputerName server33.contoso.com -Source TT-UTIL-02
```

To test connectivity of multiple computers, use a comma to separate computer names:

```
Test-Connection -ComputerName server33.contoso.com,TT-UTIL-02,TT-DC-02
```

The following command pings server33.contoso.com and returns "True" if the connection succeeded and "False" if it didn't. This is helpful for testing connectivity in scripts, such as if you wanted to run commands against remote computers but only if they are reachable on the network.

```
Test-Connection -ComputerName server33.contoso.com -Quiet
```

In this section, you learned how to use Ping and Test-Connection to check remote connectivity to remote computers as well as resolve hostnames and look for degraded network connectivity. Next, we'll look at a tool to pinpoint specific problems that exist even when you can communicate with a remote computer using Ping or Test-Connection.

### **Testing connectivity to specific ports**

Let's look at that troubleshooting order in a bit more detail, this time with some of the tools to use at each stage:

1. Check to see if name resolution is functional. For example, see if you can resolve the destination FQDN from the source computer by using Nslookup or Resolve-DnsName.

2. Check to see if the source server can communicate with other destinations servers by using the Ping command.
3. Check to see if the destination server is communicating on the network by using the Ping command.
4. Check to see if the destination service is listening for connections on the service's port. For example, by default, an FTP server listens on port 21. You can use the Telnet command to check connectivity to specific ports.
5. Check the destination server locally for active and listening network connections by using the Netstat tool.
6. Monitor and/or record network communications and look at the packet level to find authentication issues or other issues. At the packet level, you can often find detailed connectivity information and errors. You can monitor and record network communication by using Microsoft's Network Monitor tool. But monitoring and recording network communications is an advanced troubleshooting process that often involves a large amount of administrative time. We will not cover it in this book.

Now let's tackle some real-world network connectivity scenarios and see how to diagnose and fix them. We will use a tool named Telnet as part of the troubleshooting. The Telnet tool enables you to check connectivity to a specific network port.

- **You are unable to connect to your FTP server named tt-util-01.tailspintoys.com.** You check connectivity by running the following command:

```
ping tt-util-01.tailspintoys.com
```

That works. The FTP server listens on port 21. Next, you run this command:

```
telnet tt-util-01.tailspintoys.com 21
```

The command returns a "Connect failed" message. This indicates that the FTP server isn't listening on port 21 or a firewall is blocking that communication. In such a case, you would check the FTP server software to ensure that it is running and enabled and then work with the firewall team to investigate if a firewall is blocking the communication.

### **Hands-on Exercise**

Open a command prompt. Then, run the following command:

```
telnet ftp.microsoft.com 21
```

Note that you obtain a successful connection. Next, run the following command:

```
telnet ftp.microsoft.com 80
```

Note that it does not connect successfully since the FTP server listens on port 21, not port 80.

- **You get a report that users cannot get to your intranet site at <http://tt-util-02.tailspintoys.com>.** The intranet site listens on port 80. You suspect that a firewall is blocking the traffic. You decide to see if you can connect to port 80 from the intranet server by running the following command:

```
telnet localhost 80
```

If you can connect, it is an indication that a firewall or other network issue is preventing remote computers from connecting. When you run the telnet command and you obtain a connection, the command prompt clears and all you see is a black screen and a blinking cursor.

- **You want to troubleshoot email messages are not relaying through your email server.**

You can use the telnet command to connect to an email server on port 25. Then, you can run some SMTP commands to send an email message. By using a telnet session to do this, you can sometimes obtain more detailed error messages and find out exactly when a problem occurs (for example, you could find out if the problem occurs the moment you specify the destination email address or when you try to send the email). While troubleshooting email relay is outside the scope of this book, this example is important because it helps show the versatility of the telnet command.

Telnet is a very useful command. But, as of Windows Server 2012 R2 and Windows 8.1, a PowerShell alternative offers similar functionality for the port connectivity tests. We'll look at that now – the Test-NetConnection cmdlet.

### Testing connectivity to specific ports using PowerShell

You can use the Test-NetConnection cmdlet in place of Telnet, except when you want to establish a session and communicate with the remote computer during the session (such as when troubleshooting email forwarding through an email server). Let's look at some examples and show you some enhancements that Test-NetConnection offers.

- **You need to test connectivity to a web site on the internet and you want to return detailed information about the test.** For example, you want to resolve all DNS names as part of the test. Or, you want to know which network adapter was used for the test. You run this command:

```
Test-NetConnection -ComputerName www.microsoft.com -CommonTCPPort HTTP - InformationLevel Detailed
```

It tests name resolution, ping, and connectivity to port 80 in one command! This saves you time. You can also use the -Port parameter and specify a port number. This is helpful because the -CommonTCPPort parameter only accepts HTTP, RDP, SMB, and WINRM and you will often need to test other ports.

Figure 11.1 shows the output of the command.

```

PS C:\Users\brian> Test-NetConnection -ComputerName www.microsoft.com -CommonTCPPort HTTP -InformationLevel Detailed

ComputerName          : www.microsoft.com
RemoteAddress         : 96.7.167.89
RemotePort            : 80
AllNameResolutionResults : 2001:668:108:981::2768
                           2001:668:108:991::2768
                           96.7.167.89
MatchingIPsecRules   :
NetworkIsolationContext : Internet
IsAdmin               : False
InterfaceAlias        : Ethernet
SourceAddress         : 192.168.1.105
NetRoute (NextHop)    : 192.168.1.1
PingSucceeded         : True
PingReplyDetails (RTT) : 40 ms
TcpTestSucceeded      : True

PS C:\Users\brian>

```

Figure 11.1 A screen capture shows the output of a Test-NetConnection command to www.microsoft.com.

- You need to test connectivity to a web site on the internet and you only want to know if it succeeds or fails. You run this command:

```
Test-NetConnection -ComputerName www.microsoft.com -CommonTCPPort HTTP -InformationLevel Quiet
```

The only information return is "True" (indicating a successful connection to port 80) or "False" (indicating a failed connection to port 80). This command is useful for tying into follow-up commands (such as in a pipeline or a script). For example, you could run this command and if the value returned is "False" then you could run the same command with detailed output and email that information to the Helpdesk.

### Hands-on Exercise

Open a PowerShell prompt. Then, run the following command:

```
Test-NetConnection -ComputerName ftp.microsoft.com -Port 21 -InformationLevel Detailed
```

Note how the ping to ftp.microsoft.com fails but the connection to port 21 succeeds.

So far, you've learned how to test basic network connectivity and connectivity to specific ports. But sometimes, you need to troubleshoot a server's network connectivity locally to fix

problems. Next, we'll look at the Ipconfig command for troubleshooting network issues on a server.

### Reviewing the IP configuration of a computer

In most network troubleshooting scenarios, you need to find out the IP configuration of a computer so that you can track down the root cause of the problem. The IP configuration of a computer includes the IP addresses, the subnet mask, the gateway, DNS servers, WINS servers, routes, DHCP server, and physical attributes of the network adapters. The primary command-line tool to obtain this information is Ipconfig. The Ipconfig command is built into Windows operating systems. Many IT administrators have used it often in their jobs. Because you probably have some basic familiarity with it, we will review some common uses and then look at a PowerShell equivalent in this section.

- **You need to find out as much as possible about the local computer's network configuration.** You should run the ipconfig /all command to display detailed information about the local computer's network configuration.
- **You need to release a current DHCP lease.** You should run the ipconfig /release command to release a DHCP lease.
- **You need to find the local computer's current IP address.** You should run the ipconfig command to find the local computers' current IP address.

### Reviewing the IP configuration of a computer with PowerShell

One of the primary uses of the Ipconfig tool is to obtain the current IP address, gateway, and DNS servers of the local computer. In PowerShell, you can use the Get-NetIPConfiguration cmdlet to obtain that information. In fact, I prefer Get-NetIPConfiguration because it returns, by default, a minimal amount of information about the primary adapter (whereas Ipconfig returns information about NICs that are not in use, tunnel interfaces, and IPv6 transition interfaces – mostly extra information that you must weed through).

#### Hands-on Exercise

Open a command prompt. Then, run this command:

```
ipconfig /all
```

Note the large amount of information returned. Next, open a PowerShell prompt.

Run the following cmdlet:

```
Get-NetIPConfiguration
```

Note the lesser amount of data returned.

In this section, we reviewed common Ipconfig uses and introduced the Get-NetIPConfiguration cmdlet. These tools are good for working with the network configuration on

a computer. Next, we'll loop back to troubleshooting remote communication. This time, we will learn about the Tracert tool and a PowerShell equivalent so you can check how communication from a source computer travels to a destination computer.

## Tracing the network route

When computers communicate over a network, they rarely communicate directly (unless they are connected to each other with a crossover cable). Instead, the communication usually travels through switches and/or routers. Each router along the way is considered a "hop". The Tracert tool (tracert.exe) traces the route from one computer to another computer. Tracert records each hop along with the amount of time it takes to communicate with each hop. Tracert uses ICMP to perform the route tracing. Thus, if ICMP is being blocked by a firewall, some or all the hops might not be shown (or will be shown with an error saying that the request timed out). In today's networks, including the internet, ICMP is routinely blocked by firewalls. Thus, Tracert isn't nearly as effective as it once was in the early days of the internet. However, you can still use it to figure out if the route your communication is taking is the preferred route. Often, you will need to work with a network engineer who can provide some of the backend network information to correspond to your route tracing information. Let's look at a couple of troubleshooting scenarios:

- You want to trace the route from your computer to the server that hosts [www.alpineskihouse.com](http://www.alpineskihouse.com) to see if the preferred route is being used. Run the Tracert `www.alpineskihouse.com` command. This command lists all the hops along the way along with name resolution information for each hop.
- You want to trace the route from your computer to the server that hosts [www.alpineskihouse.com](http://www.alpineskihouse.com) to see if the preferred route is being used (but without name resolution). Run the Tracert `-d www.alpineskihouse.com` command. This command traces the route from your computer to the server that hosts [www.alpineskihouse.com](http://www.alpineskihouse.com). It lists all the hops along the way but does not perform name resolution on each hop. This command runs faster because it doesn't perform the name resolution.

## Hands-on Exercise

Open a command prompt. Then, run this command:

```
tracert www.google.com
```

Look at each hop while trying to figure out when the communication leaves your network and your internet provider's network (name resolution will often help ascertain that). Next, run this command:

```
tracert -d www.google.com
```

Notice the difference in speed between the two commands.

## Tracing the network route with PowerShell

Remember the Test-NetConnection cmdlet from earlier in this chapter? Well, it is back! It has a `-TraceRoute` parameter that performs the same function as Tracert. The only difference is that it doesn't resolve the IP addresses of each hop. And it runs quite fast. To perform a traceroute to `www.contoso.com`, run the following command:

```
Test-NetConnection -TraceRoute -ComputerName www.contoso.com
```

### Hands-on Exercise

Open a PowerShell prompt. Then, run this command:

```
Test-NetConnection -TraceRoute www.google.com
```

Compare the output with the output from the Nslookup command earlier in this section.

Now you know how to trace a network path. Next, let's look at some other tools that you will use regularly. One such tool is Nslookup which you use to resolve DNS names. Let's switch gears and look at how it can help you troubleshoot network issues.

### Troubleshooting name resolution

You are trying to find out why one computer is resolving an FQDN to one IP address while another computer is resolving an FQDN to a different IP address. You want to find out which IP addresses the DNS server knows to be associated with the FQDN. But how can you do that? You can use the Nslookup tool. The Nslookup tool (`nslookup.exe`) is a name resolution tool built into Windows operating systems. Nslookup is short for "name server lookup". You can use it to query any DNS server. Of course, the DNS server needs to be reachable on the network to query it. In earlier chapters, we discussed using the Ping tool for basic name resolution checks. Now, we'll expand upon name resolution checks using Nslookup which is a much more powerful tool for troubleshooting name resolution. Here are two use cases that you can use with Nslookup:

- **You want to perform one name resolution as quickly as possible.** For this use case, you would use Nslookup's non-interactive mode. In non-interactive mode, you perform a single query to the default server or a specified server. When the query finishes, you are back at a standard command prompt. For example, you can run the following command:

```
nslookup www.microsoft.com
```

This command will query your default DNS server for the A (address) record for `www.microsoft.com`. By default, if you do not specify a DNS record type, Nslookup will default to the A (address) record.

- **You want to perform several name resolution checks and have flexibility as you go.** For this use case, you would use Nslookup's interactive mode. In interactive mode, you go into an nslookup prompt. Then, you can issue multiple queries without having to run Nslookup

again. For example, if you wanted to look up www.microsoft.com, www.contoso.com, and www.adatum.com, you could just run the nslookup command without parameters. Then, the Nslookup prompt will open. From there, you can type the FQDN that you want to resolve and press the Enter key. When done, you run the quit command and you will be back at the standard command prompt.

The Nslookup command syntax is straightforward. Use nslookup plus the hostname for basic A (address) record lookups. For example, run this command to find the A (address) record(s) for the domain:

```
nslookup www.contoso.com
```

To query for other record types, you should use interactive mode. In interactive mode, you can run the following command to query for other records:

```
set type
```

For example, to lookup an MX record, run the following command interactively:

```
set type=mx
```

Then, type the domain name that you want to query for the MX record. You can switch DNS servers too. If you want to change to a different server, run the following command where <IP ADDRESS> is the IP address of the DNS server that you want to query:

```
server <IP ADDRESS>
```

## Troubleshooting name resolution with PowerShell

The PowerShell equivalent to Nslookup is Resolve-DnsName. The Resolve-DnsName cmdlet performs name resolution queries similarly to Nslookup. The following are some example commands:

```
Resolve-DnsName -Name microsoft.com -Type MX
```

This command queries your locally configured name server for the MX record of microsoft.com.

```
Resolve-DnsName -Name contoso.com -Server 8.8.8.8
```

This command uses one of Google's public DNS servers to resolve the A (address) record for contoso.com.

## Hands-on Exercise

Open a command prompt. Run this command:

```
nslookup www.contoso.com
```

Then, run the nslookup command without parameters to get into interactive mode. From there, query a domain's MX record. Next, open a PowerShell prompt and perform the same name resolution but this time use the Resolve-DnsName cmdlet.

Now, let's look at a couple of troubleshooting scenarios and see how Nslookup and Resolve-DnsName can help you.

Two different IP address for the same web site

A user reports that he can't get to www.contoso.com. You have him ping www.contoso.com and he gets an IP address of 65.55.39.10. Then, you ping it from your computer but get back an IP address of 64.4.6.100. At first, you think that there is a name resolution issue such as an old cached entry or a misconfigured HOSTS file. But before you head down that route, you should run either of the following commands:

```
nslookup www.contoso.com
Resolve-DnsName -Name www.contoso.com
```

Figure 11.3 shows the output of the Nslookup command while Figure 11-4 shows the output of the Resolve-DnsName command.

```
C:\temp>nslookup www.contoso.com
Server: localhost
Address: ::1

Non-authoritative answer:
Name: contoso.com
Addresses: 64.4.6.100
          65.55.39.10
Aliases: www.contoso.com
```

Figure 11.3 A screen capture shows the output of the nslookup www.contoso.com command.

```
PS C:\Users\brian> Resolve-DnsName -Name www.contoso.com
Name                           Type      TTL     Section   NameHost
----                         -----    ----   -----   -----
www.contoso.com               CNAME    3598   Answer    contoso.com

Name : contoso.com
QueryType : A
TTL : 3598
Section : Answer
IP4Address : 65.55.39.10
                                         ↑
                                         DNS record type

Name : contoso.com
QueryType : A
TTL : 3598
Section : Answer
IP4Address : 64.4.6.100
                                         ↑
                                         IP addresses found
```

Figure 11.4 A screen capture shows the output of the Resolve-DnsName –Name www.contoso.com command.

In both outputs, you can see that there are two IP addresses shown. And both IP addresses are for the contoso.com domain. There is an alias (CNAME record) for www.contoso.com that points to contoso.com. All this information comes from the output of the commands. Based on this information, you can conclude that receiving two different IP addresses when resolving

www.contoso.com is OK. In fact, it is by design because the administrator added two records. He might have done this because more than one server is hosting the web site and he can use DNS to load balance the web site. Or, the IP addresses could be virtual IP addresses on a load balancer and the administrator is using two different load balancers. These details aren't important for the point being made here, though. And that point is that Nslookup and Resolve-DnsName can often uncover important details that show why names are resolving the way they are.

### Finding out where the email goes

A user reports that she can't send an email to any email addresses using the microsoft.com domain name. You start to investigate. You immediately start thinking about using Telnet to test connectivity to the microsoft.com email server. However, the first step in this troubleshooting scenario is to find out exactly where the email is supposed to go. You can use Nslookup or Resolve-DnsName to do that. With Nslookup, you perform the following steps:

1. Open a command prompt.
2. Type nslookup and press the **Enter** key to go into interactive mode.
3. Run the following command:  
`set type=MX`
4. Type microsoft.com and then press the **Enter** key.

With PowerShell, you run the following command:

```
Resolve-DnsName -Type MX -Name microsoft.com
```

The output of the Nslookup command is shown below in Figure 11.5.

```
> set type=mx
> microsoft.com
Server: google-public-dns-a.google.com
Address: 8.8.8.8

Non-authoritative answer:
microsoft.com    MX preference = 10, mail exchanger = microsoft-com.mail.protection.outlook.com
> set type=a
> microsoft-com.mail.protection.outlook.com
Server: google-public-dns-a.google.com
Address: 8.8.8.8

Non-authoritative answer:
Name:   microsoft-com.mail.protection.outlook.com
Addresses: 207.46.163.247
          207.46.163.215
          207.46.163.138
```

Figure 11.5 A screen capture shows the output of the Nslookup command.

The output of the Resolve-DnsName command is shown in Figure 11.6.

```
PS C:\Users\brian> Resolve-DnsName -Type MX -Name microsoft.com
Name                               Type   TTL    Section      NameExchange
-----                         -----  --  -----  -----
microsoft.com                     MX     10    Answer      microsoft-com.mail.protection.outlook.com

Name      : microsoft-com.mail.protection.outlook.com
QueryType : A
TTL       : 10
Section   : Additional
IP4Address : 207.46.163.247

Name      : microsoft-com.mail.protection.outlook.com
QueryType : A
TTL       : 10
Section   : Additional
IP4Address : 207.46.163.170

Name      : microsoft-com.mail.protection.outlook.com
QueryType : A
TTL       : 10
Section   : Additional
IP4Address : 207.46.163.138
```

Figure 11.6 A screen capture shows the output of the Resolve-DnsName –Type MX –Name microsoft.com command.

In both outputs, you can see that there is a single MX record for microsoft.com – microsoft-com.mail.protection.outlook.com. There are 3 A (address) records for microsoft-com.mail.protection.outlook.com. Thus, email destined for microsoft.com may go to any of those 3 servers. As part of the troubleshooting, you need to test connectivity to each of the servers. It isn't uncommon to have an intermittent issue (such as occasional email delivery problems) and have the root cause of the problem be one of the several servers being unreachable or down. Nslookup helps you find out that there are multiple servers. But thereafter, you will use Ping and Telnet to troubleshoot further.

In this section, you learned how using dedicated name resolution tools can help you find detailed information about a web site or an email server. And this information applies to other services and scenarios too. As you begin using these tools, you'll become proficient quickly and start using them together to solve network issues. In your day-to-day administrative work, troubleshooting name resolution will be a routine task. In the next section, we'll look at a tool for displaying information about a server's network connections.

## Troubleshooting listening ports

As part of troubleshooting network issues, you will often need to find out if a server is listening on a specific port. For example, if you can't connect to your FTP server on port 21, you may need to connect to your FTP server and find out if it is listening on port 21. Other times, you will need to find out if the right running process is using a specific port. For example, you might have two applications that both use the same default port. So, you configure 1 application to use a non-standard port. As part of the validation of the configuration, you need to start the applications or services and then confirm the port usage. In some cases, you may want to see all a server's active or recently active network connections. This is common when you are investigating security related issues such as a virus or malware outbreak. You can use the Netstat tool to perform these tasks. The tool isn't complicated but sometimes it takes a

minute or two to understand the output and what it means to you. Let's look at some common troubleshooting scenarios and Netstat commands and explain how they can help you troubleshoot network connectivity issues.

- **You want to list all current network connections on a server.** Run the netstat command without parameters and it will list all the current and recently active connections. Figure 11.7 shows a snippet of such output. Note that active connections are listed as "ESTABLISHED" and inactive connection are listed as "TIME\_WAIT" (and, although not shown, "CLOSE\_WAIT"). Inactive connections are connections closed by one computer. In such a situation, the other computer waits to ensure that delayed packets aren't still being delivered and closes the connection after a couple of minutes.

Active Connections				
Proto	Local Address	Foreign Address	State	
TCP	192.168.1.105:135	SERVER33:63437	ESTABLISHED	
TCP	192.168.1.105:443	SERVER33:63460	TIME_WAIT	
TCP	192.168.1.105:3389	BOX1001:59598	ESTABLISHED	
TCP	192.168.1.105:49667	SERVER33:63438	ESTABLISHED	
TCP	192.168.1.105:61236	queue:https	ESTABLISHED	

Figure 11.7 A screen capture shows the output of the netstat command.

- **You want to find all active TCP connections and their associated process ID (PID).** The PID is the ID number assigned to processes as they are created. By finding a process ID, you can end the process to stop the network connection – just right-click the process in Task Manager and click the option to end the task. By finding the PID, you can also track down which application or service is using a port in a situation where more than one application or service could be using a port. Run the following command to find all active TCP connections and their PIDs.

```
netstat -p TCP -o
```

Figure 11.8 shows the output of the command.

Active Connections					
Proto	Local Address	Foreign Address	State	PID	
TCP	192.168.1.105:135	SERVER33:63385	ESTABLISHED	696	
TCP	192.168.1.105:3389	BOX1001:59598	ESTABLISHED	224	
TCP	192.168.1.105:49667	SERVER33:63386	ESTABLISHED	512	
TCP	192.168.1.105:61236	queue:https	ESTABLISHED	2764	
TCP	192.168.1.105:61597	104.40.69.64:https	ESTABLISHED	2764	
TCP	192.168.1.105:63309	SERVER33:epmap	TIME_WAIT	0	
TCP	192.168.1.105:63310	SERVER33:49667	TIME_WAIT	0	
TCP	192.168.1.105:63367	168.62.16.149:https	TIME_WAIT	0	
TCP	192.168.1.105:63372	168.62.16.149:https	TIME_WAIT	0	
TCP	192.168.1.105:63385	SERVER33:epmap	ESTABLISHED	2064	
TCP	192.168.1.105:63386	SERVER33:49667	ESTABLISHED	2064	

Figure 11.8 A screen capture shows the output of the netstat -p TCP -o command.

- **You want to find out which executable file is responsible for opening a network connection.** Often, finding the PID will help you track down the executable file because they are often the same. But that isn't always the case. Thus, you can use a parameter with Netstat to view the executable file responsible for starting the network connection. Run the following command:

```
netstat -b
```

You can combine Netstat parameters together to output as much information as you need based on your objective. To easily work with the Netstat data live, try running your Netstat commands in a PowerShell prompt and pipe the commands to the Out-GridView command. This will output the Netstat information to a window where you can easily type in a filter (such as "LISTENING") to filter the data on the fly.

### **Hands-on Exercise**

Open a PowerShell prompt. Run the following command:

```
netstat -a | Out-GridView
```

In the Filter textbox, type LISTENING and watch the output filter the data so that you only see data based on the filter. Close the output window. Then, run this command:

```
netstat -a -b -o
```

Review the data. Open Task Manager. Make sure you are showing more details. Then click the Details tab and try to find the PID of some of the network connections you see in the Netstat output.

In this section, you learned how to view your current network connections, how to find out which PIDs are facilitating the network connections, how to find the applications responsible for starting a network connection, and how to use Netstat with various parameters to output data useful in troubleshooting. Now, you can use these skills to identify the root cause of common network issues.

In this chapter, you learned how to use several network troubleshooting tools to investigate network connectivity issues. Next, we'll test your knowledge of this chapter in the hands-on lab exercises.

## **Lab**

This lab is designed to validate your retention of information from this chapter and perform some network troubleshooting steps.

### **Use Telnet to validate network connectivity**

Perform the following tasks:

- Use Telnet to connect to ftp.microsoft.com on port 21.
- Use Nslookup to find the email server for a domain. Then, connect to the email server with Telnet on port 25.

## Use PowerShell to investigate network connectivity

Perform the following tasks:

- Use PowerShell to see if www.contoso.com is listening for HTTPS connections.
- Use PowerShell to find out the route your computer takes to get to www.google.com.

## Resolve names

Perform the following tasks:

- Use the interactive mode of Nslookup to change your DNS server to 8.8.8.8.
- In interactive mode, find out if www.alpineskihouse.com is an alias. Then, find out the IP addresses of the web site.
- Use PowerShell to query 8.8.4.4 for the IP addresses of www.adatum.com.

## Use Netstat to look at network connections and listeners

Perform the following tasks:

- Use Netstat to find out the PID for network connections on port 135. Then, use Task Manager to find out the executable name for the PID.
- Use Netstat to find out if your computer is listening on port 443.

## CHAPTER 12: ACTIVE DIRECTORY BASIC ADMINISTRATION

---

Now that you know the basic networking skills, it's time to get introduced to Active Directory Domain Services (AD DS), which draws upon those skills. Active Directory is a huge topic--many books have been written *just* about AD—and we are going to spend the next five chapters skipping most of the background and theory, and focusing instead on what you need to know to perform your routine administrative tasks.

We start by learning to manage Active Directory remotely from a client computer or administrative computer (instead of from a domain controller). Why administer AD remotely? In most organizations, AD is a critical foundational technology that many applications and services rely on to function. It must always be up and running and available. By managing Active Directory remotely, you help contribute to its stability by minimizing the number of tools, applications, and services installed on the domain controllers. In addition, you free up resources that the domain controllers can use for their other operations.

After remote management, we learn to configure your administrative environment for maximum efficiency by customizing your management tools. When you first begin using remote management, it can sometimes be inefficient because your management computer may not have all the tools—or quick access to them—to manage your environment. To fix that, we look at some options for combining multiple tools into a single management console. Then, we jump into organizational units (OUs), which store most of the AD DS objects. And finally, we see how to protect your Active Directory objects from accidental deletion. This information can save you the headache of a mistaken click or drag-and-drop action that deletes an unprotected object. How to manage your environment remotely

### Managing your environment remotely

A common mistake for administrators new to managing Active Directory is to connect to a domain controller with Remote Desktop Connection and perform all their administrative tasks (such as new user account creation, GPO edits, and password resets) through that connection. Here's why is this a mistake: Only 2 Remote Desktop Connection sessions are supported at one time. So, if you have more than 2 administrators, you will sometimes not be able to connect to a specific domain controller if the other administrators are already taking up the available sessions.

Additionally, when you perform your administrative tasks through an RDP connection to a domain controller, processing of your tasks takes place on the domain controller. If you are running a processing intensive script, it might slow down other operations on the domain controller. And finally, third-party software is sometimes required or desired to perform some management tasks. While domain controllers are exceptionally stable servers, they become less so when other software is installed. The best scenario is to dedicate a domain controller to just AD DS and DNS by minimizing all additional software applications. Instead, you can install those on a management computer.

Remote management isn't just for managing Active Directory! It is also good has similar benefits for managing other areas of your IT environment. But it is especially important for a

foundational service like AD that is relied upon by many applications and services at a company. Recent estimates say that 95% of the Fortune 500 companies use Active Directory. And that estimate is probably valid for all businesses except very small organizations, such as those with less than 15 computers. Thus, it needs to be reliable. And, because it is usually the primary source for authentication and authorization, it needs to be secure.

Instead of connecting to a domain controller by using Remote Desktop Connection, you should install the required AD DS management tools on a dedicated administrative computer. Then, you can use the tools to remotely perform administrative tasks. Some administrators opt to use a server for remote administration. But you can also use a client computer. There aren't clear benefits making the server operating system or the client operating system better for all remote management tasks. Experiment and see which one you prefer. To maximize security, you should separate your administrative activities (managing servers) from your standard employee activities (e-mail, web surfing, social media). For example, you can have a dedicated administrative computer which you sign into with administrative credentials when you need to perform administrative tasks. And, you can have a dedicated computer for your e-mail, web surfing, and other non-administrative activities. By separating the activities, and using a standard (non-administrative) account for all online activity, you can minimize the chances of malware gaining a foothold into your environment. If malware infects your dedicated computer, it can damage your computer or data. But it won't have administrative rights to propagate that damage across your entire network.

Think of an Active Directory domain as a jail and your domain controllers are buildings in the jail. In a jail, everything and everybody coming in are checked for contraband, so unauthorized stuff doesn't get into the jail where it causes fights, gambling, escapes, and other trouble. In an Active Directory domain, the equivalent trouble might be a virus, malware, unauthorized access to data, or data leakage. You introduce risk by allowing stuff in. A domain controller is a high-security server that is responsible for authenticating and authorizing just about everything on a network. You don't want things coming and going. You should not surf the web or install applications on a domain controller, as your goal is to minimize change on it. You need it locked down, behind a huge wall, where it can do what it needs to do without interruption or corruption!

## Configuring remote management

By default, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 are configured with remote administration enabled. Thus, you can use Server Manager to remotely manage them by default. Older versions of Windows Server have remote administration disabled by default. In Windows, Windows Remote Management (WinRM) is responsible for facilitating remote management operations. To see if WinRM is configured and listening on a server, run the following command:

```
winrm enumerate winrm/config/listener
```

The output of the command will look like the following output:

```
Address = *
Transport = HTTP
Port = 5985
Hostname
Enabled = true
URLPrefix = wsman
CertificateThumbprint
ListeningOn = 127.0.0.1, 10.50.40.99
```

To enable remote management, perform the following steps.

1. Open Server Manager.
2. Click **Local Server** in the left pane.
3. In the right pane, click **Disabled** next to "Remote management".
4. In the Configure Remote Management window, click the **Enable remote management of this server from other computers** checkbox and then click **OK**.

A dedicated Windows service, named Windows Remote Management (WS-Management), provides the backend functionality for remote management. The service runs by default and starts automatically at boot. But if it gets disabled or is stopped, then remote management will not function.

### **Hands-on Exercise**

Open an elevated command prompt. Then, run the following command:

```
winrm enumerate winrm/config/listener
```

Next, run the following command:

```
winrm get winrm/config
```

This will display a detailed output of the server's WinRM configuration.

Windows PowerShell and Server Manager both use WinRM to perform tasks on remote computers. Many other management tools do too. Though there are exceptions, such as when DCOM is used, especially with older versions of Windows Server.

As an administrator, you should plan to enable remote management across all your servers. Otherwise, you'll have to use Remote Desktop connections and console connections to perform management tasks. And we already told you why that's a bad idea! To simplify this process, you can use Group Policy. We don't look at Group Policy in detail until Chapter 17. So, for now, just know that a Group Policy setting can help to enable remote management very easily.

Now that remote manage management has been enabled, what can you do? Let's look at a couple examples of what you can do with remote management.

- **Run a command on a remote server.** Suppose you want to check which DNS servers were configured on a server. To do that, you can run the following command on a remote server named Server1:

```
ipconfig /all
```

You can run another command to retrieve the ipconfig information from a remote computer, but display it locally. This command is:

```
winrs -r:server1 ipconfig /all
```

An example of when you would do this is if you wanted to check which DNS servers were configured on a server.

- **Use an interactive PowerShell session on a remote server (PowerShell remoting).** Let's say you are on your client computer. And you want to have an interactive PowerShell prompt targeted to Server1. With an interactive PowerShell prompt, you can run as many commands back to back as you want, just as though you were running PowerShell locally on a server. You can run the following command to start an interactive prompt on Server1:

```
Enter-PSSession -ComputerName Server1
```

The command will change the PowerShell prompt so that it is targeted at Server1. Then, all the commands you enter thereafter are run on Server1 and output to your session. This enables you to perform management tasks right from your client computer. You can check to see which hotfixes have been installed, look at the most recent event logs, or restart a service. This type of remote management is fast and efficient and all your PowerShell commands are available to run again targeting a totally different server. When you are finished with the interactive session, you can run the exit command to exit the session.

- **Run a PowerShell command on multiple remote computers simultaneously.** Imagine that you want to find out something about many of your servers. Maybe whether a certain service is running or stopped. You can use the Invoke-Command cmdlet to do that. For this example, create a servers.txt file that contains multiple server hostnames, 1 per line. Then, at the PowerShell prompt, navigate to the location of servers.txt. Then, to find all services on all the servers in the servers.txt file that can be paused and then started again, run the following command:

```
Invoke-Command -ScriptBlock {Get-Service | where CanPauseAndContinue -eq "True"} -ComputerName (Get-Content .\servers.txt)
```

If you want to see which of the servers in servers.txt runs the World Wide Web Publishing Service (W3SVC), you can run this command:

```
Invoke-Command -ScriptBlock {Get-Service | where Name -eq "W3SVC"} -ComputerName (Get-Content .\servers.txt)
```

- **Use GUI tools to manage a remote server.** Besides the command-line tasks, you can also manage servers remotely using GUI tools such as Server Manager. Some tasks are simpler or

more intuitive with a GUI. Such as changing the address for an Active Directory user account. And GUI tools are fully capable of managing remote servers!

### **Hands-on Exercise**

Open a PowerShell console. Then, run this command:

```
Enter-PSSession -ComputerName <Server1>
```

The PowerShell prompt should change to show that your session is now tied to the remote server. Then, run the hostname command to confirm that the command runs on the remote server. Try other commands too.

As you've seen, managing servers remotely is more efficient than individually connecting to servers via Remote Desktop Connection. It also increases security and minimizes the chances of additional software being installed on servers. Now, that you know how to enable remote management and you understand the types of tasks that you can perform remotely. In the next section, well, let's look at customizing your management environment to makes things even more efficient!

## **Customizing your management environment**

For many of your management tasks, you use a tool such as Active Directory Users and Computers or Computer Management. And often, that tool is specific to a task or application. In the span of a couple of hours, you might use 10 or 12 different tools: You might have an MMC open to manage your email server, a PowerShell prompt open for managing Active Directory, Computer Management open to manage disk drives, and Services open to review some service account settings. Soon, your taskbar is filled up with many tools, multiple instances of tools, and a variety of other applications. It can be hard to keep track of all the tools. Even just for Active Directory management, there are quite a few tools that you will use regularly!

You can customize your environment to cut down on the number of tools and the number of windows that you have open to perform your management tasks. In Chapter 3, we showed you how to customize Server Manager. In this section, we will focus on customizing your management environment by using the Microsoft Management Console (MMC). If you aren't familiar with the MMC, you won't find it intuitive initially.

By default, the MMC has nothing to indicate what it is, what it can do, or how to do it. That's why I said it isn't intuitive. But, it isn't difficult once you work with it a little bit. Let's start by adding some tools. Note that you can't add all MMC-based tools by default. For example, the Active Directory tools are not available until you install the role administration tools for AD DS (specifically the AD DS Snap-Ins and Command-Line Tools).

The following high-level steps show the process to add the Computer Management tool to the MMC:

1. In the MMC, click **File** and then click **Add/Remove Snap-in**.
2. In the **Add or Remove Snap-ins** window, in the left pane, click **Computer Management** and then click the **Add** button.
3. In the **Computer Management** window, you can maintain the default which will manage the local computer. Or, you can click the **Another computer** radio button and manage a remote computer.
4. Click **Finish**.

### **Hands-on Exercise**

Right-click Start, click Run, type MMC, and then press the Enter key. Add the Computer Management snap-in targeted at the local computer. Note that this is the same Computer Management tool that you use when you run it from the Administrative Tools on a server.

Now, you have an MMC with one tool. You can then add more tools. You can even add multiple instances of the same tool with each instance managing a different server. In Figure 12.1 below, the MMC has 5 tools added: Computer Management targeted at the local computer, DNS Manager, which itself has two DNS servers it is managing, Active Directory Users and Computers, DHCP Management, and Event Viewer targeted at a remote server named TT-DC-01.

An MMC with 5 tools added.

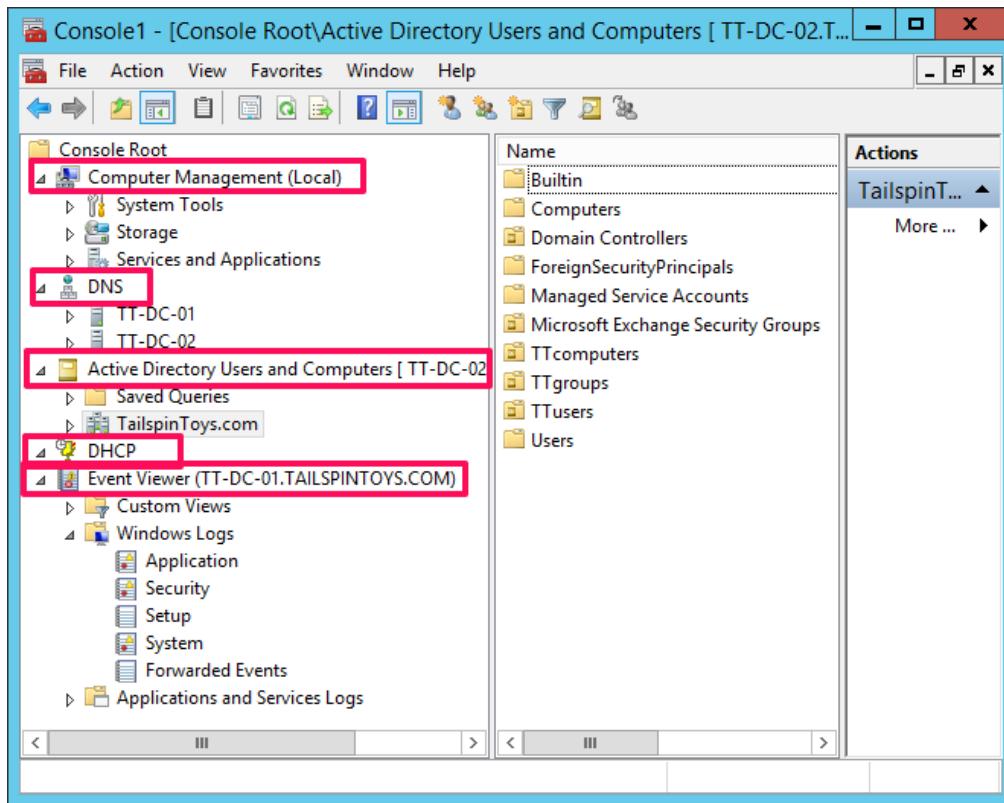


Figure 12.1 An MMC with 5 tools added

Once you spend time customizing the MMC, you need to save it. Otherwise, the next time you log off or reboot, all your customizations will be lost. To save the MMC, click **File**, and then click **Save As**. For the file name, type **%ProgramData%\Microsoft\Windows\Start Menu\Programs\Administrative Tools** and then type a name such as **Brian's MMC**. Then click **Save**. Thereafter, a shortcut for your customized MMC will be in the Administrative Tools section of the Control Panel. You can run it and all the customizations and tools will be loaded.

### Hands-on Exercise

Customize your MMC to include several tools. If you have multiple computers available, try targeting some of the tools at remote computers. Then, save the MMC. Feel free to wander a little bit, explore, look around, try a few tools that you aren't even familiar with. It will help you become comfortable with the MMC.

Now that you know how to remotely manage servers and customize your management environment, we are ready to jump into the second half of his chapter where we begin diving into Active Directory related management tasks.

## Working with Organizational Units

Active Directory Domain Services (AD DS) is the foundation authentication and authorization service that manages computers, users, and other objects for many organizations. You have probably worked in and around AD DS, even if you haven't managed it directly before. At its simplest, AD DS is a database to store all your identity information, authorization information, and a repository of policies to manage your computers. AD DS stores a lot of different objects – users, computers, and groups are the most common.

To facilitate object management and security, there is a logical structure to Active Directory. At the top of the structure is a forest. A forest is a security boundary, which simply means that Active Directory administrators have total control of everything in the forest but not in other forests. Just below the forest are domains. Often, there is just one domain and one forest. A domain is an administrative boundary. In large organizations, you might have one set of administrators manage one domain. In other organizations, administrators manage all the domains. Figure 12.2 shows a forest and domain layout.

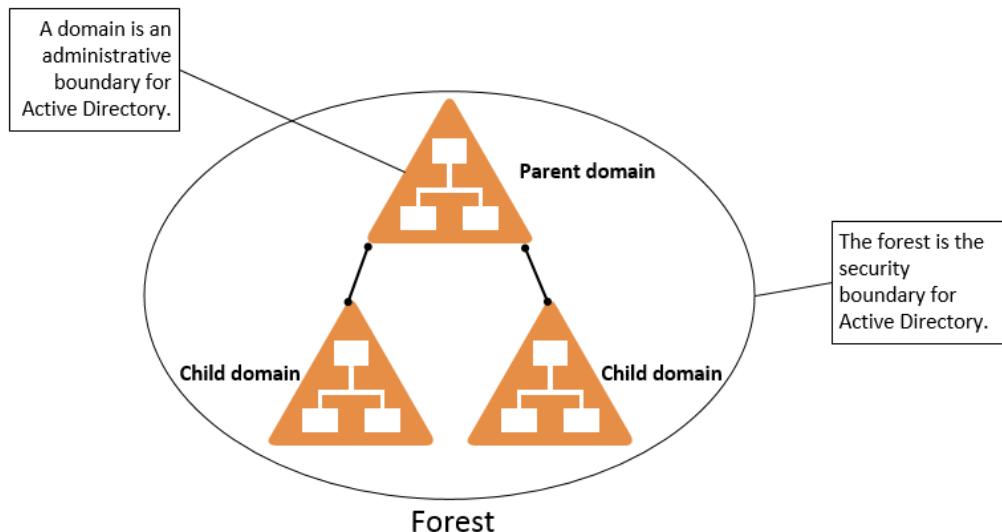


Figure 12.2 A forest and domain layout

## Creating OUs

To efficiently manage user, computer, and group objects, you need a way to store them in a method that facilitates efficient delegation. And that's where Organization Units (OUs) come in. OUs are objects in Active Directory that are used to store other Active Directory objects such as users, computers, and groups. You can store computers in OUs. Users. Groups. You can even store OUs in OUs. In such a scenario, the top-level OU is called the parent OU. OUs under a top-level OU are called child OUs. There are two primary reasons to create an OU:

- **You need a delegation point.** In many organizations, Active Directory is managed by Active Directory administrators. But, there are desktop or helpdesk administrators that help with some day-to-day administrative tasks such as creating new users, resetting user passwords, and managing group memberships. To enable those administrators to perform those tasks,

you need to give them the rights. Delegation is the act of giving them the rights to perform a specified set of tasks against a specified set of objects. For example, let's say you have an office in Toronto. And you have a helpdesk administrator that works in Toronto. You want him to be able to help Toronto users with their passwords and account lockouts. You can create an OU named Toronto, put all the Toronto user objects in the OU, and then delegate the administrator rights to reset passwords and unlock user accounts for just that OU. As part of maintaining a secure environment, it is important to assign the minimum amount of rights necessary for an admin to perform their job. Part of that is using the right scope and this is where OUs come in.

- **You need to link Group Policy Objects (GPOs) to objects stored in an OU.** We don't talk about Group Policy for a few more chapters. For now, know that GPOs are policies to customize and enforce computer settings. In most companies, you have different departments and employee types. Often, the requirements are different between departments and employee types. For example, for users with laptops, you may want to use more secure settings in case a laptop is lost or stolen. For the marketing team, you may want their desktop background to show the company logo. You can use GPOs to do this. But to make sure that the GPO doesn't configure a setting for all your company's computers, you need to associate (link) a GPO to an OU that contains the computers that you want to configure. There are other ways to do this too, but for now, we are focused only on the method of linking a GPO to an OU.

A common mistake that administrators make is to use OUs like folders in Windows. They use them to organize objects. For example, they might store computers running Windows 10 in one OU and Windows 8.1 in another OU. They might store users that are managers in one OU and non-management users in another OU. Soon, there are hundreds or thousands of OUs, just to organize all the different objects in the desired way. When it comes time to delegate permissions, it can be painful because you either must delegate in a whole bunch of different places or you must delegate at a very high level in the OU structure which often leads to delegating more access than is necessary. The same situation happens in that environment when you need to create a GPO for a subset of computers. The organization of the objects may not match the organization you need for your computer configuration. For example, imagine a scenario where all computers are stored based on the floor number of their location. There is an OU named Toronto1 for floor 1, an OU named Toronto2 for floor 2, etc. And there are 10 floors! You need to configure all Toronto computers with a GPO. You must link it to 10 different OUs. You can also use security and WMI filtering, but we won't cover that here.

To avoid the downsides of using OUs like Windows folders, it is a good practice to plan your OU structure while you plan your delegation and GPO strategy. That way, you can align them and get them working together seamlessly. It is also a good practice to minimize the total number of OUs, simplify the OU layout, and adhere to the two reasons to create OUs mentioned earlier (as a delegation point or as a GPO target). By doing so, you will have an easier time maintaining and troubleshooting the environment.

## Managing OUs

Let's look at the common OU management tasks. We will employ two tools for all the tasks: Active Directory Users and Computers and Windows PowerShell. But before we begin, let's quickly look at how you get these tools on your administrative server. Windows PowerShell is already on your server if you are running a recent version of Windows Server! But, it doesn't have the Active Directory module (named "ActiveDirectory"). To get both tools on your server, you can go to a PowerShell prompt and run the `Install-WindowsFeature -Name RSAT-AD-Tools -IncludeAllSubFeature` command. If you are using a client version of Windows, such as Windows 10, you can download the Remote Server Administration Tools (RSAT) from <https://www.microsoft.com/en-us/download/details.aspx?id=45520>. After you install them, you can enable the tools by going to the Control Panel, Program and Features, and then clicking the Turn Windows features on or off. From there, expand Remote Server Administration Tools and then expand Role Administration Tools. Check the AD DS and AD LDS Tools feature and then click OK. Now, let's look at a few common OU management tasks:

- **Create a new OU.** In Active Directory Users and Computers, right-click an existing OU or the domain name, click **New**, and then click **Organizational Unit**. Type a name and then click **OK**. You now have a new OU! In Windows PowerShell, to create a new OU named OU1, run the following command:

```
New-ADOrganizationalUnit -Name OU1
```

This creates an OU at the top-level.

- **Delete an existing OU.** In Active Directory Users and Computers, right-click an existing OU, click **Delete**, and then click **Yes** to confirm the deletion. This works if the OU isn't protected from accidental deletion and if the OU doesn't contain other objects. If the OU is protected from accidental deletion, you must first remove that protection. To remove the protection, right-click the OU, click **Properties**, click the **Object** tab, and then deselect the **Protect object from accidental deletion** option. We discuss the details of protecting objects from accidental deletion in the next section. To delete an OU named OU1 in the root of a domain named contoso.com by using PowerShell, run this command:

```
Remove-ADOrganizationalUnit -Identity "ou=OU1,dc=contoso,dc=com" -Confirm:$False
```

This command must be run from an elevated PowerShell prompt. You can't just specify the name of the OU that you want to delete because there can be several OUs with the same name. Thus, you must use the distinguished name (DN) of the OU. The DN is the path to the OU. The path to the OU is always unique in a domain and forest.

- **Move an existing OU.** In Active Directory Users and Computers, you can drag and drop an OU to move it. That is handy. But be careful that you don't accidentally drag and drop an OU in the wrong location because it can have serious impacts such as computers getting the wrong policies. Instead, you can move it a different way. You can right-click an OU, click **Move**, then browse to and click a new location, and then click **OK**. To move an OU with PowerShell, you can use the `Move-ADObject` cmdlet. For this example, there are two OUs

named OU1 and OU2. Each is in the root of the contoso.com domain. To move OU1 to be under OU2, run the following command:

```
Move-ADObject -Identity "ou=OU1,dc=contoso,dc=com" -TargetPath  
"ou=OU2,dc=contoso,dc=com"
```

Be aware that OUs protected from accidental deletion must have the protection removed before a move. After a move, you can re-apply the protection.

### **Hands-on Exercise**

Create two new OUs. Then, move one of the OUs to be under the other. Next, delete both OUs.

There aren't many tasks for managing OUs. Mostly, you'll add and remove OUs. But, there is one more setting that applies to users, computers, groups, and OUs. The setting protects the objects from accidental deletion. We talk about that in the next section.

### **Protecting Objects from Accidental Deletion**

Imagine that you are trying to delete an unused user account and you accidentally delete an OU instead. Big trouble. To reduce the chances of a mistake like this, you can protect objects from accidental deletion. You can protect user objects, computer objects, group objects, and OU objects. If any administrator accidentally runs a command to delete a protected object, the deletion will be blocked. You can

To protect an object in Active Directory Users and Computers, by selecting the option on the Object tab. To see the Object tab, you first need to go to Active Directory Users and Computers, click View, and then click Advanced Features.

In Figure 12.3, a group named App-V Core Apps is protected from accidental deletion.

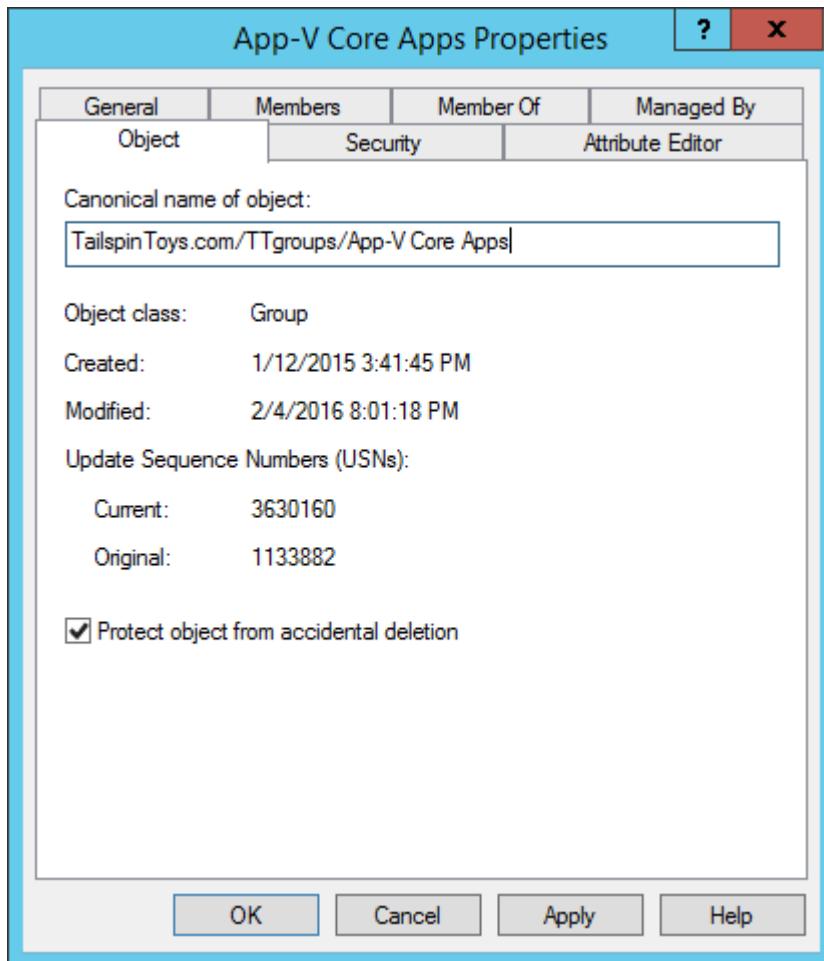


Figure 12.3 An object that is protected from accidental deletion

You can also use PowerShell to protect an object from accidental deletion. To protect an OU named OU1 in the root of the contoso.com domain, run the following command:

```
Set-ADOrganizationalUnit -Identity "ou=OU1,dc=contoso,dc=com" -ProtectedFromAccidentalDeletion $True
```

### Hands-on Exercise

Use Active Directory Users and Computers to create a new OU and protect it from accidental deletion. Next, try to delete the OU. When that fails, remove the protection and then delete the OU.

## Above and Beyond

So how does this protection work? It is interesting because of the simplicity. When an object is protected from accidental deletion, the permissions of the object are adjusted so that the Everyone group (which accounts for literally everybody) is denied access to delete the object. To do this, Windows uses the Delete and Delete subtree permissions and sets them to Deny for the Everyone group.

To remove protection, you can deselect the option in Active Directory Users and Computers. Or, you can run the PowerShell command to protect an object but change the value of the `-ProtectedFromAccidentalDeletion` parameter from `$True` to `$False`.

Besides protecting just one object at a time, you can use PowerShell to protect a large swath of objects at a time. For example, to protect all OUs in the domain from accidental deletion, run this command:

```
Get-ADOrganizationalUnit -Filter * | Set-ADObject -  
ProtectedFromAccidentalDeletion $True
```

You've now been introduced to some of the basic administrative tasks in Active Directory, including remote management, customizing your management environment, working with OUs, and protecting objects from accidental deletion. But this is just the start, and we'll be diving into other areas of AD in the next few chapters. because we've got many more chapters to dive into other areas of Active Directory. But first, let's test your skills in the lab.

## Lab

### Perform remote management

Perform the following tasks:

- Enable remote management on a server.
- From your computer, run Ipconfig on a remote server.

### Customize an MMC

Customize an MMC as follows:

- Add Computer Management targeted to a remote computer.
- Add Event Viewer targeted to the local computer.
- Add Services targeted to a remote computer.

### Work with OUs

Use PowerShell to perform the following tasks:

- Create a new OU named OU98 and another OU named OU99. Ensure that they are not protected from accidental deletion.

- Move OU98 under OU99.
- Delete both OUs.
- Repeat steps 1 through 3 by using Active Directory Users and Computers.

## Work with protecting objects from accidental deletion

Use PowerShell to perform the following tasks:

- Create a new OU named OU100.
- Protect the OU from accidental deletion.
- Try to delete the OU.
- Remove the accidental deletion protection from the OU.
- Delete the OU.
- Repeat steps 1 through 5 by using Active Directory Users and Computers.

## CHAPTER 13: CREATING USER ACCOUNTS

---

We just started the section of the book that covers Active Directory fundamentals. So far, you have learned some of the basics of AD administration such as remote management, customizing your management environment, working with OUs, and protecting objects from accidental deletion.

In this chapter, we are going to focus strictly on user accounts. In fact, working with user accounts is such a big part of an administrator's job that we will spend the next two chapters on it. User accounts are user objects in Active Directory that are most often associated with a user, such as an employee. For example, an employee named Bob might have a user account named Bob and he would use that to sign into his computer. Other times, user accounts are used for backend processes and services. For example, you might have an application such as Microsoft SQL Server that runs as a Windows service (or has multiple services). And that application might require a user account to run the service(s). In such a case, the user account is often referred to as a "service account.". And that the service account's username and password credentials are saved as part of the Windows service configuration.

In this chapter, we'll talk about many of the tasks that you will perform while working with all types of user accounts. These tasks include creating user accounts, modifying user accounts, enabling, and disabling user accounts, and using a user account template to simplify user account creation. In the next chapter, we look at some of the day-to-day management tasks for existing user accounts.

### Creating a new user account

You arrive at the office on Monday morning only to find 3 new employees waiting at your desk. They each have a laptop but none of them can sign in. That's because they don't know their credentials. Even worse, they don't even have a user account in Active Director yet! A routine task for you will be to create new user accounts. This often coincides with a new employee starting work at your organization. The process of preparing your environment for a new employee is sometimes called "onboarding.". Until an employee has a user account, they can't sign-in to their computer, access their email, or perform other routine tasks. Thus, you will usually create a new user account before a new employee starts! In a new environment, creating user accounts is often one of the first tasks you'll have. In an existing environment, it will be a routine task (in fact, in large environments with thousands of users, several user accounts are often created every day). At a high level, there are a few tasks that you need to perform as part of the new user creation process:

- **Gather information about the new employee.** At a minimum, you need their name. But, it is a good idea to store other pertinent information in Active Directory too. For example, you should store phone numbers and address information in Active Directory to make it easy for employees to find each other. In small organizations, you might get all this information in an email from the hiring manager. In large organizations, all this information may come from a Human Resources (HR) software solution.

- **Create the user account.** The second step is to use the information gathered to create a new user account. We will walk through the actual process of creating accounts shortly.
- **Test functionality.** The final high-level step is to test functionality. In small organizations, this testing may occur on the first day the employee starts. In large organizations, you might send an email to the new employee and set up a user profile on their computing device(s).

There are two primary ways to create a new user account. One way is with a graphical user interface application such as Active Directory Users and Computers. The other way is with PowerShell. Of course, like most things in computing, there are a multitude of other ways too, but we won't cover those in this book because we are going to focus on the two primary methods to not bog you down with too much unnecessary information.

### Creating user accounts with Active Directory Users and Computers

The user creation process in Active Directory Users and Computers is not complex. There is a new user creation wizard that walks you through the new user creation process. In the following example, we will create a new user named Evan Smith.

1. Open Active Directory Users and Computers.
2. Create an OU named OU101.
3. Right-click the **OU101** OU, click **New**, and then click **User**.
4. In the **New Object – User** window, type Evan for the first name and Smith for the last name. Then, type esmith for the logon name. Note that the logon name is the same as the sAMAccountName. In everyday conversation, it is often referred to as the "logon name" but the actual name of the attribute that stores the value is the sAMAccountName. Click **Next**.
5. Type a password and then type it again to confirm the password. Note that by default, you must type a password that is 7 characters or longer and contain characters from at least 3 of the following 4 categories: uppercase characters, lowercase characters, numbers (0-9), and non-alphabetic characters (symbols such as the exclamation point). Your environment may have a different set of requirements too. Then, you have 4 options, shown below in Figure 13.1.

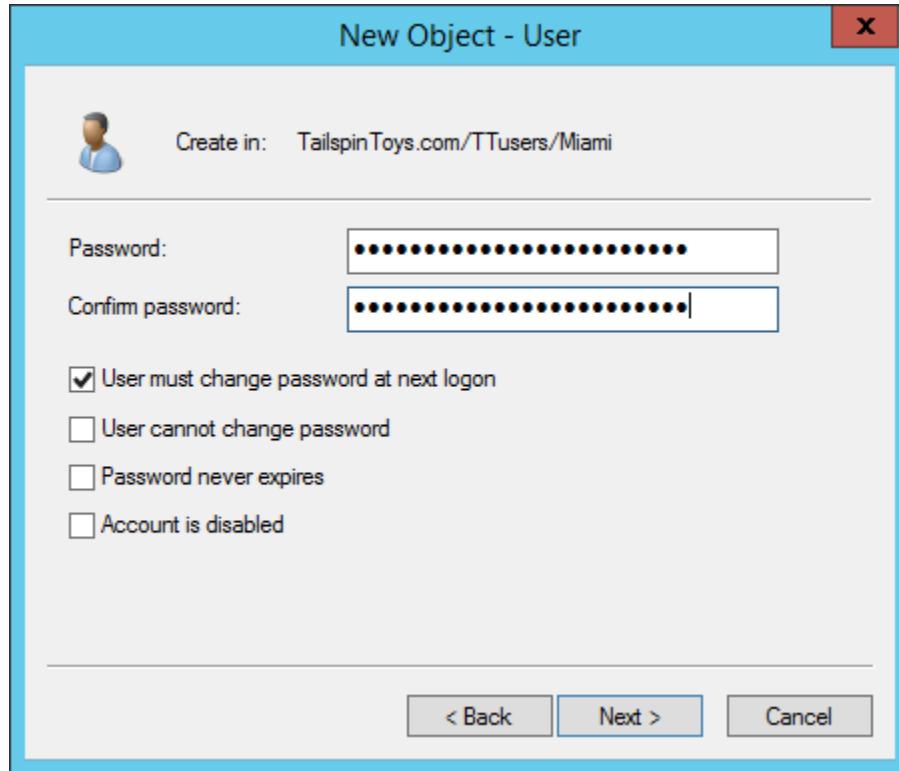


Figure 13.1 Account options when creating a new user account

For this example, we will only require a password change at next logon. Click **Next**. But, familiarize yourself with the other password and account options. You will use these regularly in your job.

- **User must change password at next logon.** This option, if enabled, requires a user to change their password at the time of their next logon. It is a good idea to use this option for your users. If you don't, they may use the temporary password that you gave them indefinitely, which might not be secure. Why? Because the administrator knows the password. And, the temporary password might be reused for other user accounts or password resets. Or, the password might be weak.
- **User cannot change password.** This option, rarely used, is sometimes useful for a shared user account or a service account that will be managed by an administrator. When this option is enabled, the user cannot change their password. However, an administrator can.
- **Password never expires.** This option, when enabled, configures a user so that a password change is never required. A user can still change their password, but it is optional. This setting reduces security because a malicious person can try to crack a password indefinitely. Imagine a scenario where a password takes 100 days to crack, using specialized hardware and software. If your password policy requires a password change every 90 days, by the time a password is cracked, it is already changed so the cracked password is worthless. But, passwords that don't expire can be helpful for service accounts because changing service account passwords can be disruptive. With service accounts, if you opt to not use password

expiration, you should use very strong passwords, such as password with 20 or more characters and complexity.

- **Account is disabled.** This option, when enabled, disables the new user upon creation. A disabled account cannot be used. This option is handy if you want to create some accounts early, long before a new user will need it. Instead of having dormant accounts active, you can disable the accounts until they are needed. We look at enabling and disabling accounts later in this chapter, in section 2.3. Dormant accounts are often targets of malicious hackers so minimizing the amount of dormant accounts you have helps minimize your exposure.

6. Review the new user account summary information. If it is accurate, click **Finish** to create the new user account.

### **Hands-on Exercise**

Open Active Directory Users and Computers. Create a new account named Evan Smith.

### **Creating user accounts with PowerShell**

If you are creating a new user and plan to populate several user properties (such as title, address, and phone number), most administrators would prefer to use Active Directory Users and Computers. But if you are planning to create many user objects, most administrators prefer to use PowerShell to save time. Constructing the right PowerShell command might take you a few tries, especially if you aren't using it often. For creating new users, you will use the New-ADUser cmdlet. Let's look at a couple of examples of how to create new users with PowerShell.

```
New-ADUser -Name "Evan Smith"
```

The above command creates a new user named Evan Smith. Interesting, right? Because we didn't specify a password, a logon name, or any other information beyond "Evan Smith". Because we didn't specify the password, the user account is disabled.

```
New-ADUser -Name "Evan Smith" -AccountPassword (ConvertTo-SecureString  
"Pa$$w0rd" -AsPlainText -Force) -Enabled $True -GivenName "Evan" -Surname  
"Smith"
```

This command creates a new user named Evan Smith. It also sets the first name, last name, and the password. The account is also enabled. There are many other account attributes you can set. But, the command quickly gets complex. For creating a single user, some administrators might not find PowerShell to be the best choice. However, when creating a lot of new users, you can use PowerShell to create new users based on information in a .txt file or Excel spreadsheet. That can be a time saver.

Let's look at creating multiple users too. For this next example, we'll create two new users but the method applies even if you need to create hundreds or thousands of users. Our users will

have a logon name, a first name, a last name, a display name, a name, a description, and a password. They will also be have enabled user accounts.

First, create a .csv file with column headers at the top and account information on each line, as shown below in Figure 13.2.

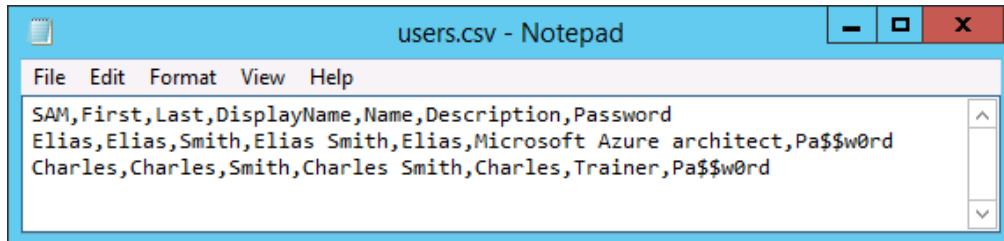


Figure 13.2 A .csv file to use for creating new users with PowerShell

Next, run the following command:

```
Import-Csv .\users.csv | foreach {New-ADUser -SamAccountName $_.SAM -  
GivenName $_.First -Surname $_.Last -DisplayName $_.DisplayName -Name  
$_.Name -Description $_.Description -AccountPassword (ConvertTo-  
SecureString $_.Password -AsPlainText -Force) -Enabled $True}
```

Don't worry about understanding the complex command. For now, the most important thing is to see what it takes. As you get more comfortable with PowerShell, commands like this one will be easy to construct.

### **Hands-on Exercise**

Use PowerShell to create a new user named Elias Smith. While constructing your command, use PowerShell's tab completion to explore some of the New-ADUser parameters:

```
Get-Help New-ADUser -Examples
```

You can also run the following command to see several example commands and helpful information about the command's parameters.

You now know how to create new users. And you know how to do it a couple of different ways. You even know how to create many new users in a single command. But creating new users is just the start of working with user accounts! Next, let's look at modifying user accounts, another routine administrative task.

### **Modifying the properties of an account**

There are a lot of reasons that you will need to modify user accounts. Some common reasons include name changes, marriages, title changes, department changes, phone number updates, and specifying a new manager. Whatever the reason, you should be comfortable modifying user accounts. We'll look at a few different ways to modify user accounts: by using the existing

Active Directory Users and Computers fields, by modifying AD DS attributes, and by using PowerShell.

### Updating user accounts with Active Directory Users and Computers

The most intuitive way to update a user account is to use Active Directory Users and Computers. You search for the account (right-click the domain object, click Find) that you want to update, get the properties for it, and then update information as needed. Figure 13.3 shows the General tab of a user account.

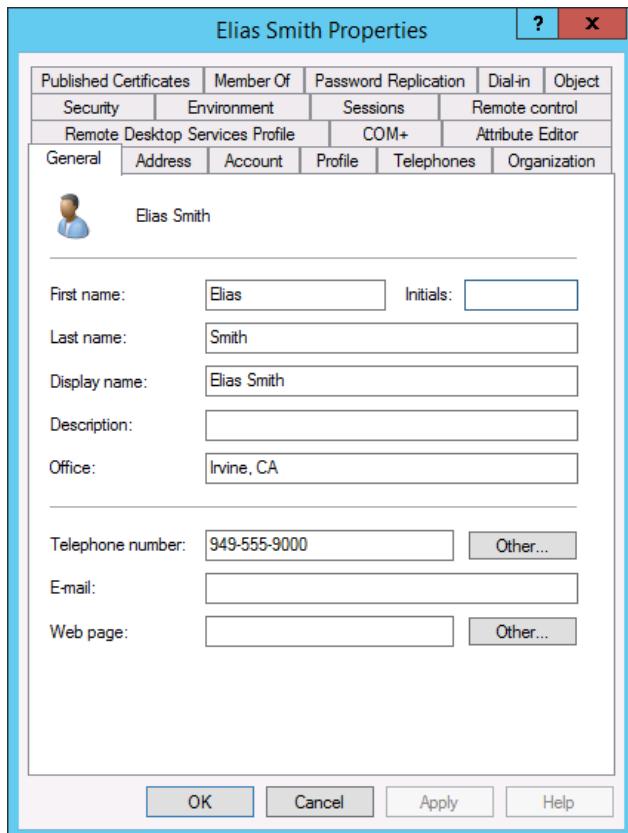


Figure 13.3 The properties of a user account show the details such as the name, address, and phone number.

As you can see in Figure 13.3, you can just highlight the information that you want to change, change it, then click OK and you are finished! As you probably noticed, there is potentially a lot of data for each user, some of which we cover in the next section.

Besides just modifying a single user at a time, Active Directory Users and Computers enables you to update multiple user accounts at the same time. But, such updates are limited to specific attributes such as the office, description, address info, organization info, and account options (such as the option to force a password change at next logon). But, this the bulk modifications feature is handy. You just need to use CTRL and left-click to multi-select user accounts, then get the properties and make the change once to update all of them.

## Hands-on Exercise

Open Active Directory Users and Computers. Select two user accounts and then get the properties for the accounts. Force both accounts to reset their password at next logon. Scroll through the other settings to get familiar with them.

Next, let's have a look at updating some of the lesser known and used user account attributes. Some are not directly exposed in all the management tools.

### Modifying AD DS attributes for user accounts

Beyond just updating the primary attributes of a user account, such as their phone number and manager, there are a myriad of other settings, many of which are not directly exposed on the user property tabs. Most of the attributes are optional – use them if you want, don't use them if you don't want to. On the Attribute Editor tab, you can look at all the user account attributes and modify them as needed. For example, you can set a value for the employeeID attribute. The attribute itself isn't important here. Your organization might or might not use employee IDs (or store them in AD DS). The key point is that AD DS can store lots of different data and you should be familiar with the types of data. In Figure 13.4, the employeeID attribute is set to 5309 for Elias Smith.

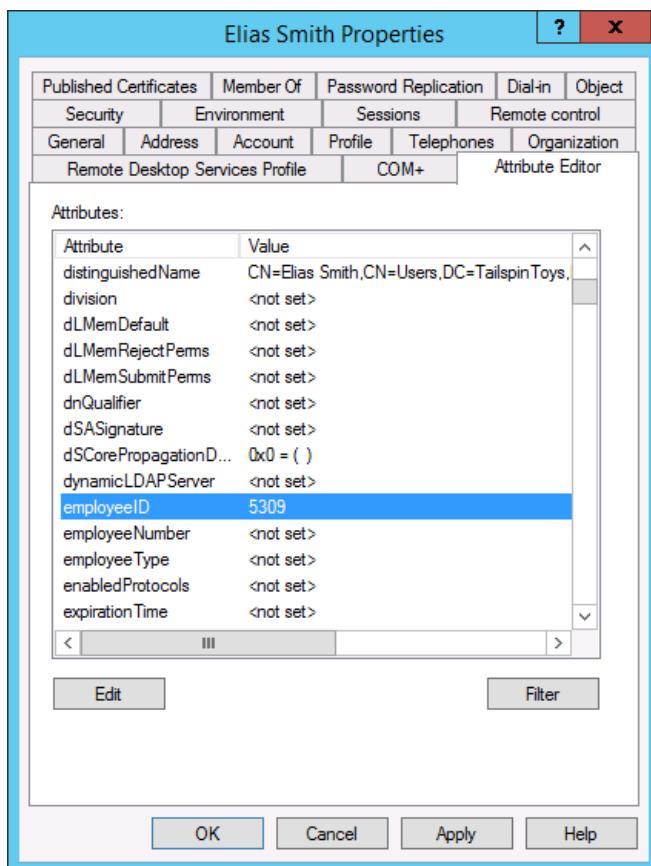


Figure 13.4 The Attribute Editor tab enables you to modify user account attributes

By default, the Attribute Editor tab, along with some other tabs, is not displayed when you view the properties of a user. To view all the tabs, you need to set the view so that advanced features are displayed. To do that, in Active Directory Users and Computers, click View and then click Advanced Features.

### **Hands-on Exercise**

Open Active Directory Users and Computers. Then, make sure you are viewing the advanced features. Find a user account and update the employeeID of that user account.

User accounts have a lot of attributes. The Active Directory schema defines the classes, objects, and attributes in AD DS. When you extend the schema for Microsoft Exchange and many other applications, more attributes are added to the schema (and thus, available for use in AD DS). Next time you are trying to figure out a place to store some identity data for users, look to the Attribute Editor tab and see if you can find an attribute.

Next, let's go outside of the graphical tools and see how you can modify user accounts with PowerShell.

## **Modifying user accounts with PowerShell**

When you are updating a single user account, the graphical tools are often a good choice because they are easy and intuitive. But, there are two things that drive administrators to PowerShell. One is becoming proficient and comfortable with PowerShell. When you are proficient and comfortable, PowerShell can often be quicker and easier than the graphical tools. When I first learned PowerShell, I performed all my management tasks with it even if it was more inefficient for a given task. My goal was to learn it and sometimes the best way to do that is to jump right in! But the other thing that drives administrators to PowerShell is when actions are required across many objects. For example, if you needed to find all users in the sales department and update their department name, that could be time-consuming with the graphical tools. But in PowerShell, it is a single command.

The primary PowerShell cmdlet for updating users is Set-ADUser. But you can also use Set-ADObject, which can work with all types of AD DS objects. Let's look at some examples of modifying user accounts with these commands:

```
Set-ADUser -Identity Elias -Description "Consultant"
```

The above command sets the description attribute to "Consultant". If the attribute already had data, that data overwritten part of this command. You might run this command when you need to add identifying information to user accounts.

```
Get-ADUser -SearchBase "OU=Chicago,OU=TTUsers,DC=tailspintoy,DC=com" -  
Filter "Department -eq 'Sales'" | Set-ADUser -Office "Downtown Chicago"
```

This command gets all the users in the Chicago OU that has a department name of "Sales" and sets their Office attribute to "Downtown Chicago". You might run such a command when you open a new office or move employees to a different office.

```
Get-ADUser -Filter * -SearchBase  
"OU=Chicago,OU=TTUsers,DC=tailspintoys,DC=com" | Set-ADObject -  
ProtectedFromAccidentalDeletion $True
```

This command gets all the users in the Chicago OU and protects the user accounts from accidental deletion. We use Set-ADObject to protect the accounts from accidental deletion because the Set-ADUser cmdlet doesn't offer a way to do this.

One of the best ways to become comfortable with PowerShell is to force yourself to use it. You can decide to use only PowerShell for all administrative tasks for a week, even if some of the tasks take a little extra time as you are learning the cmdlets and parameters. If you do this, you may find that you rarely go back to the graphical tools.

Now that we've looked at modifying users accounts with PowerShell, let's look at enabling and disabling accounts, which is another routine administrative task.

### **Hands-on Exercise**

Use PowerShell to modify all user accounts in the Users container so that the description for each user object is "Move this account to an OU".

## **Enabling and disabling an account**

For administrators, there are a couple of instances where they have a bunch of tasks to perform. I'm talking about when new employees start at a company and when existing employees leave a company. When new employees start, an administrator needs to create accounts, prepare computers, adjust group memberships, and update user accounts with information. When existing employees leave, an administrator needs to remove access and disable user accounts. The enabling and disabling are critical tasks. Without them being handled quickly, employees can't work or the organization is at risk for malicious use. First, let's look at enabling accounts.

### **Enabling user accounts**

In some companies, new user accounts are created as enabled accounts. In such cases, an administrator won't spend much time enable enabling many accounts, except when accounts are temporarily disabled (since accounts are enabled at creation). But in other companies, new user accounts are created far in advance and not enabled until needed. This can help make for a smooth first day for a new employee because their computer can be set up in advance, their email can be set up in advance, and, they can even be granted access to key resources in advance. When organizations create users in advance, administrators will spend time enable enabling quite a few accounts (since those new accounts created in advance are disabled at

creation). You can enable accounts in Active Directory Users and Computers, Active Directory Administrative Center, or PowerShell. While we've shown you Active Directory Users and Computers so far, Active Directory Administrative Center is another graphical tool for managing AD DS. We'll try to show examples from both, sometimes alternating, so you can get some experience with each. For this example, we'll look at Active Directory Administrative Center (although Active Directory Users and Computers is nearly identical for this task).

One thing that makes ADAC interesting is that there is a PowerShell history section at the bottom of the console. You can click it to expand it. Then, you can click the option to show all the PowerShell. The window will then display all the PowerShell commands that are being run in the background while you perform administrative tasks using the GUI. In Figure 13.5, Elias Smith's user account is disabled, indicated by the small down-arrow icon.

Name	Type	Description
DnsAdmins	Group	DNS Administrators Group
DnsUpdateProxy	Group	DNS clients who are permitted to perform d...
Domain Admins	Group	Designated administrators of the domain
Domain Computers	Group	All workstations and servers joined to the do...
Domain Controllers	Group	All domain controllers in the domain
Domain Guests	Group	All domain guests
Domain Users	Group	All domain users
Elias Smith	User	Consultant
Enterprise Admins	Group	Designated administrators of the enterprise

Figure 13.5 Active Directory Administrative Center shows a disabled user account (Elias Smith) which has an icon with a downward arrow.

To enable Elias' account in ADAC, right-click the account and click **Enable**.

Name	Type	Description
DnsAdmins	Group	DNS Administra...
DnsUpdateProxy	Group	DNS clients who...
Domain Admins	Group	Designated adm...
Domain Computers	Group	All workstations...
Domain Controllers	Group	All domain cont...
Domain Guests	Group	All domain gues...
Domain Users	Group	All domain user...
Elias Smith	User	Consultant
Enterprise Admins	Group	Designated administr...
Enterprise Read-or...	Group	
Exchange Online-A...	Group	
FederatedEmail.4c...	Group	

Figure 13.6 Enabling a disabled user account is as simple as right-clicking an account (in this case Elias Smith) and then clicking Enable.

As you can see, enabling accounts is straight forward in the graphical tool. Elias Smith's logon name is Elias. To enable the account with PowerShell, run the following command:

```
Enable-ADAccount -Identity Elias
```

### Hands-on Exercise

Use PowerShell to enable a user account. If you don't have any disabled accounts, skip to the next part of this topic to learn how to disable accounts. Then, loop back to this exercise.

That's all there is to enable accounts! Let's look at disabling accounts now.

### Disabling user accounts

You disable accounts when you need to immediately disable the ability for a user to access resources. You would do this in situations such as when an employee leaves the company or when a user account has been flagged as performing malicious actions. For this example, we'll look at Active Directory Users and Computers (although performing the task in Active Directory Administrative Center is almost identical). To disable an account, navigate to the location where the account is stored, right-click the account, and then click Disable Account. A dialog box will be displayed indicating whether the object has been disabled. Figure 13.7 shows the menu.

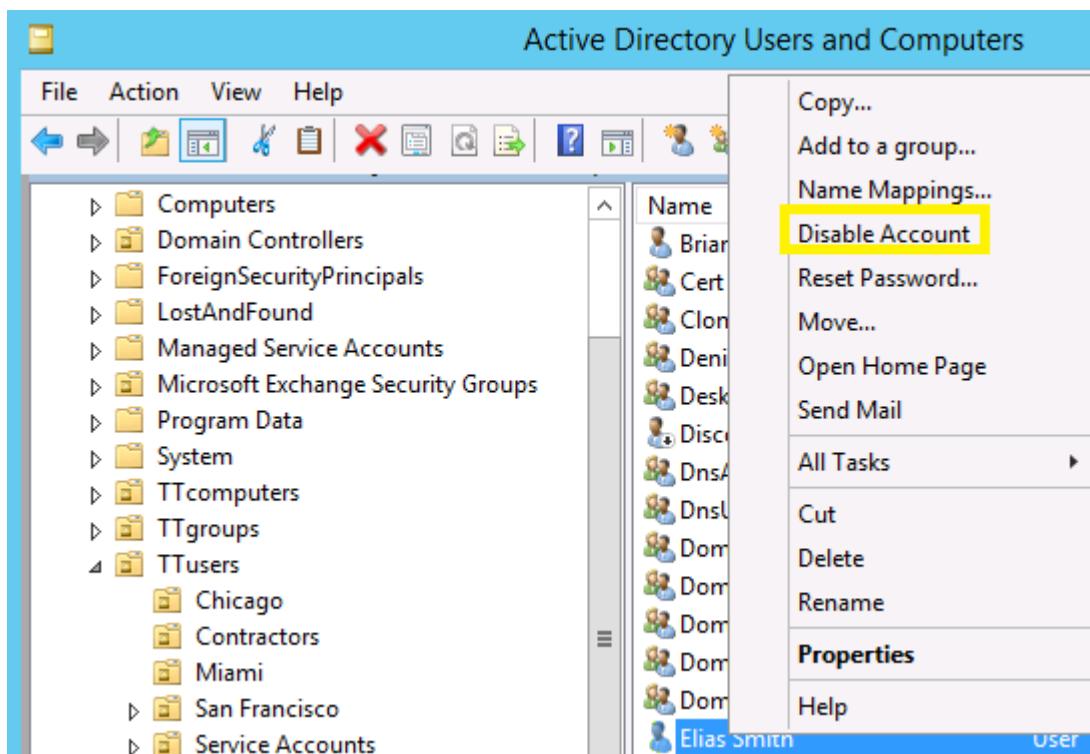


Figure 13.7 Disabling the Elias Smith user account in Active Directory Users and Computers by right-clicking the account and then clicking Disable Account.

You can also disable user accounts in PowerShell. To disable Elias Smith, you can run the following command:

```
Disable-ADAccount -Identity Elias
```

As you can see, disabling user accounts is also very straight forward. It is a good practice to move disabled user accounts to a new OU to make them easily identifiable, easy to report on, and delegate the access to delete the accounts (without delegating access to anything else). Many organizations choose to have an OU to temporarily store disabled user accounts. Then, after a set amount of time has passed, such as 30 days, the accounts can be permanently deleted.

### **Hands-on Exercise**

Use PowerShell to disable a user account.

So far, we've looked at creating and modifying user accounts. For our final section of this chapter, we are going to look at user account templates. Templates are handy because they enable you to create accounts that have some attributes already populated!

### **Creating a template for user creation**

Imagine that you needed to create 5 new user accounts for some new sales employees. For each new account, you must type all the information that you want to populate (such as names, addresses, offices, departments, and other information). It is time-consuming. A template helps reduce the administrative overhead. A template has some account information pre-populated on a user account that is used as a template. The account is disabled. When it comes time to create a new user, you can just copy the template account, type the name and password, and you are ready to go. The copy process does not copy all the user attributes from the template to the new user account. By default, only a subset of the most commonly used attributes are copied over. (You can change which attributes are copied over by modifying the schema, although we won't discuss that process in this book.)

In Figure 13.8, an existing user account named \_TEMPLATE is used as a template. You can create a template by creating a regular user account, populating the needed attributes, and then disabling the account. Using an underscore as the first character of a template account is helpful because the user account shows up first in the list of users. Once you have a template account, you can create an account from it. To create a new user from the template, right-click the template user and then click Copy. From there, you complete the new user wizard with the name, logon name, and password.

### Copying a user template.

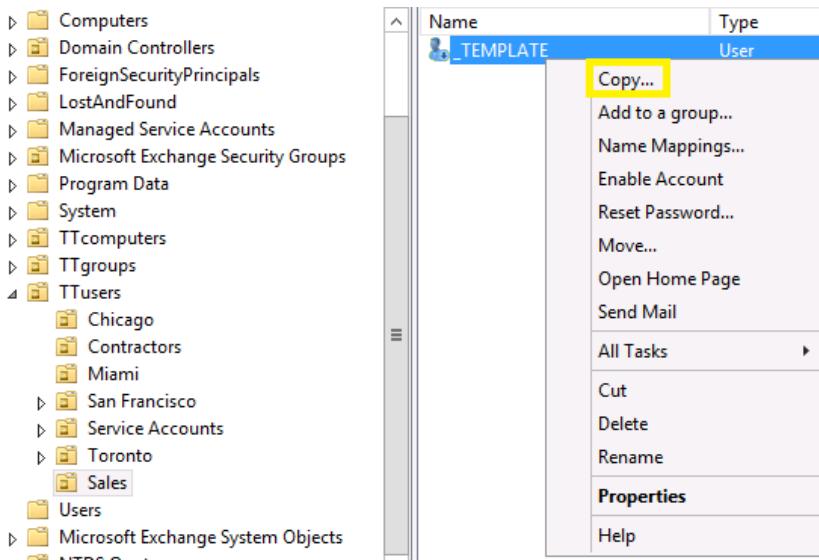


Figure 13.8 Copying a user account from a template is as simple as right-clicking the account (in this case "\_TEMPLATE" and then clicking Copy).

Some of the account information that is copied over is the department, the company, the address information, and account settings such as logon hours and account expiration.

You can use PowerShell to create a new user from a template too. To create a new user named Michael from a \_TEMPLATE account that is stored in the TTUsers/Sales OU, run the following command:

```
New-ADUser -Name "Michael" -SamAccountName "Michael" -AccountPassword
(ConvertTo-SecureString "Pa$$w0rd" -AsPlainText -Force) -Enabled $True -
Instance "CN=_TEMPLATE,OU=Sales,OU=TTUsers,DC=tailspintoys,DC=com"
```

The –Instance parameter points to the distinguished name (DN) of the template. The DN is the path to the object in AD DS.

### Hands-on Exercise

Create a new user account named \_TEMPLATE. Disable the account. Populate several attributes, such as the address information and country. Then, use PowerShell to create a new user account based on the template.

In this chapter, we looked at creating and modifying user accounts with a focus on some very common administrative tasks. I bet you are feeling comfortable with this information by now. But, we are just getting started! In the next chapter, we continue talking about user accounts. But we'll focus on managing passwords, account lockout, stale accounts, and account deletion. First though, we are going to test this chapter's information in a lab.

## Lab

### Create a new user account

Perform the following tasks:

- Use PowerShell to create a new user account. Use "Test" as the first name, "User" as the last name, and "Test User" as the name. Use "testuser" as the logon name. Store the user account in the Users container.

### Modify existing users

Perform the following tasks. Note that you need two user accounts to proceed. If you don't have at least two user accounts, create two test accounts before proceeding.

5. Use Active Directory Users and Computer to force more than one account to change their password at next logon. Do this in a single action in the tool.

### Enable and disable users

Use PowerShell to perform the following tasks. To proceed, create a test OU and create two test accounts in the OU.

6. Disable all user accounts in your test OU with a single command.
7. Enable all user accounts in your test OU with a single command.

### Work with user account templates

Use Active Directory Administrative Center to perform the following actions:

- Create a new user account template.
- Populate at least 20 attributes for the account, including account options.
- Create a new AD user based on the account template.
- Look at the new AD user and see which attributes were copied over from the template.

## CHAPTER 14: MANAGING USER ACCOUNTS

---

In the last chapter, you learned how to create new user accounts, including how to modify some of the user account properties, enable accounts, disable accounts, and create user account templates to ease the administrative burden of creating new user accounts.

In this chapter, we continue our focus on user accounts. That's because working with user accounts is something that you'll do just about every day so you want to know the ins and outs. We'll start by looking at resetting passwords. This is a task that is fun for the first few times you do it. But after that, not so much. We'll see if PowerShell can at least make it more interesting for you. Then, we'll look at unlocking user accounts. Often, user accounts will get locked out after too many bad password attempts. You'll want to know how to remedy that situation quickly.

Next, we'll deal with stale user accounts (accounts that have not been used in 60 days or 90 days). You should clean up stale accounts because stale accounts because they can be targets of hackers. Why? Because, if a stale account is hacked, who would notice? Usually nobody notices, since nobody is using it. Finally, we'll learn about deleting user accounts. Often, deleting these accounts means that users are leaving a company. That's no fun. But it is an important part of the job because you need to ensure that departing users don't have continuing access to the company network and data.

### Resetting Passwords

Elias, the sales guy, just got into the office. He has a meeting to try to close a big sale. But, he can't sign in to his computer! The meeting starts in 5 minutes and he needs to gain access to the meeting information. What's the fix? Usually, it is as simple as a password reset. And luckily, password resets are quick and painless. But, there are some key details about password changes that you should know about so you don't accidentally expose the company to undue risk. Let me show you what I mean by walking through a quick scenario:

The new administrator started a couple of weeks ago. He has been buried in work and struggling to keep up with all the new trouble tickets coming in. He looks for ways to be more efficient. One way that he figured out is by using Changeme! as the password for all password resets. It makes things easier. He no longer needs to remember unique passwords for every password reset. To make it even more efficient, the new administrator sets the account with the "Password never expires" option. Now, he rarely hears from users for password resets.

So, what's wrong with this situation? A few things:

- **Users will eventually realize that the new administrator always uses Changeme! for password resets.** Thus, whenever somebody overhears another employee having trouble signing into their computer, they'll know that they are soon going to have a password of Changeme!. This could lead to malicious activity.
- **The password Changeme! isn't very secure.** It is easily cracked. And can even be guessed because it routinely shows up as a commonly used password in publicly available password databases. By setting user accounts to have passwords that don't expire, many users will opt

to keep the Changeme! password forever, leaving those accounts susceptible to password compromises.

- **Users do not have to change their password at next logon after the new administrator resets their password.** That means they can use a password that the administrator (and possibly other users) know. Instead, it is a good practice to force a password reset on the next logon. That way, the user is the only person that knows their password. In a security incident, you want to be able to tie account usage back to an individual, when possible. And forcing users to have a unique password that only they know is one step in achieving that goal.

You should now have a feel for why password resets need to be taken seriously, even if the actual task is quick and easy. Now, let's look at performing password resets.

The steps to reset a password in Active Directory Users and Computers are:

1. Navigate to the location of the user object or search for the user object.
2. Right-click the user object and then click **Reset Password**.
3. In the Reset Password window, type the new password in the **New password** text box and in the **Confirm password** text box.
4. Keep the default option to force the password change at next logon enabled.
5. Click **OK** to complete the password reset.

You can also reset the password in PowerShell. For example, to reset Elias' password to **What is 2x 8euriERE**, run the following command from a PowerShell prompt:

```
Set-ADAccountPassword -Identity Elias -Reset -NewPassword (ConvertTo-SecureString -AsPlainText "What is 2x 8euriERE" -Force)
```

As you can see, resetting password is quick and easy!

### Above and Beyond

For high-security accounts, such as users with access to sensitive or confidential data, you should use 15 characters (or longer) passwords. Pass phrases, such as a complete sentence with spaces are good to use because they are easy to remember, fast to type, and often long. And a 15 character or longer password will not have a valid password hash stored. And password hashes, which are stored in the AD DS database, are what malicious users try to crack to gain access to your plain text password. Thus, password cracking becomes drastically more difficult with 15 character or longer passwords.

In most environments, you will have a mix of standard user accounts (accounts that do not have access to sensitive or privileged data) and high-security user accounts (accounts that have access to sensitive or privileged data or have administrative access to backend computers). Using a single password policy is difficult. Standard users will be taken aback when presented with 15 character passwords. By using a feature known as fine-grained password policies, you can have multiple password policies which have different requirements and are targeted at

different groups of users. For example, you could have one policy for standard user accounts with an 8-character password length minimum. Then, for high-security user accounts, you could have another policy that dictates that the minimum password length is 15 characters. We don't cover fine-grained password policies in this book but look at [https://technet.microsoft.com/en-us/library/cc770394\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc770394(v=ws.10).aspx) for more information.

### Hands-on Exercise

Open Active Directory Users and Computers and reset a user's password. Then, use PowerShell to reset another user's password. Compare the methods to see which you like better for password resets.

OK, enough about passwords! Time to talk about unlocking user accounts. Unlocking user accounts is closely associated with password resets. That's because you often must reset a password and unlock an account at the same time. This happens when an account lockout policy is configured to lock out accounts after a specified number of bad password attempts. By default, an account will be locked out if 5 invalid logons are attempted within 30 minutes.

## Unlocking User Accounts

User accounts have two possible states regarding locking. They are either unlocked and operating normally. Or they are locked out and thus cannot be used to sign in. But when does this happen? This happens when there is a Group Policy Object (GPO) that sets what's called an "account lockout threshold". An account lockout threshold is a GPO setting that dictates how many invalid password attempts are allowed in a specified time frame before a user account becomes locked out (and thus unable to sign in). There are 3 account lockout GPO settings, as shown in Figure 14.1 below.

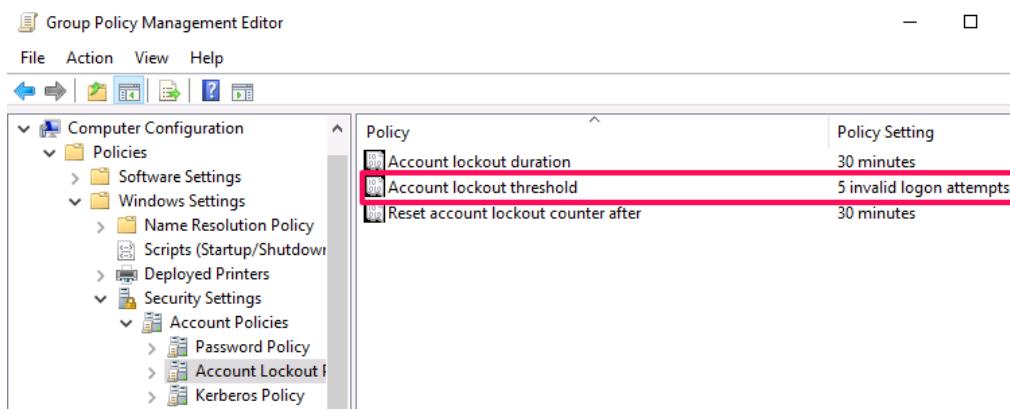


Figure 14.1 The three account lockout GPO settings

Let's talk about what the 3 GPO settings do:

- **Account lockout duration.** This setting dictates how long an account remains locked out. After the elapsed time, the account automatically unlocks. However, if the value of the

setting is 0 then an account stayed locked out until an administrator unlocks it. In high-security environments, it is a good practice to set the value of the setting to 0. It limits password guessing attempts and it allows the IT administrator responsible for unlocking accounts to talk to the account holder and verify their identity.

- **Account lockout threshold.** This setting dictates how many invalid password attempts are allowed before an account gets locked out. If you set this to a low value, such as 3, users are likely to get locked out more often. If you set the value to 0, accounts never get locked out. There are varying opinions on this setting. Personally, I prefer to set this to 999 (the highest number allowed) to reduce and/or eliminate account lockouts in all but the most extreme conditions (which typically involve some automated process or malicious activity). A commonly used value is 5, which is usually enough chances for a user to get their password entered correctly (if they know it).
- **Reset account lockout counter after.** AD DS maintains a lockout counter which tracks invalid password attempts. If you enter 2 bad passwords in a row, the counter is set to 2. If this setting is configured, then the counter resets when the elapsed time is reached. If the setting is set to 30 minutes, then the counter resets to 0 at 30 minutes.

Now you have a good idea of how account lockout works and understand which settings are available. Let's look at the process of finding locked accounts and unlocking them.

### Finding users that are locked out

Imagine that you come back to the office from lunch and are bombarded with account unlock requests. Immediately, you might wonder if there was malicious activity on your network causing all the accounts to become locked out. To prepare for that situation or similar situations, you need to know how to find all the locked-out users. To find all the currently locked out users, run this command:

```
Search-ADAccount -LockedOut
```

### Hands-on Exercise

Open an elevated PowerShell prompt and search for locked out users. Not finding any? That's OK, you can lock one or more out. Before proceeding, make sure you are working in a non-production environment with test user accounts. If you have a test user named Test5 in a domain named contoso.com, you can lock the account out by performing the following steps. Go to a command prompt. Run the following command:

```
net use X: \\contoso.com /USER:contoso\test5
```

When prompted for the password, type random keys and then press the **Enter** key. Repeat steps 2 and 3 until the command notifies you that the account is locked out. Run the following command again to view the locked-out accounts:

```
Search-ADAccount -LockedOut
```

## Unlocking user accounts with ADUC and ADAC

Using the graphical user interface tools such as Active Directory Users and Computers (ADUC) and Active Directory Administrative Center (ADAC) to unlock accounts is simple. For example, in ADUC, navigate to the account, get the properties, go to the Account tab, and click the checkbox to unlock the account. Figure 14.2 shows the checkbox.

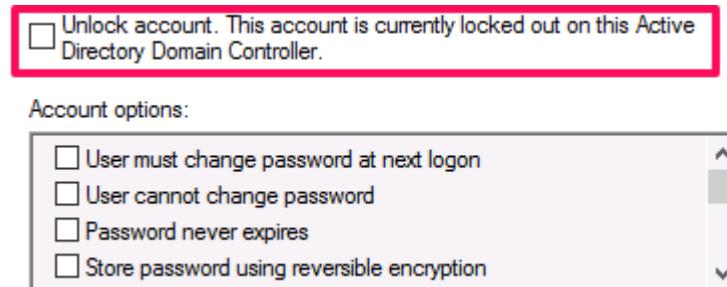


Figure 14.2 The process to unlock a user account in ADUC

In ADAC, the process is even simpler. Navigate to the account, right-click it, and then click **Unlock**. Figure 14.3 shows the process.

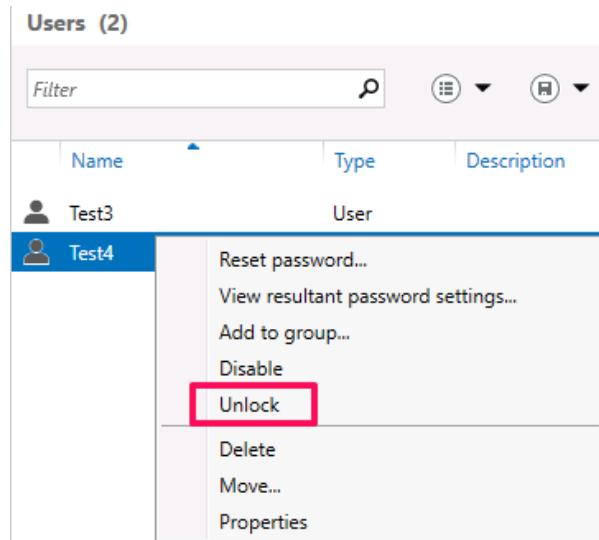


Figure 14.3 The process to unlock a user account in ADAC

## Unlocking a user account by using PowerShell

You can also quickly unlock a user account by using PowerShell. For example, to unlock a user named Charles, run the following command:

```
Unlock-ADAccount -Identity Charles
```

The `-Identity` parameter accepts the DN of the user, the GUID, or the `sAMAccountName`. If you ever run into a mass case of account lockouts, you can also use the pipeline to quickly fix that.

To unlock all accounts in the domain at once, run the following command:

```
Search-ADAccount -LockedOut | Unlock-ADAccount -PassThru
```

You probably don't want to do this in a production environment. It is better to look at which accounts are locked and how many are locked before you blindly unlock them. That way, you can discover if you have a serious issue on your hands.

### **Warning**

Make sure that you verify a person's identity before you unlock their account.

Failing to do so could lead to account misuse or worse.

### **Hands-on Exercise**

Open an elevated PowerShell prompt and unlock a locked-out user. If you don't have a locked-out user, refer to the previous Hands-on Exercise for steps on how to lock out a user.

That covers account lockouts and unlocks. Straight forward stuff. We are going to switch things up a little bit by jumping to stale user accounts and how to work with them.

## **Managing Stale Accounts**

A stale user account is a user account that hasn't been used in a long time. A long time usually means at least 30 days. Sometimes 60 days. But certainly at 90 days, most administrators consider a user account stale. Stale user accounts are sometimes user accounts that were not removed when a person left the organization or a service account that is no longer being used. Stale accounts, especially when they remain enabled, are prime targets for malicious users because nobody usually notices if a stale account is being used. As an administrator, you should have a process to remediate stale accounts on a regular basis, such as quarterly. In this section, I'll show you two ways to find stale accounts and give you some ideas on how to deal with them once you find them.

### **Finding stale user accounts in ADUC**

You can use ADUC to find stale user accounts. However, compared to PowerShell, the options are a bit limited. ADUC enables you to search for accounts that haven't signed in for 30, 60, 90, 120, or 180 days. To find stale accounts, perform the following steps:

1. In ADUC, right-click the domain and then click **Find**.
2. In the **Find Users, Contacts, and Groups** window, click the **Find** dropdown menu and then click **Common Queries**.
3. At the bottom of the window, click the **Days since last logon** dropdown menu and click the desired number of days since last logon.

#### 4. Click **Find Now**.

Figure 14.4 shows the process of searching the tailspintoys.com for user accounts that have not logged in for 30 days. 8 user accounts were found.

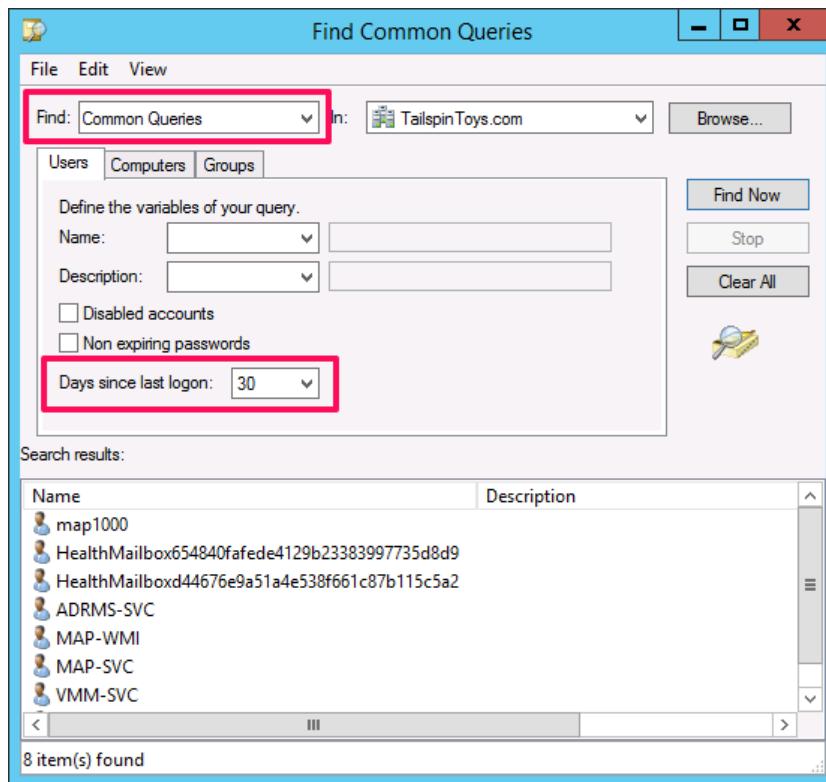


Figure 14.4 Finding stale user accounts in ADUC requires you to use the Common Queries feature in the Find dialog box. Then, you can specify the number of days since the last logon.

### Hands-on Exercise

Open ADUC and search for users that haven't signed in for 30 days. Then try 90 days and 180 days. Don't worry if you don't find any, especially in a test environment.

While it is quick and easy to find stale user accounts in ADUC, the interface limits the filtering that you can do. Let's look at some of the filtering that you can do in PowerShell so you can see how it is more flexible and powerful.

### Finding stale user accounts by using PowerShell

Now that you know how to find stale users in ADUC, let's look how to find stale users in PowerShell. In the following example, the command finds all users that haven't signed in for at least 30 days.

```
Search-ADAccount -AccountInactive -TimeSpan 30 | where LastLogonDate -ne $NULL | Select-Object Name,LastLogonDate
```

There are a couple of differences between what we just did in PowerShell and what we did a few minutes ago in ADUC:

- **The PowerShell option shows the actual date of last logon.** This is good information to have, especially if you are generating a report. With ADUC, you do not get the last logon date.
- **The PowerShell command only found user accounts that have logged in at least once.** We did that by using LastLogonDate -ne \$NULL. That's also handy. If you have user accounts that haven't been used yet, you don't want them coming up in your stale user search.

But, you can also use PowerShell to do a few more things in a search for stale users:

- **You can specify an exact number of days.** For example, if you wanted to find users that hadn't signed in in 25 days, you could do that.
- **You can output the stale users to a file for later viewing.** For example, if you found 25 stale users and you wanted to go through them later to see which can be deleted, you could output to a .csv file and go through them later. You can pipe the output of our example command to the Export-Csv output.csv -NoTypeInformation command to save the output in a file named output.csv.
- **You can schedule the stale user search so that it runs as a scheduled task.** You could opt to run it monthly and save the output file to a shared folder. Or, you could even opt to have the PowerShell command email the results to you.

### Hands-on Exercise

Open PowerShell. Find stale users by using a command like the example shown.

Start by searching for users that haven't signed in for 30 days. If you don't find any, drop the number down to 10, 5, and even 1 until you can find some. Then, re-run the command and export the results to a .csv file.

As you can see, the PowerShell method is quite an improvement over the ADUC method. Now, you know how to find stale users. In multiple ways. But, what do you do with them after you find them? Let's find out.

### What to do with stale users after you find them

So, you've found some stale users. And some of them haven't signed in for 6 months! What now? The short answer is to delete them! But there are a couple of key considerations:

- **It is a good practice to disable user accounts before you delete them.** Whether you are preparing to delete a stale user account or an active user account that you don't need any longer, you should first disable user accounts. You can dedicate an OU for all your disabled

accounts. After you disable an account, delete it after 30 days. This helps reduce risk in case an account is only occasionally used or somehow tied to something that you weren't aware of. Often, disabling an account will immediately find dependencies. And at that point, you can enable an account, put it back to its original location, and everything will be "back to normal".

- **It is a good practice to add a description to a user account when you disable it.** For example, you could add "Disabled by Brian due to stale account on 8/18/16". If another administrator gets a call to enable the account or stumbles across it and thinks it should be enabled, he will have information to help him. And he'll know who to contact for more information.
- **Be careful with special use cases.** For example, what if a user is out on maternity leave or long-term medical leave? In such cases, it is a good practice to add a description indicating that on the user account. And for these accounts, you don't want to delete them!

So far, you've learned how to find stale user accounts. And you've learned how to deal with them once you find them. If you are following the recommendations, you are disabling accounts, moving them to an OU, and adding a description to them. Let's fast forward 30 days. Now, it is time to delete the user accounts! So, in the next section, we'll show you how.

## Deleting User Accounts

You delete user accounts after they are no longer needed. Often, this coincides with an employee leaving a company, an application being retired (and thus its service accounts too), or a stale user check. Whatever the reason, like what we've described earlier in this chapter, you should perform the deletion in the following order:

1. Verify that the account isn't needed.
2. Disable the user account and move it to a dedicated OU.
3. Wait 30 days, or however long your company standards dictate.
4. After the elapsed number of days, if the account remained disabled (nobody needed it or used it), delete the user account.

Like the other tasks in this chapter, you can perform deletes using a graphical tool such as ADUC and ADAC or you can use PowerShell. We'll look at ADUC and PowerShell now.

### Deleting user accounts in ADUC

To delete a user in ADUC, perform the following steps:

1. In ADUC, search for or navigate to the user object that you want to delete.
2. Right-click the user object and click **Delete**.
3. In the confirmation window, click **Yes** to complete the deletion. Don't forget about protection from accidental deletion! If the user accounts are protected, you need to remove the protection before the deletion.

Figure 14.5 shows the ADUC user account context menu, where you click **Delete** to delete a user account.

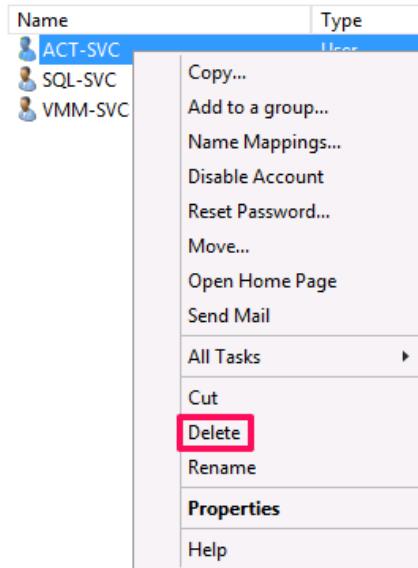


Figure 14.5 Deleting a user in ADUC

Besides deleting just one user at a time, you can also delete user accounts in ADUC a couple of other ways:

- **Delete multiple users.** To delete multiple users, you just navigate to where they are located (or use a query to search for them). Then, select all the users that you want to delete, right-click one of them, click Delete, and confirm the deletion.
- **Delete an OU and all user objects inside of it.** If you are dealing with many users, you can delete the parent OU and all the child objects. For example, if you move all your stale user accounts to an OU each month, it might hold many user accounts. You can delete the OU itself. When you delete an OU with objects inside of it, it will notify you that other objects are inside of it. You can proceed anyway (although it will fail if any of the child objects are protected from accidental deletion). Or, you can click the Use Subtree Server Control option. By doing so, you force the OU and child object deletion, even if objects underneath have been protected from accidental deletion.

### Hands-on Exercise

Open ADUC. Create a test user account. Then, delete the user account.

Technically, deleting user accounts in ADUC is easy. The hardest part is knowing when to disable them and when to delete them. Let's stick with deleting users for another section by showing you how to perform the same tasks in PowerShell.

## Deleting user accounts in PowerShell

To delete a user in PowerShell, perform the following command:

```
Remove-ADUser -Identity Elias
```

Besides just deleting one user at a time, you can also delete multiple users. But be careful doing so without testing. Because you might accidentally delete more user accounts than you planned to. In the following example, the command finds all users with a first name of "Bob" and deletes them without any confirmation:

```
Get-ADUser -Filter 'GivenName -eq "Bob"' | Remove-ADUser -Confirm:$False
```

Before you start deleting objects like this, consider using the `-WhatIf` parameter. Just add `-WhatIf` to the end of the previous command. Instead of deleting the objects that it finds, it will report back the objects that it would've deleted (if the `'WhatIf` parameter wasn't used). If the output looks good, you can remove the `-WhatIf` parameter and run the deletion command.

Finally, you can also delete an entire OU and all its child objects. This is another command that can be dangerous, so consider using the `-WhatIf` parameter before you run real commands in a production environment. The following command deletes the Chicago OU and all its child objects in the contoso.com domain:

```
Remove-ADObject -Identity "OU=Chicago,DC=contoso,DC=com" -Recursive
```

As you can see, PowerShell is powerful when it comes to deleting objects. Just be careful and use the `-WhatIf` parameter!

## Hands-on Exercise

Open PowerShell. Create a test user account. Then, delete the user account.

In this chapter, we've looked at some very common user management tasks. As you probably noticed, many are tied directly or indirectly to the security of your environment. That makes these common tasks very important. We covered password resets, unlocking user accounts, managing stale accounts, and deleting accounts. And we showed you a good mix of the GUI tools and PowerShell. I hope you took the time to do the Hands-on Exercises. Because we are about to jump into the lab to test your skills.

## Lab

### Reset a password using PowerShell

Perform the following tasks:

- Use PowerShell to reset the password for a user account to **Fef7yur3#e9ruyras**.

### Unlock all locked out users using PowerShell

Perform the following tasks:

- Use PowerShell to find all locked out users and then unlock them. Your solution should be a single line of PowerShell.

## Find stale accounts

Perform the following tasks:

- Use ADUC to find user accounts that haven't signed in for 90 days.
- Use PowerShell to find user accounts that haven't signed in for 30 days and output the account name and last signed in date to stale-users.csv.

## Delete user accounts

Perform the following tasks:

- In PowerShell, create a new OU named Test99 that is not protected from accidental deletion.
- In PowerShell, create two new user objects in the new OU.
- Use ADUC to delete the OU and the two user objects by using the Delete Subtree server control option.

## CHAPTER 15: MANAGING ACTIVE DIRECTORY COMPUTER OBJECTS

---

In the last two chapters, we took a close look at creating and managing user accounts. User accounts are a fundamental component of Active Directory and are often part of your daily administrative tasks for activities such as onboarding, off boarding, and updating a user's account information, such as their address or phone number.

Now we are going to shift over and look at managing computer accounts. In many ways, computer accounts are very like user accounts. For example, both account types are usually stored in OUs, both are used for authentication with the domain, both have many attributes, and both can be delegated in the same way. These are just a few examples, and as we proceed through the next few sections you will see other similarities along the way.

In this chapter, we will be dealing with computer account management in Active Directory, covering common administrative tasks, such as joining a computer to the domain. While the process of joining a computer to the domain is relatively straightforward, there are some key considerations to think about beforehand, like whether you should create computer accounts before you join them to the domain (referred to as "pre-staging"). We will also discuss how computers in a domain securely communicate (over a secure connection called a "secure channel"), especially how secure channels are used for establishing connections with domain members. Then, we will spend some time addressing scenarios where when the secure channel isn't working and prevents a computer from participating in the domain, requiring you to reset the account. Resetting a computer account may not be an everyday occurrence, but knowing what it does and when to use it can be a vital resource when troubleshooting client communication issues. Lastly, we will deal with managing stale computer accounts, something that every environment is susceptible to. While many organizations require that user accounts be disabled and deleted upon termination, stale computer accounts tend to be forgotten about.

At the end of this chapter, we have a lab. We will have you perform all the core activities we cover in this chapter and see if you mastered the topics.

### Prestaging Computer Accounts

Tailspin Toys has recently expanded their operations to a new sales office in Chicago. The office will house 200 new employees. In preparation for the office opening, the purchasing department for Tailspin Toys has ordered 200 laptops. Your manager has asked you to prepare Active Directory for these new devices so that the help desk staff can quickly bring them on the network. How do you proceed?

This scenario may or may not be on the same scale as the environment you manage, but the concept of preparing Active Directory for new computers is something that you should be familiar with. In this situation prestaging the computer accounts will help prepare for the new deployment of laptops. So, what is prestaging? Prestaging is creating the computer accounts in Active Directory before the devices are joined to the domain. For reference, if you don't do that, computer accounts are created in Active Directory during the domain join process. Prestaging is

an ideal solution for situations where you can expect more than a handful of new computers to be joined to your domain around the same timeframe. Some common examples include refreshing a classroom of computers, delegating the ability to join the new computers to the domain to employees that don't normally have rights to join computers to the domain (although, by default, all users can join 10 computers to the domain), or receiving a new shipment of computers that need to be deployed quickly.

Often the alternative to prestaging is granting elevated permissions to various members of your help desk team or having an admin with rights to join computers work on the task. The problem with this is consistency and security. You might be deploying 10 new computers or 200. In either case, knowing that they were created correctly and in the right location is an important objective.

There are several benefits for prestaging computer accounts, including:

- **Automation.** Many environments leverage automated deployment solutions for deploying their operating system images to the devices on their network, such as Windows Deployment Services (WDS). Prestaging the computer accounts in Active Directory can enhance the security of these deployment solutions by limiting new deployments to pre-staged accounts. If you don't set your deployments to pre-staged devices only, it is possible that anybody on the network can deploy an image to any device.
- **Location.** By prestaging the computer account, you can choose where that account will be stored in Active Directory. This eliminates the likelihood of accounts being left in the default Computers container. If computers get left in the Computers container, they won't receive the GPOs that you apply to the OUs that contain your computer accounts.
- **Naming.** By prestaging the computer account, you can choose the name of the computer. This eliminates naming inconsistencies or names that do not follow your computer naming standard.
- **Permissions.** By prestaging the computer account, you can assign domain join permissions to a less privileged account, such as somebody that doesn't even work in the IT department. This can be beneficial in situations where a user may need to rejoin their own computer to the domain due to a broken secure channel. This is also handy for a remote branch office that doesn't have an IT person or just has an IT person come by every week or two. Remember earlier, I mentioned that all users, by default, can join up to 10 computers to the domain. However, in many environments, the default configuration is changed to prevent that. In most environments, to maintain control and maximize security, you should have IT administrators join computers to the domain. In environments that are not running the default, pre-staging computer accounts is helpful to enable less privileged accounts to join a computer to the domain (such as in situations where an IT administrator isn't at a remote branch office).

At this point you should have a clear understanding at why prestaging is a good practice, and in what situations it would be beneficial. Now let's look at how to pre-stage a computer account in Active Directory.

## Prestaging a computer account using ADUC and ADAC

1. Navigate to the OU where you want to create the pre-staged computer account.
2. Right-Click the OU, click **New**, and then click **Computer**.
3. In the **New Object – Computer** window, type a name for the computer account. At this step, you have the option to adjust the default join domain permissions for this computer account. So, if you wanted to have the branch manager at a remote location join the computer to the domain, you could configure that at this step.
4. Click **OK** to create the computer account.

Figure 15.1 shows the right-click menu for creating a new computer account in ADUC.

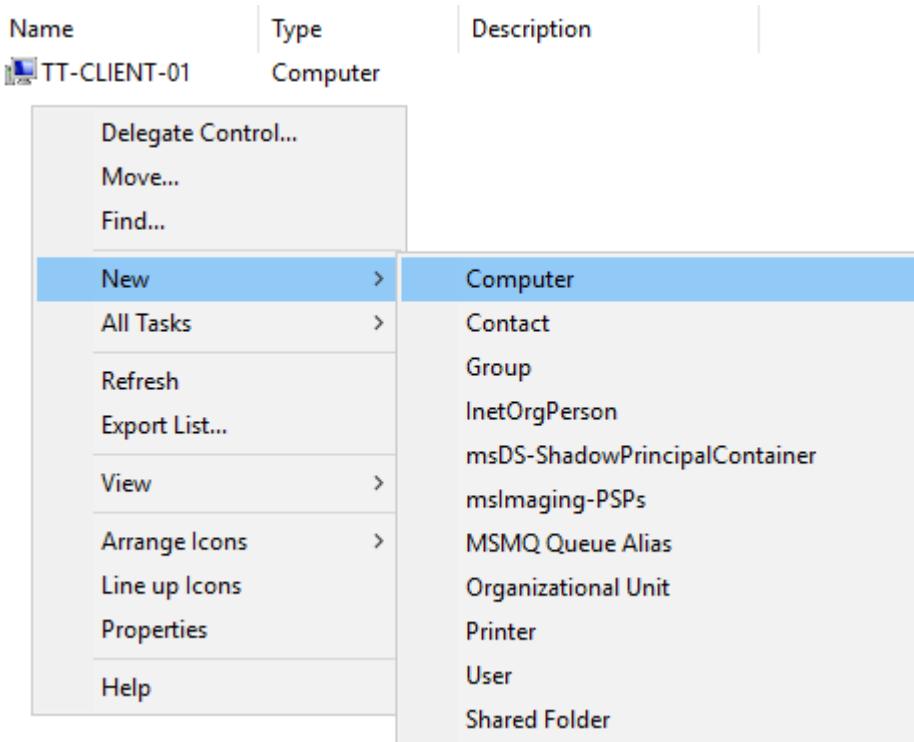


Figure 15.1 You can create a new computer account in ADUC by using the right-click menu. Right-click the OU where you want to create the new computer account, click New, and then click Computer to start the new computer creation wizard.

In ADAC, the process is very similar, but the setup page includes some additional options for assigning group membership and management information.

Figure 15.2 shows the right-click menu in ADAC for creating a new computer account.

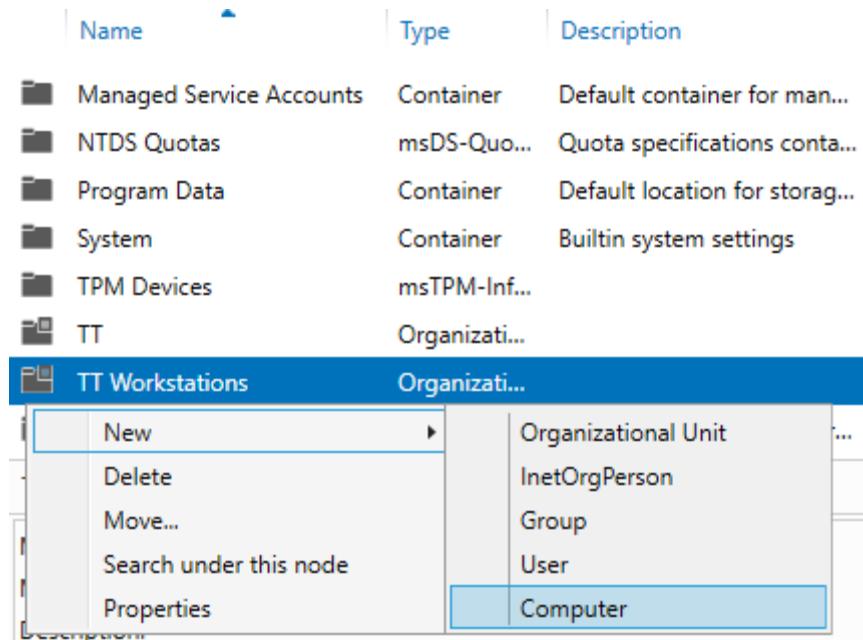


Figure 15.2 Creating a new computer account in Active Directory Administrative Center. Right-click the OU where you want to create the computer object, click New, and then click Computer to begin the process.

## Prestaging a computer account using PowerShell

The process for creating a single new computer account in ADUC or ADAC is easy, but what options do you have when you need to create accounts in bulk? For those situations, PowerShell is a great solution. For example, to create a pre-staged computer account named Computer1 using PowerShell, run the following command from an elevated PowerShell prompt (note that you will be prompted for your domain credentials):

```
New-ADComputer -Name "Computer1"
```

You should now have a new computer account in your Computers container (or whichever OU you use for newly created computer objects). If you have a list of computers that you would like to pre-stage, save them to a CSV file and import them. Run the following PowerShell commands from an elevated PowerShell prompt to import a list of computers and create them in a single pass. In this example, the column header for the list of computers is "Name". And the file with all the computer names is C:\Temp\computers.csv.

```
$Source = "C:\Temp\computers.csv"
$OU = "OU=TT Workstations,DC=tailspintoys,DC=com"
Import-Csv -Path $Source | ForEach-Object {New-ADComputer -Name $_.Name -Path $OU}
```

## Hands-on Exercise

Open Active Directory Users and Computers and create a few computer accounts. Then, use PowerShell to accomplish the same task. Compare the methods to see which you like better for prestaging computer accounts.

So, pre-staging is a good thing, when a situation can benefit from it such as when you want to have a non-administrative user join computers to the domain. Even when you use prestaging, you will routinely still have to join computers to the domain during troubleshooting and unplanned situations. So, for situations where pre-staging doesn't make sense (such as working with an individual computer during a troubleshooting situation), you need to be intimately familiar with joining a computer to the domain. Let's talk about that in our next topic.

## Joining a Computer to a Domain

You will need to join computers to the domain regularly. For example, if you purchase a new physical server, you'll need to configure it for your network and then join it to your domain. If you order new laptops, you'll need to join them to the domain. Domain joining is rarely something that end users can assist with, usually due to a lack of knowledge about the process and a lack of permissions to do so.

The process for joining a computer to the domain is generally done from a standalone server or a client computer. The following steps will walk you through joining a Windows 10 computer to the domain using the Windows interface. These steps are similar for joining a server running Windows Server 2016 to a domain.

### Join a computer to the domain using the GUI

1. On the client computer, open the **System** properties page in Control Panel.
2. In the left pane, click **Advanced System Settings**.
3. In the **System Properties** window, click the **Computer Name** tab and then click **Change**.
4. On the **Computer Name/Domain Changes** window, select the **Domain** radio button and then type the name of your domain.
5. At the **Windows Security** prompt, enter the account credentials that have permission to join the computer to the domain and then click **OK**.
6. Reboot the computer to complete the domain join process.

Figure 15.3 shows the Computer Name/Domain Changes dialog window for adding a computer to the domain. In this example, we have provided a computer name and the name of the domain.

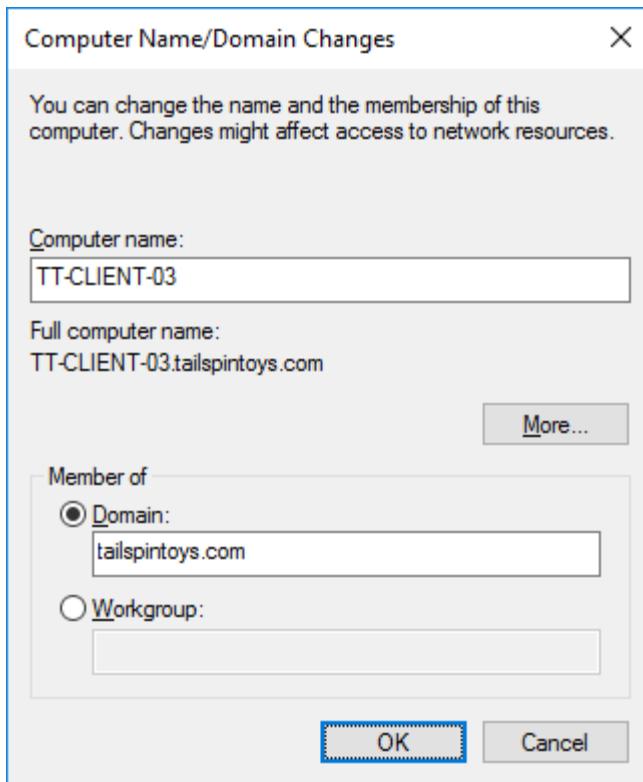


Figure 15.3 Joining a Windows 10 computer to the domain starts with going to the Systems properties and selecting the domain option.

### Join a computer to the domain using PowerShell

Joining a computer to the domain through the GUI is a simple process, and in some cases programmed into muscle memory for many veteran IT administrators. That said, there is a faster method available. After you have joined a few computers to the domain you may notice that it takes roughly 9 clicks of the mouse and multiple windows to complete the task. You can use PowerShell to speed things along or assist with an automated deployment process.

To join the local computer to the tailspintoys.com domain, run the following command from an elevated PowerShell prompt:

```
Add-Computer -DomainName tailspintoys.com -Restart
```

## **Hands-on Exercise**

Sign into a standalone computer using a local administrative account and join the computer to the domain. Then reboot. Once the computer is joined to the domain, sign in using a domain account. Confirm the sign in was successful, remove the computer from the domain, and then join it to a workgroup. Then reboot. Lastly, try joining that same computer to the domain again, but this time use the PowerShell method discussed in this section.

So far, you've learned how to pre-stage computers in the domain which is useful when you are planning to join computers in bulk or have non-administrators join computers to the domain. You've also learned the process to manually join individual computers to the domain which is useful when deploying new computers or troubleshooting. But once computers are joined to a domain, sometimes you need to troubleshoot them. And one such situation is when a computer has a problem with its secure channel (trust) with the domain. If you are wondering what a secure channel is (or how to fix a broken secure channel), jump straight to the next topic where we walk through it.

## **Computer Secure Channels**

Active Directory is designed to provide secure authentication between its members and other domain resources. When you join a computer to your domain, a trust relationship is set up between the domain controller and the domain member. This connection is established through a secure channel. The secure channel is based on a rotating password policy associated with the member computer and Active Directory. Every computer account has its own password that it uses to maintain secure communication with the domain. On a periodic basis, domain-joined computers will compare their password with Active Directory (which also stores a hash of the computer account password) to confirm the secure channel is still valid. If for any reason the passwords don't match, the trust relationship between the domain member and the domain controller will fail. Common reasons are Active Directory replication issues, restoring a client computer to a restore point, or restoring a domain controller to a previous point in time. In this situation, users will be unable to sign in to the computer or authenticate with other network resources. The service responsible for establishing the secure channel is named NetLogon. The Netlogon service is configured to start automatically at boot and will attempt to communicate with the domain as soon as a computer starts.

As an administrator, your primary interaction with secure channels will be troubleshooting them and fixing them. So, let's jump right into those tasks.

### **Check the health of secure channels using PowerShell**

Before we talk about how to fix broken secure channels, let's take a quick look at checking the health first. Because, if you check the health and the secure channel is functional, you won't have to waste time trying to fix it! The following PowerShell command can be run from a

domain-joined computer to check the status of the secure channel. This can be useful for troubleshooting trust relationship issues between domain members and the domain.

```
Test-ComputerSecureChannel -Verbose
```

The output from this command will provide a true or false status, and in verbose mode a message, to indicate whether the secure channel is healthy. If it isn't healthy, then you need to act to fix it. You have a few choices. Let's look at some choices now.

## Resetting Computer Accounts

Now that we've reviewed how secure channels work, the next step is understanding the tools you have at your disposal for remediating a broken secure channel with a member computer. You can resolve a broken secure channel in a couple of different ways:

- **Rejoin the domain.** This process involves rejoining the computer to the domain, which isn't often the first choice because it is time-consuming. This typically involves joining the computer to a workgroup, rebooting, and then rejoining the domain (which requires another reboot). While this method is effective, it isn't always ideal because of the downtime required.
- **Reset the account.** You can often fix the secure channel by resetting the computer account password and the domain account password so that they are back in sync.

Earlier in this chapter, we walked through the process of joining a computer to the domain. Rejoining the domain is very similar, so we will not go into detail on those steps. Instead, we will look at how to reset the computer account by using the GUI and by using PowerShell. The GUI and PowerShell options for resetting a computer account enable you to maintain computer account metadata (data about the computer stored in AD DS) and maintain group memberships. Additionally, resetting an account often takes less time than rejoining the domain.

### Resetting a computer account using ADUC

Computer accounts can be reset by using ADUC or PowerShell. Both offer similar functionality so the tool choice is often a personal preference with this task. By using ADUC to reset a computer account, you are only resetting the account password in Active Directory. This action will still require you to rejoin the computer to the domain. You might ask, when would I ever need to do this? One possible circumstance is to preserve an existing computer account's security groups and attributes. If you reset the account and rejoin the domain, those items are preserved.

1. Navigate to the location of the computer object or search for the computer object.
2. Right-click the computer object and then click **Reset Account**.
3. On the **Active Directory Domain Services** window, click **Yes** to confirm the change.
4. Click **OK**.
5. Rejoin the computer to the domain.

It is a good idea to try the PowerShell method first since it is more efficient. Then, if that doesn't work, resort to this method.

### **Resetting a computer account using PowerShell**

Using PowerShell, you can reset the local account password and domain account password in a single step. This enables you to avoid disjoining and rejoining the domain. This is a helpful command to memorize for those situations where you need to quickly restore the secure channel with a computer. To repair the secure channel connection by using PowerShell, run the following command from an elevated PowerShell prompt on the member computer:

```
Test-ComputerSecureChannel -Repair -Credential tailspintos\brian
```

After authenticating, the command will run and the output will provide a true or false response based on success. In our example, we use the -Credential parameter. Because the command is resetting the computer account in the domain, you need to provide credentials that have access to accomplish the task.

### **Hands-on Exercise**

Use ADUC to reset a computer account for a computer that is currently joined to the domain and has a healthy secure channel established. Restart the computer and attempt to sign in using a domain account. It will fail. Now sign in to the computer using a local account and run the PowerShell command for repairing a secure channel connection. Restart the computer once more and try logging in again with a domain account. As a final step, check the secure channel connection using PowerShell.

Now that you know how to investigate and reset broken secure channels, let's look at dealing with computer account when they are no longer in use.

## **Managing Stale Computer Objects**

Like stale user accounts, stale computer accounts are objects that you will need to deal with from time to time. Both account types get added to Active Directory frequently, but user accounts have a higher tendency to be audited and removed when access needs to be terminated for one reason or another. Computer accounts, on the other hand, are not always given the same level of importance. After all, an orphaned computer account can't cause any harm, right? Wrong! They can affect reporting or computer management. And, there might be licensing impacts for software that is licensed per computer account. In an example covering reporting or licensing, if your manager asked for a count of current Windows Server 2012 R2 computers in the domain, a query in Active Directory would be inaccurate due to the stale computer accounts. In another scenario, you might use System Center Configuration Manager to install software updates on your computers. Because Configuration Manager looks to Active Directory for computer accounts, any inactive objects are going to impact your success rate for

patch deployment. Finally, from a licensing perspective, you don't want to pay for licenses that you don't need and licensing sometimes is handled by counting objects in Active Directory.

In this section, we will review two ways to find stale computer accounts and give you some ideas on how to deal with them after you find them. As you can imagine, there are always several ways to do these types of tasks. Another way that we aren't walking through in this book, is by using Active Directory Users and Computers' Find feature, which has some common queries such as the number of days since the last logon.

### Finding stale computer accounts by using dsquery

Dsquery is a command line utility included with Windows Server as part of the Active Directory Domain Services (AD DS) management tools. From a command prompt, run the following command to identify computer accounts that haven't been active in 24 weeks:

```
dsquery computer -inactive 24
```

The output from the command will display the Distinguished Name of all the computer accounts in the forest that are inactive based on the number of weeks you provided (in our example, 24 weeks). In Figure 15.4 you can see an example of the dsquery command and the expected output. If you do not receive any output, you can try lowering the value. There are a few ways to save the results from this query. One solution is to use PowerShell. From a PowerShell prompt, run the same command with the Out-File parameter. For example, to output the command results to C:\Temp\stale-computers.txt, run the following command:

```
dsquery computer -inactive 6 | Out-File C:\Temp\stale-computers.txt
```

```
C:\>dsquery computer -inactive 6
"CN=TT-CM-01,OU=Servers,OU=SF,OU=TT,DC=tailspintoys,DC=com"
"CN=TT-SQL-01,OU=Servers,OU=SF,OU=TT,DC=tailspintoys,DC=com"
"CN=TT-CLIENT-10,OU=Workstations,OU=SF,OU=TT,DC=tailspintoys,DC=com"
"CN=TT-WIN10-01,CN=Computers,DC=tailspintoys,DC=com"
"CN=TT-DC-02,OU=Domain Controllers,DC=tailspintoys,DC=com"
"CN=TT-DC-03,OU=Domain Controllers,DC=tailspintoys,DC=com"
"CN=TT-UTIL-01,OU=Workstations,OU=SF,OU=TT,DC=tailspintoys,DC=com"
```

Figure 15.4 Run a dsquery to identify inactive computer accounts.

### Hands-on Exercise

Open a command prompt. Find stale computers by using a command like the example shown. Start by searching for computers that haven't signed in for 6 weeks. If you don't find any, drop the number down to 3, 2, and even 1 until you can find some.

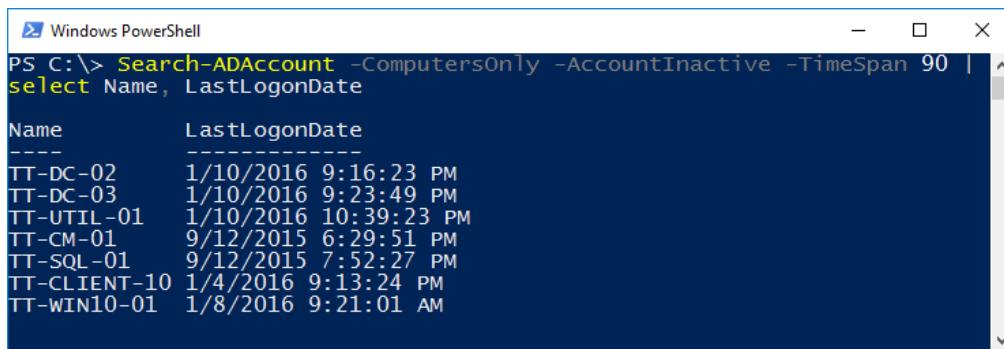
Next, let's look at how you can find stale computers in PowerShell.

## Finding stale computer accounts by using PowerShell

Using PowerShell, you gain the flexibility of gathering some additional attributes when running your query. This can be very useful for reporting. Run the following PowerShell command to identify inactive computer accounts in Active Directory with a last logon date greater than 90 days:

```
Search-ADAccount -ComputersOnly -AccountInactive -TimeSpan 90 | select Name, LastLogonDate
```

The output from this command will report back the computer name for all the computer accounts that are inactive based on the value of 90 days. In Figure 15.5 you can see an example of a PowerShell command finding computers that are inactive for 90 days. If you do not receive any output, you can try lowering the timespan value.



The screenshot shows a Windows PowerShell window titled "Windows PowerShell". The command entered is: PS C:\> Search-ADAccount -ComputersOnly -AccountInactive -TimeSpan 90 | select Name, LastLogonDate. The output displays a table with two columns: "Name" and "LastLogonDate". The data is as follows:

Name	LastLogonDate
TT-DC-02	1/10/2016 9:16:23 PM
TT-DC-03	1/10/2016 9:23:49 PM
TT-UTIL-01	1/10/2016 10:39:23 PM
TT-CM-01	9/12/2015 6:29:51 PM
TT-SQL-01	9/12/2015 7:52:27 PM
TT-CLIENT-10	1/4/2016 9:13:24 PM
TT-WIN10-01	1/8/2016 9:21:01 AM

Figure 15.5 Run a PowerShell command to identify inactive computer accounts.

### Hands-on Exercise

Open PowerShell. Find stale computers by using a command like the example shown. Start by searching for computers that haven't signed in for 90 days. If you don't find any, drop the number down to 30, 10, and even 1 until you can find some. Then, re-run the command and export the results to a .csv file.

OK, so you know what stale computers are. And you know how to find them. But, that's just the first couple of steps. The information you want to know is what you are supposed to do with stale user accounts. Let's find out next.

### What to do with stale computer accounts after you find them

So, you've found some stale computers. And some of them haven't signed in for 6 months! What now? The short answer is to delete them! But there are a couple of key considerations:

- **It is a good practice to disable computer accounts before you delete them.** Whether you are preparing to delete a stale computer account or an active computer account that you don't need any longer, you should first disable it. You can dedicate an OU for all your disabled accounts. After you disable an account, delete it after 30 days. This helps reduce

risk in case an account is only occasionally used or somehow tied to something that you weren't aware of. Often, disabling an account will immediately find dependencies. And at that point, you can enable an account and everything will be "back to normal".

- **It is a good practice to add a description to a computer account when you disable it.**

For example, you can add "Disabled by Brian due to stale account on 2/18/16". If another administrator gets a call to enable the account or stumbles across it and thinks it should be enabled, he will have information to help him. And he'll know who to contact for more information.

So far, you've learned how to find stale computer accounts. And you've learned how to deal with them once you find them. If you are following the recommendations, you are disabling accounts, moving them to an OU, and adding a description to them. Then, 30 days later you are deleting them. We won't cover deleting computer objects because the process is so like deleting user accounts, which you already know how to do. However, I want to end this chapter with a Hands-on Exercise that challenges you to delete a computer account using PowerShell! To help you, go back to the previous chapter and review the method to delete a user account. Instead of using Remove-ADUser, you will use Remove-ADComputer to delete a computer account.

### **Hands-on Exercise**

Open PowerShell. Use the Remove-ADComputer cmdlet to delete a computer account.

Now that you've finished reading this chapter, it is time to test your knowledge in the lab. If you haven't already performed the Hands-on Exercises, you should do so now. Then come back and complete the lab. Good luck!

## **Lab**

### **Pre-stage 10 new computer accounts**

Perform the following tasks:

- Use ADUC to create 5 new pre-staged computer accounts in the Laptops OU. Use the following naming convention: LAPTOP-01 through LAPTOP-05.
- Use PowerShell to create 5 new pre-staged computer accounts in the Desktops OU. Use the following naming convention: DESKTOP-01 through DESKTOP-05.

### **Join a new computer to the domain**

Perform the following tasks:

- From a client computer, use the Windows GUI to join the computer to the domain.
- From a client computer, use PowerShell to join the computer to the domain.

## **Reset a computer account**

Perform the following tasks:

- Use ADUC to reset the computer accounts created in lab 5.6.2.
- Login to the computers created in lab 5.6.2 and verify the computer account has been reset.

## **Check secure channel status**

Perform the following tasks:

- Login to the computer accounts from lab 15.6.3 using a local account. Check the status of the secure channel connection with the domain.

## **Find stale computer accounts**

Perform the following tasks:

- Use dsquery to find inactive computer accounts that haven't connected to the domain in 3 weeks. Output the results to a text file named stale-computers.txt.
- Use PowerShell to find inactive computer accounts that haven't connected to the domain in 60 days. Output the computer account name and last logon date to a .csv file named stale-computers.csv.

## CHAPTER 16: MANAGING GROUPS

---

The Active Directory chapters leading up to this point have focused on the creation and management of user and computer objects. These objects make up the security identities for authentication in the domain. Without a valid user account, users cannot sign in and authenticate with domain resources. Without a valid computer account, member computers cannot establish a secure connection with the domain.

After you have user accounts and computer accounts, you need an efficient way to grant access to resources such as shared folders and applications. While you can use user and computer accounts for that, it is unwieldy. That's because assigning rights to users is repetitive when you have more than 1 user that needs access to the same resource (since you must perform the task once for each user that needs access). So, in this chapter, we are going to look at groups. Granting access to resources by using groups is efficient because you only assign rights once. So, you'll want to know the details! A big part of your day to day administrative work will be granting access to resources, troubleshooting resource access, and managing groups.

First up in this chapter, we are going to look at some real-world situations where groups play an important role. We will then examine the available group types and scopes in a domain. The type of group you deploy and the scope of the group will impact how it works. These fundamentals are important prerequisites before you grant access to a new application or service. Next, we will outline some of the default groups that make up Active Directory and what role they play in your day-to-day management. And finally, we will go over the process for adding accounts to groups, querying those groups for current membership, and removing accounts when necessary. As a system administrator, you will see these types of management tasks on a routine basis, and often it's a bulk set of changes, which provides a great opportunity to strengthen your PowerShell skills!

At the end of this chapter, we have a lab. We will have you perform all the core activities we cover in this chapter and see if you mastered the topics.

### Group Types and Scopes

You are a system administrator at Tailspin Toys. Within the organization, a new product development team has been created to focus on hoverboard technology. Jason, the team lead for this group, has submitted a ticket requesting a new SharePoint team site for project documentation and task management. All the members of Jason's team, which has 15 members, need access to the team site. To accomplish this task, we are going to implement a security group for assigning access to the SharePoint team site. Security groups are one of two types of groups, and we'll look at the details of each next. User accounts can be added to the group and the group can then be assigned the required permissions in SharePoint. Compare that with the time it would take to individually add each member of Jason's team to the SharePoint site!

In Figure 16.1, we create the group for the new hoverboard development team. In ADUC, you create groups in the same way that you create user and computer objects. Right-click on an OU or container, click **New**, and then click **Group**. The New Object – Group window prompts you for

a name, a group scope, and a group type. Before you start creating groups, you should understand the available options for group scope and group type, so let's look at the options next.

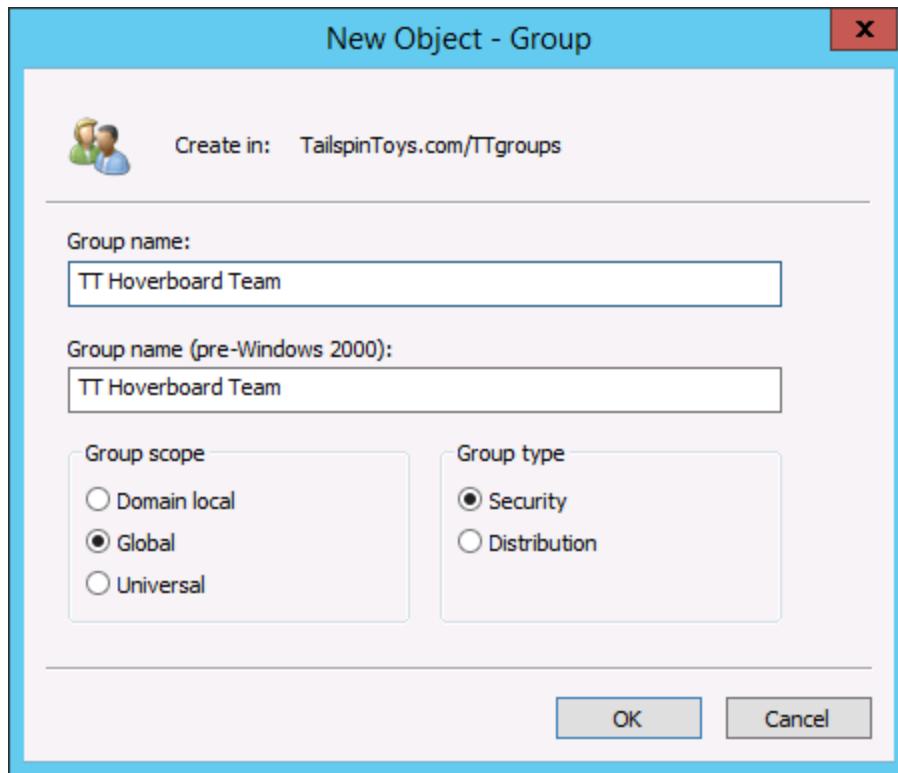


Figure 16.1 Creating a new group using Active Directory Users and Computers

## Group types

There are two types of Active Directory groups. Let's break down both types and look at a scenario where one might be beneficial over the other.

- **Security groups.** When you create a new group in Active Directory, the default group type will be a security group. Security groups are used to give access to resources. For example, you can use a security group to grant access to a file share or to web site. Let's look at how using security groups makes your job of assigning access to resources quick and easy. The following real-world example illustrates:
  - **A team manages 8 applications.** Imagine that you have a team of 4 administrators. And the team manages 8 applications. Each application has its own permissions. Without security groups, you must manually grant each administrator access to each application. In this case, you have 32 tasks. How do security groups help? Imagine instead that you create a security group named IT Admins. And you add your admins to the security group. Now, you just must give the security group access to each application. Much less work. Next, a new administrator starts. You have 1 task – add him to the IT Admins security group which gives him access to all the applications!

- **Distribution groups.** The other group type is the Distribution group. Distribution groups are used to send an email to a group of people. Distribution groups are used with email servers such as Microsoft Exchange Server. Distribution groups should be used in situations where the only requirement is email communication. For this chapter, we will mostly focus on security groups because distribution groups are often managed by the email team, and because we don't cover Microsoft Exchange Server as part of this book.

Group types is one facet of Active Directory groups. But there is one more facet that we'll talk about... group scopes. Group scopes dictate when and how you can use a security group. Let's look at group scopes now.

## Group scopes

An Active Directory forest is the top-level container in an AD DS implementation. As such, it is known as the security boundary for AD DS. Just underneath the forest are domains. A domain is a container which contains domain controllers and other objects. A domain is an administrative boundary for objects and objects share the same security policies with other domain members. A forest can contain a single domain (common) or can contain multiple domains (less common). When you create a group, the scope of that group dictates its reach in the forest as well as which other objects can be members of the group. Let's look at the key differences between the three group scopes – Domain local, Global, and Universal.

- **Domain local groups.** Domain local groups can have members from any domain in the forest. However, if other domain local groups are members, those groups must be in the same domain. You can assign permissions to domain local groups for resources in the same domain.
- **Global groups.** Global groups can have members from only the same domain. However, global groups can be assigned permissions in any domain in the forest.
- **Universal groups.** Universal groups can have members from any domain in the forest. And, they can be assigned permissions in any domain in the forest. The membership of a universal group adds AD DS replication overhead to your environment. For groups that will have routine membership changes, you should use domain local or global groups if they meet your requirements.

For most single-domain forest environments, you should use global and domain local groups for your group needs. A common practice is to use global groups as role groups. Role groups are used to group users that have a similar job role. For example, if you have a Helpdesk team with 5 members, you might have a role group named "Helpdesk". You add users to the role groups. Then, you create domain local groups as groups that will be granted permissions to resources. For example, you might have a domain local group named "AD DS Password Reset". Finally, you add the role groups as members of the domain local groups (this is referred to as "group nesting"). For example, the "Helpdesk" group would be a member of the "AD DS Password Reset" group.

In Figure 16.2, a diagram shows how a user and groups tie together to give the user permissions to an application named Application 1.

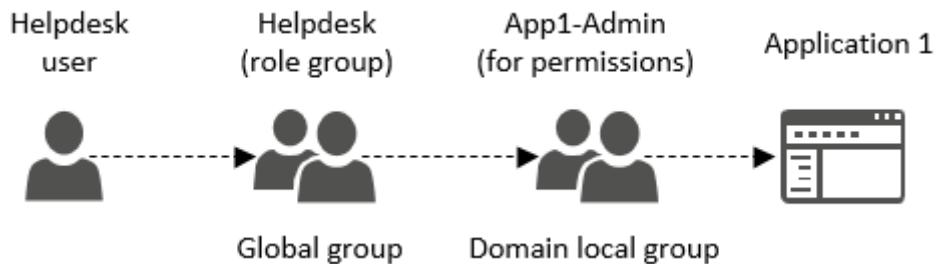


Figure 16.2 The Helpdesk user is a member of a Helpdesk role group. The Helpdesk role group is a member of a domain local group which is assigned administrative permissions in Application 1.

In some smaller organizations, it is also quite common to use only global groups. And those organizations often add users to the global groups and assign permissions to the same global groups. This can simplify the configuration.

### Hands-on Exercise

Open ADUC and navigate to the Users container. Create three new security groups, one with each scope. Name the groups: Group1-DomainLocal, Group2-Global, and Group3-Universal.

### Above and Beyond

There are a few details about group scopes. For your immediate day-to-day management, you don't need to know all the details. But, if you are curious, here is a bit more info about them:

- **Domain local.** A group set to the domain local scope can assign permissions for resources in the local domain. In a forest with multiple domains, a domain local group can contain members from any domain in the forest. For example, If Joe and Tom have user accounts in different domains, but need access to a resource in Joe's domain, both users can be added to a domain local group in Joe's domain that has access. Domain local groups can contain the following member types: user accounts, computer accounts, universal groups, and global groups.
- **Global.** A group set to the global scope can be assigned permissions for resources in the local domain, like a domain local group. However, a global group can only contain members from the local domain. The global group scope is the default when you create a new security group and is used most often for assigning permissions to resources in your local domain. Global groups can contain user accounts, computer accounts, and other global groups in the same domain.
- **Universal.** A group set to the universal scope can assign permissions for resources across multiple domains. You can directly add account objects to

universal groups, or nest other groups that contain the associated objects. For example, you have a universal group called HR that contains four members: HR North, HR East, HR South, and HR West. Each of these members is a global group in a separate domain. The global groups control HR permissions in the local domains, whereas the HR universal group provides access to core resources, such as the corporate HR SharePoint site. Universal groups can contain user accounts, computer accounts, universal groups, and global groups.

Now you have a good overview of groups. But, besides working with new groups and groups that you or another administrator created, there are also default groups. And the default groups are critical because some of them grant high-level administrative rights to everything in the Active Directory domain. We'll examine default groups now.

## Default Groups

Imagine that you deploy a brand new Active Directory environment. And now, you need to give all the Active Directory administrators access to perform their jobs. Starting from scratch like that, it would take quite a while to dole out all the different permissions. There are a ton of places in Active Directory where you would need to manually grant access and assign permissions. Luckily, Active Directory has some built-in security groups just for this purpose! Let's look at some of the default groups and when you would use them.

- **Administrators.** The Administrators group is a domain local group. The Administrators group provides full control to the domain and domain controllers. This group isn't often used, mostly because the Domain Admins group is a member. Most AD administrators are members of the Domain Admins group instead. The Domain Admins group has all the rights that the Administrators group does, plus administrative rights on all domain-joined computers.
- **Domain Admins.** Domain Admins is a global group, and is a member of the Administrators domain local group. This group is commonly used for managing the local domain and accomplishing tasks, such as installing a new domain controller and configuring replication. It has the same rights as the Administrators group but also has local administrative rights to all domain member computers. In the real-world, this group is one of the most commonly misused groups. In many cases, it is used as a "catch all". For example, when an application team is installing their application and the application needs a service account, it isn't uncommon for the application team to request that the service account be a member of the Domain Admins group. In most situations, the service account only needs local administrative access on the applicable member servers. As the administrator, you should be wary any time a request is made for membership to the Domain Admins group. Membership in the Domain Admins group should be strictly limited only to Active Directory administrators to maximize security and reduce risk.
- **Schema Admins.** Schema Admins is a universal group. The group enables members to extend the Active Directory schema. It is a good practice to only add users to the group temporarily such as when they need to extend the Active Directory schema. This can help prevent administrators from accidentally extending the schema without intending to (since,

if they aren't members of the group, they won't have rights to do it). When they are added to the group and they extend the schema, they should be removed from the group when the schema extension is complete. Schema extensions should be strictly controlled because a malicious schema extension or a bad schema extension can cause catastrophic damage to your environment. Before extending the schema, you should look at the details of the schema extensions (what is changing, what is being added), test the extensions extensively in a non-production environment, and perform a complete backup of your AD DS environment.

- **Enterprise Admins.** Enterprise Admins is a universal group. This group provides full control of the entire forest. This is another group that administrators will temporarily add themselves to for administrative tasks across multiple domains. This group is sometimes a prerequisite for some administrative tasks but is mostly useful in multi-domain forests, where the group gives rights to all the domains (whereas Domain Admins gives rights only to the domain where the Domain Admins group resides). For example, imagine that you have a domain with 4 forests. Each domain is managed by a small team of AD administrators that belong to their domain's Domain Admins Group. At your headquarters, you have the most senior level AD administrators. They are escalation points for all the Domain Admins. They can be members of the Enterprise Admins group to facilitate their role.

There are many more default security groups in Active Directory and you should take the time to familiarize with each of the group descriptions. As you work your way through the various default groups, take a moment to review the current group membership. As a security precaution, it is recommended that you routinely audit the membership of groups that provide administrative access in your domain. For example, check the group membership of Domain Admins and Enterprise Admins periodically. Or, you can opt for third-party software which can notify you by email or SMS when the group memberships change. Some products can even roll back membership changes automatically as part of the alerting.

Knowing high-level information about group types and scopes is important. But, even more importantly, you need to know how to manage the groups. And the first step in managing groups is knowing how to add members to a group, so we'll look at that next. Then, we'll work our way through other group tasks.

## Adding Members to a Group

Now that you know the different group types and scopes, you need to know how to add members to a group. In this section, we will review two methods for doing so – one by using the GUI and one by using PowerShell.

### Add members to a group using ADUC and ADAC

In Figure 16.3, we have added one member to the TT Workstation Administrators security group using ADUC. Use the following steps to add members to a group using ADUC.

1. In ADUC, navigate to the group that you want to update.
2. Right-click on the group and click **Properties**.

3. In the properties window, click the **Members** tab.
4. On the **Members** tab, click the **Add** button.
5. Type the user, computer, and/or group objects that you want to add and then click **OK**. If you needed to add multiple members, you can. Just add a semicolon between all the names.
6. Click **OK** to apply the membership changes to the group.

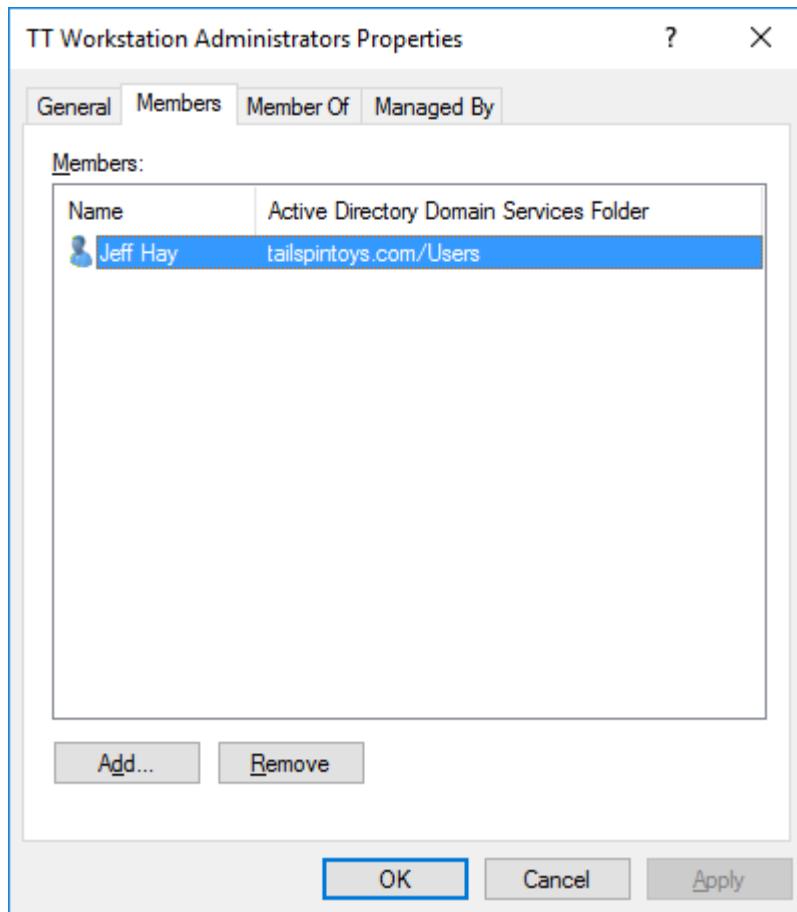


Figure 16.3 A group's properties showing the Members tab

### Managing group membership with ADAC

In Figure 16.4, we have added three new members to the TT All Users security group using ADAC. Perform the following steps to add accounts to a group using ADAC.

1. In ADAC, navigate to the group that you want to update.
2. Right-click on the group and click **Properties**.
3. In the properties window, click the **Members** section in the left pane. By the way, if you just wanted to view the members of the group, you would stop at this step.

4. Click the **Add** button under the **Members** section.
5. Type the user, computer, and/or group objects that you want to add and then click **OK**.
6. Click **OK** to apply the membership changes to the group.

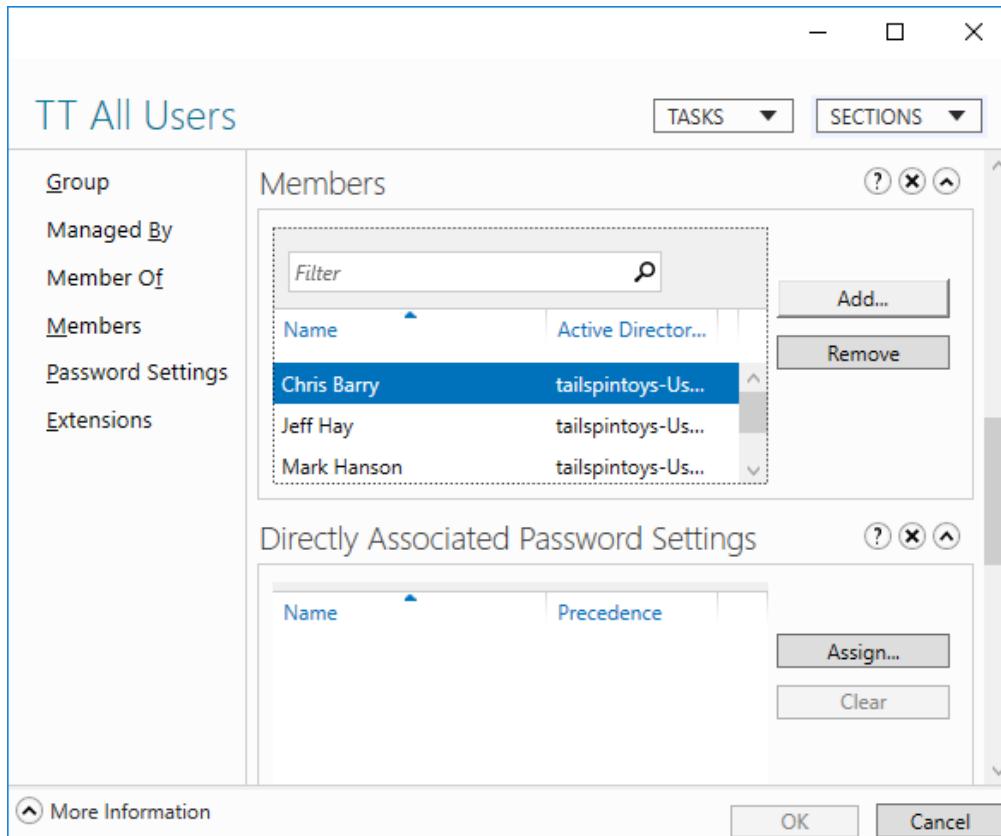


Figure 16.4 A group's properties in ADAC, showing the Members tab

### Add members to a group using PowerShell

Managing groups using the GUI is a perfectly suitable solution. In fact, many administrators use the GUI for most of their group management tasks. But, there are some tasks that are best suited for PowerShell. For example, if you need to add 25 users to a group, the GUI would be inefficient. Most administrators prefer PowerShell when they add many users to a group. Personally, the cutoff for me is when PowerShell can save me time. That's when I opt for PowerShell. So, let's see how we can use PowerShell to manage group membership.

Let's say that you have 25 users that you want to add to the TT Workstation Administrators group. The users, for this example, have sAMAccountNames of User1, User2, and so on with the last users having a sAMAccountName of User25. Perform the following steps to add them to the TT Workstation Administrators group.

1. Create a .txt file named users.txt with each user's sAMAccountName on one line of the text file. Figure 16.3 2 shows a sample users.txt file with 6 users.

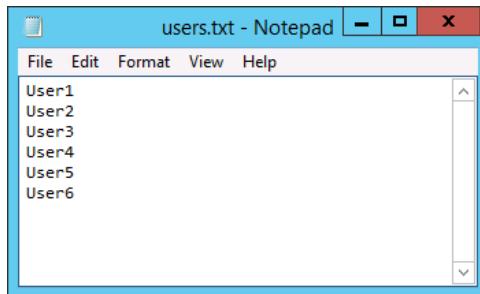


Figure 16.2 An input file for automating group membership changes

Open a PowerShell prompt and navigate to the directory where users.txt is located.

Run the following command:

```
Get-Content .\users.txt | ForEach-Object {Add-ADGroupMember -Identity "TT Workstation Administrators" -Members $_}
```

That's it – all 25 users are now members of the group. As you can see, using PowerShell for bulk tasks is a time saver.

As part of adding members to groups, a common follow-on task is to view the group memberships. To view the group membership using PowerShell, you can use the Get-ADGroupMember cmdlet. For example, to view the membership of the TT Workstation Administrators group and output the name of each member, you can run the following command:

```
Get-ADGroupMember -Identity "TT Workstation Administrator" | Select Name
```

### **Hands-on Exercise**

Open ADUC and create a new security group named Finance. Add some objects as members by using ADUC. Next, create a text file named users.txt and add several existing users' sAMAccountNames to the file, one on each line. Then, open a PowerShell prompt and add the members to the Finance group.

Whenever you come across administrative tasks that are repetitive and/or time consuming, stop for a second and consider whether PowerShell can help you automate the task or reduce the administrative time required to perform the task. My rule of thumb is that I use PowerShell whenever it saves me time compared to using the GUI.

Now that you know how to add members, let's look at removing members. As the administrator, you'll be doing a little bit of both.

## Removing Members from a Group

Removing members from groups is a common task when access permissions need to be rolled back or when an employee is switching to a new role. First, let's look at how to remove members by using the GUI.

Remove members from a group using ADUC and ADAC

Perform the following steps to remove members from a group by using ADUC:

1. In ADUC, navigate to the group that you want to update.
2. Right-click on the group and then click **Properties**.
3. In the properties window, click the **Members** tab.
4. Click to select the member you want to remove. Note that you can use CTRL+click and SHIFT+click to select multiple members.
5. On the **Members** tab, click the **Remove** button.
6. Click **Yes** to approve the change.
7. Click **OK** to apply the membership changes.

Perform the following steps to remove members from a group using ADAC:

1. In ADAC, navigate to the group that you want to update.
2. Right-click on the group and then click **Properties**.
3. In the properties window, click the **Members** section in the left pane.
4. Click to select the member that you want to remove. Note that you can use CTRL+click and SHIFT+click to select multiple members.
5. Click the **Remove** button in the members section.
6. Click **OK** to apply the membership changes.

Now, let's switch to PowerShell.

Remove accounts from a group using PowerShell

The process for removing accounts from a group is very like adding accounts (but with a different cmdlet). To remove Mark Hanson (with sAMAccountName of mark.hanson) from the TT Workstation Administrators group, run the following PowerShell command:

```
Remove-ADGroupMember -Identity "TT Workstation Administrators" -Members  
"mark.hanson" -Confirm:$false
```

By using the `-Confirm:$false` parameter, you can avoid being prompted to confirm the membership removal.

## **Hands-on Exercise**

Open ADUC and locate the Finance group that you created in earlier. Remove an account from the membership list of the group. Next, open a PowerShell prompt. Remove all the accounts that you added to the TT Workstation Administrator group in the previous Hands-on Exercise. For reference, refer to the last Hands-on Exercise.

In this chapter, we looked at the common tasks associated with Active Directory groups. By now, you should be comfortable with creating groups and managing group members in both the GUI tools and with PowerShell. This chapter wraps up our look at the common Active Directory objects – users, computers, and groups. But, before we move on from Active Directory, we have two more chapters. And those chapters cover Group Policy, a configuration management technology built into Active Directory and the Windows operating system. First though, let's test your group management skills in the lab.

## **Lab**

### **Group Types and Scopes**

Perform the following tasks:

- Use ADUC to create to the following security groups in the Users container: HR, Finance, and Marketing. The groups should be called FileShare-HR, FileShare-Finance, and FileShare-Marketing.
- Use ADUC to create a new email distribution group for the sales team under the Users container. The name of the group should be Sales Staff.

### **Default Groups**

Perform the following tasks:

- Use ADUC to locate the Domain Controllers security group. Which container is this group in, and what is the group scope?

### **Adding Accounts to a Group**

Perform the following tasks:

- Use ADUC to locate the Remote Desktop Users group. Add your account to the group.
- Use ADAC to review the membership of the Remote Desktop Users group.
- Create a test user named test98 and a test user named test99. Then, use PowerShell's Get-Content cmdlet in a command to add the two accounts to the Print Operators group.

## Removing Accounts from a Group

Perform the following tasks:

- Use ADUC to locate the Remote Desktop Users group. Remove your account from the group.
- Use PowerShell's Get-Content cmdlet in a command to remove the two test accounts from the Print Operators group.
- Use PowerShell to output the membership of the Remote Desktop Users group. Only return the name attribute in the output.

## CHAPTER 17: WORKING WITH GROUP POLICY

---

We've been working with Active Directory related tasks for 5 straight chapters! By now, you should be feeling comfortable with the core administrative tasks such as creating and managing users, groups, and computers. For the next two chapters, we are going to look at Group Policy. Group Policy which provides configuration management for your domain-joined computing devices. And the reason we are talking about it now is because you need users, groups, and computers to use it! So now that you have some users, some groups, and some domain-joined computers, let's get acquainted with Group Policy.

So, what is "configuration management" anyway? It is a nifty way to say "configure, and optionally, enforce settings on computers". Without Group Policy, you must individually configure each computer. For example, if you wanted to enforce a password-protected screen saver after 5 minutes of inactivity, you would have to go to each computer and individually configure that setting. With Group Policy, you configure that setting one time and a policy is used to push that setting out to the targeted computers. And that can be time-consuming and error prone. With Group Policy, you can configure desired settings once and push those settings out to your computers automatically. The more computers that you have, the more time you save with Group Policy.

But, Group Policy is complex. There is a lot to it. In fact, entire books have been written just about Group Policy. We aren't trying to condense an entire book of knowledge into 2 chapters. Instead, we'll focus on the core concepts and tasks that you can use regularly as an administrator.

In this chapter, we are going to introduce Group Policy, look at the different types of settings (policies and preferences), examine how Group Policy is pushed out to computers, how methods to granularly control how and when settings are or aren't applied to computers, we'll also walk through some common troubleshooting scenarios. At the end of the chapter, we will test your newly acquired Group Policy skills in the hands-on lab.

### What is Group Policy?

Group Policy is the built-in configuration management technology that comes with Windows operating systems. There are two forms of Group Policy:

- **Domain-based Group Policy.** Domain-based Group Policy is a function of Active Directory Domain Services. It is centralized management because you configure it per AD DS domain and it can be used across some or all your domain-joined computing devices. When people generically refer to "Group Policy", then are usually referring to domain-based Group Policy. Thus, it is common to hear admins only use the term "Group Policy" and the assumption is that it is the domain-based version.
- **Local Group Policy.** Local Group Policy is computer-based. Each Windows computer – client computer or server – has local Group Policy available. This is decentralized management because local Group Policy only applies to the local computer. Thus, if you want to configure a setting by using local Group Policy, you need to configure that setting on every single

computer. As you can imagine, in any environment but the smallest environments, using local Group Policy to configure all your computers is inefficient at best. As such, local Group Policy is not recommended if you have AD DS. If you do not have AD DS, local Group Policy is a valid option, though.

In this book, we will focus specifically on domain-based Group Policy and refer to it from here out just as "Group Policy". Group Policy offers configuration management by way of Group Policy objects (GPOs). GPOs define the computer settings that you configure. After you create a new GPO, you configure it to apply to computers or users (or both). That process is known as linking. You link a GPO. You can link a GPO to a domain, an AD DS site, or an OU. In Figure 17.1, a diagram shows two GPOs, each configuring different settings, linking to different OUs.

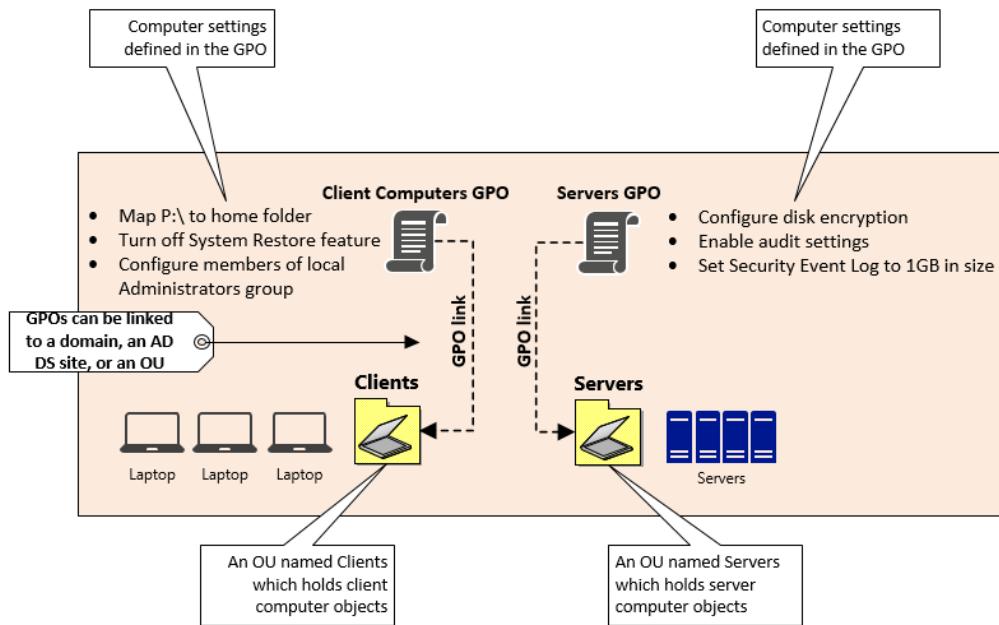


Figure 17.1 A diagram shows two GPOs linking to two different OUs. Each GPO sets unique computer settings and is targeted to a specific set of computers such as all the computers in the Clients OU or all the computers in the Servers OU.

We'll look at linking in more detail in the upcoming section titled "Understanding LSDOU".

You use the Group Policy Management Console (GPMC) to manage most aspects of Group Policy. Over the next couple of chapters, one of our goals will be to get you comfortable working with the GPMC.

In Figure 17.2, the GPMC is shown along with some of the key components annotated – the domain, GPOs, a GPO link, and an OU.

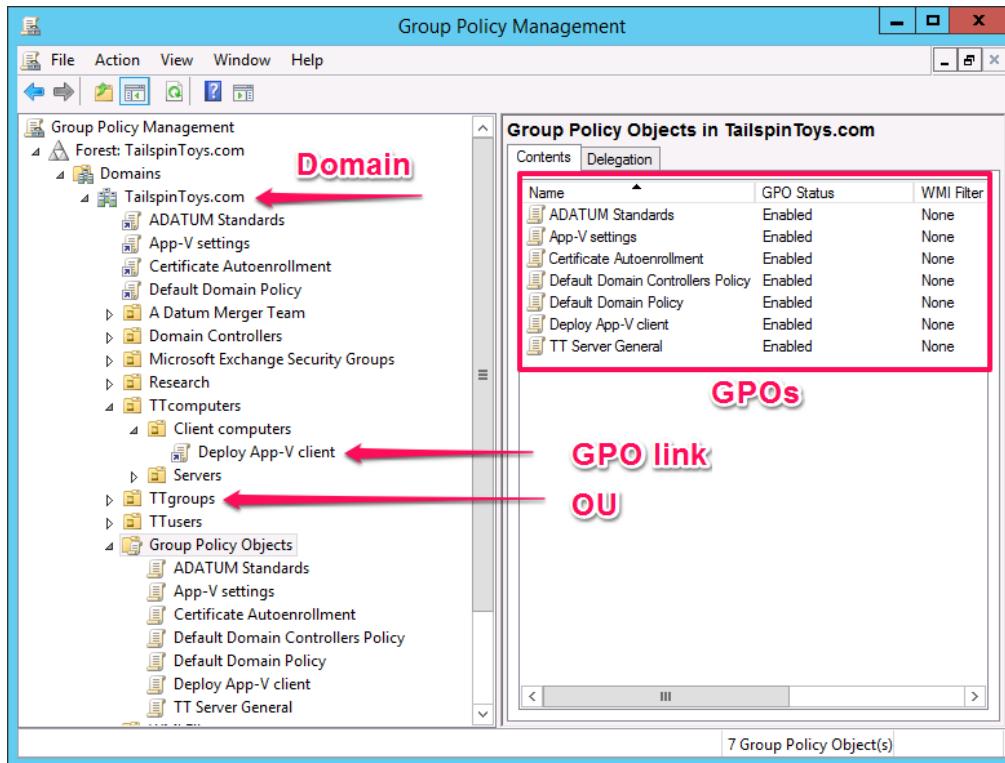


Figure 17.2 The Group Policy Management Console showing key components such as the domain, GPOs, a GPO link, and an OU.

When you configure a GPO, you have the option to configure thousands of different settings. The settings range from security settings (such as restricting which applications can be run on a computer) to look and feel settings (such as the desktop background image and the Start menu layout).

For many of the settings, you just choose whether a setting is enabled, disabled, or not defined (not configured), as shown below in Figure 17.3

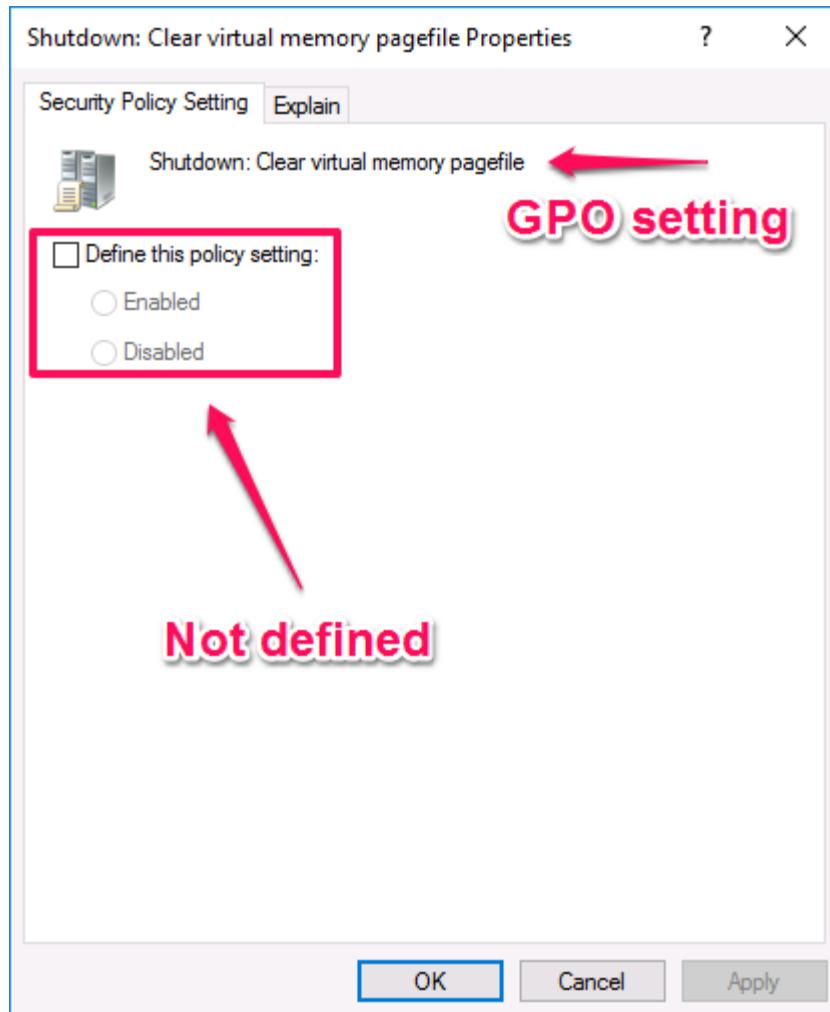


Figure 17.3 An undefined GPO setting

Whenever you have a setting that is not defined as enabled or disabled, you have an undefined (or not defined) GPO setting. You can define a GPO setting as enabled or disabled. When enabled, a setting applies. When disabled, a setting does not apply. But watch out for some settings that have names that can be confusing with the settings names! For example, there is a GPO named "Disable the Advanced page" which controls whether the Internet Explorer Advanced tab is displayed. If you disable that GPO setting, the tab is displayed. If you enable that setting, that tab is not displayed. For some GPO settings, you can enable a setting and choose specific values for the setting. We won't look at those details in this chapter, but you should take some time to explore some of the settings as you work through this chapter. Next chapter, we will look at some of the available security settings.

## Hands-on Exercise

Open Server Manager on an AD DS domain controller. From Server Manager, click Tools and then click Group Policy Management. Explore the console at a high level. Expand all the items in the left pane. Click the Group Policy Objects container and then double-click a GPO in the right pane. Continue to explore for a few minutes.

You now know, at a high level, what Group Policy is. But enough with the introductions. We have a lot of content to cover (2 chapters worth), so let's keep moving!

## Policies vs. Preferences

When you use Group Policy to configure your computers, you can opt to use policies or preferences. While both configure settings on computers, they do it in different ways.

- **Policies.** Policies are settings that are enforced. And often, the user interface to change the settings from the local computer is disabled. You use policies to enforce settings on computers when you do not want the settings to change. For example, you might want to configure some Internet Explorer security settings to maximize your organization's browser security. You certainly don't want an end user changing the Internet Explorer settings to reduce security. For example, your company might want to set the Internet Explorer security level for the Internet zone to "High" to reduce the risk of malware. If a user can change that setting to reduce security, it may leave your organization susceptible to malware. Thus, you should use a policy to enforce the settings.
- **Preferences.** Preferences are settings that are often not enforced. And the user interface to change the settings from the local computer is not disabled. Thus, users, even non-administrative users, can change the settings that you've deployed by using preferences. When you want to push "nice-to-have" settings to computers, such as adding a desktop shortcut to a shared folder, you should use preferences. In such a case, you are saying "Hi users – we've configured some settings on your computer to make things easier for you but feel free to change the settings (or remove them) if you prefer". Many of the settings available in policies are different from the settings available in preferences. Preference settings are mostly tied to Control Panel settings or environment settings. For example, you can configure power plans, mapped drives, scheduled tasks, and local users and groups with preference settings. Policy settings are much more wide ranging, covering just about every facet of a Windows and domain configuration. For example, you can configure all the Windows services behaviors for startup and management permissions. You can add custom registry keys. You can completely configure the Windows firewall. And you can configure granular auditing and logging.

### Hands-on Exercise

Open GPMC. Under the Group Policy Objects container in the left pane, right-click Default Domain Policy and then click Edit. Expand Policies and then look through some of the policy settings by double-clicking them. Read the explanations. Next, expand Preferences and then look through some of the available settings categories. Compare the type of settings available in each.

In Group Policy, settings are categorized as computer settings or user settings. Computer settings are initially applied when a computer starts up. User settings are initially applied when a user signs in. While some of the same settings are available as a computer setting or a user setting, other settings are only available in one or the other, depending on whether the setting is applicable. For example, there are some desktop related settings in Group Policy such as settings to manage the desktop icons. Those settings are only available as user settings because computers don't have a "desktop". On the computer settings side, there are some event log settings which dictate how the event logs handle data. Those settings are not available as user settings because users do not have an "event log". We won't cover all the intricacies of the settings or some advanced topics in this book. Instead, it is preferable that you get comfortable with the core concepts of Group Policy first. And core concepts are differentiating between policies and preferences, understanding linking, understanding blocking and enforcing, and troubleshooting common Group Policy problems.

By now, you have a good overview of Group Policy. You know that it is made up of GPOs, policies and preferences, and computer and user settings. Now, before we go any further, we need to make sure you know how to create a GPO. Because the tasks we talk about after GPO creation require some GPOs!

## Creating GPOs

Creating GPOs is quick and easy. The configuration is the tough part. Let's look at two methods to create new GPOs – using the GUI (GPMC) and using PowerShell (the New-GPO cmdlet).

### Creating GPOs using GPMC

Before we begin, go to Server Manager, click Tools, and then click Group Policy Management to run the GPMC. Then, perform the following steps to create a new GPO named Corp Servers.

1. Expand your forest, the Domains container, and your domain in the left pane.
2. Navigate to the Group Policy Objects container.
3. Right-click **Group Policy Objects** and then click **New**.
4. In the **New GPO** window, type Corp Servers as the name and then click **OK**.

That's it! You have a new GPO. But, it doesn't have any settings. And it isn't linked anywhere. Thus, at this point, it is worthless. But at least it is a start! Next, let's look at creating a new GPO using PowerShell.

### **Hands-on Exercise**

Open GPMC. Create a new GPO named Corp Servers.

### **Creating GPOs using PowerShell**

You can use the New-GPO cmdlet to create new GPOs. To create a new GPO named Client Computers, run the following command from a PowerShell prompt:

```
New-GPO -Name "Client Computers"
```

That's pretty much all there is to it. There are a few parameters that you can use, such as the –Comment parameter to add a comment (you might do this to describe the GPO goal so future administrators have an easier time in understanding the environment configuration). To see all the available parameters, run this the following command:

```
Get-Help New-GPO -Detailed
```

Now, let's say you have a GPO and you've defined some settings. And you want your computers to apply the settings. What's next? Linking your GPOs to the right locations in AD DS! Linking the GPOs effectively applies the settings! We are going to talk about that next while introducing a new acronym – LSDOU.

### **Hands-on Exercise**

Open a PowerShell prompt. Use the New-GPO to create a new GPO named Client Computers.

As you can see, creating GPOs is quick and easy. Now, let's look at taking those GPOs and linking them. Before GPOs are useful, they must be linked. In case you are wondering when we'll configure some GPO settings, be patient – we tackle that next chapter.

### **Linking GPOs**

Once you have GPOs and you configure the desired settings, you need computers to apply the settings. The process of associating a GPO with a target location is known as linking. You can link a GPO to a domain, Active Directory sites (objects in Active Directory that represent the physical sites of your company), and to OUs. You can link a GPO to more than one place at a time. And, you can have GPOs without any links (although, in such a case, the GPO settings are not applied).

By default, all GPOs have a security filter set to the Authenticated Users group. That group is made up of all domain users and domain computers that have authenticated. While this makes it simple to link GPOs, it may not be sufficient for all scenarios. In some cases, you can adjust the security filter so that only a specific group of users or computers will apply a GPO. For example, imagine that you have an OU that contains all your client computers. Half of them run Windows 8.1 and half of them run Windows 10. You need to push a policy to all the Windows 10 computers. But if you link the GPO to the OU, all the computers will get the policy. In this case, you can use a security filter so that only the Windows 10 computers get the policy. In addition to security filters, you can use Windows Management Instrumentation (WMI) filters to target specific computers based on a WMI query. We won't look at WMI filtering in this book but wanted to introduce the feature. Now, let's look at linking GPOs in GPMC and in PowerShell.

## Linking GPOs in GPMC

The process to link a GPO is quite simple. You use the Group Policy Management Console, navigate to the location where you want to link the policy, right-click the OU, site, or domain and click **Link an Existing GPO**. To unlink a GPO, you can click to highlight the link and then delete it. Only the link is deleted, not the GPO.

## Link a GPO by using PowerShell

Imagine that you have a GPO named "SQL Security Settings" and you want to link it to an OU named SQL Servers under a parent OU named Servers in the root of the tailspintoys.com domain. To do that in PowerShell, run the following command:

```
New-GPLink -Name "SQL Security Settings" -Target `  
"OU=SQL Servers,OU=Servers,DC=tailspintoys,DC=com"
```

In your daily routine, you won't often link a GPO to enough locations (such as 20 or more) to warrant the GUI inefficient. But it may come up occasionally. In such cases, you can add all the DNs of the link locations to a text file. Then, you can use the Get-Content cmdlet to retrieve that list and use PowerShell to add a link for each link location. Even in this case, creating the text file itself isn't that efficient.

## Hands-on Exercise

Create a text file with the DN to 3 link locations (such as 3 different OUs). Then, use PowerShell to link a GPO to the 3 locations in one line.

You may be wondering...how do you remove GPO links with PowerShell? Use the Remove-GPLink cmdlet.

Now you know how to link and unlink GPOs. And that will make up a portion of your Group Policy work. But, there are times when you link a GPO and it has some unintended consequences for some computers. In such a case, there is an advanced feature known as

inheritance blocking. Along with inheritance blocking, you have another advanced feature known as GPO enforcement to override blocking! Let's look at both in the next section.

## Blocking and Enforcing

By default, a GPO that is linked to a parent OU is automatically inherited by child OUs. Effectively, all child OUs have the same GPOs linked to them as the parent OU.

Now, imagine that you store all your server's computer objects in an OU named Servers. And, you have some child OUs for storing specific types of servers, such as SQL servers, so that you can easily delegate control of the server objects to the various IT teams. You implement a new GPO for servers to configure disk encryption. You link the GPO to the Servers OU.

A short while later, one of the SQL administrators reports that the SQL servers have some problems with their backups. It turns out that the disk encryption settings in your GPO are not compatible with their SQL backup software. But, you need the GPO to apply to all the rest of the servers under the Servers OU. What can you do to apply the GPO to the rest of the servers but not the SQL servers?

In this case, you can use a Group Policy feature known as inheritance blocking. With inheritance blocking, you can configure an OU to block GPO inheritance. If a GPO is linked above an OU, it won't be inherited by the child OU.

Figure 17.4 shows such a situation with inheritance blocking.

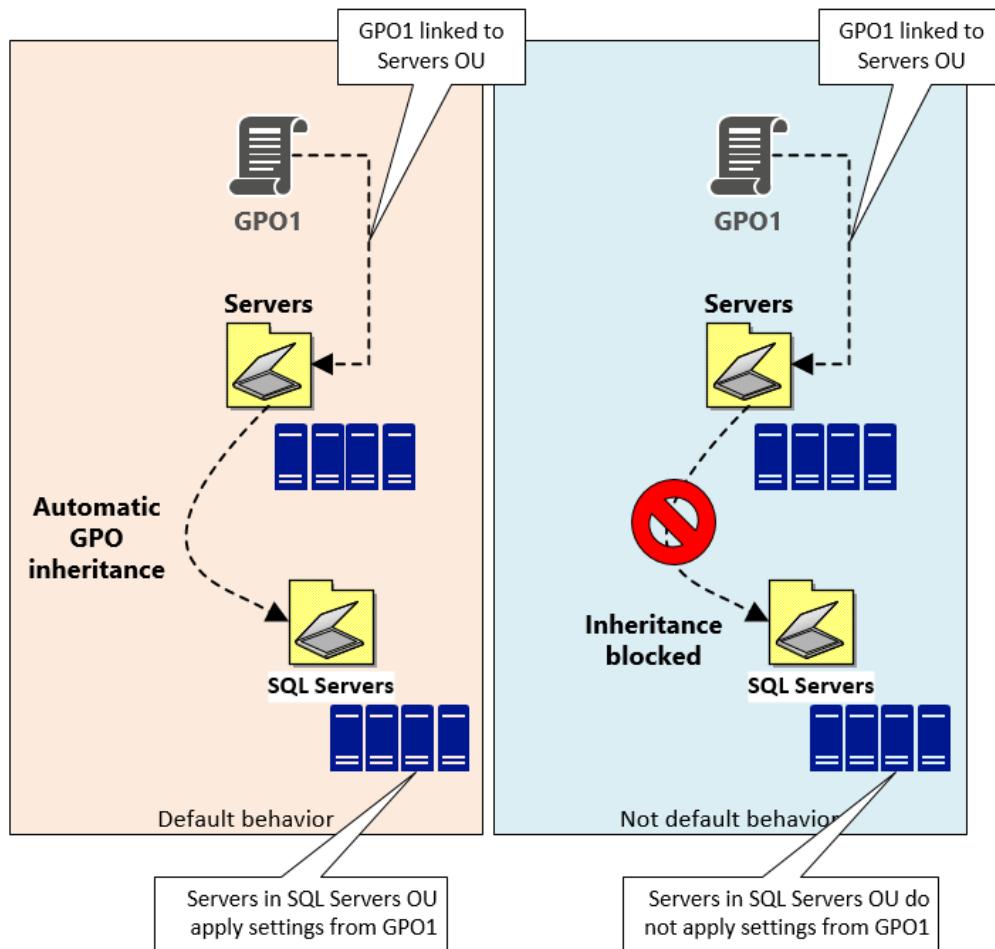


Figure 17.4 You can use GPO inheritance blocking to ensure that a child OU does not inherit GPOs linked to a parent OU.

One downside to inheritance blocking is that, for the child OU, you block all GPOs that are linked to the parent OU. Imagine that 3 months have passed since you solved the SQL backup problem. Your GPO is working for your servers and the SQL servers are functional because you blocked inheritance. Now, a new requirement comes in that says that all servers must be configured with a banner message during logon. You configure a GPO to display a banner message and link it to the Servers OU. All servers then have a banner message. Except for SQL servers. Because you configured inheritance blocking on the SQL Servers OU!

How can you ensure that the GPO with the banner message applies to all servers, including servers in the SQL Servers OU? You can use a Group Policy feature known as GPO enforcement. You can enforce a GPO link and when you do, that GPO overrides any inheritance blocking! While it is a good practice to reduce, or eliminate inheritance blocking and enforcement, sometimes you don't have a choice. For example, if you inherit an environment that already has inheritance blocking and get a request to implement a new setting across all computers, you may find yourself needing to use the enforcement feature.

Figure 17.5 builds on Figure 17.4 by showing how GPO2, a GPO that sets a logon banner, can use enforcement to get around the inheritance blocking.

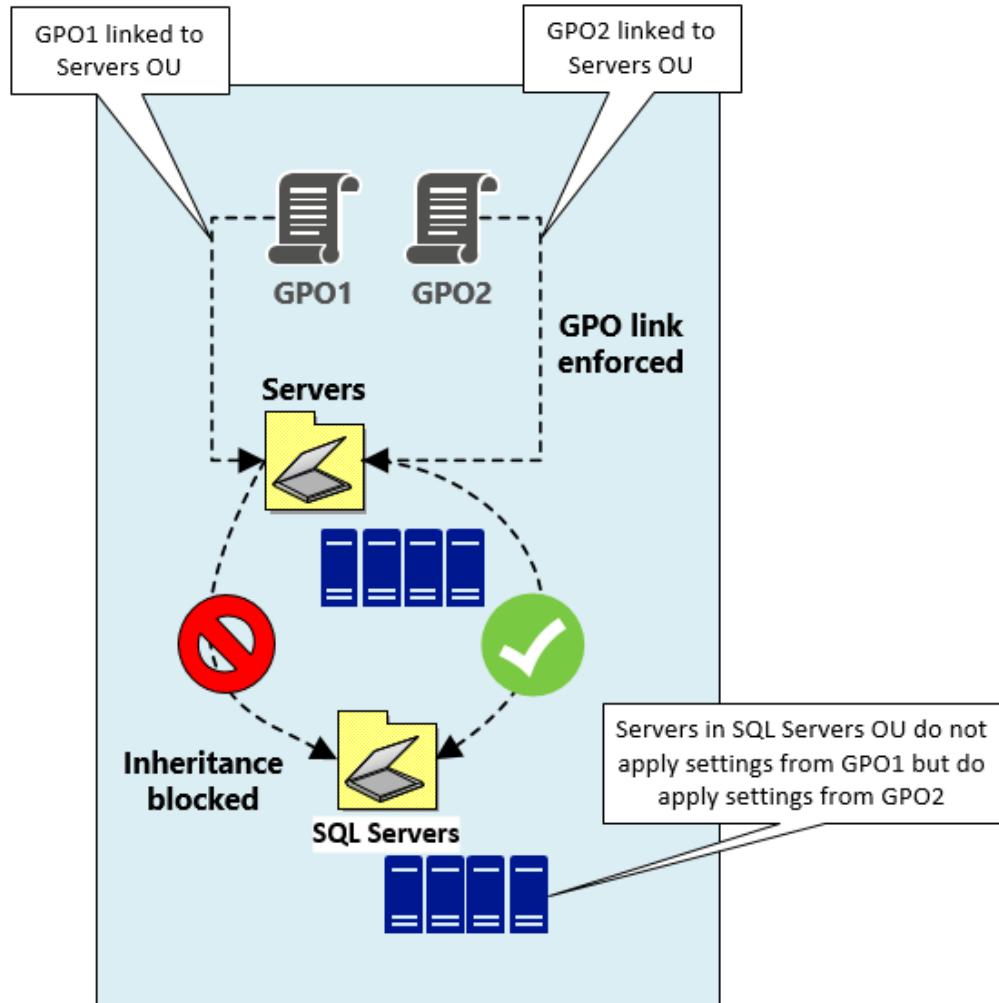


Figure 17.5 GPO enforcement overrides inheritance blocking so that a GPO link with enforcement is not blocked by inheritance blocking in Figure 17.5, the GPO2 GPO link is enforced. Thus, GPO2 applies to servers in the Servers OU and in the SQL Servers OU. Meanwhile, GPO1 only applies to servers in the Servers OU because inheritance is blocked.

To block inheritance on an OU, perform the following steps in the GPMC:

1. Navigate to the OU where you want to block inheritance.
2. Right-click the OU and then click **Block Inheritance**.

When you block inheritance, the OU icon changes to include a blue exclamation point to make it easier to identify where inheritance blocking is being used.

To enforce a GPO, perform the following steps in the GPMC:

1. Navigate to the GPO link that you want to enforce.
2. Right-click the GPO link and then click **Enforced**.

When you enforce a GPO link, the GPO link icon changes to include a small lock icon to make it easier to identify where GPO enforcement is being used.

### **Hands-on Exercise**

Open the GPMC. Create a test GPO named Test1 and a test GPO named Test2. Create a test OU named TestOU1. Then, create a child OU under TestOU1 named ChildOU1. Link GPO1 to the TestOU1 OU. Then, block inheritance on the ChildOU1 OU. Finally, link the Test2 GPO to the TestOU1 OU and enforce the link. Review the icons in the GPMC to verify your configuration.

As you probably guessed, inheritance blocking and GPO enforcement are features that can quickly complicate your environment and make troubleshooting difficult. For example, imagine that you inherited your environment. And, you are working day by day trying to get up to speed on everything. You get a request to configure a new GPO to set auditing levels on all servers. You create a GPO, configure it, and link it. But it only works on some servers. It will take a long time to go through all the troubleshooting steps and come up with a solution. Complexity adds things to look at and consider. And that takes time. Thus, you should complexity (in this case, inheritance blocking and enforcement) when you can. Now, let's show you some troubleshooting techniques to help you figure out why GPOs aren't doing what you think they should be!

## **Troubleshooting Group Policy**

In this section, we are going to walk through a couple of real-world situations and show you some techniques to help you troubleshoot and solve common GPO problems.

### **How to get around inheritance blocking**

Imagine the same scenario from earlier in this chapter. You have a Servers OU which contains many computer objects. You have a child OU named SQL Servers which holds the SQL Server computer objects. And, you have an OU named SQL QA Servers under the SQL Servers OU. You blocked inheritance at the SQL Servers OU to fix an earlier problem.

Now, you need to apply some security settings from a GPO to all servers. You could enforce the GPO link, as we mentioned in the last section. But, inheritance blocking and enforcement complicate your environment.

So, another way to fix this is to link the GPO to the Servers OU and link it to the SQL Servers OU. When you link the GPO to the SQL Servers OU directly, the inheritance blocking isn't applicable to it so the GPO is applied. And this method is a bit simpler than using GPO enforcement. It is common to link a GPO to multiple locations to simplify your environment. And, linking a GPO to multiple locations is a good way to get around inheritance blocking.

## What to check when GPO settings aren't applying as you intended

Occasionally, you will have a scenario where you have multiple GPOs linked to your server OUs but the settings aren't applying as you intend. Let's look at some of the configuration items that can help you pinpoint the cause. Note that these configuration items below aren't listed in a specific order. Often, in your troubleshooting, you need to gather information and check multiple configuration items before you can track down the problem. But the good news is that you can check these items relatively quickly.

- **Security filtering.** Earlier in this chapter, we mentioned that all GPOs start off with a security filter specifying that the Authenticated Users group can read and apply all new GPOs. When a GPO isn't doing what you think it should be, you should check the security filter to ensure that the group(s) being used include the target computers for the GPO.
- **Link order.** When a computer has multiple GPOs to process, it processes them in a specific order. That order is known as the link order. Sounds simple, right? Except there are two link orders! One link order handles all the GPOs linked to an OU. The GPO that is processed last "wins". By that, I mean that the settings from that GPO take precedence over the same settings from any other GPOs. Now, let's quickly cover the second link order – this is the order that dictates the order that GPOs are applied when they are linked at various levels. The default processing order is:
  - local GPOs apply first
  - GPOs linked to a site are processed second
  - GPOs linked to the domain are processed third
  - GPOs linked to parent OUs are processed fourth
  - GPOs linked to child OUs are processed last (and thus take precedence)
- **The default processing order has an acronym to make it easier to remember: LSDOU (local/site/domain/OU).** If multiple GPOs are linked to the same OU, the GPOs are processed based on their link order. While there are some exceptions to all of this, the exceptions aren't important now. For now, try to get comfortable with the link order and processing order. When a GPO isn't applying, check the link order and see if it is the cause of the problem.
- **Settings must be enabled.** Each GPO has user settings and computer settings, which we introduced earlier in this chapter. You can disable the user settings, the computer settings, or both computer and user settings. For example, if you don't have any user settings in a GPO, you can disable the user settings. You don't gain much from doing so, although there is possibly a tiny performance improvement (although there is some debate as to whether that exists). But, if you have a GPO that doesn't seem to be applying as you think it should, check to see if the user or computer settings are disabled – if so, you may be able to fix the problem by enabling the settings.

## How to minimize the need for inheritance blocking

While you know how to use inheritance blocking and some techniques to bypass it, you are better off avoiding it altogether. A simple environment and configuration is the easiest to implement and maintain! When you have a Servers OU as a parent OU for all server related OUs, do not store any computer objects in the parent OU. When you do, it leads to an occasional need to use inheritance blocking and or enforcement, as we saw in the earlier example.

Now, let's look what happens when the parent OU only contains child OUs. When it comes time to link a GPO to all servers except computers in the SQL Servers OU, you don't need inheritance blocking. Instead, you can link the GPO to the child OUs directly (all of them except the SQL Servers OU). In such a scenario, your settings apply to all servers except those in the SQL Servers OU and you didn't need to use inheritance blocking and GPO enforcement. This shows you the importance of a proper OU design, one that simplifies your GPO implementation.

### Hands-on Exercise

Open the GPMC. Create a test GPO named Test3 and a test GPO named Test4. Create a test OU named TestOU2. Then, link both GPOs to the TestOU2 OU. In the GPMC, expand the TestOU2 OU in the left pane. In the right pane, look at the Linked Group Policy Objects tab and review the link order. Change the link order so that the GPO in link order #2 is in link order #1. Remember, the GPO with the lowest link order (#1) is processed last and thus takes precedence.

In this chapter, we introduced you to Group Policy and covered quite a bit of content such as GPO policies vs. GPO preferences, link order, inheritance blocking and enforcing, and troubleshooting GPOs. These tasks are what we often refer to as the core concepts of Group Policy. You now know how to work with Group Policy as well as manage and troubleshoot your existing GPOs. All this information helps prepare you for our next chapter which looks at securing Windows servers with Group Policy. But before we go there, let's test your knowledge in the lab. If you haven't already performed the Hands-on Exercises, you should do so now before you start the lab tasks.

## Lab

### Create and link a GPO by using the GUI

Perform the following tasks:

- Create a GPO named Test10.
- Link the GPO to the domain level.

### Create and link a GPO by using PowerShell

Perform the following tasks:

- Link the Test10 GPO to a test OU by using PowerShell.

### **Block policy inheritance on a child OU**

Perform the following tasks:

- Create a child OU under an existing OU, if necessary. Otherwise, use an existing child OU.
- Block policy inheritance for the child OU.

### **Enforce a GPO link**

Perform the following tasks:

- Enforce the Test10 GPO link to your test OU.

### **Disable the user configuration settings for a GPO**

Perform the following tasks:

- Disable the user configuration settings for the Test10 GPO.

## CHAPTER 18: SECURING WINDOWS SERVER WITH GROUP POLICY

---

Last chapter, we looked at Group Policy, the built-in configuration management feature of Active Directory and the Windows operating system. You learned that you can use Group Policy to configure your domain-joined computers and saw the power and complexity of Group Policy, such as its ability to enforce specific settings on every computer in your organization.

In this chapter, we are going to expand your Group Policy knowledge. We are going to focus on securing your Windows servers with Group Policy. We will mainly focus on the core security settings that are commonly used. We won't look at every policy and every option. There are thousands of them in Group Policy so we won't have time to do that! If you are interested in exploring further after this chapter, see <https://technet.microsoft.com/en-us/windowsserver/bb310732.aspx> for a good starting point.

First up in this chapter, we are going to define security settings which can help you enhance the security of your server environment. From there, we will show you password and account lockout policies which ensure that your user account passwords meet or exceed your password requirements and control what happens if too many bad passwords are attempted for a user account. Then, we'll look at specific user rights that you can enable or deny which can help you better control which tasks user accounts can perform in your domain. Thereafter, we look at auditing policies. If a user attempts to perform a change that is unauthorized or somebody tries to sign in as somebody else, how do you find out about it? Auditing! To wrap up the chapter, we will look at how you can use Group Policy to control the behavior of the Windows firewall which you can use to open or close different types of network communication in your environment. This is starting to feel like a jam-packed chapter, isn't it?! Then, at the very end, we will have a lab to test your newfound security skills.

### What are security settings?

In the last chapter, we introduced Group Policy and described how it is a built-in configuration management technology that comes with Windows operating systems. Group Policy enables you to configure thousands of settings, as you have now seen. In this chapter, we are going to narrow the focus to just security related settings. But what do we mean by security related settings? Security settings are any settings that help to increase the security of your domain-joined computers. For example, a password complexity setting, available in Group Policy, can help to increase the security of users' passwords. A Group Policy setting to ensure that computers digitally sign network communications can help to increase the security of your environment. But, we need to be realistic. There are way too many security-related settings to cover in a chapter. We would need an entire book to cover them all. So, we are going to focus on some of the most widely used security settings, settings that are often applicable to most environments (and thus, it is likely that you are using some of the settings or will want to use them at your organization).

Let's start off looking at password and account lockout policies, because these policies are widely used and necessary in just about every environment. And because the concepts are straight forward, it will be a good way to introduce security settings.

## Password and Lockout Policies

Imagine that a malicious individual is trying to break into the email account of one of your IT administrators. The malicious individual uses a script to try common passwords at a rapid pace until one of the password attempts works. This process is referred to as brute forcing. If the password is simple, such as a dog's name, the password will likely be compromised with such a script. And beyond just random, over the internet attacks, you need to beware of a compromise of your AD DS database which contains password hashes that are more easily cracked. One compromise of your AD DS database is when a malicious individual maliciously obtains a copy of the AD DS database. How do you protect against such situations? You use password and account lockout policies to reduce the chances of brute forcing and reduce the likelihood of password hashes being cracked. Here's why password policies and account lockout policies can help improve your organization's security:

- **You can use password policies to mandate a long and complex password.** For example, if you wanted to mandate that all users use a 14-character password with a mix of characters (3 of the 4 categories: uppercase, lowercase, number, and symbols), you can do that! Brute forcing a long and complex password becomes much tougher with long and complex passwords (even if the malicious individual has a copy of the AD DS database).
- **You can use account lockout policies to reduce the amount of bad password attempts that are allowed.** Without an account lockout policy, a malicious user can attempt passwords indefinitely. With an account lockout policy, you choose how many bad password attempts are allowed before an account is locked out (and not usable until unlocked). Some administrators choose a small number of attempts, such as 5. After 5 bad attempts, accounts get locked out. Other administrators (including myself), prefer a larger number such as 50. This reduces user induced account lockouts but still provides protection against malicious password guessing.

OK, so password policies and account lockout policies are effective and you should use them. But how? That's what we'll look at now. There are two methods to configure password and account lockout policies in your domain:

- **Standard password and account lockout policies.** These policies are the original policies available in Active Directory since its inception. With these policies, you get one password policy that applies to all user accounts and one lockout policy that applies to all user accounts. These are a bit limiting. For example, I think user accounts with access to sensitive or confidential data should have stronger passwords than user accounts that have very limited access. But with a standard password policy, I can't do that.
- **Fine-grained password and account lockout policies.** These policies have been available as an optional feature since Windows Server 2008. You can create as many policies as you need and apply them based on security groups. So, if I want to ensure that the IT

administrators have a strong password policy, I can create one just for them and apply it to a group that the IT administrators are a member of.

Most of the time, you'll choose to use one method or the other. For example, you might have a standard password and account lockout policy for your users that do not work with sensitive data or have administrative access to your computing infrastructure while using fine-grained password and account lockout policies for your users that work with sensitive data and manage your computing infrastructure.

## Standard password and account lockout policies

Standard password and account lockout policies are configured in a GPO. The Default Domain Policy is the most often used GPO for the policies. The password settings are in the Computer Configuration/Policies/Windows Settings/Security Settings/Password Policy node. The account lockout settings are in the Computer Configuration/Policies/Windows Settings/Security Settings/Account Lockout Policy node.

### Password settings

Figure 18.1 shows the password settings available in a GPO. Normally, the GPO is the Default Domain Policy or a custom policy. Either way, the GPO must be linked at the domain level to function properly. In the screen capture, I show the settings configured with commonly used policy settings.

Policy	Policy Setting
Enforce password history	10 passwords remembered
Maximum password age	120 days
Minimum password age	12 days
Minimum password length	14 characters
Password must meet complexity requirements	Enabled
Store passwords using reversible encryption	Disabled

Figure 18.1 There are 6 password settings available in a GPO.

Let's look at the use of each password setting to see when and how you use it:

- **Enforce password history.** When enabled, you can ensure that users cannot use passwords maintained in the password history. In Figure 18.1, the setting is configured for 10 passwords which means when a user changes their password, they cannot use any of the previous 10 passwords.
- **Maximum password age.** This setting, configured for 120 days in Figure 18.1, is the maximum amount of time that a user can have the same password. Once the maximum password age has been reached, a user must change their password.
- **Minimum password age.** This one, at first glance, seems counterintuitive. After all, as the administrator, you do want to ensure users are changing their passwords. But, imagine a scenario where a user decides to changes his password 11 times in a row, all within a few minutes. He could do that to bypass the password history and then end up "changing" his

password back to his original password! Minimum password age makes that much tougher (or impossible, depending on the setting). In Figure 18.1, the minimum password age is set to 12 days. Thus, to bypass the password history, a user would have to change their password every 12 days for 120 days! But be aware, a minimum password age also impacts the ability for a user to immediately change their own password after a password reset by an administrator. So, there is a balance to think about. In this example, a minimum password age of 12 is high! In most environments, a minimum age of 1 or 2 days is sufficient. You don't want users to bypass password change requirements but you also don't want administrators to know user passwords! It is a good practice to have administrators that reset passwords to not keep track of the passwords, whether in a secure file or database or just on paper.

- **Minimum password length.** This setting sets the minimum length allowed for passwords. The longest minimum length supported in a standard password policy is 14 characters. Longer passwords are more secure than shorter passwords. But, as the administrator, you need to balance security with the user experience. In my experience, I've found success using an 8-character minimum for the general user population and a 15-character minimum for high-security users (IT administrators and other employees with access to sensitive or confidential data). With service accounts, I generally opt to match them with high-security users. To mandate more than a 14-character minimum length, you must use fine-grained password policies, which we look at next.
- **Password must meet complexity requirements.** This setting can be disabled or enabled. When disabled, users can use any password if the minimum length, password history, and password age meet the policies. When enabled, users must use characters from 3 of the 4 character categories: lower case characters, upper case characters, numbers, and special characters (such as #, &, and \*). Additionally, the password cannot match the user's account name or more than two consecutive characters of the user's full name. I recommend enabling password complexity for all environments.
- **Store passwords using reversible encryption.** This setting, when enabled, greatly reduces the security of the stored password. This setting was sometimes required for legacy applications. I recommend disabling this setting in all environments unless absolutely required.

### Hands-on Exercise

Open GPMC. Navigate to the Default Domain Policy and then edit it. Set a password policy so that all passwords meet the following requirements: 5 passwords are remembered, the maximum password age is 365 days, the minimum password age is 1 day, the minimum password length is 14 characters, password complexity is required, and passwords are not stored using reversible encryption.

## Account lockout settings

Let's now look at account lockout settings. Account lockout settings go hand in hand with password policies. They complement each other. Figure 18.2 shows the available account lockout policies in a GPO. Like password settings, these settings are often in the Default Domain Policy or another GPO. In either scenario, the account lockout policies must be linked at the domain level to be effective.

Policy	Policy Setting
Account lockout duration	0
Account lockout threshold	999 invalid logon attempts
Reset account lockout counter after	30 minutes

Figure 18.2 There are 3 account lockout policy settings available in a GPO.

Let's see how you can use these settings.

- **Account lockout duration.** When an account gets locked out, you have a choice. Do you want the account to automatically unlock? Or, do you want to require an administrator to deliberately unlock the account? To reduce administrative overhead, you should automatically unlock accounts by using the account lockout duration setting. For example, to automatically unlock locked accounts after 30 minutes, set the account lockout duration to 30. But, in high-security environments (or with the highly secure user accounts), require an administrator to unlock user accounts. To require an administrator to unlock a locked account, set the account lockout duration value to 0.
- **Account lock threshold.** The threshold setting dictates how many times a user can enter a bad password before their account gets locked out. It is generally a good idea to use a threshold to prevent unlimited bad password attempts. This reduces the chances of an automated process attempting to brute force a password indefinitely. However, if you use a very low threshold, such as 3 password attempts, you may find that users are routinely being locked out and the administrative overhead for IT increases. So, you need to balance security with user experience and productivity. I've found that in most environments, 10 invalid logon attempts is a good balance. In other environments, I've proposed 999 invalid logon attempts. The pros of using 999 is that accounts rarely, if ever, get locked out. That reduces administrative overhead and reduces end user downtime since their accounts rarely get locked out. But the downside is that an automated process or a malicious person can try 999 password attempts without interruption (account lockout). Thus, to use a large threshold such as 999, you need to have long and complex password requirements so you can be confident that a password won't be compromised within 999 password attempts. 999 invalid logon attempts is a bit progressive for most environments because administrators are used to using 5 or 10 invalid logon attempts and 999 is quite a bit higher. You should have monitoring in place to detect many bad password attempts, especially when you have a large account lock threshold.
- **Reset account lockout counter after.** This setting dictates how long bad password attempts are remembered. In Figure 18.2, the setting is set for 30 minutes. That means if I type 1 bad password in, that will be remembered for 30 minutes. After that, the account

lockout count resets to 0. 30 minutes is a very typical setting which balances security with functionality. If you go with a very low number of minutes, such as 2 minutes, a person can try a password attempt every 2 minutes and it doesn't increment the lockout counter. Thus, they could target a user account and try 720 passwords a day without the account being locked out. On the other hand, if you go with a very high number, such as 1440 minutes, users could get locked out based on bad password attempts from a few days ago. And that can be confusing and frustrating for users.

### Hands-on Exercise

Open GPMC. Navigate to the Default Domain Policy and then edit it. Set an account lockout policy so that the user accounts are locked out for 30 minutes, 10 bad password attempts can be attempted before lockout, and account lockout counters are reset after 30 minutes.

### Fine-grained password and account lockout policies

Now that you know about the standard password and account locked policy settings, let's look at the second method – fine-grained password policies. But before we dive in, I want you to know that fine-grained password and account lockout policies are not GPOs. Instead, they are policies that you create and associate with users and/or groups. I've found that fine-grained password policies are very valuable and I try to use them in every organization. The information we've gone over so far on password and account lockout policies applies to fine-grained password policies too. Fine-grained password policies can do everything that the standard policies can do, but more. They work a little bit differently in the way that you configure them and use them. First, password and account lockout settings are in a single fine-grained password policy. In Figure 18.3, there is a policy named IT High-Security Password Policy.

Figure 18.3 A fine-grained password policy has password and account lockout settings in a single policy.

As you can see in Figure 18.3, we have many of the same settings we discussed earlier such as the minimum password length and the password history. So, I won't repeat information about those settings here. But I do want to point out some key differences:

- **There is a policy precedence.** In our example in Figure 18.3, the precedence is set to 10. The precedence is only applicable if more than one policy applies to a user. In such a case, the policy with the lowest precedence wins (and its settings are applied to the user).
- **There is a place to apply the policy to security groups or users.** Unlike GPOs, which can be linked to domains, sites, or OUs, fine-grained password policies are applied to security groups or users. To keep things simple and easy to manage, you should try to apply fine-grained password policies to groups, not users. Then, you must add somebody to the group to have the fine-grained password policy apply.
- **Some of the values allowed for the settings are higher in a fine-grained password policy** than in a standard password policy. For example, the maximum length you can use for the minimum password length is 14 characters when you use a standard password policy. However, in a fine-grained password policy, you can set the minimum password length much higher – up to 255 characters! Some of the other settings also allow a much larger value, such as the password history and account lockout threshold.

I think fine-grained password policies are the best choice for most organizations. Because, they allow you to enforce much stronger password settings and set password and account lockout policies based on the user account type or security level (which means you can mandate the strongest password and account lockout settings for your highest security user accounts while not degrading the user experience for users that don't warrant the strongest password and account lockout settings). If you don't use fine-grained password policies, then the Default Domain Policy's password and account lockout settings will apply. Because of this, it is common to maintain a password and account lockout policy in the Default Domain Policy to apply to users that are not being handled by a fine-grained password policy.

### Hands-on Exercise

Open Active Directory Administrative Center. In the left pane, click the domain. In the middle pane, double-click the System container. Then, in the middle pane, double-click the Password Settings Container. In the right pane, click New and then click Password Settings. From there, create a new policy to meet the following requirements: minimum password length of 20 characters, 10 passwords remembered, minimum password age of 5 days, maximum password age of 365 days, account lockout policy such that accounts are locked out after 10 attempts and an administrator must unlock accounts if they get locked out. Apply the policy to any existing security group, name the policy "IT Password Policy" and set the precedence value to 10.

When thinking about security for your environments and your servers, password policies and account lockout policies are one aspect. But we need to look beyond those to explore other areas that are important for securing your servers. Next, we'll look at some user rights that can be assigned by using a GPO.

## User Rights Assignment

There is a section in a GPO for user rights assignments and it is named "User Rights Assignment"! ". In total, there are 44 available settings. So, we won't have room to cover all of them here but I want to look at some of the most used settings here. To see more information about each of the settings, see <https://technet.microsoft.com/en-us/library/dn221963.aspx>. To get to the settings in a GPO, navigate to Computer Configuration/Policies/Windows Settings/Security Settings/Lock Policies, and then look at User Rights Assignment. Figure 18.4 shows a snippet of a GPO with some of the rights shown.

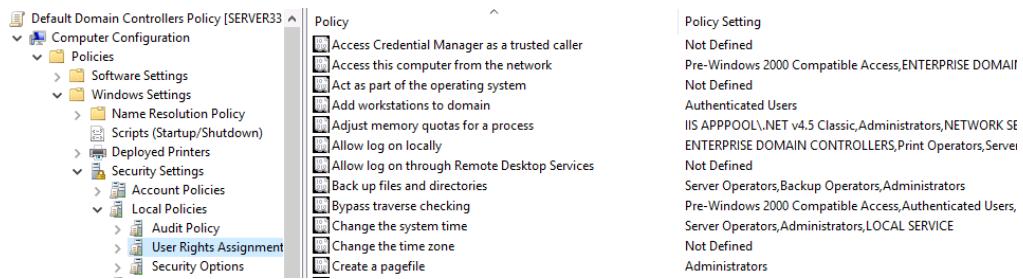


Figure 18.4 A GPO shows some of the User Rights Assignment settings available.

The following bullet points outline some of the most used and important settings under the User Rights Assignment node:

- **Act as part of the operating system.** This optional setting enables any running process to impersonate any user without authentication. In organizations that have legacy operating systems, this was occasionally needed. However, with the introduction of the LocalSystem account (which can access the same local resources as any user), there is no longer a need to use this optional setting. It may still be found out there because administrators may have kept it in place after upgrading to newer operating systems. You should opt to not use this setting because it introduces additional security risks.
- **Add workstations to the domain.** Did you know that, by default, any AD DS user can add computers to the AD DS domain? Not many administrators do. The default setting allows for up to 10 domain joins per user. Many organizations prefer to maintain control over which systems can join the domain and what the system requirements are (such as having up-to-date anti-virus and other security software installed). It is a good idea to adjust this setting so that only IT administrators that are responsible for adding computers to the domain can add computers to the domain.
- **Allow log on through Remote Desktop Services.** Remote Desktop Services is a key remote connectivity tool. As such, access to your domain-joined computers via Remote Desktop Services must be controlled. By default, only Administrators and members of the Remote Desktop Users group can use Remote Desktop Services to access client computers and

servers. But, you can change the default based on your requirements by modifying this GPO setting. As part of an audit, this setting is a good one to check to ensure that the right people have access.

- **Debug programs.** By default, only Administrators can debug programs. The right to debug programs opens potential vulnerabilities because malicious users can inject malware into a server or gain access to sensitive areas of the operating system. In high-security environments, you should remove everybody from this setting and enable the debugging of programs on an "as needed" basis. At a minimum, you should ensure that only Administrators can debug programs so that you do not expose your organization to unnecessary risk. Note that this setting is applicable to debugging Windows system components, not applications that they wrote.
- **Log on as a service.** By default, any user account can be used to run a service. When a user is specified to run a service, you can grant them the ability to log on as a service to a specific server. It is a good practice to only allow service accounts to log on as a service. This prevents user accounts from being used for running services, which can sometimes lead to issues. For example, if the user changes their password and forgets about a service that is using their account, the service may stop running or not start upon next boot. In addition, if the user leaves the company or is terminated from the company, the user account may be deleted and the service will stop. You can create a security group named "Service Accounts", create a GPO (or modify an existing GPO), and give the group rights to log on as a service. Then, whenever you create a service account, you can add it to this group. This helps to minimize the use of non-service accounts for running services. Malicious software packages sometimes try to add a new service to a compromised computer. The new service sometimes run as a user. This setting, when used as described, can help reduce and/or prevent such a situation.

A key principle in computing is to give people the minimum amount of permissions necessary to perform their jobs. This principle is known as the "principle of least privilege" and it is a cornerstone of an effective security strategy. As you look through the available user rights under the User Rights Assignment node, think about the principle of least privilege when figuring out whether a setting is needed. Aren't sure what a setting is? Double-click it, then look at the Explain tab. The Explain tab summarizes what the setting is, what the default setting is, and often calls out when a setting can be a security risk. Don't worry about memorizing these settings (or other GPO settings). In time, you'll become familiar with the most important settings. Additionally, you can look up information on the internet to learn about the rest of the settings.

### **Hands-on Exercise**

Open the Group Policy Management Console. Edit the Default Domain Policy. Navigate to the user Rights Assignment node. Modify the settings to meet the following requirements: only Domain Admins can add computers to the domain, nobody can debug programs, and only members of the Administrators group can log on through Remote Desktop Services.

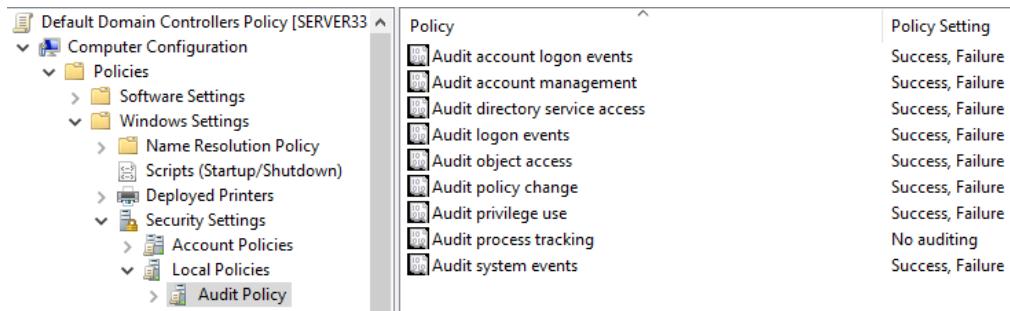
So far, we've looked at password policies, account lockout policies, and user rights. And we've only touched on some of the key points about each. But we still have more to go. Next up is all about auditing. After that, we'll dive into managing the Windows Firewall. As you can see, security related settings cover quite a bit of the operating system!

### **Auditing Policies**

When people think about security, they often think about the specific strategy and tactics to keep the bad guys out. Or, they think about how to limit the damage that bad guys can do if they get into a network. But there is another area that sometimes goes unnoticed or doesn't get the attention it deserves. How do you find out if a bad guy got in? Or how do you find out what damage was done in such a situation? That's what we're here to talk about in this section. Auditing. Auditing is all about recording the activities of users and computers so that you can create an electronic trail. The goal is to figure out when something first began, how long it lasted, and what transpired. By doing so, you can figure out a plan of attack to fix the damage, close any open security holes, and train the staff to avoid the same type of attack in the future. Auditing is nice to have in the event of a security incident but it is also nice to have for day-to-day administration. If one of your servers has a major issue, you can use auditing to help find out what happened just prior to the issue. If a file is accidentally deleted, you can track down who deleted it and when they deleted it.

You can configure auditing in two ways in a GPO. The old way, which is often referred to as standard (or legacy) auditing policies, enables you to audit your servers but doesn't give you granular control over the information that you capture. Instead, you have 9 categories and you enable them for Success or Failure (or without auditing). The new way, which is referred to as advanced audit policies, enables you to audit your servers with granular control. With granular control, you can opt to capture just what you need which can be time-saving when it comes to reviewing your audit logs. For example, instead of auditing all the account management events, you can audit just parts of it. No matter which way you choose, all the auditing that you enable goes to your Security event log.

Figure 18.5 shows the available standard audit settings in a GPO.

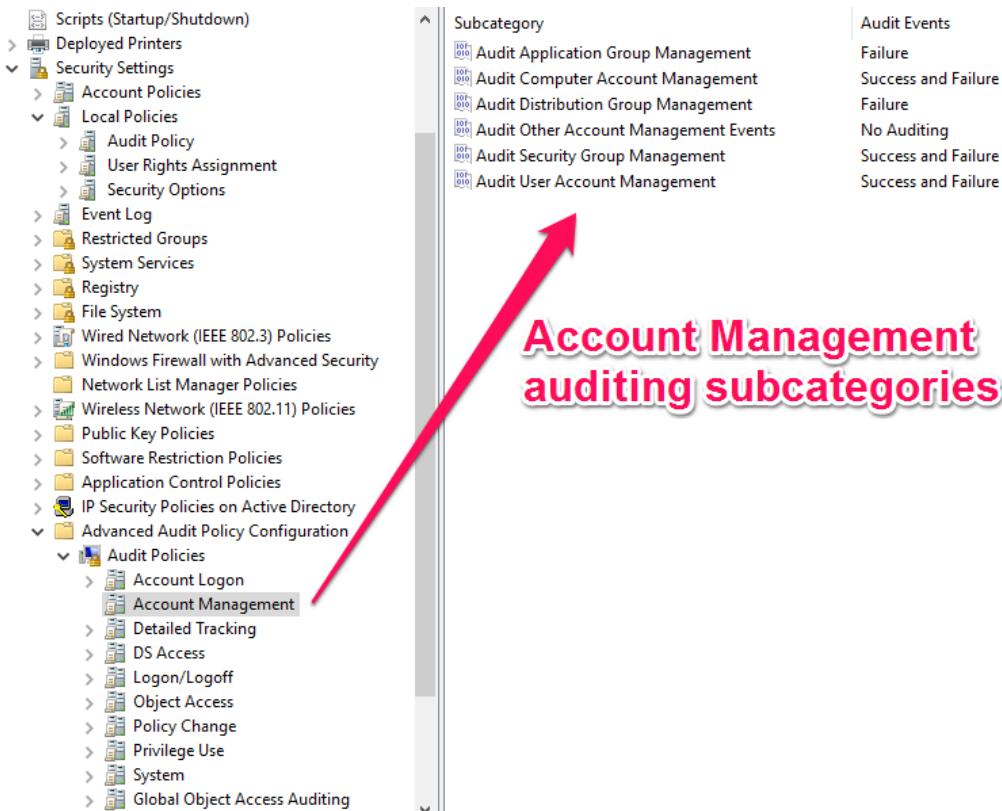


The screenshot shows the 'Default Domain Controllers Policy [SERVER33]' in the left pane. Under 'Computer Configuration' > 'Policies' > 'Windows Settings' > 'Security Settings', the 'Audit Policy' node is selected. To the right, a table lists 9 standard audit policies with their policy settings:

Policy	Policy Setting
Audit account logon events	Success, Failure
Audit account management	Success, Failure
Audit directory service access	Success, Failure
Audit logon events	Success, Failure
Audit object access	Success, Failure
Audit policy change	Success, Failure
Audit privilege use	Success, Failure
Audit process tracking	No auditing
Audit system events	Success, Failure

Figure 18.5 There are 9 categories of standard auditing available in a GPO.

Figure 18.6 shows the advanced audit policy settings in a GPO and how the auditing categories are divided into subcategories.



The screenshot shows the 'Default Domain Controllers Policy [SERVER33]' in the left pane. Under 'Computer Configuration' > 'Policies' > 'Local Policies' > 'Audit Policy', the 'Account Management' subcategory is highlighted. A red arrow points from this subcategory to the right pane, which displays its subcategories:

Subcategory	Audit Events
Audit Application Group Management	Failure
Audit Computer Account Management	Success and Failure
Audit Distribution Group Management	Failure
Audit Other Account Management Events	No Auditing
Audit Security Group Management	Success and Failure
Audit User Account Management	Success and Failure

**Account Management auditing subcategories**

Figure 18.6 Advanced audit policy settings have categories that are divided into subcategories, enabling you to granularly select what you want to audit.

As you might be able to tell based on the names of the audit categories and subcategories, auditing captures information about user and computer activity. The following activities are some of the key activities that you can log by using auditing:

- **User sign in and sign out.** This setting, named Audit Logon, is in the Logon/Logoff subcategory. Auditing will capture when a user signs in, the computer being used, and the date and time stamp. Same with sign out information. This information is helpful to

help you track down the source of account lockouts, find unauthorized sign in activity, and helps you narrow down the source of configuration changes.

- **Creation and deletion of objects in AD DS.** This setting, named Audit Directory Services Changes, is in the DS Access subcategory. Imagine that a service account is deleted in AD DS. Besides restoring the service account, you need to find out what happened. Who did it? When did it happen? This info is important for examining the root cause and trying to minimize the recurrence of the issue. Beyond just creation and deletion, you can also capture changes to attributes in many cases.
- **Successful access to specific files.** This setting, named Audit File System, is in the Object Access subcategory. Confidential information about an upcoming product launch was accidentally copied to a shared folder that everybody at your company has access to. However, the information was only meant for a specific group of people because the product launch was a secret. To find out if the information was seen by others, you can use auditing and look at all successful accesses to the file(s).
- **Unsuccessful access attempts to specific files.** This setting, named Audit File System, is in the Object Access subcategory. Besides just finding out when somebody accesses a file, you can also find out when somebody tries to access a file but is denied due to lack of access. This is important because it could be indicative of somebody attempting to look at information that they are not authorized to look at.
- **A computer was shut down or started.** This setting, named Audit Security State Change, is in the System subcategory. With auditing, you can look at the date and time of each computer shutdown or started up. You can also find out which user or process initiated the action. While many shutdowns or startups are routine, you should investigate unexpected shutdowns and startups because they could be indicative of malicious activity or unauthorized activity.

Auditing tracks a large amount of activity. If you audit several categories or everything, you will capture a massive amount of data. As such, as part of your auditing strategy, you need to ensure that your event log sizes are configured to hold the extra information. For domain controllers and servers, I often opt for 4 GB log sizes for the Security event log, 256MB for the Application event log, and 512MB for the System event log. This enables me to go back for at least an entire day in the Security event log in a busy environment (and several days in smaller and less busy environments). In addition, you should consider an event log archiving solution so that once an event log fills up, the log is archived and a new log is started. Without such a solution, once your log fills up, each new log entry will be written while the oldest log entry will be deleted. There are third-party solutions that offer many features. But, Windows Server has a method for archiving too! It isn't feature-packed but it can still be quite helpful. We won't cover the intricacies here but if you are interested in exploring it, there are two Group Policy settings that you should look at: "Back up log automatically when full" (Computer Configuration/Policies/Administrative Templates, Windows Components, Event Log Service – then see the individual event log folders) and "Retain <log name> log" (Computer Configuration/Policies/Windows Settings/Security Settings/Event Log). You need to use both settings to set up archiving.

Configuring auditing is a straight forward process. Edit a GPO, navigate to the auditing policy setting, and then edit the setting. You can configure a setting for no auditing, for success auditing, for failure auditing, or for both success and failure auditing.

### **Hands-on Exercise**

Open the GPMC. Create a test GPO named Auditing. Edit the GPO. Use advanced audit policies to enable the following auditing policies: Audit Logoff and Audit Logon under Logon/Logoff (Success and Failure), Audit Security State Change under System (Success only), and Audit Process Creation under Detailed Tracking (Success only). Link the GPO to your domain. Then, restart a domain-joined computer, sign in, and open Notepad. Then, restart the computer again. Then, review the computer's Security event log to review the event log entries associated with the audited activities.

Now you have a good overview of auditing. It isn't a complex topic. But, it is an important topic. Most of what you need to know and learn from here is about the individual audit settings so you can figure out exactly which ones to use based on your organization's requirements. Part of learning that is trial and error. Next, we are going to look at using Group Policy to manage the Windows firewall.

## **Managing the Windows Firewall**

When you think about the security settings for your organization's computer infrastructure, you should be thinking about ways to manage and control the settings. This enables you to have a consistent and predictable configuration across your environment. With computer settings, you can also prevent users from unknowingly reconfiguring computers to decrease security. The Windows firewall is one of the key security features of the Windows operating system. Group Policy can manage the Windows firewall to help you configure and control it. There are two primary objectives that Group Policy can help with in regards to the Windows firewall:

- **Manage the firewall service and service settings.** At a high level, you can enable the firewall and turn it on for all network profiles or just some network profiles. A network profile is a profile that Windows uses to configure your firewall. There is a profile for while your computer is connected to a domain, a public profile for while you are at a public place such as an airport, and a private profile for when you are on a private network such as your home network. You can also configure whether notifications are sent to a user when communication is blocked. You can configure computers to log dropped packets and log success connections. You can also configure IPsec settings (IPsec is a technology to help sign and encrypt network communication). I prefer to run the Windows firewall on all computers and have it protect all network profiles. However, this often results in a little more administrative overhead for setup and day-to-day management.

- **Manage the firewall rules.** The Windows firewall, like virtually all other firewalls, has rules to control which communication is allowed and which communication is blocked. For example, you can have a rule that enables a web server to receive inbound communication for HTTP so that visitors can reach a web site. The Windows firewall separates firewall rules into inbound rules (for incoming communication from a different computer) and outbound rules (for communication originating from the computer but going to a different computer). You can create individual rules that are then pushed out to all computers that are applying the GPO that contains the firewall settings.

By default, the Windows firewall will enable outbound communication. For example, if you enable the firewall and then try to go to a web site on the internet, it will function. On the other hand, the Windows firewall blocks inbound communication by default. For example, if you install an application on a server that has the Windows firewall enabled, other computers won't be able to initiate communication to the application because the Windows firewall will block it by default. Some operating system components and third-party applications will detect that the Windows firewall is enabled and create firewall rules during the installation! This saves you the time to create the needed rules.

So how do you configure the firewall settings? Let's cover the two areas – service and service settings and firewall rules:

- **Service and service settings.** Navigate to Computer Configuration/Policies/Windows Settings/Security Settings/Windows Firewall with Advanced Security and expand it. On the node, right-click on **Windows Firewall with Advanced Security** and then click **Properties**. There are 4 tabs available. 1 tab for each network profile and 1 tab for IPsec settings. You can click through the tabs and use the dropdown menus to configure the desired settings. For example, to enable the firewall for the Private Profile, click the **Private Profile** tab, click the **Firewall state** dropdown menu, and then click **On (Recommended)**. When finished, click **OK**.
- **Firewall rules.** To configure firewall rules, right-click **Inbound Rules** or **Outbound Rules** in the left pane and then click **New Rule**. The wizard begins and walks you through the process. For example, to create a rule to allow file and printer sharing, begin the wizard, click the **Predefined** rule type, click the **File and Printer Sharing** predefined rule, click **Next**, click **Next**, and then click **Finish**.

The Windows firewall also has some advanced capabilities, such as domain isolation. With domain isolation, you can configure computers so that they can only communicate with other domain-joined computers. Another advanced capability is enabling a feature that allows local administrative users to add firewall rules which end up merging with any rules that you have configured in a GPO. We won't get into the advanced capabilities in this book because it is important to learn and become comfortable with the base functionality (enabling the firewall and configuring firewall rules) first. But I wanted you to realize that those advanced capabilities are there, and maybe you can look sometime in the future.

## Hands-on Exercise

Open the GPMC. Create a test GPO named Server Firewall Policy. Edit the GPO. Navigate to the firewall settings area of the GPO and then configure the GPO to meet the following requirements: enable the firewall for all profiles, display a notification to users if inbound communication is blocked, and enable a predefined rule to allow Remote Desktop communication.

In this section, we looked at how to manage the Windows firewall with Group Policy. You should now know how to enable the firewall and configure firewall rules. And, you have a pretty good idea about the capabilities of the Windows firewall.

In this chapter, we showed you how to use Group Policy to secure Windows servers. We touched on password policies, account lockout policies, user rights, auditing, and the Windows firewall. These topics represent some of the most used security settings in Group Policy. But, we barely scratched the surface of what's out there. And that was the intention. Introduce you to some common security settings, get you comfortable with them, and then enable you to explore. Security is an ongoing endeavor, a never-ending philosophy that you must think about every time you work on a project. Now, let's see if you can put your newfound security knowledge to work by going through the lab.

## Lab

### Configure a fine-grained password policy

Perform the following tasks:

- Create a new security group in AD DS named IT.
- Create a fine-grained password policy to meet the following requirements:
  - Name: IT Password Policy
  - Precedence: 10
  - Minimum password length: 15
  - Minimum password age: 1 day
  - Maximum password age: 180 days
  - Number of passwords remembered: 10
  - Password complexity: Yes
  - Enforce account lockout: Yes
  - Number of failed logon attempts: 10
  - Lockout duration: Until an administrator unlocks

- Directly applies to: IT (the security group)

## Configure user rights

Perform the following tasks:

- Create a new security group in AD DS named Service Accounts.
- Edit the Default Domain Policy GPO to meet the following requirements:
  - Only allow local Administrators to shut down a computer
  - Only allow members of the Service Accounts group the right to log on as a service.
  - Only allow local Administrators to force shutdown from a remote system.

## Configure advanced audit policy settings

Perform the following tasks:

- Edit the Default Domain Controllers Policy GPO.
- Configure advanced audit policy settings to meet the following requirements:
  - Audit directory service changes (success and failure)
  - Audit computer account management (success)
  - Audit user account management (success and failure)

## Configure the Windows firewall

Perform the following tasks:

- Create a new GPO named Web Server Firewall Policy.
- Edit the GPO to meet the following requirements:
  - Enable the Windows firewall for just the Public Profile
  - Block inbound connections
  - Allow outbound connections
  - Create a rule to allow inbound communication for HTTP
  - Create a rule to allow inbound communication for HTTPS

## CHAPTER 19: MANAGING WINDOWS SERVICES

---

In its simplest form, a Windows service is an application that runs in the background. These applications do not usually have a graphical interface, but they do provide core functionality to applications and services on the system. For example, an FTP application might have a service and the service is responsible for network connectivity (opening ports, listening for connections, handling authentication). Services can be configured to start in a variety of ways (when the computer starts up or manually on demand), run under specific conditions (such as only if another service is running), and automatically recover when problems occur (such as restart itself or restart the server).

In this chapter, we are going to look at managing Windows services so that you are comfortable adding services, starting and stopping services, and configuring or troubleshooting services. The order of this chapter is laid out so that we start with the simplest concepts and move toward advanced concepts by the end of the chapter. We will start by looking at the Services management console, the primary graphical interface tool for managing services. Next, we will walk through the process of starting and stopping services, both from the management console and in PowerShell. Some of the most basic management tasks dealing with services will involve checking the status of a service and starting or restarting a service to resolve an issue. To wrap up the chapter we will address how to run a service using a dedicated service account and a managed service account, and why you would want to do so.

### Using the Services Management Console

The Services management console provides you with a graphical interface for managing the services on your Windows servers. You can access the management console through Server Manager by using the Tools menu. Alternatively, you can type services.msc on the Start screen and press Enter. In Figure 19.1, you can see the Services management console. One service, the Print Spooler service, is highlighted.

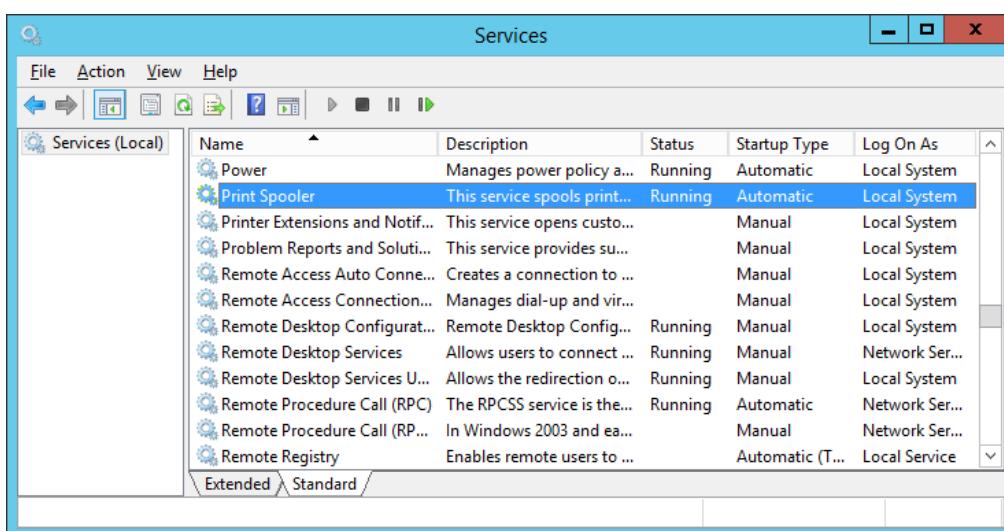


Figure 19.1 Windows services can be managed through the Services management console.

The default view provides you with some basic information about the services installed on your server. These include the following:

- **Name.** This column displays the display name of the service (sometimes known as the "friendly name"). These names are meant to be descriptive and easily identifiable. Services also have a short name, ironically and officially known as the "name". The short name is used when making changes from a command line or from PowerShell. For example, Print Spooler is the display name, and Spooler is the name.
- **Description.** This column displays a description of the service.
- **Status.** This column displays the status of the service. In the management console, services that are stopped will show a blank status. Services can be stopped, started, paused, stopping, or starting.
- **Startup Type.** This column displays the startup type for the service. Services can be configured to start automatically, manually, use a delayed start, or be disabled (won't start at all).
- **Log On As.** This column displays the assigned user account for the service to run as. All services must run using an account with adequate permissions. If a service is instructed to run using an account that does not have permission, the service will fail to start or could fail to operate properly. To run a service with a service account, the service account must have the "Log on as a service" rights, which can be granted in a GPO. The default Windows services are configured to use an account by default (Local Service, Local System, or Network Service). With many third-party services, you will need to specify an account instead. We'll walk through that a bit later in this chapter.

Every service in the management console has a series of properties. Through the service properties, you can configure the attributes of the items listed above. As an example, let's open the Print Spooler properties window and look at where these settings can be adjusted. To access the properties window, double-click on the service in the management console.

In Figure 19.2, you can see the properties window for the Print Spooler service.

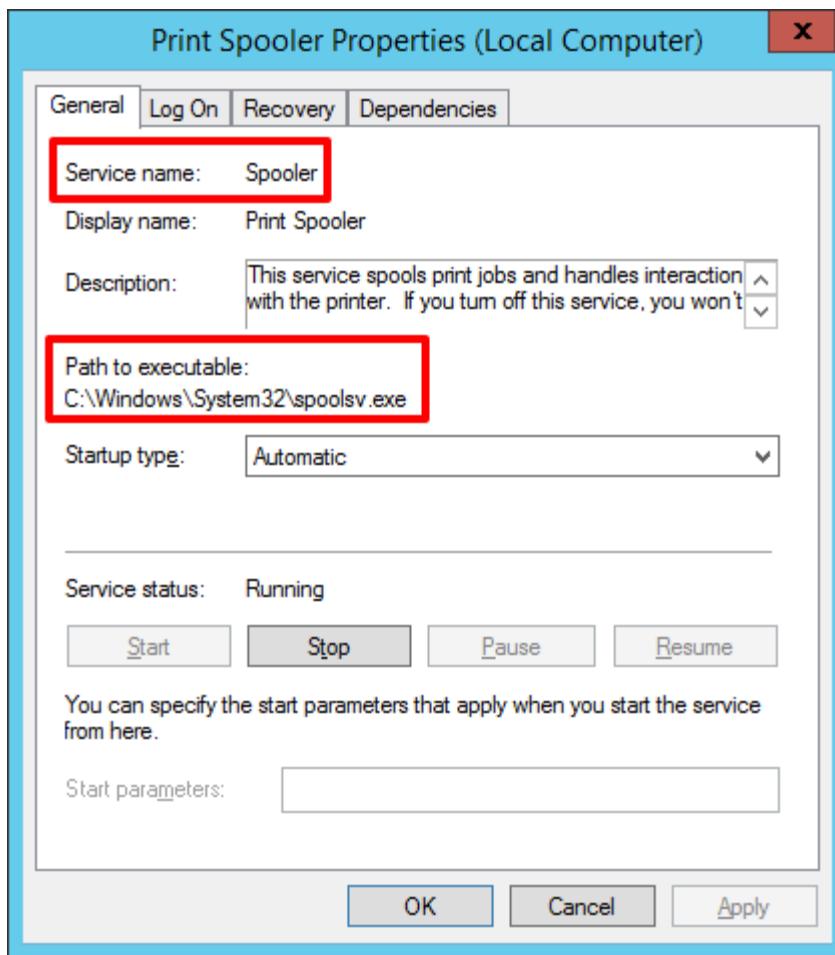


Figure 19.2 Working with the Print Spooler service properties window.

The first tab on the properties window is the General tab. This tab provides you with some basic information about the service. You will already be familiar with the display name and description attributes from the main management console, but take note of the service name (also known as the short name) and the executable path. Remember, the short name is important when you want to manage the service using a command line or PowerShell. The executable path is an important data point in basic troubleshooting scenarios when you need to find out which application is associated with a service (which is mostly helpful when you have multiple third-party services running on a server).

You also can adjust the startup behavior using the Startup type dropdown. There are four startup types:

- **Automatic (Delayed Start).** This startup type configures the service to run shortly after the startup process (after services set to Automatic have already started) to improve logon performance. You usually won't use this setting unless an application calls for it or possibly during troubleshooting.

- **Automatic.** This startup type configures the service to run during the boot and logon process. This is the most common startup type.
- **Manual.** This startup type requires an administrator to manually start the service. You won't use manual often. But, if you were troubleshooting a server and didn't want a service to start up automatically on a reboot, you can set the service to Manual during troubleshooting. When finished, you can set it back to Automatic and start the service.
- **Disabled.** This startup type prevents the service from starting. You use this startup type if a service is causing a problem and you want to ensure that the service doesn't start.

## Startup parameters

In the lower half of the General tab you can adjust the run state of the service, and lastly, you can enter a start parameter(s) for the service. For example, you could have a service that has a configuration file. The default configuration file is used when the service starts without a start parameter. But, you could specify a custom configuration file as the start parameter to troubleshoot an issue or test new functionality. This functionality isn't often used, but it is nice to know it is there.

## Log on for services

The second tab on the Properties window is the Log On tab. This tab gives you the ability to assign a user account for the service to run as. There is an optional setting to enable a service to interface with the desktop. You should avoid enabling that setting because it introduces weakened security on the computer and a malicious user might be able to take over the service without authorization. In an upcoming section, we will take a closer look at assigning a service account to run a service and why it is beneficial.

## Service recovery options

The third tab on the properties window is the Recovery tab. This tab provides you with a few options to help recover a service that has failed to start or stopped unexpectedly after starting. This can help you automate the recovery of a service. For example, you might find a service is sensitive to starting after Windows updates are applied to your server. In this situation, you discover that simply forcing the service to start a second time will resolve the issue. If this is the case, you can navigate to the Recovery tab and instruct the service to restart automatically after the first failure, second failure, or subsequent failures.

You also have the option to initiate a reboot or run a program or script that might help resolve the issue without manual interaction. For example, if a service doesn't start after a couple of tries, instead of restarting the service, you could run a script to alert your monitoring system, send an email to the server administration team, or log an entry to the event log.

## Service dependencies

The last tab on the Properties window is the Dependencies tab. This tab displays the service dependencies for a service. Some services depend on other services to run. For example, the

DNS Client service depends on NetIO Legacy TDI Support Driver service and the Network Store Interface Service services. So, if those two services aren't started, the DNS Client service can't start. The dependencies can be a helpful data point when troubleshooting why a service won't start or to assess which other services might be impacted when a service with dependencies is restarted.

### **Hands-on Exercise**

Open the Service management console. Navigate to the Windows Firewall service. Open the Properties window for the service. Change the startup type for the service to Manual. The default is Automatic. Reboot the server and confirm that the Windows Firewall service did not start automatically.

Now that you know the basics of what a Windows service is and where to manage services, let's look at the process for starting and stopping service.

## **Starting and Stopping Services**

Imagine it's the end of the fiscal year and you are an administrator for Tailspin Toys. Your finance department is working around the clock to close out their files before the end of the day. A trouble ticket arrives in your queue stating that the printer in finance is not working. You open the print queue and see a dozen jobs waiting to be processed. You ping the printer and confirm it is responding on the network. A few moments later you receive another ticket from your HR department. They are also having trouble printing. Where do you begin? Well, one of the first things to check is the Print Spooler service on the print server. You can stop the service and start it. And, you can start it if it isn't running. In general, when troubleshooting, if you find a service that is set to Automatic (whether it has a delayed start) and the service isn't running, start the service. Let's look at how to stop and start services now.

### **Starting and stopping services in the management console**

In the previous section, we looked at the Services management console. You may have noticed a few areas where the run state of a service can be manipulated. Here are a few locations in the management console where a service can be started, stopped, or restarted:

- **Menu bar.** With a service selected, you can manage the run state from the menu bar. There are icons for start, stop, pause, and restart.
- **Action pane.** With a service selected, you can manage the run state from the action pane on the left. Note that you must be on the Extended tab to see the options. The actions will dynamically display based on what the service will allow. For example, a stopped service will only have a link to start the service. A running service will have a link to stop, restart, and pause the service (although pause is only supported for a limited number of services).
- **Right-click menu.** Right-click on any service and the run states will be available in the context menu.

- **Properties window.** Open the Properties window for any service. On the General tab you have access to each of the available run states.

### Hands-on Exercise

Open the Services management console. Locate the Windows Update service.

Using your preferred method walk through the following actions: start the service, restart the service, and stop the service.

Now you know how to work with services from the Services management console. But it is also handy to be able to work with services from PowerShell because PowerShell enables you to automate service management based on events and work with a service across multiple servers. For example, if you need to stop the Windows Update service on 100 servers, you can do that easily with PowerShell but doing so with the Services management console would be very time-consuming. Let's look at how to work with services by using PowerShell now.

### Starting and stopping services with PowerShell

The run state of services can be managed with PowerShell. PowerShell continues to improve automation and is an excellent resource for quickly managing services. PowerShell will output notably more information about the services on your system than the Services management console and enable you to manipulate them in an easier fashion. To manipulate the run state of a service using PowerShell, run the following commands from an elevated PowerShell prompt. Note that the examples below, when referring to a service, are using the Spooler service or the W32time service. You can substitute other service names for the commands instead.

The following command displays the status, name, and display name for all the services on your system:

```
Get-Service
```

The following command displays the status, name, and display name for a specific service:

```
Get-Service -Name 'Spooler'
```

The following command starts the Spooler service if the service isn't running:

```
Start-Service -Name 'Spooler'
```

The following command stops the Spooler service if the service is running:

```
Stop-Service -Name 'Spooler'
```

The following command retrieves a list of all services that are not currently running:

```
Get-Service | where Status -ne 'Running' | Select DisplayName
```

The following command retrieves a list of all services that support pausing and resuming from pause:

```
Get-Service | where CanPauseandContinue
```

The following command stops the W32Time service on all computers in the servers.txt file located in the C:\temp directory. The servers.txt file is a text file with one server hostname per line:

```
Get-Service W32Time -ComputerName (Get-Content C:\temp\servers.txt) | Set-Service -Status 'Stopped'
```

### **Hands-on Exercise**

Open an elevated PowerShell prompt. Stop the W32Time service and then start the W32Time service. Next, create a text file named servers.txt and use PowerShell to stop the W32Time service on multiple servers by using the servers.txt file as part of your command.

As you can see, PowerShell offers some additional power and automation when it comes to managing Windows services. Now that you know how to manage services by using the GUI and PowerShell, it is time to look at service accounts. Custom service accounts, while only used on a small number of services, are an important part of your role managing services.

## **Service Accounts**

A service account is a dedicated user account that you can use to run schedule tasks, scripts, and services.

Earlier in this chapter, we pointed out that the built-in Windows services are configured to use a few different built-in accounts. The following accounts are the built-in local service accounts:

- **Local System.** The local system account is a privileged account with high-level privileges on the local server. It has the equivalent of combining local administrative privileges and NT AUTHORITY\SYSTEM privileges (which has access to most aspects of a server).
- **Local Service.** The local service account is a restricted account that handles processes local to the associated server. It has minimal permissions on the server.
- **Network Service.** The network service account is a restricted account that handles network communication and authentication. It has minimal privileges on the local server.

These built-in local service accounts are designed to support the core functions of the Windows services. But servers are also deployed to run and host applications. Many of the applications that you install add additional services to Windows. Depending on the application, those services may require a dedicated service account for more control of user rights. Microsoft SharePoint and Microsoft SQL Server are two examples of applications that often have dedicated service accounts.

As an administrator, whenever you assign permission to a new user account or service account, you should grant the least amount of access required. This behavior improves the security of your environment. Under this context, Windows services need to be managed in the

same way. When a new application is installed, you should identify the requirements of the associated service(s). Often, the requirements are documented by the vendor. From there you can create a service account, configure the service to use the account, and only grant the access required by the application. You should avoid granting all service accounts local administrative access on servers, by default.

### Choosing which account you should use for a service

So, which type of account should you use? You should avoid using administrative accounts such as Administrator because those accounts often have more rights than needed. Using administrative accounts for services reduces your environment security. You also should avoid using a user's account. For example, you wouldn't want to use your own account to run the MSSQLSERVER service on a critical database server. Guess what happens if you change your password but forget to update the service? The service would fail to start. The best path is to create a dedicated service account.

When creating a service account, you should assign a strong password, especially because the service account will often have elevated access to the application. Also, configure the account so that the password never expires and cannot be changed by the account itself. These adjustments will limit disruption to the service and corresponding application. Don't forget to document the services and servers the service account is configured for!

In Figure 19.3, the MSSQLSERVER service is using the SCCM-SQL-SVC service account. In this example, the service account was assigned during the installation of SQL Server. The installation program prompted for the service account information and then automatically configured the service with the specified service account.

In some cases, after an installation of a third-party application, you might have to manually configure a service with the desired service account. Doing so is straight forward and easy:

1. Navigate to the service and open the Properties window.
2. Go to the **Log On** tab and specify the option to use a specified service account.
3. Enter the service accounts credentials and then click **OK**.
4. To finish, you need to restart the service. Be careful though, because restarting the service will probably make it temporarily unavailable.

A service uses a specified service account.

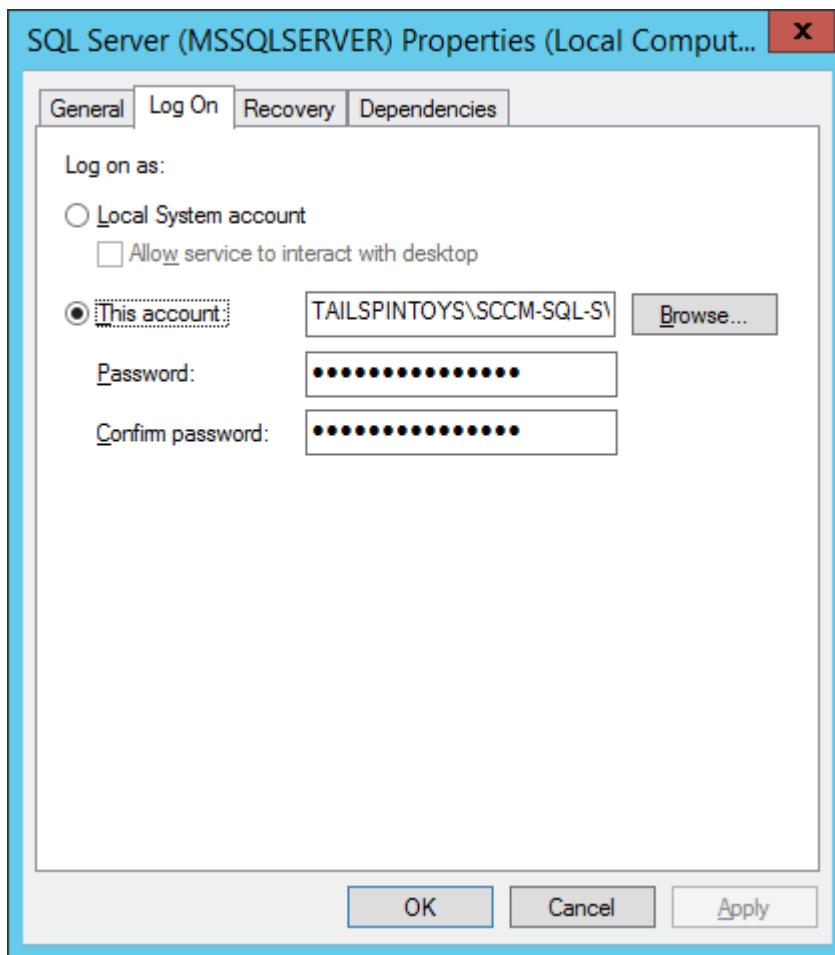


Figure 19.3 The SQL Server service is using a dedicated service account named SCCM-SQL-SVC.

Using a dedicated service account is a relatively common practice. But there are still areas where this solution is difficult to manage. For example, imagine that you are in the process of upgrading an application for your organization. There is a service account associated with the legacy version of the application and you need to use that same service account for the new installation. Unfortunately, you cannot remember the password. In this situation, you might have to reset the password for the service account, which likely means an outage to the legacy application. This becomes a broader issue if the same service account is used for several applications.

Or, imagine that you have 100 database servers. Each uses the SQL service account to run SQL-related services. The administrator that initially configured everything just left the company. The security team wants to reset the SQL service account password since the administrator left the company. To do this, you must reset the password, configure the new password on 100 servers for the service in question, and then restart the service on all 100 servers.

This problem (and others like it) can be solved by using managed service accounts. Because with managed service accounts, you no longer need to know or manage passwords!

## Managed Service Accounts

At this point in the chapter, we have reviewed what a Windows service is, how to manipulate the running configuration of a service, and how to use a service account to run a service.

In this final section, we are going to be discussing managed service accounts (MSAs) and group managed service accounts (gMSAs). They provide service accounts with automated password management which helps to reduce administrative overhead and increase security. The MSA was introduced with the release of Windows Server 2008 R2. The core benefit was to eliminate password management issues among service accounts. While a dedicated service account is a great solution for access management, the accounts are still vulnerable to password attacks and administrative overhead (like any regular user account). In contrast, an MSA password is more secure because the password is not known by anybody (thus a person won't write it down, store it in a spreadsheet, or attempt to manually use it). And an MSA is changed more often than a user's password, at least in most environments.

The following are some of the key benefits of an MSA:

- MSAs are created and stored in Active Directory.
- MSAs are created and managed using PowerShell.
- MSAs can be used by a single computer.
- MSAs are automatically assigned a complex password.
- MSA account passwords are automatically updated every 30 days, like computer accounts.
- MSAs are exempt from account lock out policies.

## Group Managed Service Accounts

Unfortunately, MSAs did not live up to their potential. There were several drawbacks that drastically reduced their adoption. Extremely limited application support and the inability to use MSAs for scheduled tasks crippled their value. Industry-leading applications like SQL Server were not supported. The good news is, this was remedied with the release of Windows Server 2012 and the introduction of gMSAs. All the characteristics of MSAs, such as their password management and storage in AD DS, are characteristics of gMSAs too. When working with service accounts, you should always try to use a gMSA when supported and avoid MSAs at all costs.

The following are a few of the key improvements made with gMSAs:

- **Multi-server.** Unlike MSAs, a single gMSA can be used across multiple servers.
- **Scheduled tasks.** A gMSA can be used to run scheduled tasks.
- **Application support.** Unlike MSAs, gMSAs are supported across applications like SQL server and IIS application pools.

## Creating Group Managed Service Accounts

The MSA and gMSA are created and installed using PowerShell. But they are manageable in ADUC and ADAC. Before creating a gMSA you need at least one Windows Server 2012 or later domain controller. Additionally, the server that will use the gMSA must run Windows Server 2012 or later. During the creation process, gMSAs are the default account type unless explicitly changed through the PowerShell parameters.

The following steps outline the process for creating and assigning a gMSA:

1. Prepare your AD environment by adding a Key Distribution Services (KDS) root key. A KDS root key is required to generate gMSA passwords.
2. Wait a minimum of 10 hours. After adding a KDS root key, it is recommended that you wait 10 hours. The command to create the root key builds in a 10 hour wait time by default. While there are methods to get around the 10 hour wait time, you should avoid those methods in a production environment to ensure full functionality of newly created gMSAs.
3. Create a security group. The security group is assigned to the gMSA to restrict usage of the gMSA to only computers that are members of the group.
4. Create a gMSA.
5. Assign the gMSA to a member server. Servers cannot utilize the gMSA until it is assigned using PowerShell.
6. Test the gMSA. Use a PowerShell command on the member server to confirm the process completed successfully.

In the following example, we will create a gMSA for SQL server, using the steps mentioned above.

1. On your Windows Server 2012 Domain Controller, open an elevated PowerShell window and run the following PowerShell command to add a KDS root key:  
`Add-KdsRootKey -EffectiveImmediately`
2. Wait 10 hours.
3. Run the following PowerShell command to create an AD security group:  
`New-ADGroup -Name TT-SQL-gMSA-Group -GroupScope Global`
4. Run the following PowerShell command to create a gMSA:  
`New-ADServiceAccount -Name TT-SQL-gMSA -DNSHostName tt-sql-gmsa.tailspintoys.com -PrincipalsAllowedToRetrieveManagedPassword TT-SQL-gMSA-Group`
5. Run the following PowerShell command to install the gMSA on the local server:  
`Install-ADServiceAccount -Identity 'TT-SQL-gMSA'`

6. On your member SQL server, run the following PowerShell command to confirm the gMSA has been installed successfully. Note that the Active Directory PowerShell module is required to run the command:

```
Test-ADServiceAccount -Identity TT-SQL-gMSA
```

In Figure 19.4, you can see an example of the SQL Server gMSA that we created above. By default, MSAs and gMSAs are stored in the Managed Service Accounts container in AD.

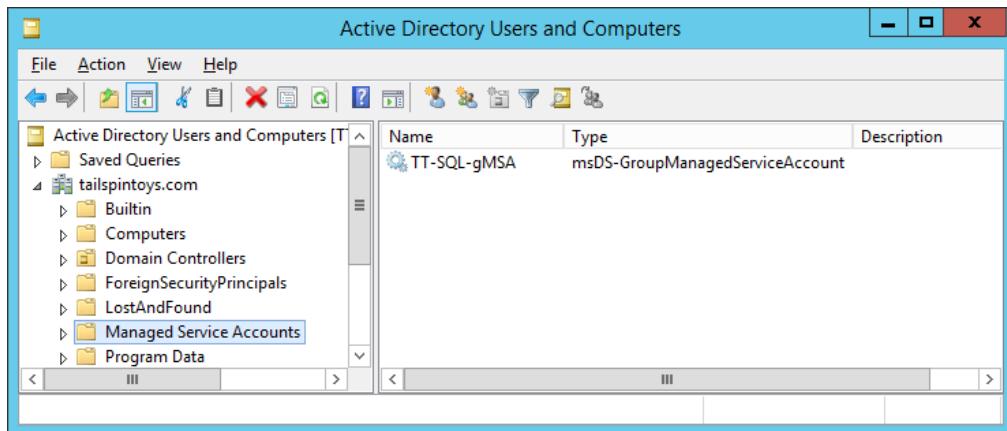


Figure 19.4 A group managed service account for SQL Server in Active Directory Users and Computers

In this section, you learned about gMSAs. They provide service account with automated password management which helps to reduce administrative overhead and increase security. Earlier in the chapter, we talked about standard service accounts and how to manage them. At the beginning of the chapter, we showed you the ins and outs of the Services management console which you can use to start, stop, and configure services. That a lot of information around working with services. Let's move to the lab now and test your knowledge.

## Lab

### Using the Service Management Console

Perform the following tasks:

- Use the Services management console to locate the Remote Desktop Services service. What is the short name for the service?
- Use the Services management console to change the recovery options for the Remote Desktop Services service. Configure the second failure to restart the computer.

### Starting and Stopping Services

Perform the following tasks:

- Use the Services management console to stop the Netlogon service.
- Use PowerShell to start the Netlogon service.

- Use the Services management console to pause the Winmgmt service.
- Use PowerShell to resume the Winmgmt service.

## Service Accounts

Answer the following question:

- Which steps would you take to use the service account, Tailspintos\SQL-SVC, to run the MSSQLSERVER service?

## Group Managed Service Accounts

Perform the following tasks:

- Prepare your AD environment for gMSAs. Add a KDS root key. Wait 10 hours before proceeding.
- Create a gMSA for SQL Server. Create a security group named SQL Servers and configure the gMSA to restrict access to only the group.

## CHAPTER 20: WORKING WITH LOCAL STORAGE

---

While we've covered quite a few topics so far in this book, we haven't talked too much about storage. You got a peek at some storage fundamentals in Chapter 2. Now, we are going to spend the next few chapters on storage and storage-related technologies such as local storage (this chapter), NTFS permissions (directly following this chapter) and shared folders/file services (directly following the NTFS permissions chapter). The order that we are using introduces you to the foundational storage technologies first and then builds on that information by showing you how to secure it and share it thereafter.

By the time you finish the next few chapters, you should be quite comfortable managing storage and storage-related technologies in Windows Server.

In this chapter, we look at the uses and configuration of local storage for a server. First, we will go through the disk management options for local disks and volumes. Disks and volumes are two components that you will routinely work with when managing servers. Then we will discuss how to configure those disks and volumes to obtain higher performance, redundancy, or both. As an administrator, you should regularly configure your environment for higher performance and redundancy, because doing so makes your customers happy (even if your customers are just other employees and departments at your company). Thereafter, we will transition into how disk drives can be used with Storage Spaces, a new(ish) feature for grouping storage on Windows Server to ease management and provide redundancy. Storage Spaces makes providing performance and redundancy easier and has a more flexible storage model compared to the traditional way of doing it in Windows Server (which we'll cover in this chapter first). To finish up the content of the chapter, we will discuss the command line and Windows PowerShell tasks that you can use to manage local storage. At the end of the chapter, we will test your storage knowledge in the lab!

### Manage Disks and Volumes

Every server has disk drives. Every server also has volumes. But before we begin talking about them in detail, I want to provide a very brief definition of some key terms that we'll be using throughout this chapter. If you are already familiar with these terms, feel free to skip ahead!

- **Physical disk.** A disk drive made up of physical components. You can buy one at the store. You can hold it. When you want to use it, you need to install it in a computer and connect it with cable(s). If you are anything like me, then you avoid physical disks whenever possible. Physical disks have a bit of overhead managing them (installing, replacing, troubleshooting) so I prefer virtual disks!
- **Virtual disk.** A software-based disk drive made up of a single file (usually a .vhdx or .vhd file). You can't buy one and you can't hold one. But, you can freely create them on virtualization platforms (such as Hyper-V and VMware ESXi). To most operating systems, a physical disk and a virtual disk are identical – you manage them the same way. But, virtual disks are often presented to your server by a storage team, which helps offload some of your storage management tasks!

- **Partition.** Each disk drive is partitioned before use. A disk drive can have a single partition (very common) which takes up the entire disk drive. Or a disk drive can have multiple partitions with each taking up a portion of the disk drive. Each partition is sometimes referred to as a logical disk.
- **Volumes.** To use a disk drive, you create a volume. A volume is all or part of a partition that has been formatted with a file system and usually has a drive letter (such as E:\). I prefer to use a single partition and a single volume, when possible. I find that simple configurations require the least amount of maintenance and are easiest to troubleshoot.

One of the reasons why you'll work with disks and volumes so often is because every server has them! Next, we will look at how to manage disks and volumes.

## Managing Disks

The first step in providing storage to a server is acquiring disks (or a server that has disks). Disks, without any configuration or preparation, are not ready to store an operating system or data. In this section, we'll walk through preparing a disk (physical or virtual) for use. We'll do so by using a tool named Disk Management, which is an MMC snap-in like previous tools we worked with such as DNS Manager. You can run the Disk Management MMC in a few different ways:

1. Right-click the Start button and then click **Disk Management**.
2. From Administrative Tools, launch Computer Management, and then click **Disk Management**.
3. From the Run window or command line, run **diskmgmt.msc**.

I recommend that you start all your disk management activities from Disk Management. After you become comfortable with it, then start exploring some of the other management methods such as Diskpart and PowerShell (which we discuss a bit later in this chapter).

Figure 20.1 shows the Disk Management MMC snap-in with two disks. Disk 0 is in use and contains the operating systems files. It is the disk that the computer will be booted from so it is labeled as the Boot volume. Disk 1 is an unallocated disk that is not ready for use. It has not been marked online, initialized, or formatted. This is the typical state when you add disks for the first time. In this chapter, we'll be looking at preparing a disk for use.

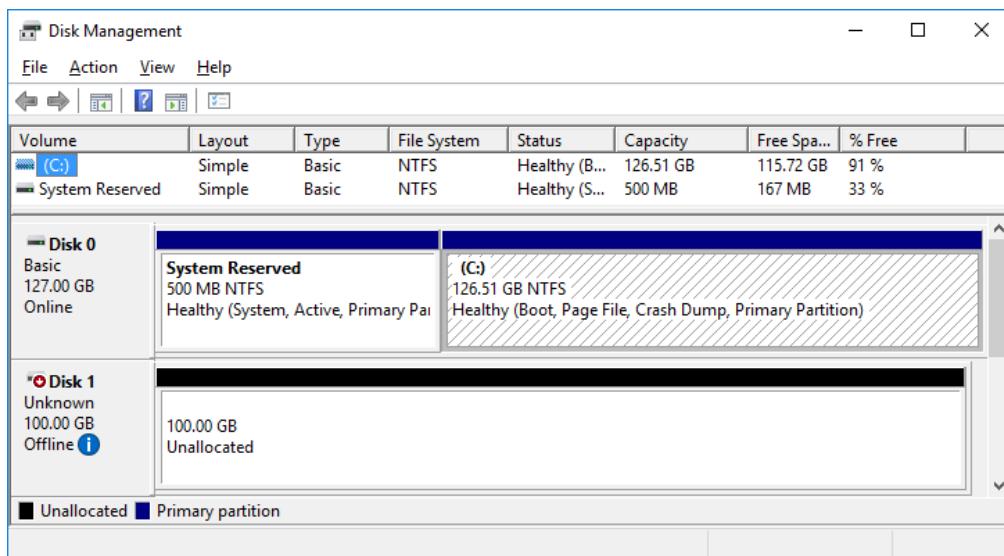


Figure 20.1 A screen capture showing the Disk Management MMC snap-in.

## Above and Beyond

The easiest way to experiment with disks and volumes at home or in a lab is to use VHDX files. VHDX files are virtual hard disk files. Each file represents a virtual disk drive. If you're running Windows Server 2008 or later, you can enable the Hyper-V role on your computer, if you have hardware that supports virtualization. By running virtual machines (VMs) and virtual disks (.VHD or .VDHX), you can easily test and perform hardware-type actions within a VM. To add a VHDX to a VM, modify the settings of the VM and add a hard drive to the VM configuration. To configure and test the topics that are presented in this chapter, a VM with three additional virtual disks will provide a good testing environment.

Let's walk through the process of preparing a newly added disk for use.

1. The first step to being able to use a disk is to mark it online. To mark a disk as online, right-click the disk number in the left pane and then click Online.
2. After marking the disk online, you must initialize the disk. To initialize a disk, right-click the disk number and then click Initialize Disk. Figure 20.2 shows the Initialize Disk window. When initializing a disk, you will be prompted to select the partition style. There are two options for the partition style:
  - **Master Boot Record (MBR).** MBR is the default partition style for new disks under 2 TB. MBR has a 2 TB limit, so GPT is required for any disk

larger than 2 TB. MBR also only supports creating up to 4 primary partitions on the disk, while GPT supports up to 128 partitions.

- **GUID Partition Table (GPT).** GPT will also support disks that go up to almost 1 Zettabyte (ZB) – which is about 1 billion TB! I usually choose GPT partitions which enables me to use very large partitions, if needed.

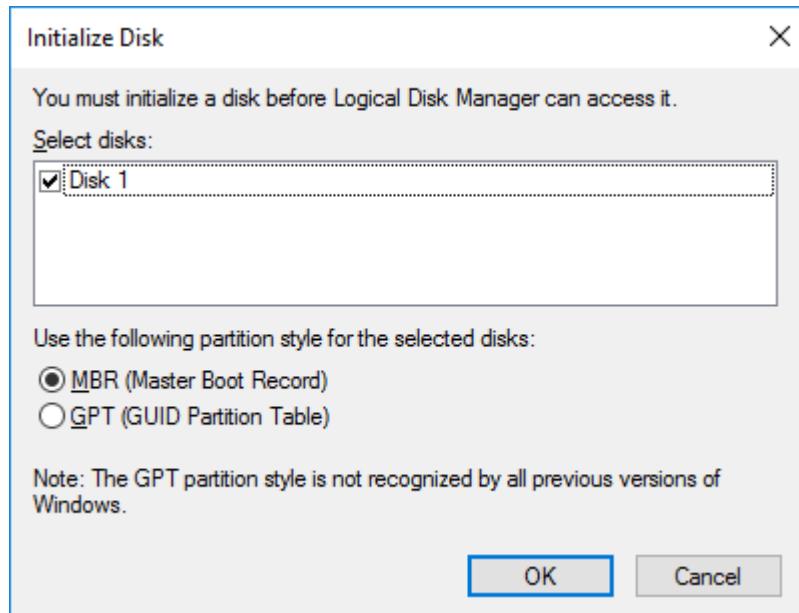


Figure 20.2 A screen capture shows the Initialize Disk window where you choose the partition style that you want to use for a disk.

3. After you initialize the disk, you create a volume on the disk. You can choose to create one volume (which takes up the entire disk) or multiple volumes (with each taking up a portion of the disk). Using one volume simplifies the configuration. But sometimes you will want to use multiple volumes to segment data. For example, you might create one volume for your Windows operating system and another volume for all your data – this makes it easy to work with your data (backups, encryption, reporting) because it is all on a dedicated volume. Creating a volume is the final step in preparing your disk for use! Creating a volume formats the partition and enables the partition to be presented to the operating system. This is the point that you can assign a drive letter and begin using the disk. To create a volume, right-click in the unallocated space of the disk and then click **Create Simple Volume**. The New Simple Volume wizard will be displayed, and will prompt you for the size of the volume (you can choose the entire size available or a portion of the space), the driver letter that you want to use (any drive letter that is not currently in use), and the file system that you want to use (NFTS, FAT32, ReFS). Figure 20.3 shows the completed New Simple Volume Wizard, with some common options configured. For the file system, you have three options:

- **File Allocation Table (FAT32).** FAT32 is a Microsoft-proprietary file system that was regularly used for an older version of Windows client

computers, such as Windows XP. You should avoid using FAT32 unless you must (such as in a case where you are running a legacy client computer and it only supports FAT32).

- **New Technology File System (NTFS).** NTFS is the default file system and most commonly used file system in Windows Server. NTFS supports popular features such as file compression and encryption. You can create volumes of up to 256TB with NTFS.
- **Resilient File System (ReFS).** ReFS is also a Microsoft-proprietary file system that was introduced in Windows Server 2012. It offers increased scalability compared to NTFS with support for volume sizes up to 1 YB. If you are wondering what a YB is, don't worry – many others are too! It is a Yottabyte. 1 Yottabyte is equivalent to 1,000,000,000,000 TB. It is hard to comprehend, at least for me!

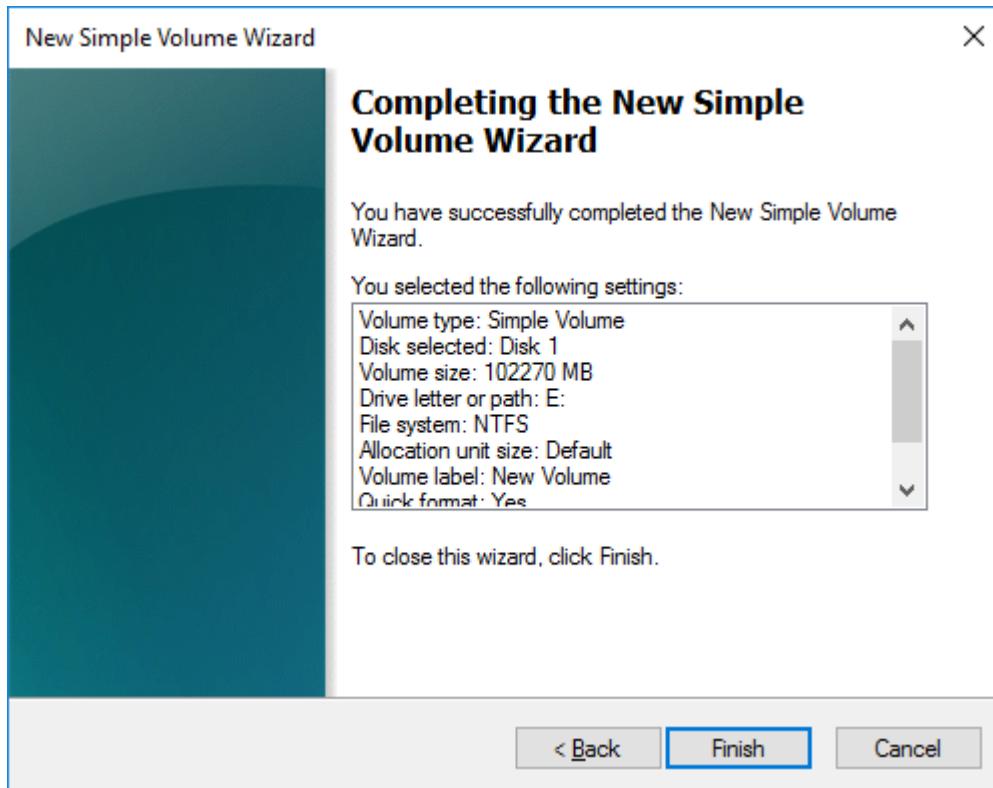


Figure 20.3 A screen capture showing the New Simple Volume Wizard completed with the most common options configured.

### Hands-on Exercise

Open Disk Management and initialize a disk as MBR. Create a new simple volume and format the volume with NTFS.

Now you know how to work with single disks. Let's build on that in the next section by looking at special disk configurations that use multiple disks that offer higher performance and/or redundancy compared to single disks.

## RAID Levels

A redundant array of inexpensive disks (RAID) is a grouping of disks (using software or hardware) to provide higher performance or redundancy for disks (so that if a disk fails, a server can stay running and data access remains functional). RAID offers different RAID levels and each level has characteristics that are tied to performance or redundancy (or both). RAID can be configured via software or hardware. RAID with hardware usually provides the best performance but also requires hardware that supports RAID and physical components (such as physical disk drives). Some hardware vendors even have proprietary methods of storing data for a custom RAID level or a combination of RAID levels. Keeping this in mind, we will discuss only the most commonly used RAID types that can be configured from Windows Server:

- Stripe (RAID 0)
- Mirror (RAID 1)
- Parity (RAID 5)

Windows Server supports software RAID and hardware-based RAID (which is usually managed by the storage hardware attached to the server). For my physical servers, I always try to use hardware-based RAID when a server supports it. Mostly, I use RAID for the operating system volume. My data volumes are often SAN-based so the redundancy is provided by the SAN and I don't need to concern myself with it (beyond that it meets the requirements and performance requirements that I need).

### RAID 0

Every RAID type has certain benefits and/or drawbacks. The method that RAID 0 uses is striping, meaning that each block of data that is written is striped across the disks that are members of the stripe set. For example, if you had four disks that were members of the stripe set, a 64 MB file could be striped across each of the disks. Each disk could receive 16 MB ( $64 / 4$ ) of the file. The benefit to this is that instead of performing a large write to a single disk, the system is performing four smaller simultaneous writes. RAID 0 typically provides an increase in performance, which improves with the number of disks that are members of the stripe set (more disks = more performance).

Another advance of RAID 0 is that it enables you to use the full capacity of the drives. If you have two 1 TB disk drives, you can use 2 TB of space. However, because each disk is only receiving a portion of the file, RAID 0 is also very susceptible to data loss. Regardless of the number of drives in a RAID 0 configuration, corruption or failure of a single drive can cause the loss of all the data in the stripe set (all data on all disks). Therefore, I generally avoid RAID0 for production workloads. In fact, I generally avoid it whenever I can and opt for other RAID levels that provide redundancy and more protection of the data. But there are use cases for it. For example, imagine you have a web application that has a need for high-performance storage in a small form factor. You have 50 servers that are providing the exact same web application, each

being self-contained and load balanced. If 1 server's storage fails, it has no impact on the web application and instead only impacts a single web session which can be restarted on another server. In such a scenario, RAID 0 might be highly effective.

Figure 20.4 illustrates how RAID 0 stripes data across multiple disk drives by breaking the data up into blocks and storing them on the drives that are members of the stripe set.

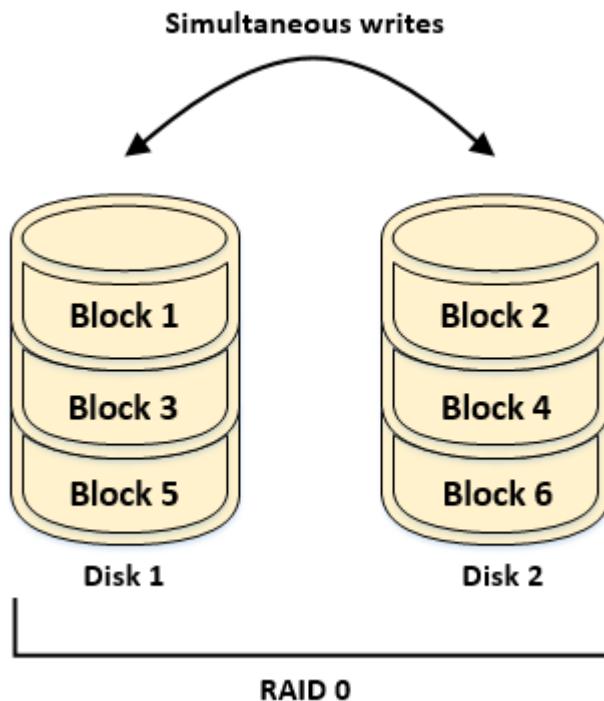


Figure 20.4 A diagram that illustrates how RAID 0 breaks data up to be stored across multiple disk drives.

### Creating a RAID 0 volume

Now that you know what RAID 0 is, let's look at how to create a software-based RAID 0 volume. Perform the following steps in Windows Server:

1. Insert two disks (or more) and prepare them by bringing them online and initializing them.
2. Open Disk Management. Right-click the disk number for one of the disks that you want to use and then click **Create Striped Volume**.
3. In the **New Striped Volume** window, on the Welcome to the New Striped Volume Wizard page, click **Next**.
4. On the **Select Disks** page, the disk you right-clicked will already be selected as part of the striped volume. Click the other disk, listed as an available disk in the left portion of the page, and then click **Add**. At the bottom of the window, configure the space that you want to use on the volume (by default, all the space is configured) and then click **Next**.
5. On the **Assign Drive Letter or Path** page, configure the drive letter that you want to use (by default, the first available drive letter will be configured) and then click **Next**.

6. On the **Format Volume** page, configure the file system that you want to use (NTFS or ReFS only for RAID volumes), configure a volume label (optional step), and then click the **Perform a quick format** option (this is optional but desired because it prepares the volume for use much faster than a standard format does). Then click **Next**.
7. On the **Completing the New Striped Volume Wizard** page, review the settings and then click **Finish**.

That's all there is to it. It is like creating a standard volume without RAID. Let's now look at RAID 1.

### RAID 1

RAID 1 provides fault tolerance if a single drive fails. In other words, if one drive fails, you are still operating with the other drive! This makes it popular for use for system volumes which store the operating system. RAID 1 uses a mirroring technique, which writes the same block of data to at least two drives in a mirrored configuration. Therefore, if a 64 MB file is written to Disk 1, then the same 64 MB file is also written to Disk 2. This enables you to lose an entire drive without losing access to the data. Because of this redundancy, RAID 1 is incredibly popular. I've seen it used in literally every organization where I've done any work.

Most mirroring implementations require an even number of drives: 2, 4, 6, or more, although using more than two drives is typically named RAID 10. RAID 10 combines the striping of RAID 0 with the data protection of RAID 1. Some controllers or software can use an odd number of drives. For example, if you create a RAID 1 set with three disk drives, the software may provide double redundancy by copying the same write to all the drives in the configuration.

The drawback of RAID 1 is that it effectively doubles the cost per TB of storage. If you have two 1 TB disk drives in a RAID 1 configuration, you will only be able to use 1 TB of the 2 TB total. The "missing" 1 TB is an exact replica of the data that is being written on the other disk.

Creating a mirror can be performed in Windows by using Disk Management. Like a RAID 0 volume, you can use Disk Management to create a RAID 1 volume (shown as a mirrored volume in Windows).

Figure 20.5 illustrates how RAID 1 mirrors data between two drives that are members of the RAID configuration.

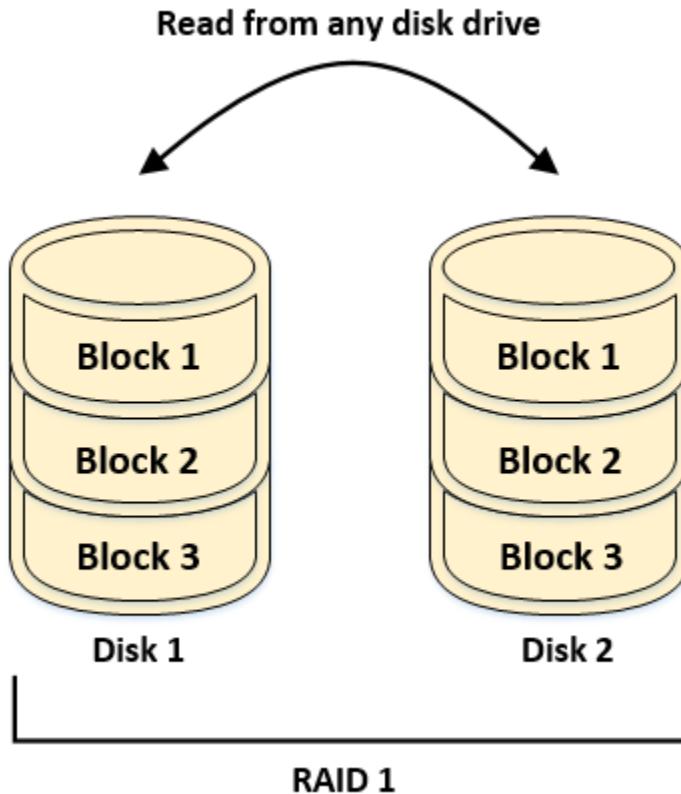


Figure 20.5 A diagram that illustrates how RAID 1 mirrors data that is being written across two drives.

### Hands-on Exercise

Create a RAID 1 volume from two virtual disks using Disk Management.

The final RAID level that we'll look at is RAID 5, another popular RAID level in wide use in organizations today.

### RAID 5

RAID 5 tries to provide the best of both RAID 0 and RAID 1. RAID 5 requires a minimum of three drives and calculates a parity bit based on the data that is being written. For example, data is written to two of the three disks and then software or the RAID controller calculates a parity bit based on the data that was written, and writes that parity information to the third drive. Note that the order in which the data is written is randomized, so that parity information is spread across all three drives. This parity information enables you to lose a drive to failure, while still retaining access to the data. The software or controller can still identify the data that is missing from the drive by reverse-calculating the parity information that was stored for that data.

From a performance perspective, raw data is primarily written across two drives, but could be slowed due to the additional processing in calculating the parity information. The performance of data reads is increased, as reading can be performed across multiple drives without the parity information. Imagine that you have an application that enables customer service representatives to look up customer orders to find dates, order information, and historical pricing. In such a scenario, virtually all the storage access is read-only. RAID 5 is an excellent fit in such a scenario. The capacity of a RAID 5 configuration is always  $N-1$ , where  $N$  is the number of drives. If you have a RAID 5 configuration with five 1 TB disk drives, you would have 4 TB of usable space. 1 TB of space would be reserved for storing the parity information.

A RAID 5 can be created in Windows Server by using Disk Management. Simply right-click an unused disk and then click **New RAID-5 Volume**. The process is like what we walked through earlier, so we won't cover the step-by-step instructions here. Figure 20.6 illustrates how RAID 5 stores data with parity information across multiple drives.

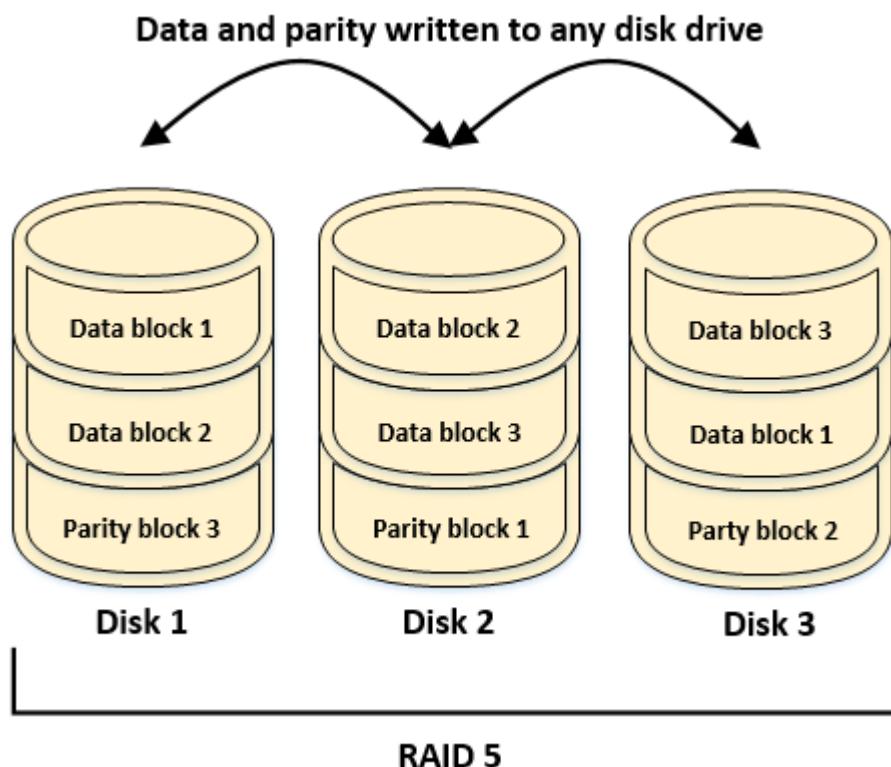


Figure 20.6 A diagram that illustrates how RAID 5 writes data with parity information across disk drives.

## Storage Spaces

What if you wanted to achieve the benefits of RAID but without the complexity? Storage Spaces is the answer. It is a storage feature built into Windows Server that enables you to configure RAID-like storage by grouping available disk drives in a server. Storage Spaces is intended to take the place of software and hardware RAID, when feasible. However, you can't use Storage Spaces for your boot/system volume so there are still some uses of RAID without

Storage Spaces. When creating new redundant volumes, it is a good practice to use Storage Spaces. This is because future development of storage redundancy in Windows Server is focused on Storage Spaces. Only use software RAID when you are working with the boot/system volume and do not have access to hardware RAID.

The first step to using Storage Spaces is to create a storage pool. A storage pool is just a grouping of disks. To create a storage pool, the desired drives must be online and initialized but cannot be formatted. Additionally, disks of any size can be used to create the storage pool. However, depending on the resiliency type that is selected, the storage pool may not use the entire space that is available on all the disks.

After creating the pool, you create the volume that will be stored in the storage pool. Like creating a volume from Disk Management, creating a Storage Spaces volume has a few resiliency types to choose from, as shown in Figure 20.7.

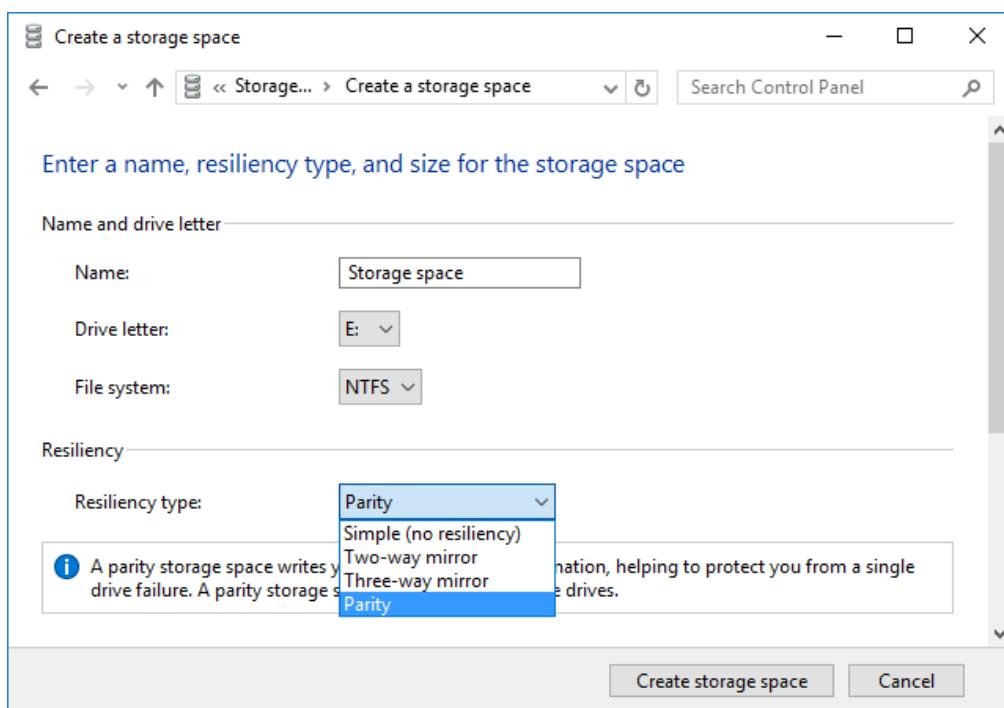


Figure 20.7 A screen shot that shows creating a storage pool from unformatted drives.

Each of the resiliency types can be thought of as variations of the RAID levels that we discussed earlier:

- **Simple (no resiliency).** A Simple storage space is like RAID 0, where all the data that is stored on the volume will be striped across the disks that were configured in the pool. A Simple storage space does not provide data protection but allows for the maximum amount of storage and performance.
- **Two-way mirror.** A two-way mirror requires a minimum of two drives and will mirror the data that is written on one drive to the second drive.
- **Three-way mirror.** A three-way mirror can be thought of as a 1:2 configuration of RAID 1. Data that is written to one disk is then mirrored (copied or written again) to the two other

drives in the storage pool. A three-way mirror also requires a minimum of three disk drives in the pool. If you want the mirror to continue to operate in the event of two disk failures, you must use at least 5 disk drives. Personally, I prefer a configuration that supports one disk failure and have spare disks in stock in case of failure.

- **Parity.** Parity can be thought of as RAID 5. A Parity storage space will protect against a single drive failure and provide the most available storage with redundancy.

### Hands-on Exercise

Using the Control Panel, create a two-way mirror storage space. Assign the storage space a drive letter and format the storage as NTFS.

The above examples were taken by using the Storage Spaces manager from the Control Panel. For advanced options, the File and Storage Services node in Server Manager should be used to manage the storage pool. The File and Storage Services server role enables you to specify which disks should be used for a pool and which disks (if any) should be allocated as spare disks. Additionally, after the pool is created, the server role also enables you to configure advanced options for virtual disks that are created within the pool.

One of the advanced options is storage tiers. Storage tiers enable you to use both spinning hard disk drives (HDDs) and solid-state drives (SSDs), and maximize the performance between the two drive types. Traditionally, spinning hard drives are slower than solid-state drives. Storage tiers will automatically move data that is being accessed more frequently from the slower media (HDDs) to the faster media (SSDs).

By using the server role to configure the virtual disks, you can also customize the size of individual disks, as well as the provisioning type. The provisioning type enables you to select from either Fixed or Thin disks.

- Fixed disks pre-allocate the storage from the pool when you create the disk and offers the highest level of performance.
- A Thin disk enables you to over-provision the storage pool, by only allocating data that is written. Therefore, you could configure multiple disk drives that total more than the available storage in the pool. Thin disks impact performance a little but maximizes the capacity of the storage. However, it is also very easy to maximize the actual capacity of the storage pool sooner than expected by overprovisioning the available space. In the real world, I typically opt for fixed disks for applications that require high performance (database, compute farms). For applications that do not require high-performance storage, such as a utility server or a Windows Update server, I consider thin disks.

A storage pool and virtual disk are shown in Figure 20.8.

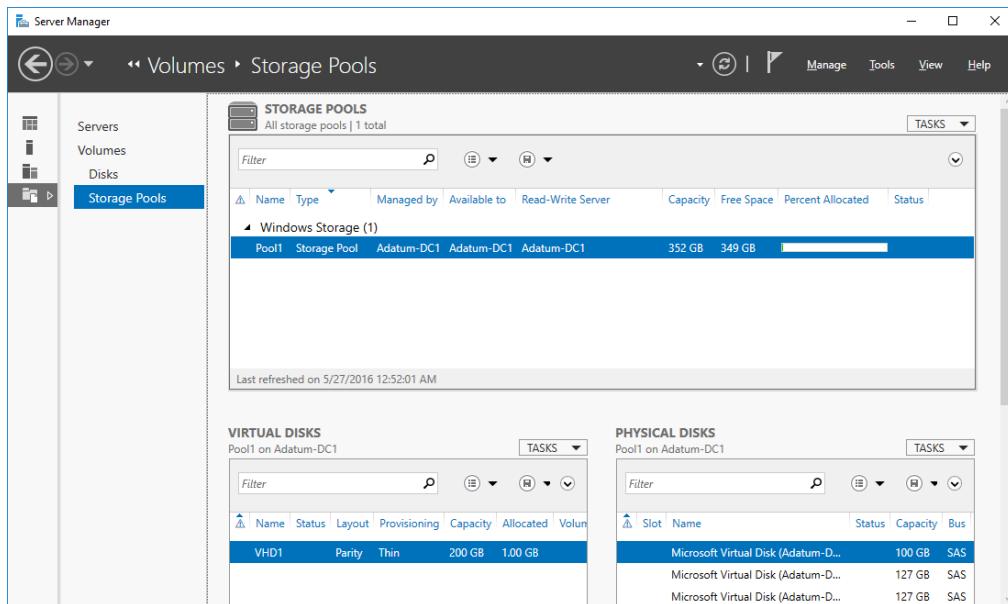


Figure 20.8 A screen shot shows the File and Storage Services Storage Pool management interface.

## Managing storage from the command line and Windows PowerShell

So far, we've looked at the graphical user interface tools for managing storage. While the GUI tools are often the primary storage management tools for administrators, we'll also take a brief look at the command line and Windows PowerShell options. Both the command line and Windows PowerShell options provide for a way to automate repetitive storage tasks or many tasks. For example, imagine that you just provisioned a new SAN and you've allocated storage to 50 servers. Now, you need to create a data drive on each of the servers! In this section, we are going to look at a command-line tool named Diskpart as well as some of the important storage related cmdlets in Windows PowerShell.

### Diskpart.exe

The diskpart.exe utility is a powerful command-line application that enables you to manage disk drives. Any action that can be performed from Disk Management can also be performed by using diskpart.exe. For example, you can mark disks online, initialize disks as MBR or GPT, create partitions, create simple volumes, create RAID volumes, and delete partitions and volumes. Diskpart is falling out of favor because of the expanded functionality of the Disk Management tool and PowerShell. Today, I find myself using Diskpart mostly when I work with legacy operating systems such as Windows Server 2003. That's because, since the arrival of Windows Server 2008, much of the functionality of Diskpart has been built into the Windows disk management GUI tools and PowerShell. But with Windows Server 2003, Diskpart is often the only tool to do the job.

## Using Diskpart

To start using the Diskpart utility, run the diskpart.exe command from a command prompt. Thereafter, you'll be at a Diskpart prompt. Use a question mark (?) to view the help contents. To see the complete Diskpart syntax, see <https://technet.microsoft.com/en-us/library/bb490893.aspx>.

The following Diskpart uses show some command storage management commands using Diskpart:

- **You can view the current disks on a computer using Diskpart.** The disk list shows you a disk number for each disk which helps you target specific disks for other actions, such as creating a partition. To list the current disks, run the command from the DISKPART prompt:

```
list disk
```

The output of the command is shown in Figure 20.9 below.

Disk ###	Status	Size	Free	Dyn	Gpt
Disk 0	Online	22 GB	1024 KB	*	*
Disk 1	Online	10 GB	9 GB	*	*
Disk 2	Online	10 GB	9 GB	*	*

Figure 20.9 A screen capture shows the disks on a server, as reported by the Diskpart command-line tool.

- **You can select a specific disk to manage once you obtain the disk number.** To select a disk with a disk number of 2, run the following command:

```
select disk 2
```

- **You can create a new partition in a single command.** After you've selected a disk, you can manage it. In the following command, we create a primary partition on a GPT style disk using all the available space

```
create partition primary
```

After the creation of the partition on disk 2, the free space changes to 0, as shown below in Figure 20.10.

Disk ###	Status	Size	Free	Dyn	Gpt
Disk 0	Online	22 GB	1024 KB	*	*
Disk 1	Online	10 GB	9 GB	*	*
* Disk 2	Online	10 GB	0 B	*	*

Figure 20.10 A screen capture shows the disks on a server, as reported by the Diskpart command-line tool.

Diskpart can perform a ton of other storage management tasks too. But, before you explore too much with it, let's have a look at Windows PowerShell options. Windows PowerShell will eventually replace the legacy command line for the majority (if not all) of command line work.

## Windows PowerShell

By default, in Windows Server 2016, there are approximately 140 cmdlets/functions that are included in the Storage module for Windows PowerShell. Most of these are also available in PowerShell in earlier versions of Windows Server back to Windows Server 2012. These cmdlets provide you with the ability to manage all aspects of storage, including all the tasks that we have discussed in this chapter. For example, to set a disk number 1 online, you run the following command:

```
Get-Disk -Number 1 -IsOffline $False
```

However, as with using the Disk Management interface, simply turning the disk online does not initialize or format the disk. To initialize disk number 1 with the GPT partition type, run the following command:

```
Initialize-Disk -Number 1 -PartitionStyle GPT
```

Finally, to format the disk as NTFS and create a volume, run the following command:

```
New-Volume -DiskNumber 1 -FileSystem NTFS -DriveLetter E -FriendlyName Volume1
```

You can also use the Get-Disk cmdlet to view the properties of the disk before and after formatting it. Figure 20.11 displays the process of setting a disk online and presenting it as storage to the operating system.

```
PS C:\Users\Administrator> Set-Disk -Number 1 -IsOffline $False
PS C:\Users\Administrator> Initialize-Disk -Number 1
PS C:\Users\Administrator> New-Volume -DiskNumber 1 -FileSystem NTFS -DriveLetter E -FriendlyName Volume1
DriveLetter FileSystemLabel FileSystem DriveType HealthStatus OperationalStatus
----- -----------
E       Volume1      NTFS     Fixed    Healthy   OK

PS C:\Users\Administrator> Get-Disk -Number 1 | FL

UniqueId          : 60022480D2F249B30D6756F339100E21
Number            : 1
Path              : \\?\scsi#disk&ven_msft&prod_virtual_disk#000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}
Manufacturer      : Msft
Model             : Virtual Disk
SerialNumber      :
Size              : 127 GB
AllocatedSize     : 136365211648
LogicalSectorSize : 512
PhysicalSectorSize: 4096
NumberOfPartitions: 2
PartitionStyle    : GPT
IsReadOnly        : False
IsSystem          : False
IsBoot            : False
```

Figure 20.11 A screen capture shows the process of managing a disk with Windows PowerShell.

While we don't have the room in this book to dive deeper into managing storage with Windows PowerShell, you should explore Windows PowerShell's Storage module to see what else you can do. And, find out whether you like managing storage more with the GUI or with Windows PowerShell.

In this chapter, we looked at basic storage management for servers. You learned all about disks, partitions, and volumes. You learned that a physical disk and a virtual disk are managed

the same way. You learned that a volume is a portion of a disk (or an entire disk) that is formatted with a file system so that you can use it with Windows. Then, we looked at RAID which combines multiple disks together to enhance performance and redundancy (at least sometimes). You learned about RAID 0 (striping), RAID 1 (mirroring), and RAID 5 (parity) so you now know which RAID level to use based on redundancy and performance requirements. We looked at Storage Spaces, which is a feature in Windows Server to simplifies the process of using RAID-like storage. Finally, we looked at some basic storage management capabilities that you can perform from a command line and from Windows PowerShell. You now know that Diskpart and PowerShell provide all the functionality (and sometimes more) of the GUI tools such as Disk Management and Server Manager. Next, we'll test your newfound storage knowledge in the lab.

## Lab

This lab is designed to validate your retention of information from this chapter and perform some storage management tasks. If you haven't already completed the Hands-on Exercises in this chapter, do that now and then come back to perform the lab exercises.

### Initialize and format disks

Perform the following tasks:

- If necessary, add two virtual disks to a VM.
- Using Windows PowerShell, initialize the disks and set the partition type to GPT.
- Using Windows PowerShell, format the disks as ReFS and assign a drive letter to the volume.

### Create a software RAID

Create a software RAID 1 by using Disk Management:

- If necessary, add two virtual disks to a VM.
- Initialize two disks and set the partition type to MBR.
- Create a new mirrored volume by using Disk Management.

### Create Storage Spaces

Using the Control Panel, create a Storage Space:

- Using the tool of your choice, initialize three disks as GPT or MBR.
- From the Control Panel, open the Storage Spaces manager.
- Create a Storage Space that uses Parity for the resiliency type.

### Manage Storage Pools

Using Storage Manager, create a virtual disk:

- Use Server Manager to create a 50 GB virtual disk that is Thin provisioned.

## CHAPTER 21: MANAGING NTFS PERMISSIONS

---

In the last chapter, we looked at managing local storage such as disks and volumes, RAID, and Storage Spaces. After you have configured your storage, it is time to use it! Using it often means storing data in the storage, configuring access to the data, protecting the data with encryption, and monitoring access and access attempts to the data. For example, imagine that the Tax department need to store company tax returns in a folder. They don't want anybody else accessing the data because it contains sensitive information. And they would want to know if anybody tried to access the data unsuccessfully as well as know when anybody successfully accesses the data. This scenario is common. Not just for a tax department but for all departments across organizations. In this chapter, we will show you how to handle a scenario like the Tax department data.

First, we will discuss how you can use NTFS permissions to grant or deny access to files and folders on NTFS-formatted volumes. In our Tax department scenario, you would use NTFS permissions to ensure that only users that are a member of the Tax group can read or edit the tax data. Permissions can get a bit complex. Because a user might be a member of the Tax department but also be a member of the Executive department and the Legal department too. To help you figure out the actual access level somebody has, we will show you what's called "effective permissions". Basically, effective permissions are the permissions that a user has after Windows has accounted for all their group memberships and all the NTFS permissions configured. After that, we look at encryption. Encryption obfuscates data so that it can only be read by authorized users. For sensitive data like tax data, you should use encryption to minimize the chances that a malicious individual gains access to the data. After you have all your configuration in place for the Tax department, you need to capture file and folder access information such as who tried but failed to read the data and who did read the data. We will show you how to do that with auditing, a feature built into the Windows operating system to track access to data.

### What are NTFS Permissions?

NTFS permissions enable you to define the user accounts, or more commonly, the security groups, that can or cannot access a file or folder. Imagine that you have a folder named Tax. Under the Tax folder, there are folders for each year – 2013, 2014, 2015, and 2016. If you configure the Tax group to have Full Control permissions to the Tax folder, then they will also get Full Control permissions to the 2013, 2014, 2015, and 2016 folders. This is because of inheritance, which is enabled by default (but can easily be disabled as needed). You can configure permissions at the file level, at the folder level, or at the volume level. These permissions are known as explicit permissions because you explicitly set them at the object level. By default, the permissions are inherited by child objects. Inherited permissions are sometimes referred to as implicit permissions.

Figure 21.1 shows an example of inheritance.

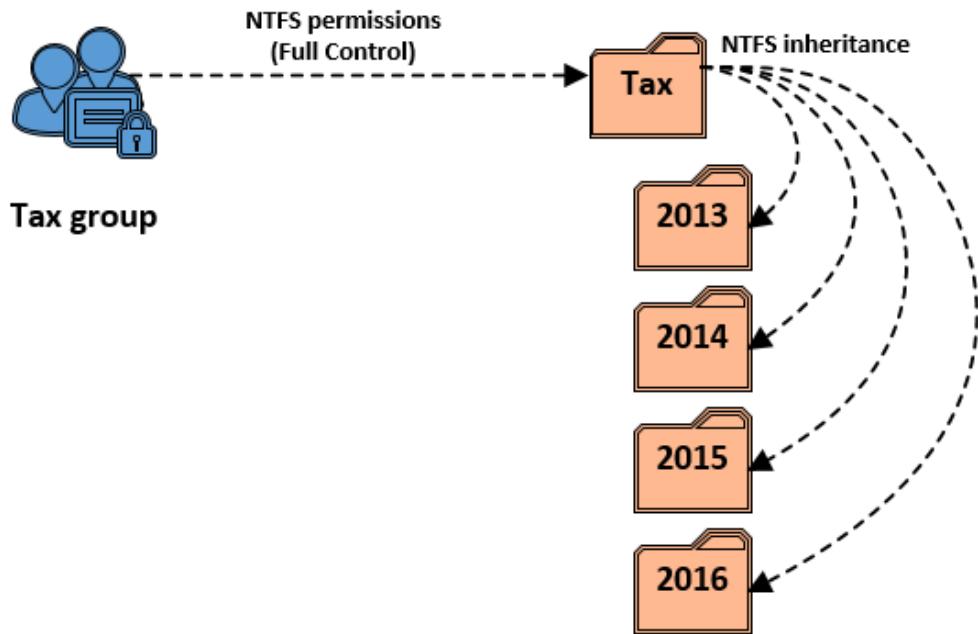


Figure 21.1 A diagram shows how NTFS permissions applied at a parent folder are automatically inherited by child folders.

For most volumes, files, and folders, the default NTFS permissions grant the built-in Administrators security group and the SYSTEM account Full Control permissions. Also by default, the Users group is granted read and execute permissions. There are other permissions too, but they aren't important for our discussion. You can look at those permissions by looking at the permissions of the C:\ disk on a default build of Windows Server.

In Figure 21.2, a screen capture shows the permissions for the E:\Tax folder. Note that the permissions shown are the default permissions.

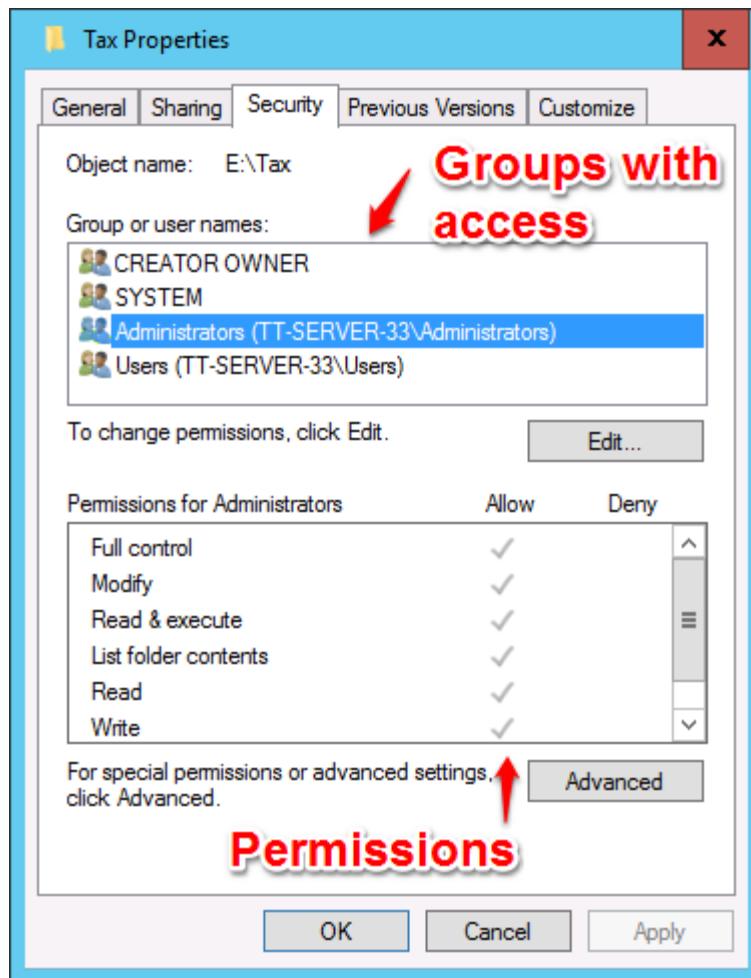


Figure 21.2 A screen capture shows the default NTFS permissions for a folder.

Giving permissions is very straight forward. Perform the following steps to grant permissions:

1. Right-click the folder for which you want to grant access and then click **Properties**.
2. Click the **Security** tab.
3. Click the **Edit** button.
4. Click the **Add** button, type the name of the user or group, and then click **OK**.
5. Accept the default permissions for the user or group, or adjust the permissions, and then click **OK**.

Revoking permissions is like giving permissions. Perform the following steps to revoke permissions:

1. Right-click the folder for which you want to grant access and then click **Properties**.
2. Click the **Security** tab.

3. Click the **Edit** button.
4. Click the group or user from which you want to revoke access and then click **Remove**.  
You can only remove a group that has explicit permissions to the folder. If a user has access to a folder through inherited permissions (which we covered earlier in this chapter), then you can't remove the access without first removing the inheritance (which you can do in the advanced permissions section).
5. Click **OK** to close the edit window and then click **OK** to close the folder properties window.

### **Hands-on Exercise**

Create a new folder in the root of the C:\ drive named Test123. Grant the Authenticated Users group Full Control permissions on the folder. Remove permissions for the Users group. Don't forget about inheritance when you try to remove the permissions!

### **Denying permissions**

One final thing about permissions. We often talk about granting permissions and revoking permissions. These permissions are referred to as Allow permissions. But there is one more scenario that is a bit different. Imagine that you have a sales team. They have a sales folder with sales-related data. The Sales security group has access to read/write/modify data. The Sales security group is also used to grant access to a variety of other resources such as printers, web sites, and databases. The sales manager asks you to ensure that one member of the sales team, Paul, cannot get to the sales data in the folder. But, if you remove that person from the Sales security group then you'll remove virtually all their access to other resources (which they still need). Instead, you can use Deny permissions, which deny access instead of grant access. For example, if you give Paul the Deny Read permission for the sales data, then he will not be able to read the file and/or folder, but he will still be able to print and access web sites and databases. Deny permissions are very powerful because they take precedence over Allow permissions. The Windows operating system evaluates Deny permissions first, before evaluating Allow permissions. You should minimize the use of Deny permissions when you can because they add complexity to your environment and make it harder to troubleshoot access issues. I personally try to avoid the use of Deny whenever possible. Instead, I exhaust all other possibilities first such as restructuring data to simplify permissions assignments, disabling inheritance to avoid the use of Deny permissions, and splitting up security groups and using group nesting to avoid the use of Deny permissions.

### **Standard and Advanced Permissions**

Now let's look a little deeper into NTFS permissions and show you that there are two sets that you should be familiar with – standard permissions and advanced permissions. Right-click a folder, click Properties, and then look at the Security tab of the Properties window. Those are

standard permissions. Most of the time, you can use those when granting permissions because they cover the typical use cases (user needs to read documents or user needs to modify documents).

The standard permissions are:

- **Full Control.** Full control is complete control over the file or folder including the ability to change existing permissions and change the ownership of files.
- **Modify.** Modify permissions enable users to read, write, and delete files, but not modify permissions.
- **Read & execute.** The Read & execute permission enables users to open files.
- **List folder contents.** The List folder contents permission only enables a user to see the directory structure of a folder, including its files. Users can also open files with this permission.
- **Read.** Like list folder contents, the Read permission enables users to read the contents of a folder and open files.
- **Write.** Write permissions enable you to create files in a folder and modify existing files.

### Ownership and the impact on permissions

Closely related to permissions is ownership. Every file has an owner. An owner can change NTFS permissions (give himself permissions or others permissions). A local administrator can take ownership of any file and become the owner. Imagine a scenario where a department manager has full control permissions and ownership of all their files in a single folder named HR. The manager removes most groups from the permissions so that only the HR group has access to the data. The manager leaves the company. 3 months later, a new manager starts and can't update the folder permissions. How do you fix that? You, as the administrator, take ownership of the top-level folder and then update the permissions so that the new manager has full control permissions. I've also used ownership changes during file server migrations. In one case, an old file server was being migrated to a new file server. During the data copy, some folders didn't copy to the new file server due to a lack of permissions on the data. I took ownership of the top-level folder and then gave myself permissions to fix the problem and complete the file server migration.

Figure 23.3 shows that the Administrators group is the owner of the E:\Tax folder. Note the Change button to the right – you use that to change the ownership.

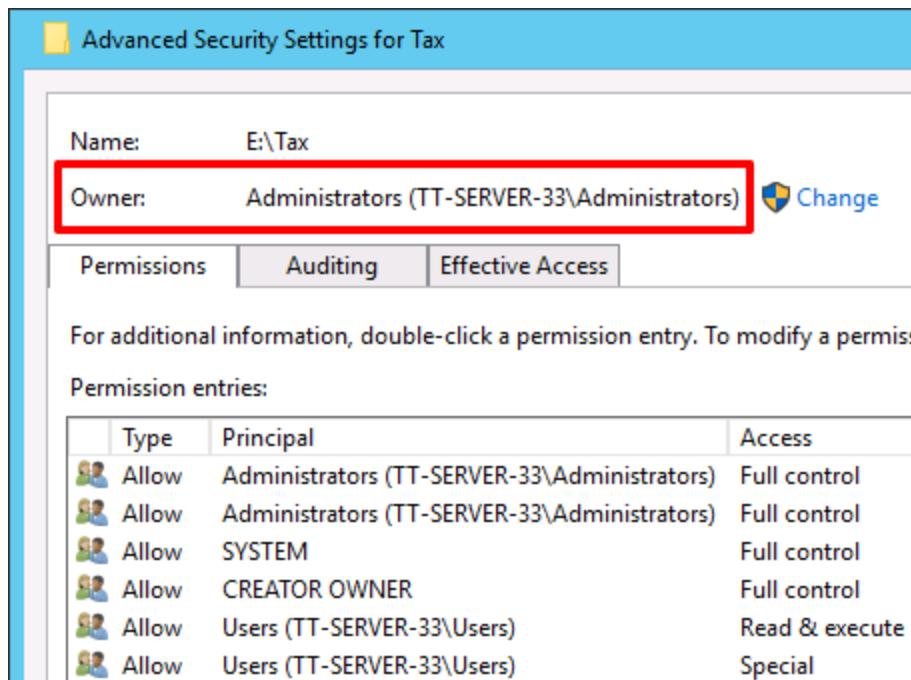


Figure 21.3 A screen capture shows that the Administrators group is the owner of the E:\Tax folder. You can modify the owners with the Change button to the right.

### Hands-on Exercise

Take ownership of the %SYSTEMROOT%\Temp folder.

The ability to change ownership and file permissions are defined as advanced or special permissions. There are more of these.

Advanced permissions include:

- **Attributes.** Attributes are metadata properties for a file or folder. You can assign read or write permissions for the attributes and extended attributes of an object.
- **Delete subfolders and files.** Deleting a file or folder is included with the Modify permission. To delete subfolders and files, you must also be granted this special permission.
- **Change permissions.** To change the permissions for an object, you must be granted this permission.
- **Take ownership.** This gives a user the ability to take ownership of a file or folder, which can lead to having full control permissions.

Figure 21.4 shows the advanced permissions for a folder named HRData. You can view advanced permissions by right-clicking a file or folder, clicking Properties, clicking the Security tab, and then click the Advanced button.

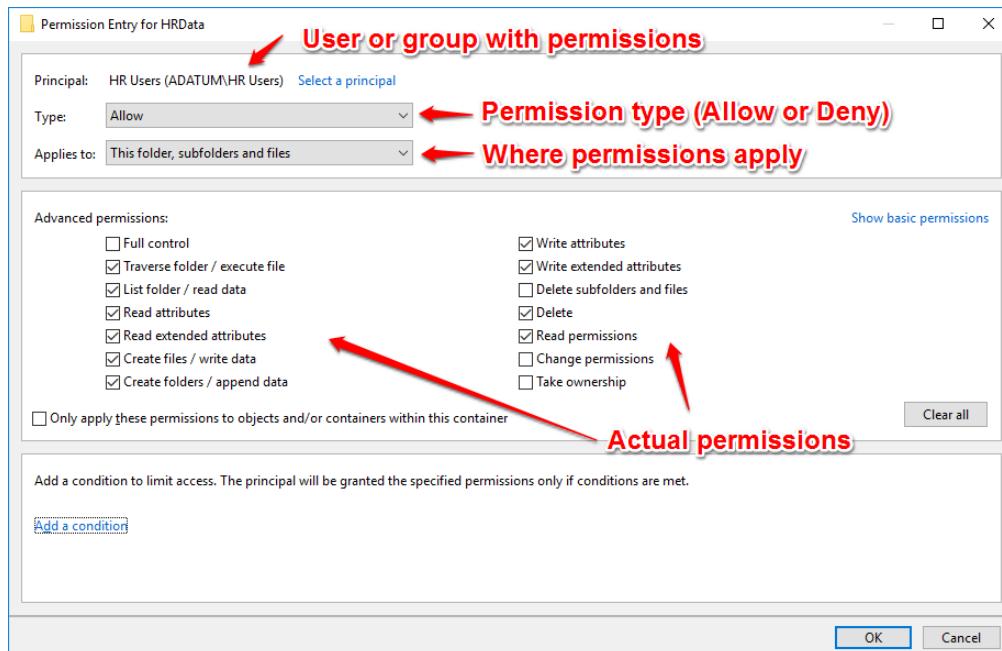


Figure 21.4 A screen capture shows the advanced permissions for a folder named HRData.

At the top of the window, the group that has permissions to the folder is shown along with the type of permissions and whether the permissions apply to just the folder or if inheritance is enabled (and thus the permissions apply to the folder, subfolders, and files). The individual permissions are shown in the middle pane.

## Effective Access

Imagine that a user notifies you that he can't access a folder. But yesterday, the user could access the folder. You check the permissions on the folder and notice 15 groups have varying permissions. You check the user's group memberships and notice that the user is a member of 43 groups. Some of the groups are nested into other groups. It sounds complex, doesn't it? That's because it is. And without a tool to help you look at the permissions, you must spend quite some time to figure out what's wrong. Luckily, Microsoft built a tool named Effective Access that calculates and displays all the permissions for a user. You can quickly figure out the access that the user has (or doesn't have) based on their group memberships.

Figure 21.5 shows an administrator checking the effective permissions for a group named HR Users for a folder named HRData.

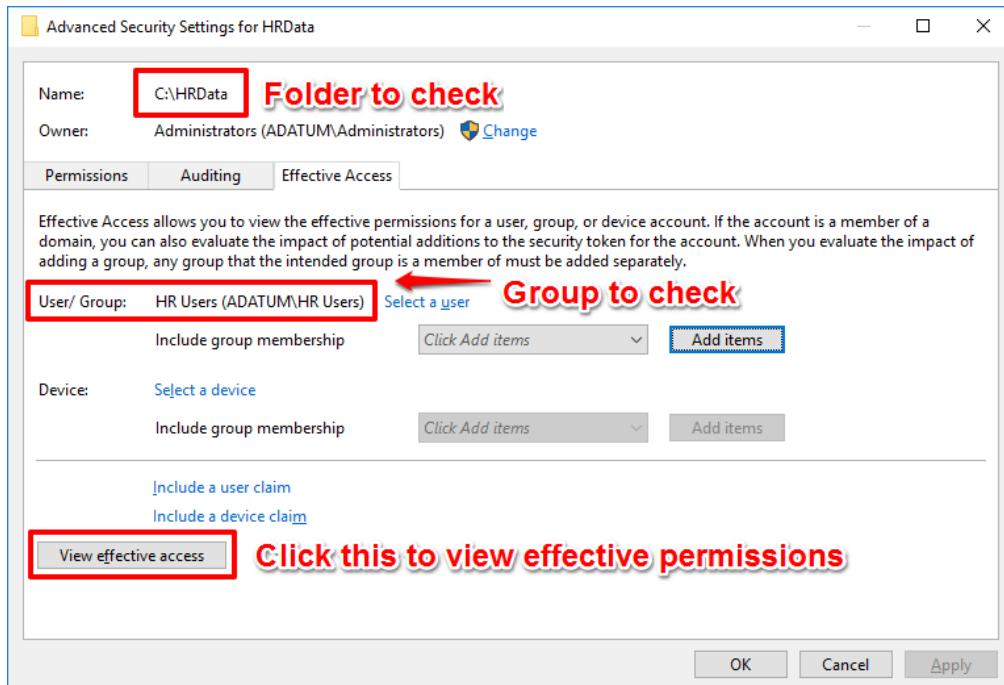


Figure 21.5 A screen capture shows the Effective Access tab of the advanced security settings. The administrator is checking effective permissions for the C:\HRData folder for a group named HR Users.

To figure out the effective access for a group:

1. Right-click the folder that you want to check for effective permissions and then click **Properties**.
2. Click the **Security** tab, click the **Advanced** button, and then click the **Effective Access** tab.
3. Select a user or group for which to view the effective permissions. You can also include group membership or specific devices.
4. Click the **View effective access** button to display the effective permissions.

### Hands-on Exercise

View the effective access for the Everyone group for the %SYSTEMDRIVE%\Users folder.

Beyond NTFS permissions, there are other factors that dictate whether somebody can access data on NTFS-formatted volumes. One of the key factors is whether the data is encrypted. Let's look at file and folder encryption with Encrypting File System next.

## Encrypting Files and Folders

Imagine that you have sensitive data stored on laptop computers in your company. Some data is personal user data such as compensation information. Other data is company information such as customer lists and proposals. Now imagine that a malicious person steals one of the laptops. The person copies all the data from the hard drive to his own computer. Without encryption, the individual has access to all the data. But if the data has been encrypted, he cannot access it! Windows offers a couple of encryption technologies, but we will focus on Encrypting File System (EFS), which encrypts at the file and folder level. Another technology, BitLocker, encrypts at the volume level, but we don't cover it in this book because it is an advanced topic that should be tackled after you are familiar with file and folder encryption with EFS. EFS is built-into Windows operating systems. You can use just BitLocker, just EFS, or you can use both together! Let's look at how you use EFS to protect data. Let's start by encrypting a single file.

To encrypt a file with EFS, perform the following steps:

1. Right-click the file that you want to encrypt and then click **Properties**.
2. On the **General** tab, click the **Advanced** button.
3. In the **Advanced Attributes** window, click the **Encrypt contents to secure data** checkbox and then click **OK**.
4. In the file's **Properties** window, click **OK**. An Encryption Warning window will prompt you to find out if you want to encrypt just the file or the file and its parent folder. In this scenario, click the **Encrypt the file only** radio button and then click **OK**.

Figure 21.6 shows the Advanced Attributes window where you can choose to encrypt a file.

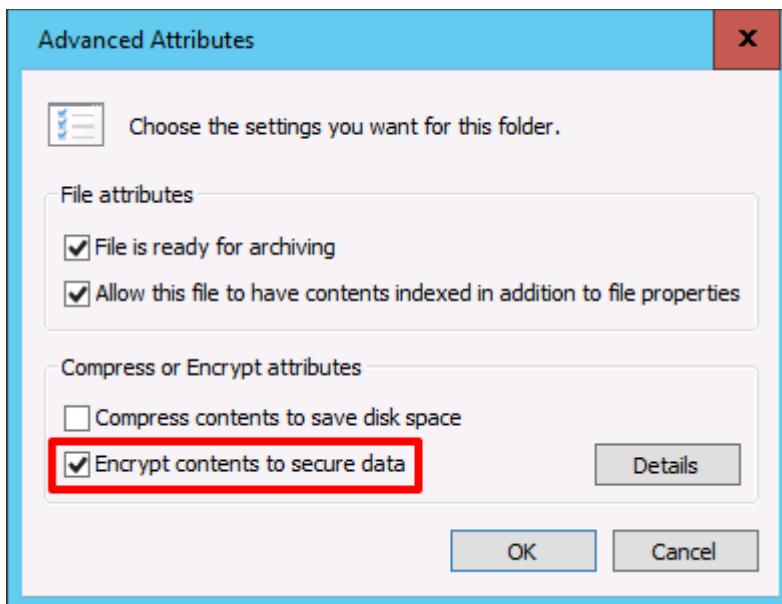


Figure 21.6 A screen capture show the option to encrypt a file with EFS.

After you encrypt the file, the filename will be displayed in green text to visually indicate that it is encrypted. Depending on the File Explorer view that you have, a lock icon will also be displayed on encrypted files. You can configure File Explorer so that encrypted files are displayed just as unencrypted files. However, I like when encrypted filenames are displayed in green because I can quickly see if a file is encrypted (or not) which helps me during troubleshooting situations.

Anybody that can log on as a user that uses EFS can decrypt files so it is important to protect the user account (a strong password is one way to protect the user account). When you first encrypt a file with EFS, Windows will generate a self-signed digital certificate that is used for EFS. The certificate is added to the user's certificate store. The private key, stored as part of the certificate, should be protected (don't share it and don't export it to an insecure location). You can back up the private key to a secure location to ensure that you can gain access to your data if the private key is lost.

Some companies have a public key infrastructure (PKI) which issues digital certificates to users and computers. In such cases, EFS can use certificates from the PKI. This centralizes the management of the certificates and certificate backups. It also enables a company to create a way to gain access to anybody's encrypted files by using a special account called a recovery agent. A recovery agent is used when a user is unable to access their encrypted data or when encrypted data must be decrypted but the user no longer has a user account associated with the encrypted data. A centralized system also enables administrators to use key archival where private keys are archived for easy recovery if something goes wrong.

### **Hands-on Exercise**

Enable EFS on a folder. How did the contents of the folder change?

## **Auditing Files and Folders**

You recently helped your company set up a file server to store company data. You used NTFS and share permissions so that only authorized users can gain access to sensitive data. You even encrypted some of the data with EFS. A few months go by. Some internal users are discussing a company merger that they aren't supposed to know about. The management team suspects that an unauthorized user found data about the merger on the file server. How can you find out? In this scenario, you can use auditing. Auditing maintains logs of data access. You can configure your servers to audit just a little bit of data, which makes it easy to work with (smaller amount of data, faster to work with, easier to search and filter). Or, you can configure your servers to audit just about everything, which makes it a bit harder to work with (huge amounts of data, slower to work with, harder to search and filter). In many companies, I recommend a balanced approach which gives you enough data to meet your requirements without overwhelming the administrators with excessive data.

## Usage scenarios for auditing

There are many scenarios where auditing the access to files and folders is necessary. Before you are ready to use auditing in these scenarios, you must turn it on and configure it so that it is capturing data. Some common reasons to use auditing are:

- **Find out who deleted data.** What do you do if a critical file suddenly goes missing from a file server? Restore it – that's one answer. But it is still a good idea to find out the root cause so you can minimize the chances of it happening again. Auditing can tell you who deleted a file and when the file was deleted.
- **Find out who changed permissions.** Imagine that a sensitive document that was restricted to the HR department was suddenly viewable by the entire company. You would quickly change the permissions to fix the issue. Then, you can use auditing to find out who changed permissions and when they changed.
- **Find out who edited a file.** A proposal is being prepared for a new customer. The quantity and price in the proposal change unexpectedly. You are trying to finalize the proposal so that you can send it to the customer. But first, you need to figure out who edited the file so you can see if the changes were authorized and approved. An administrator can use auditing to find out who edited the file.
- **Find suspicious data access attempts.** For a typical file server, the auditing logs will show a large amount of file reads, file updates, and file deletions. But what would be unexpected is a large amount of unsuccessful access attempts to various data on the file server. Such attempts might indicate malicious activity in your network. You can use auditing to figure out which user account is the cause of the unsuccessful access attempts.

There are two types of auditing in Windows Server. Standard auditing, which has been around in Windows Server since Windows 2000 Server, and Advanced auditing, a newer and more powerful version of auditing (introduced in Windows Server 2008 but made more easily accessible in Windows Server 2008 R2). When you first learn to configure and use auditing, you should use standard auditing to gain experience and a comfort level. Then later, you can explore advanced auditing. In a nutshell, advanced auditing does everything standard auditing does but it provides for more granular control of what is audited. And this can help you reduce the total amount of auditing data captured while still meeting your audit requirements. For this book, we are going to show you how to enable standard auditing but we will only briefly mention advanced auditing. To learn more about advanced auditing, see [https://technet.microsoft.com/en-us/library/dd408940\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/dd408940(v=ws.10).aspx) (note that the content was written for Windows Server 2008 R2 but is still valid today).

Like permissions and effective access, auditing can be performed from the advanced security properties of a file or folder. By default, when you add a user or group object to be audited, it will audit successful reads of a file and folder. This can cause several auditing events to be generated because anytime a user opens the folder or a file, an auditing event is written to the Security event log. If you audit too many categories, you can quickly fill up your log file. So, it is important to have event log archiving enabled so that you can maintain auditing data for an extended period. I typically recommend having about 30 days of auditing data (but this will vary

based on compliance and/or security requirements). This usually requires event log archiving because a Security log often has multiple gigabytes of data written to it daily (when auditing is broadly enabled). A better solution is to adjust the auditing configuration to capture the specific events that you are interested in. Additionally, you should consider whether you need to audit the entire file server or just folders containing sensitive data.

Figure 21.7 shows a configuration where deletions are audited, permissions changes are audited, and ownership changes are audited for the Authenticated Users group on the E:\Tax folder. Notice the exception at the bottom. In this case, an exception has been created. While auditing applies to the Authenticated Users group (basically all company users), the IT administrators have an exception because they are in the middle of a file server migration and the migration would generate an unwieldy amount of auditing data.

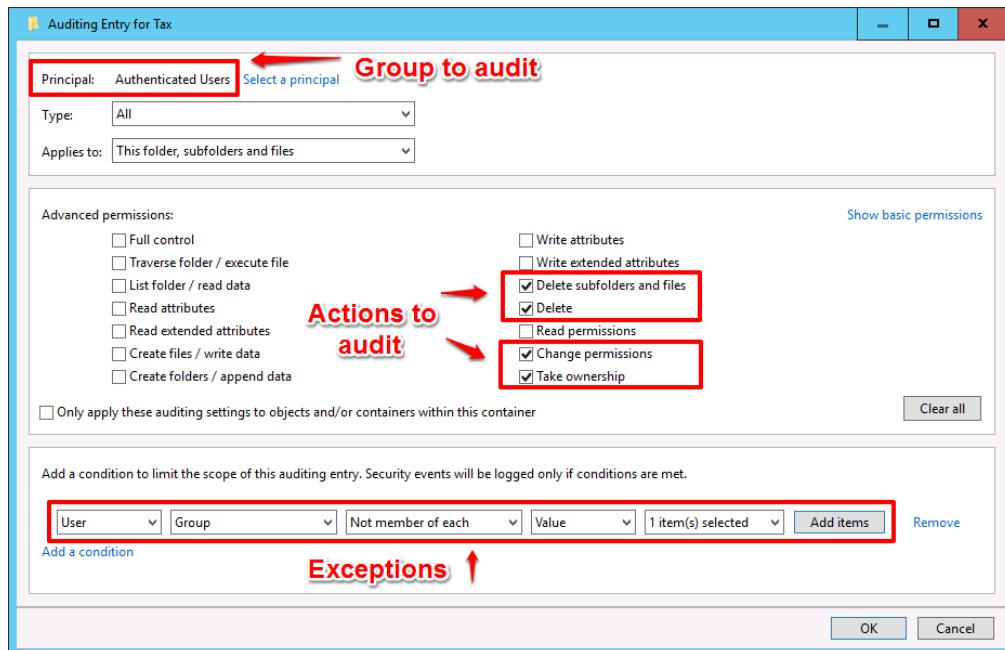


Figure 21.7 A screen capture shows an auditing configuration for the Tax folder. In this case, the Authenticated Users group is being audited for deletions, permissions changes, and ownership changes, unless the user is part of the defined exception.

By the way, remember how at the beginning of the chapter we talked about NTFS inheritance? Well, inheritance is also enabled by default for auditing! When you enable auditing for a parent object, any child objects with inheritance enabled will also receive the auditing configuration.

Figure 21.8 shows the 2016 folder with auditing enabled for the Authenticated Users group. However, instead of the auditing explicitly being listed for the 2016 folder, it has been inherited from the parent folder (Sales).

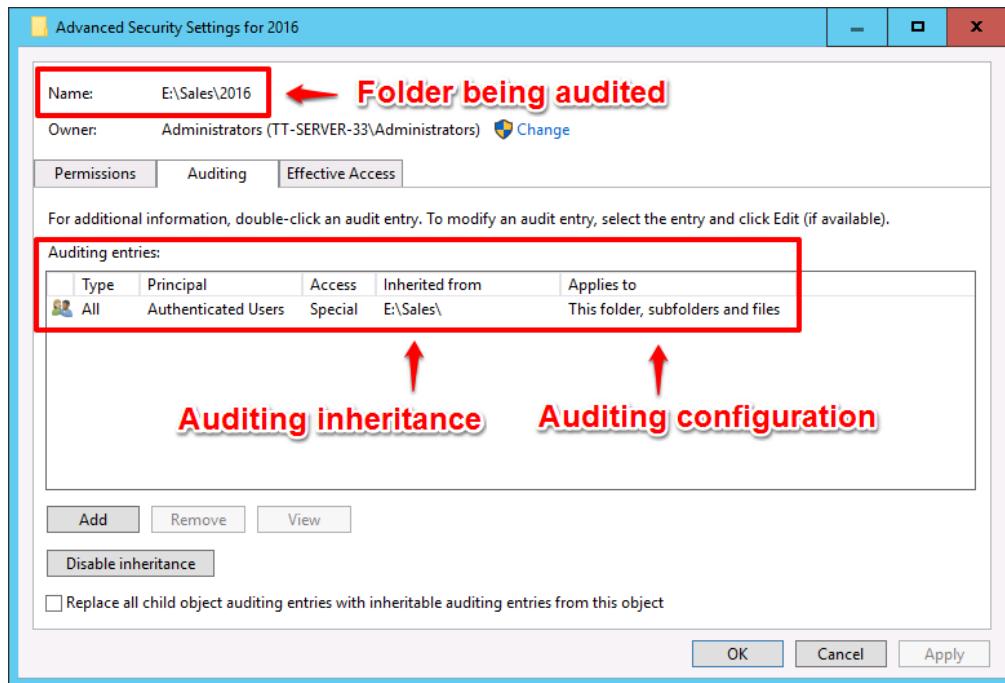


Figure 21.8 A screen capture shows the Sales folder that has an auditing entry inherited from the parent folder (Sales).

## How to enable auditing at the server level

Now that you have an idea what auditing is, let's enable it. First, you need to enable auditing for object access on the server. You can do this in the Local Security Policy tool (under Administrative Tools). Run the tool and then perform the following steps:

1. In the Local Security Policy, expand Local Policies in the left pane and then click **Audit Policy**.
2. In the right pane, double-click **Audit object access**. In the Audit object access Properties window, click the **Success** checkbox and the **Failure** Checkbox and then click **OK**.

This is a one-time task per server. When you are working with many servers, you can use a GPO to do the same thing (but across many servers or all servers). You can also use advanced auditing and enable global object access auditing (although we don't cover that in this book).

## How to enable auditing at the folder level

Once a server is configured for auditing object access, you can enable auditing at the folder level. Enabling auditing on a single folder on a single server is simple.

To enable auditing on a folder for the Authenticated Users group and all actions, perform the following steps:

1. Right-click the folder that you want to audit and then click **b**.
2. In the folder's **Properties** window, click the **Security** tab.
3. On the **Security** tab, click the **Advanced** button.
4. In the folder's **Advanced Security Settings** window, click the **Auditing** tab.
5. On the **Auditing** tab, click the **Add** button.
6. In the folder's **Auditing Entry** window, click **Select a principal**.
7. In the **Select User, Computer, Service Account, or Group** window, type Authenticated Users and then click **OK**.
8. In the **Type** dropdown menu, click **All**.
9. In the **Basic permissions** section, click the **Full control** checkbox and then click **OK**.
10. In the folder's **Advanced Security Settings** window, click **OK**.
11. In the folder's **Properties** window, click **OK**.

That might seem like a lot of steps. And it is. That becomes clear when you want to enable auditing on several folders across many servers! For now, get experience with auditing and become familiar with it. Then later, try using advanced auditing for more granular control.

### **Hands-on Exercise**

Enable auditing for a folder using the default settings. Create and open a few files in the directory, and then review the Security event log to look at the auditing information.

In this chapter, we showed you what NTFS permissions are. We showed you how to control access to your data with NTFS permissions. We walked through file and folder ownership and how you can become the owner of a file or folder to enable you to change file and folder permissions. We showed you how to protect your data with EFS. And then we showed you how to audit when people are accessing data, changing data and permissions, deleting data, or attempting to access data to which they don't have access. Let's jump to the lab and test some of your NTFS-related knowledge.

### **Lab**

This lab is designed to validate your retention of information from this chapter and perform some storage management tasks. If you haven't already completed the Hands-on Exercises in this chapter, do that now and then come back to perform the lab exercises.

## Set permissions

Perform the following tasks:

- If necessary, create a folder and two user accounts. Add three text files to the folder.
- Grant Full Control permissions for the first user account.
- Configure the folder to deny read permissions to the second user account.
- Using each user account, test the permissions.

## Viewing effective permissions

Using the existing folder, perform the following tasks:

- Use the Effective Access tab and view the permissions for the first and second user account.
- Create a third user account without granting any permissions and view the effective permissions.

## Using EFS

Using the existing folder, perform the following tasks:

- Enable EFS on the folder.
- Sign into each user account and attempt to view the contents of the folder.

## Auditing files and folders

Using the existing folder, perform these tasks:

- Enable the Audit object access setting in the Local Security Policy on the server.
- Enable auditing for all permissions in the folder for the Authenticated Users group.
- Using one of the user accounts, write data to the file and then review the Security event log.
- Using a different user account, delete a file from the folder and then review the Security event log.

## CHAPTER 22: MANAGING SHARED FOLDERS AND FILE SERVICES

---

Over the last couple of chapters, we've talked about storage and NTFS permissions. Both technologies play a key role in shared folders and file services. That's why we covered them first. Now you have a good foundation on which we can build upon! In just about every network, there are folders that are shared amongst teams or across an entire organization. These folders enable people to share data and collaborate. On the backend, Windows Server has file services that provide the sharing functionality. As an administrator, your job is to manage and troubleshoot these shared folders and file services. And that's what we'll show you in this chapter.

We start with an introduction to administrative shares, which are default shares available to administrators. Then we look at share permissions, which work a little differently than NTFS permissions. Then, we learn what happens when you combine NTFS and share permissions--because it can get a little tricky. We also look at a technology that prevents users from seeing shared folders if they don't have access to them. For example, if a user in IT doesn't have access to a shared folder named Payroll, then the user won't see it when looking at available folder shares. Lastly, we learn about a key management tool named the File Server Resource Manager, so you can work with quotas and quota templates (to ensure that users are limited in how much data they can store on a file server volume), file screening (to ensure that users do not store prohibited data such as .mp3 files), storage reports (which provide an overview of your shared folder environment), and file classification (which enables you to classify data so that you can dynamically assign permissions or more easily report on the data).

The sections in this chapter are based on the following prerequisites:

- The File and Storage Services role is already installed on your server.
- The file server is a member of an Active Directory Domain Services domain.

### Administrative Shares

Windows automatically creates a series of hidden shares as part of the operating system installation and configuration. These shares provide administrators (like you!) with access to administrative resources on remote systems. For example, as an administrator, you can use the administrative shares to upload a new security patch to the C:\Temp folder on a remote server. Any shared folder shares can be hidden. A You can identify a hidden share can be identified by the \$ character following the share name. Hidden shares are not displayed when you browse to a server. But, if you know that a hidden share is available and you have rights to it, you can manually type the path to the hidden share and use it. As an administrator, you can see hidden shares in the administrative tools such as Computer Management and Server Manager. As you create new shared folders, you can optionally mark them as hidden using the same hidden naming convention. The following list covers describes some of the default administrative shares and their purpose:

- <**DriveLetter**\$>. These shares provide access to the root of a volume on a computer. For example, the C:\ volume can be accessed by navigating to \\server\c\$. Many administrators commonly use these administrative shares to copy installation files to servers or copy log files from servers to the administrative computer.
- **ADMIN\$**. The admin share provides access to the Windows directory on the remote system. This share is sometimes used for remote administration of the system. Mostly, this share is used by some legacy applications and tools. If you can't get to the ADMIN\$ share, the tools don't work.
- **PRINT\$**. The print share provides access to the print drivers on the remote system. This share is used to administer the printers on the remote system.

Adding new roles to a Windows Server can introduce additional administrative shares. It is worth noting that not all administrative shares are hidden. For example, when you promote a member server to a domain controller, the computer adds the following shares:

- **SYSVOL**. The sysvol share is responsible for delivering policy settings to domain members.
- **NETLOGON**. The netlogon share is responsible for delivering logon scripts to domain members.

## Viewing shared folders

As an administrator, it is helpful to see which shares are active on a server. For example, if a user reports that she can't get to a share, one of your first checks is to see if you can get to it. If not, you might want to check if the share exists on the server at all! There are a few ways to accomplish this. The first is through the Computer Management console.

1. On the file server, open the Computer Management console.
2. In the left pane of the Computer Management console, expand **System Tools**, expand **Shared Folders**, and then click the **Shares** folder.

Share Name	Folder Path	Description
ADMIN\$	C:\Windows	Remote Admin
CS	C:\	Default share
CertEnroll	C:\Windows\system32\...	Active Directory Certificate ...
DeploymentShare\$	F:\DeploymentShare	MDT Deployment Share
FS	F:\	Default share
IPCS		Remote IPC
print\$	C:\Windows\system32\...	Printer Drivers
temp	C:\temp	
TT Accounting	F:\tt-accounting	
TT Engineering	F:\tt-engineering	
TT HR	F:\tt-hr	
TT Marketing	F:\tt-marketing	
TT Sales	F:\tt-sales	

**Share names**      **Storage location of shares**

Figure 22.1 The Computer Management console shows all shared folders on a system.

### **Hands-on Exercise**

On a Windows server, run Computer Management and view the current shares. Then, attempt to connect to one of the hidden administrative shares from a remote computer.

The second method for viewing a list of shared folders is through the command line. There are commands for both PowerShell and the command prompt, and they include the following:

From a command prompt, run the following command:

```
net share
```

From a PowerShell prompt, run the following command:

```
Get-SmbShare
```

### **Hands-on Exercise**

On a Windows server, run the Get-SmbShare command to view the current shares.

Now that you have a better understanding of what an administrative share is and how to view them, let's look at standard shared folders and how to manage the permissions.

## **Share Permissions**

You are an administrator for Tailspin Toys. Your manager requests that you create a new home folder structure on the file server. The project includes the following requirements:

- Each employee must have their own home folder.
- The home folders must be labeled to match the employee's AD DS account name.
- Each employee must only have read, write, and execute permission to their home folder.
- Employees must only see their home directory when browsing the "Users" file share.

This scenario is very common among many organizations. Home folders are still widely used and proper permissions are important for protecting employee data. Over the next few sections we address each of the requirements and how to solve them. To begin, let's look at how permissions are applied to shared folders.

Share permissions are assigned to a shared folder and only control access when a shared folder is accessed over the network (as compared to signing in locally to a server and navigating to the folder in File Explorer). There are three basic levels of permissions that can be assigned through share permissions: Full Control, Change, and Read. By default, in Windows Server 2016, the Everyone group has share permissions of Read when you create a share with File Explorer.

Or, when you create a share using Server Manager (and the SMB Share – Quick profile), the Everyone group is granted share permissions of Full Control. Note that default share permissions and behavior across Windows versions can vary so always double check when performing your administrative work.

To assign or modify the share permissions for a shared folder:

1. In Server Manager, navigate to File and Storage Services and then click **Shares**.
2. Right-click on the share that you want to change permissions on and then click **Properties**.
3. In the **Properties** window, click **Permissions** and then click **Customize permissions**. This opens the Advanced Security Settings for the shared folder.
4. To update the Share permissions, apply your changes to the **Share** tab. You have the option to add, remove, or edit share permissions.
5. Click **OK** to save the changes.

Figure 22.2 illustrates the default share permissions for a new shared folder created through Server Manager. In this example, we created a root directory named Users for the Tailspin Toys employee home directory structure. The share permissions are only assigned to the shared folder. We adjust access by using a different type of permissions, which we'll discuss in the next section.

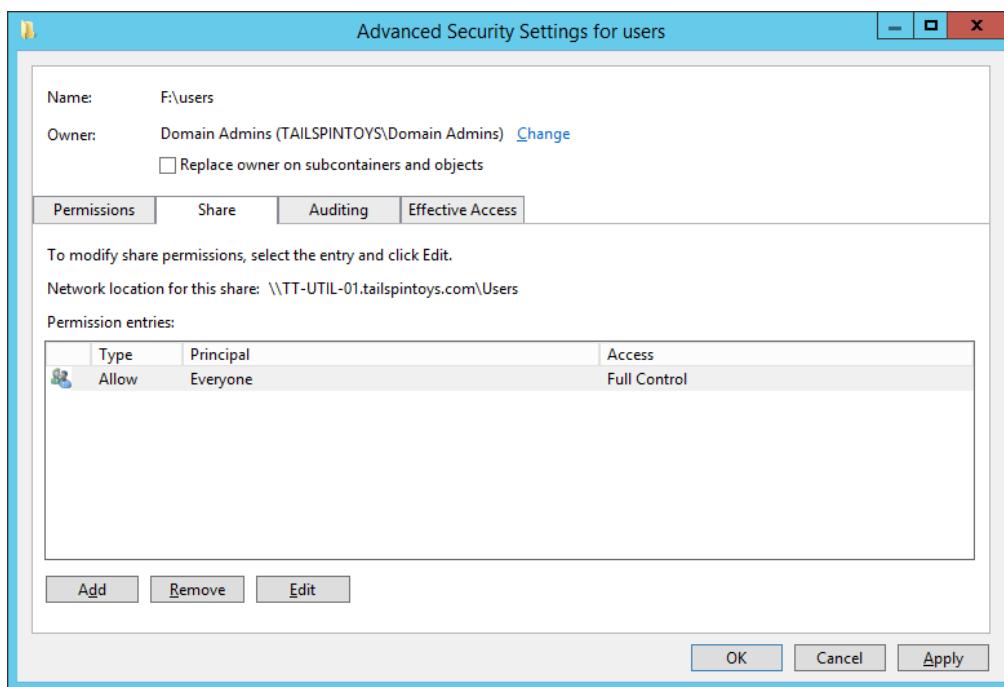


Figure 22.2 Share permissions are managed through Advanced Security Settings of the shared folder.

## Hands-on Exercise

Use Computer Management to create a new share using any existing folder and the default settings. Run Server Manager and modify the share permissions so that only Domain Admins can access the share over the network.

Share permissions are a core component for assigning access to a shared folder, but there is another permission type that makes up the overall access control list. In the next section, we look at NTFS permissions and how they interact with share permissions.

## Combining NTFS and Share Permissions

While we talked about NTFS permissions in the last chapter, here we focus on their role in shared folders. NTFS permissions are used for every standard folder and every shared folder. Unlike share permissions, NTFS permissions affect access both locally and over the network. When you combine share permissions with NTFS permissions, there are some important rules to be aware of:

- **Share permissions and NTFS permissions are both taken into consideration when determining shared folder access over the network.** If a difference is found, the most restrictive permissions are applied. For example, if a user has NTFS permissions that grant read and write access and has share permissions that grant only read permissions, then the user cannot write to the shared folder over the network. However, if the same user accesses the folder locally (such as from the console of a server), the user's NTFS permissions are used to calculate permissions, so the user can read and write.
- **Deny permissions always override allow permissions.** Imagine that there is a user named Sue. And Sue is a member of a bunch of security groups. One group gives Sue full control to a folder. Another group gives Sue the Deny Read permission to the same folder. Since deny overrides allow, Sue cannot gain access to the folder!
- **You can effectively manage shared folder access by exclusively using NTFS permissions.** To do so, set the share permissions to Full Control for the Authenticated Users group. In a high-security environment, you should opt to set the minimum amount of permissions for share permissions and NTFS permissions though.

To assign or modify the NTFS permissions for a shared folder, perform the following steps:

1. In Server Manager, navigate to File and Storage Services and then click **Shares**.
2. Right-click on the share that you want to change permissions for and then click **Properties**.
3. On the **Properties** window, click **Permissions** and then click **Customize permissions**. This opens the Advanced Security Settings for the shared folder.
4. To update the NTFS permissions, make your changes on the **Permissions** tab. You have the option to add, remove, and edit share permissions.

- Click **OK** to finish.

### Hands-on Exercise

Use Server Manager to customize the NTFS permissions on an existing share. Grant the Domain Admins group Full Control and remove all other groups and users from the NTFS permissions.

Figure 22.3 shows an example of the NTFS permissions assigned to the Users folder. In this example, we have applied a few changes. First, we disabled permission inheritance. This prevents NTFS permissions of parent folders from inheriting down to our shared folder (which might change the permissions that we want). Second, we granted the Domain Admins group Full Control over the folder and subfolders. This allows the domain administrators to manage the folders in the future. And finally, we granted the Authenticated Users group Read and Execute permissions to just the folder. This enables domain users to access the folder but prevents them from accessing any subfolders unless they have permissions assigned at the subfolder level. We come back to why this is in the next section.

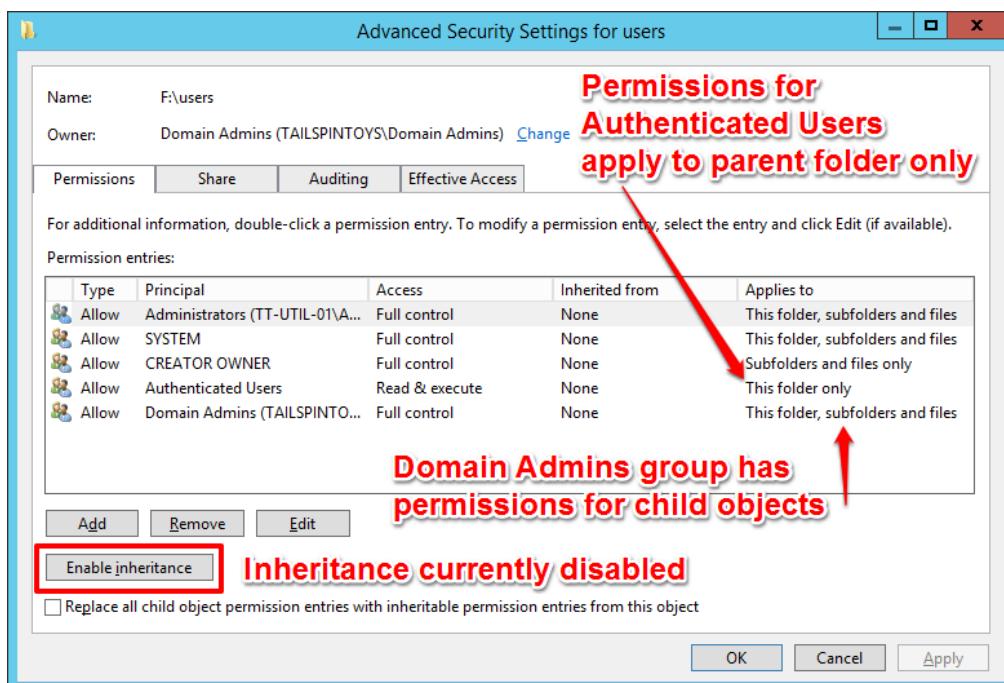


Figure 22.3 NTFS permissions are managed through Advanced Security Settings of the shared folder.

With the Users shared folder configured, we can now create the various home folders underneath. To limit access to folders, we grant each user Modify access to just their assigned home folder. This prevents users from accessing other users' folders.

At this point in the chapter, we now have our home folder structure created and you have a good understanding of how share permissions and NTFS permissions work side-by-side. Next,

we look at a technology that helps limit visibility to shared folders that users do not have access to – Access-Based Enumeration (ABE).

## Access-Based Enumeration

Often a shared folder is accessed by more than one user, and it is likely that there are subfolders that some users need access to while others don't. In these situations, you can leverage share permissions and NTFS permissions to limit access, but the restricted folders remain visible. This presents a few negative effects. First, your users see a list shared folders that have no importance to them. Second, you may have important folders that you wish to remain hidden from the general population. For example, you might have a shared folder for an unannounced merger or a secret company project.

ABE is a setting that can be enabled on your shared folders. When enabled, this setting displays only the files and folders that a user has permissions to access. If a user does not have Read (or equivalent) permissions for a folder, Windows hides the folder from the user's view.

To enable access-based enumeration on your shared folder:

1. In Server Manager, navigate to File and Storage Services and click **Shares**.
2. Right-click on the share that you want to enable access-based enumeration on and then click **Properties**.
3. In the **Properties** window, click **Settings**.

4. Click the **Enable access-based enumeration** checkbox, as shown in Figure 22.4, and then click **OK**.

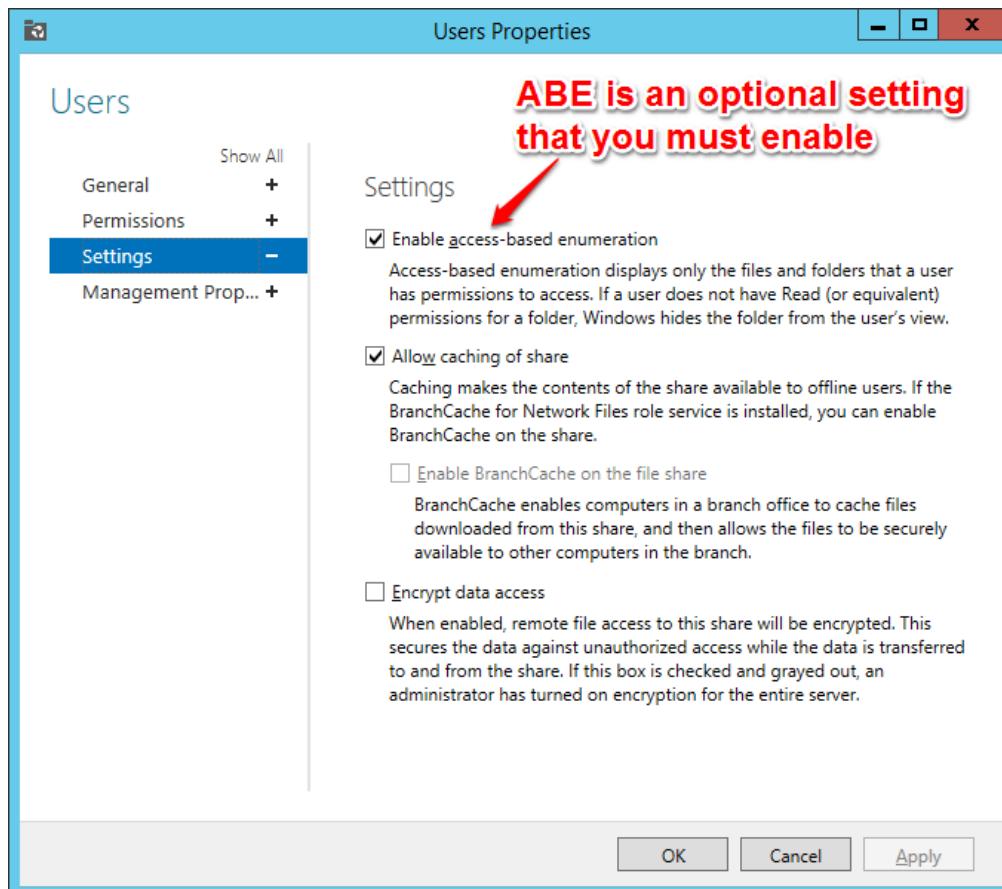


Figure 22.4 You enable access-based enumeration through the settings page of the shared folder properties window.

### Hands-on Exercise

Enable ABE for an existing share. After ABE is enabled, find out if you can still see the shared folder over the network by navigating to the server with a user account that does not have access to the folder. For example, if your server name is Server01 and the shared folder is named Share01, navigate to `\server01` and ensure that you cannot see Share01.

If you recall, earlier in this chapter, we granted the Authenticated Users group Read and Execute permissions to the Users shared folder. Now that we have enabled access-based enumeration, the individual home folders that we created under the Users folder are not visible to everybody. Instead, each user sees only their own folder under the Users folder.

Now that you have a good understanding of ABE, we are going to shift gears and dive into some of the other technologies surrounding shared folders and file services.

## Using Shadow Copies

Shadow copies, which is a feature that can be enabled on shared folders, enables users to view and restore the contents of shared folders from previous points in time. This provides users with self-service IT, which is a good thing by most users! Shadow copies are not meant to replace backups, though. That's because backups are generally kept for a period and include all data and all configuration information from a server. If a server fails, shadow copies are not available. But backups are, and you would use those to restore the server and data. Shadow copies can be beneficial with:

- **Deleted files.** Shadow copies can be used to restore files that have been deleted.
- **Overwritten files.** Shadow copies can be used to restore files that were mistakenly overwritten.
- **Compare versions.** Shadow copies can be used to compare a current version of a file with a previous version. This is a helpful option when you are tracking changes.

Enabling and managing shadow copies is performed through the Computer Management console. To enable shadow copies on your shared folders:

1. On the file server, open the **Computer Management** console.
2. In the left pane of the Computer Management console, expand **System Tools**.
3. Right-click on **Shared Folders**, click **All Tasks**, and then click **Configure Shadow Copies**. This opens the Shadow Copies dialog window.
4. Select the volume that you wish to enable shadow copies on and then click **Enable**. Additional settings can be configured, including the location where shadow copies are stored, the maximum size, and the schedule that is used to create new shadow copies.

In Figure 22.5 you can see an example of how shadow copies are accessed by the user. In this example, we are looking at a user's home folder, which has two snapshots available. The contents of each snapshot can be reviewed and restored.

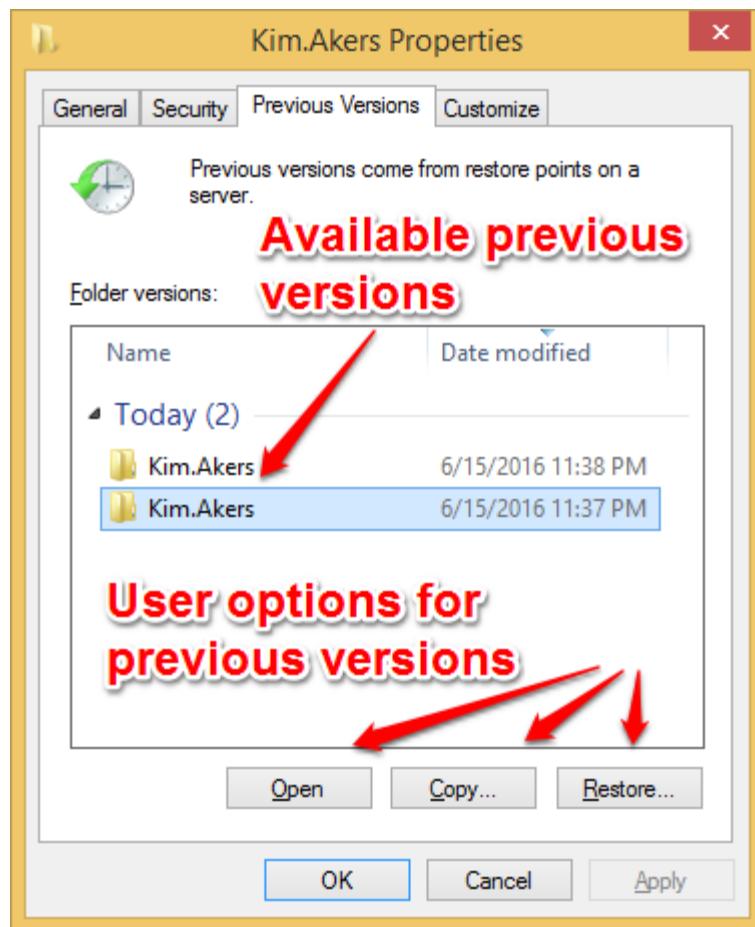


Figure 22.5 Shadow copy versions can be reviewed and restored by viewing the Previous Versions tab of the file or folder properties window.

### Hands-on Exercise

Use Computer Management to enable shadow copies on a system volume or data drive. Then, set the schedule so that a new shadow copy is created once per day, at 12:00 PM.

## The File Server Resource Manager

The file server resource manager is an additional server role that should be installed on your file server for optimal visibility and management of your file services environment. It provides a series of features that assist in the management of your server and the data it holds. There are five key features that the file server resource manager provides. These include the following:

- **Quota Management.** This feature enables you to assign file and folder size restrictions on your shared folders. You also can create quota templates, which can be referenced when new volumes or folders are created. Imagine that you have 100 users and 1TB of disk space for their home folders. Each user gets about 10GB of space. What if a couple of users decide to use 50GB or 100GB of space instead of 10GB? That's where quotas come in. Quotas enable you to run reports to find people using more than 10GB. Or, optionally, you can block users from using more than 10GB.
- **File Screening Management.** This feature enables you to designate which file types are not allowed to be stored in your shared folders. For example, you can restrict .mp3 files so that they can't be stored on the file server.
- **Storage Reports Management.** This feature provides you with historic visibility on usage trends and data analysis. These reports can assist with managing growth and responding to unauthorized file types. For example, if your manager asks you to come up with a budget for storage in the next fiscal year, you can use storage reports to estimate the amount of new storage you will need and set the budget accordingly.
- **Classification Management.** This feature enables you to classify files based on type. Those file classifications can then be used to assign policies and tasks. For example, you can assign a confidential classification to files that contain sensitive company information.
- **File Management Tasks.** This feature utilizes the file classifications. Once defined, you can assign actions to files. For example, you can encrypt all .docx files or files in a specific folder.

After installing the File Server Resource Manager role, you can access the management console by opening **Server manager**, clicking **Tools**, and then clicking **File Server Resource Manager**.

Now let's take a closer look at each of the individual features.

## Quotas and Templates

Quota Management is the first node in the File Server Resource Manager console. Quotas are important when you need to restrict usage on a file share or monitor usage for future growth. The available sections under Quota Management include the following services:

- **Quotas.** The quotas section is where you assign quotes to your shared folders. For example, you can assign each user home folder a 10 GB quota limit and send a warning notification if the user exceeds 85% of the space.
- **Quota Templates.** The quota templates section is used to create new templates or modify existing ones. A quota template defines the quota limit and notification thresholds. In most environments, you should create templates based on your needs. While there are built-in templates that you can use, they probably won't meet all your needs.

## File Screening

File Screening Management is the second node in the File Server Resource Manager console. File screening is a feature that can greatly assist with monitoring the data types on your file

server, and reclaim space when unauthorized content is discovered. The available sections under File Screening Management include the following services:

- **File Screens.** The file screens section is where you assign the screening rules for allowing, blocking, or monitoring of file types. For example, you can assign a file screen that monitors and warns you when .mp3 files are stored on the file server. You can use file screens to block all types of media files (such as music and movie files) or other types of files that are disallowed by your company.
- **File Screen Templates.** The file screen templates section is used to create new templates or modify existing ones. A file screen template defines the file types that you are monitoring, and the behavior that takes place when they are discovered.
- **File Groups.** The file groups section is used to define which file types you are monitoring for. By default, there are a handful of pre-defined groups. You have the option of modifying these groups or creating new ones to meet your needs.

## Storage Reports

Storage Reports Management is the third node in the File Server Resource Manager console. Storage reports offer a great amount of visibility for the data that resides on your file server. Common management tasks like reclaiming disk space can be greatly simplified with these reports. From this location, you can create a series of scheduled report tasks, or generate specific reports on demand. There are several built-in reports available. These include the following:

- **Duplicate Files.** Lists all files that are marked as duplicates, based on file size and last-modified date. For example, imagine that your company sent out a free e-book to all employees. And most employees decided to store it in their home folder. That isn't very efficient use of your storage space. Instead, you can store the e-book on a web site or a SharePoint site. Looking at the duplicate files report can give you an idea of how people are using their storage which can help you plan future storage requirements.
- **File Screening Audit.** Lists all violations that correspond to existing file screening policies. If you use file screening, you can either flag file screen violations OR you can block the storage of files that are in violation. Often, it is a good idea to flag violations first so you can see the impact a block would have.
- **Files by File Group.** Lists all files that belong to a specified file group.
- **Files by Owner.** Lists all files, grouped by owner. Imagine that a user leaves the organization. And he stored files all over the file server. You can run a report to track those files down and then put them into an archive folder for that user in case they are needed in the future.
- **Files by Property.** Lists all files based on a specific classification property. This report requires that you specify a classification property.
- **Folders by Property.** Lists all folders based on a specific classification property. This report requires that you specify a classification property.

- **Large Files.** Lists all files that are equal to or greater than a specified size. If you ever find yourself running critically low on disk space and you can't expand the space right away, you can opt to move some large files to temporarily free up space. The large files report helps you find files that you can potentially move.
- **Least Recently Accessed Files.** Lists all files that have not been accessed within a specified number of days. In some environments, administrator opts to store active project files on high performing storage and store past project files on lower performing storage. The Least Recently Access Files report can help you find files that are no longer actively being used and might be able to be moved to lower performing storage.
- **Most Recently Accessed Files.** Lists all files that have been accessed within a specified number of days.
- **Quota Usage.** Lists quotas that exceed a specified percentage.

## File Classification and Management

Classification Management and File Management tasks are the fourth and fifth nodes in the File Server Resource Manager console. File classification is an advanced set of features that can assist with attending to manual data management tasks. For example, a classification rule can be defined to identify files with specific strings of text (such as a string that look like credit card numbers). Those files can then be removed or encrypted based on the defined tasks. The services for these sections include the following:

- **Classification Properties.** The classification properties section is used to assign a value to files, based on the rules you define. For example, you can set a property to show whether a file has a credit card number. A property must be combined with a rule for any actions to take place.
- **Classification Rules.** The classification rules section is used to create rules for classifying files, and thus applying the values defined in your classification properties. For example, you can set a rule to mark a file as Confidential if it contains a numerical string that matches a credit card number. You first need a classification property and then you create the rule. After the rule runs (manually or on a specified schedule), files that match the rules have a custom property such as "Credit Card Data" and a value such as "Yes". You can view the properties in File Explorer. Optionally, you can use a file management task to take specific actions on such files.
- **File Management Tasks.** The file management tasks section is used to define scheduled tasks that act based on certain criteria. For example, a management task can be created that encrypts files that contain credit card numbers.

## Hands-on Exercise

Use Server Manager to install the File Server Resource Manager role service. Then, open it and navigate through the tool to see the options and tasks available.

This was a big chapter! We had a lot of topics to cover to show you all about shared folders and file services. You should now have a good understanding of administrative shares, share permissions and how they work with NTFS permissions, hiding shared folders from those without access, providing users with self-service file recovery with shadow copies, and using FSRM to manage and report on your file services. In the lab, we will test your knowledge on topics from this chapter.

## Lab

### Administrative Shares

Perform the following tasks:

- On your file server, use PowerShell to output the current list of shares.
- On a client computer, access the C-Drive of your file server and create a folder named "Patches".

### Shared Folder Permissions

Perform the following tasks:

- Create a shared folder on your file server named Meeting Minutes"
- Create a subdirectory under Meeting Minutes named Staff Meetings.
- On the share permissions for Meeting Minutes, grant the Everyone group Full Control.
- On a client computer, confirm that you can access the folders named Meeting Minutes and Staff Meetings.
- On a client computer, confirm that you can create, view, and delete files and folders in both directories.

### Combining NTFS and Share Permissions

Perform the following tasks:

- Create a global security group in Active Directory named Meeting Minutes – Editors
- On the NTFS permissions for Meeting Minutes, grant Authenticated Users Read & Execute permissions for this folder only. Grant Meeting Minutes – Editors Modify permissions for this folder, subfolders and files.
- On a client computer, confirm that you can access the Meeting Minutes folder.
- On a client computer, confirm that you cannot access the Staff Meetings folder.
- On a client computer, confirm that you cannot create, view, or delete files and folders.

## Access-Based Enumeration

Perform the following tasks:

- Enable access-based enumeration on the Meeting Minutes shared folder.
- On a client computer, confirm that you can access the Meeting Minutes folder.
- On a client computer, confirm that you cannot see the Staff Meetings folder.

## Using Shadow Copies

Perform the following tasks:

- On your file server, turn on Shadow Copies for the volume that contains the Meeting Minutes shared folder. Use the default settings.
- On a client computer, connect to your file server and review the Previous Versions of the Meeting Minutes shared folder. You should see a single entry.

## The File Server Resource Manager

Answer the following questions:

1. Your file server is running low on disk space. You need to identify data that can be removed to free up space. In the File Server Resource Manager, where would you look for this information?
2. You need to identify all files on your file server that contain credit card information. In the File Server Resource Manager, where would you apply this configuration?
3. Your manager has requested that you block the following file types from being written to the file server: .MP3, WMA, .WAV. In the File Server Resource Manager, where would you apply this change?
4. You have recently created a shared folder for HR. You need to limit the amount of usable space. In File Server Resource Manager, where would you apply this change?

## CHAPTER 23: MANAGING PRINT SERVICES

---

As a server administrator, you get to work with a variety of technologies. That's part of what makes a server administration job a fun job. One day you might be working with Active Directory, another day might be security, and another day could be managing print services. In most organizations, there are what are often referred to as "foundational services". Such services are common to virtually all organizations and are key technologies that enable users in organizations to get their job done. For example, Active Directory provides authentication and authorization to resources. File services, covered in the previous chapter, are handled by Windows Server. DNS enables users to easily find resources. Print services, another foundational service, enable users to take what's on their computer screen and print it out to paper. Often, printouts are used to sell services, promote events, and enable contract signings. So, print services are important!

In this chapter, we are going to look at managing print services using the Print Server role service. A print server is a server that runs a service to enable printing! Along the route, we will walk through some of the basics of setting up a network printer, such as installing and managing print drivers, assigning print permissions, and working with print queues. But remember, like the rest of this book, we approach this chapter with an assumption that you have inherited an existing environment so our focus will be on maintaining that environment (instead of building it from the ground up). At the end of the chapter, we will look at printer pooling and the benefits it provides, along with how to publish your printers in Active Directory for easier visibility and access.

The sections in this chapter are based on the following assumptions:

- The Print Server role service (under the Print and Document services role) is installed on at least one server.
- The print server is a member of an AD DS domain.

### Installing Print Drivers

You are the systems administrator for Tailspin Toys. The office you support has 150 active users and three multifunction printers that are shared over the network. Currently, your support team is manually adding the printers on everybody's client computer. However, they are using an outdated print driver that is stored on the file server. Occasionally when an issue arises, they install the latest driver from the manufacturer's website. As time goes on more users start reporting problems with print reliability and missing features. You need to provide a solution that unifies print management, starting with driver version control. For this project, you setup a new server running Windows Server 2016. You install the Print Server role service (under the Print and Documents services role) and add your multifunction printers to the new server.

One of the biggest benefits of leveraging a centralized print server is the ability to control which drivers your clients use. Using a common set of validated drivers will greatly limit any print inconsistencies between users. When you support a single driver version, it simplifies the support because the same experience is expected across all client computers.

The first step in installing a new print driver is obtaining the driver from the printer vendor – often from their web site. You should test drivers on a test computer before importing them to a print server. Windows Server includes a library of print drivers, with the ability to check Windows Update for newer versions. This can be a helpful resource, but it's a good practice to check the vendor's website for the latest versions and release notes. You might be troubleshooting a compatibility issue with a Windows 10 client computer, and the notes on the vendor's website can be helpful in acknowledging whether the existing drivers support Windows 10.

Once a compatible driver is identified you can add it to the print server. There are two ways – using the GUI (in this case, the Print Management console) and using PowerShell. First, let's look at the GUI method.

### Installing print drivers using the Print Management console

Adding You add drivers to a print server is often handled by using the Print Management console because working with drivers in the GUI is simpler than from with PowerShell, especially if it is the first time you have worked with the driver.

Perform the following steps to install a new print driver using the Print Management console:

1. Open Server Manager, click **Tools**, and then click **Print Management**.
2. In the left pane of the Print Management console, expand **Print Servers**, and then expand the name of your print server.
3. Right-click **Drivers** and then click **Add Driver**.
4. In the **Add Printer Driver Wizard** window, on the **Welcome** page, click **Next**.
5. On the **Processor Selection** page, select the processor type of your client computers, and then click **Next**.
6. On the **Printer Driver Selection** page, choose a printer from the existing library, or click **Have Disk** to choose a new driver. Click **Next** to continue.
7. On the **Completion** page, click **Finish**.

After the drivers are installed, they will be displayed in the Print Management console under Drivers, as shown in Figure 23.1. For additional information and management of the drivers, right-click **Drivers** and click **Manage Drivers**. From this window, you can add or remove drivers and view the properties of each driver.

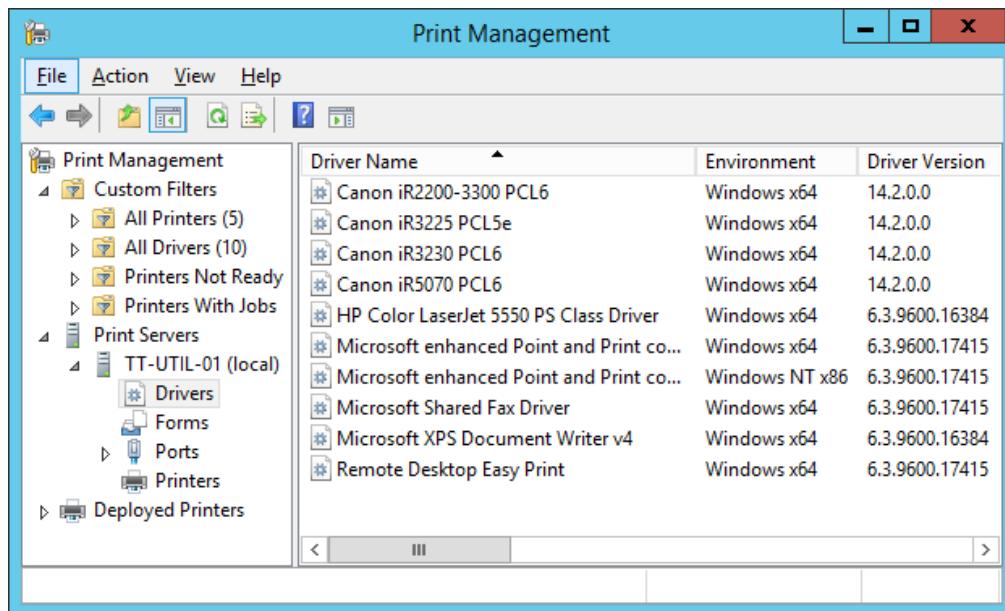


Figure 23.1 The Drivers section of the Print Management console displays all installed print drivers

### **Hands-on Exercise**

Download a new printer driver from HP or Dell. Add it to your print server using the Print Management console.

Now that you know how to work with the Print Management console to install and manage print drivers, let's look at how to do it by using Windows PowerShell.

### **Installing print drivers using the command line and PowerShell**

Sometimes, the GUI tool provides the best administrative experience. Other times, PowerShell provides the best administrative experience. In the case of print drivers, I think the GUI provides the best administrative experience unless you are trying to add many drivers or automate driver management.

While PowerShell can be used to add print drivers, the `Add-PrinterDriver` cmdlet will only install print drivers that exist in the driver store. If the driver does not exist in the driver store, you can add new drivers to the driver store by using the `PNPUTIL` command line tool.

To add a new print driver package located at E:\Drivers\PCL\_v14.02\_x64\pcl5e\_5c\P564USAL.INF to the driver store on a print server, run the following command from an elevated PowerShell command prompt:

```
pnputil -a E:\Drivers\PCL_v14.02_x64\pcl5e_5c\P564USAL.INF
```

To add a print driver named Canon iR3225 PCL5e, run the following command from an elevated PowerShell prompt:

```
Add-PrinterDriver -Name 'Canon iR3225 PCL5e'
```

To retrieve a list of installed printer drivers on the local print server, run the following command from a PowerShell prompt:

```
Get-PrinterDriver
```

Now you know how to add and manage drivers for a print server. But managing drivers is just one aspect of managing print servers. Once you have a printer and the print drivers added on a print server, you need to enable users to print to the printer and enable administrators to manage the printer. We'll look at managing printer permissions in the next section.

## Managing Printer Permissions

The marketing department at Tailspin Toys has invested in a high-quality color printer for printing marketing material and documentation. To reduce supply costs, the VP of marketing has requested that only specific employees have access to print to the new printer. To accomplish this, you choose to adjust permissions on the print server.

Perform the following steps to restrict printing privileges to a specific AD DS security group:

1. Create a new security group in ADUC for the marketing department color printer.
2. Open the Print Management console.
3. In the left pane of the Print Management console, expand **Print Servers**, expand your print server, and then click **Printers**.
4. Right-click the desired printer and select **Properties**.
5. Click the **Security** tab.
6. Remove the **Everyone** group and add the new marketing security group. Confirm that **Allow** is checked for the **Print** permission for the marketing security group.
7. Click **Apply**.

In Figure 23.2 you can see an example of the changes outlined in the steps above. After applying the new set of permissions, printing privileges for the TT-MKTG-COLOR printer are restricted to administrators and members of the TT-Marketing-Printer security group.

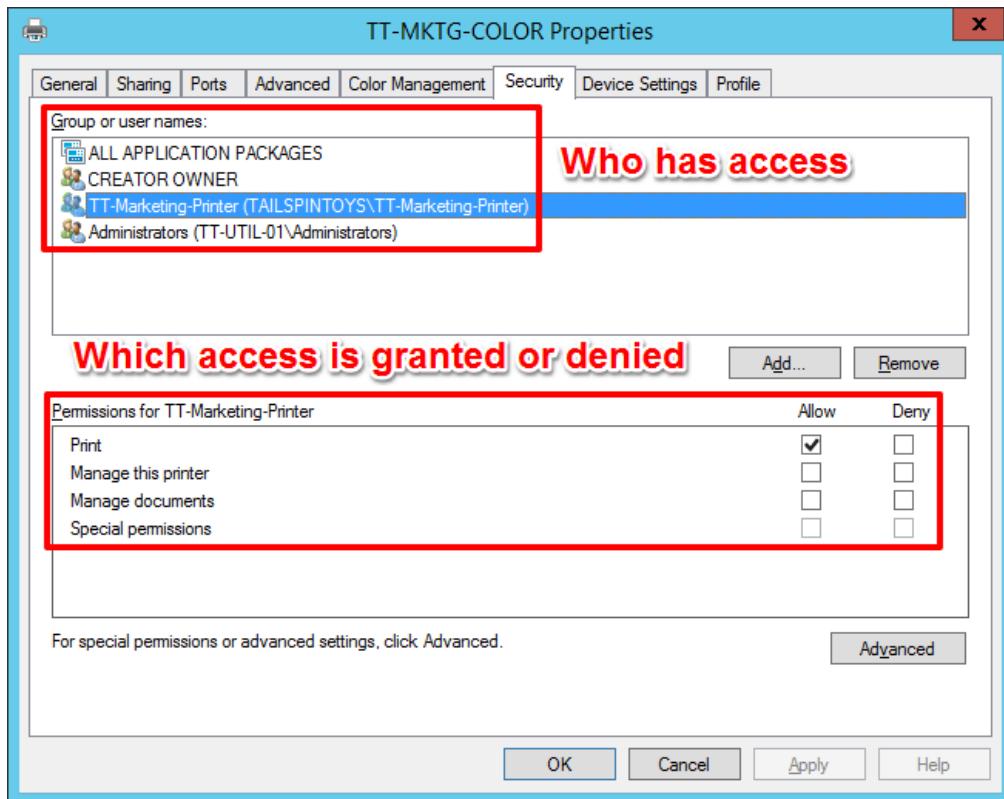


Figure 23.2 Printing permissions can be locked down to a specific user or security group through the printer security settings.

As you probably noticed in Figure 23.2, there are a few different permissions. The Print permission is used to enable users to print to a printer (or prevent users from printing to a printer). The other two permissions that you will occasionally use are the permissions for managing the printer and permissions for managing documents. The permission for managing the printer enables you to perform administrative actions on a printer such as sharing it and changing the permissions on it. This permission is mostly used for IT administrators. The permission for managing documents enables you to perform actions such as cancel a print job, rearrange print job orders (one option being that you can move a print job at the end of the print queue to start printing next), and restart print jobs (such as after a printer stops due to paper jam). It is a good idea to delegate the management of documents to a non-IT user so that departments can manage their document printing without having to rely on the IT department.

### Hands-on Exercise

Change the permissions on a printer so that the IT-Marketing-Printer group can cancel print jobs and rearrange the order of print jobs in the queue.

Now you have a printer, a driver, and appropriate permissions. The printer can be used by the users! From here out, your administrative focus will be keeping the printer and print server running (what's sometimes referred to as "daily care and feeding"). One of the tasks will be to manage print queues, which we'll look at next.

## Managing Print Queues

The process of adding a printer to a server leads to the creation of a print queue. Print queues contain a list of current print jobs that are waiting to be printed by a printer. In a typical environment, print queues will clear relatively quickly (as a job prints, it clears), unless a large job is being processed, or something is wrong with the printer. You can view all the print queues for a specific print server by navigating to the Printers node in the Print Management console, as shown in Figure 23.3.

Printer Name	Queue Status	Jobs In Queue	Driver Name
TT-MKTG-COLOR	Ready	0	Canon iR C3200 PCL5c
Canon iR3225 PCL5e	Ready	0	Canon iR3225 PCL5e
Canon iR3230 PCL6	Error	3	Canon iR3230 PCL6
Fax (redirected 2)	Ready	0	Microsoft Shared Fax Driver
Microsoft XPS Docu...	Ready	0	Microsoft XPS Document Wr

Figure 23.3 The printer queues will provide an at-glance status of the printers on your server.

In this example, we can see that one of the print queues is showing an error in the Queue Status column. This printer also has 3 jobs in queue and they won't print due to the error. As the print operator, it will be your job to resolve any errors with print queues.

You have some basic management options available for each print queue. These are available by right-clicking on the print queue that you want to manage. You should know the following options because they will be important during day-to-day management of your print queues:

- **Open Printer Queue.** This option opens the print queue for the selected printer. The print queue window displays all existing jobs for the destination printer. This is helpful when troubleshooting print jobs that are hung.
- **Pause Printing.** This option pauses the print queue for the selected printer. All jobs will stop processing until the queue is resumed. This is helpful when maintenance is required and you want to avoid clearing the print queue because that requires users to resubmit print jobs.
- **Cancel All Jobs.** This option cancels all jobs in the print queue for the selected printer. Canceled jobs cannot be restored. This is helpful when you need to clear the queue quickly.

However, users are not notified of the cancellation. They may walk over to the printer to look for their printout and be confused when they don't find it.

- **Set Printing Defaults.** This option displays the default printer settings. As the print operator, you can define the default printer settings. This is helpful in cost saving scenarios, such as enabling double-sided printing or assigning black and white as the default print style.
- **Print Test Page.** This option prints a test page to the selected printer. This is helpful when troubleshooting print queue issues. I like to print test pages to validate that the printer is functioning properly.

You can also view print queues from Windows PowerShell. To display details about a server's print queues, run the following command from a PowerShell prompt:

```
Get-Printer | select Name,PrinterStatus,JobCount,PortName
```

In Figure 23.4 you can see an example of the print queue that is reporting an error.

Document Name	Status	Owner	Pages	Size
Meeting Minutes.txt - Notepad	Error - Printing	tom.jones	N/A	
Friday schedule.txt - Notepad	N/A	chris.thompson	N/A	
https://technet.microsoft.com...	N/A	tom.jones	N/A	

3 document(s) in queue

Figure 23.4 Detailed job status can be seen from the print queue window.

As the print operator, you have the option of clearing the queue completely or canceling the one job that is reporting an error. If the printer is in good health, the remaining jobs should complete successfully. Outside of the queue, there could be another issue such as a paper jam or toner/ink issue. The individual print jobs can be paused, restarted, or canceled. To do this, right-click on the job and click the desired option.

These print jobs can also be managed using PowerShell. To retrieve a list of print jobs for a printer named Canon iR3230 PCL6, run the following command from a PowerShell prompt:

```
Get-PrintJob -PrinterName "Canon iR3230 PCL6"
```

To restart print job #4 for a printer named Canon iR3230 PCL6, run the following command from a PowerShell prompt:

```
Restart-PrintJob -PrinterName "Canon iR3230 PCL6" -ID 4
```

To remove print job #4 for a printer named "Canon iR3230 PCL6", run the following command from a PowerShell prompt:

```
Remove-PrintJob -PrinterName "Canon iR3230 PCL6" -ID 4
```

### **Hands-on Exercise**

Take the paper out of a shared printer. Send a couple of print jobs to the printer. Use PowerShell to list the current print jobs. Load paper into the printer and then use PowerShell to restart both print jobs in the queue.

You should now have a good general feel for basic print server management. In the next two sections, we'll look at a couple of advanced topics – printer pooling (combining more than one printer together for better performance) and publish printers to Active Directory (to enable users to easily find shared printers).

## **Printer Pooling**

The marketing team at Tailspin Toys ends up exceeding their expected usage on their new color printer. Many new products are being released by the company, resulting in a large volume of printed marketing material. New print jobs are taking up to 15 minutes to print due to many print jobs in the queue. The VP of marketing wants to purchase another printer to increase printing performance. But he's concerned about user confusion having two identical printers shared with different names. You decide to enable printer pooling.

Printer pooling is a feature designed for environments with demanding print needs. The idea behind printer pooling is to leverage a single print queue that has multiple printers assigned to it through different ports. When a job is submitted to the queue it goes to the first available printer. If a printer is processing a large job or has a failure, the next job will move to the next available printer.

Before enabling printer pooling, you should consider the following data points:

- **Location.** The printers assigned to the pool should be relatively close in proximity to one another. As jobs are submitted to the queue, there is not a way to insure which printer will process the job. So, users should be able to walk up to the printer location and look at the printers in the pool (not walk to building #1 to not find a printout and then go to building #2).
- **Drivers.** The printers assigned to the pool must all use the same driver. Because you are using a single printer queue, you only have the option to assign a single driver. Thus, the printer makes and models should be identical or at least share the same driver.
- **Ports.** The ports used for each printer in the pool can be the same type or mixed. The ports for most network-based printers are IP addresses, with each printer having a unique IP address.

In Figure 23.5 you can see an example of how a printer pool is created.

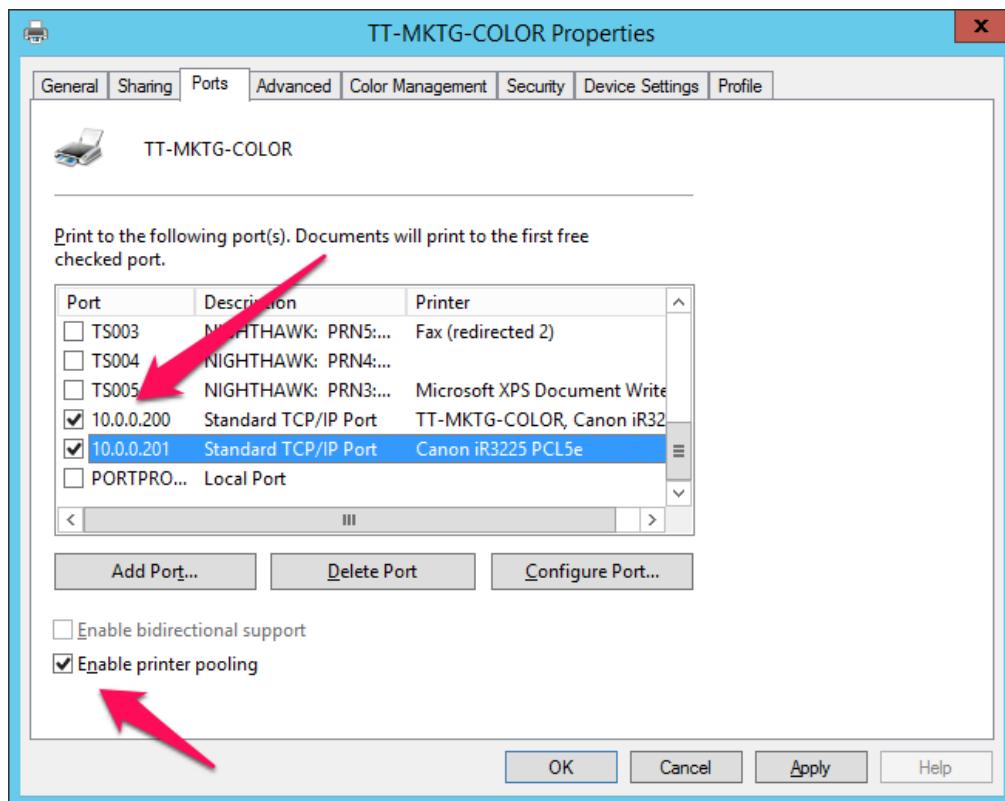


Figure 23.5 Printer pooling is enabled on the ports tab of the desired printer.

In this example, we have modified the port configuration for the marketing printer. We now have two ports assigned, and we have enabled the "Enable printer pooling" option. With these configuration changes, the next job submitted to TT-MKTG-COLOR will be sent to one of the two printers, depending on availability.

While printer pooling is a cool feature that provides potential benefits, it isn't as popular as you might think. There are a few reasons for this – admins aren't familiar with the printer pooling feature and many environments don't have demanding printing needs requiring more than one printer per area. You should examine your environment to find out if printer pooling could improve overall printing performance. Now that you are familiar with printer pooling, we will change gears and look at a way to make finding printers easier for your end users. Publishing printers in Active Directory is one way to do it and we talk about it next.

Now that you are familiar with printer pooling, we will change gears and look at a way to make finding printers easier for your end users. Publishing printers in Active Directory is one way to do it and we talk about it next. Publishing Printers in Active Directory

The final topic in this chapter looks at making it easier for users to find printers to print to. You can publish printers in Active Directory which enable users to find printers and search for printers with specific capabilities. This saves IT administrators time because users can find and add their own printers without having to rely on the IT administrators. And it saves users time because they don't have to open a trouble ticket and wait for the IT administrator to contact

them. Publishing printers in Active Directory is an action that extends the sharing capabilities of the print server. When a printer is shared, it is listed on a server when you browse to it (for example, \\print-server). When a printer is published to Active Directory, the printer information is stored in AD DS, and users can discover the printer using the Add Printer wizard in Windows. How does this help? If you have 5 print servers, users can find a printer no matter which server it is shared from (whereas without publishing printers to Active Directory, a user would have to browse to each individual server to look for the printer).

In Figure 23.6, you can see an example of the printers published in AD DS. The Find Printers window is part of the Add Printers wizard. All published printers will show up here, with helpful information like location and model. Users can filter the list based on the search criteria they enter.

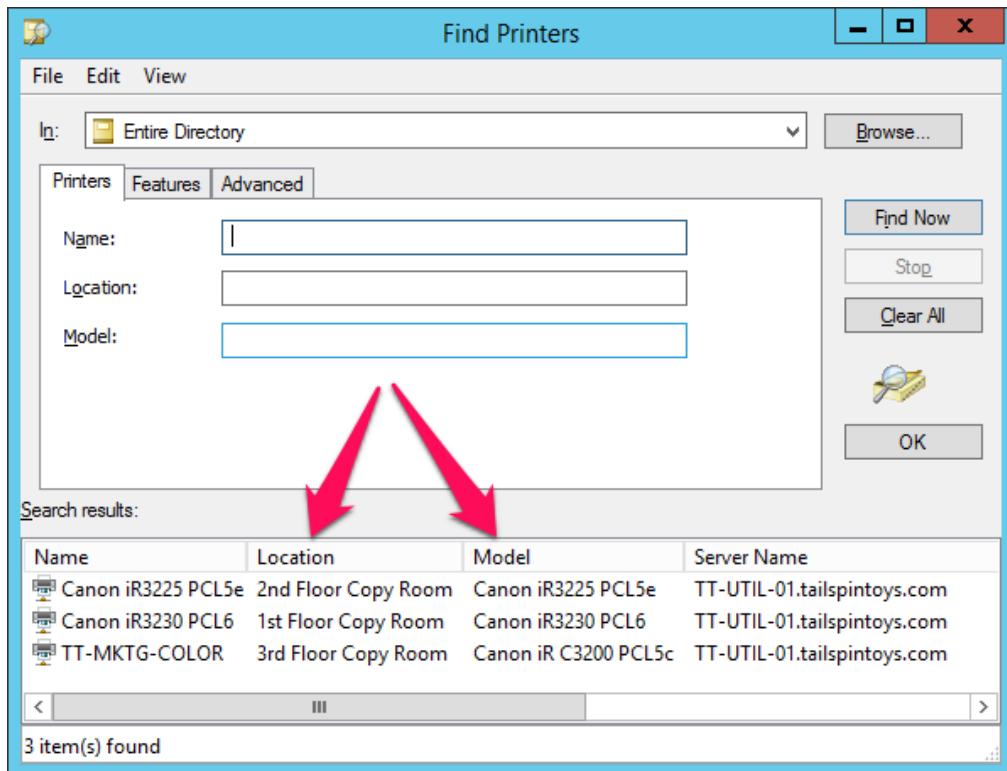


Figure 23.6 Published printers will show up in the Find Printers window and can be filtered based on search criteria.

As you can see, publishing shared printers to AD DS is beneficial. Now, let's look at how you do it.

## Publish a printer in AD DS using the Print Management console.

Perform the following steps to publish a printer in AD DS using the Print Management console:

1. Open the Print Management console.

2. In the left pane of the Print Management console, expand **Print Servers**, expand your print server, and click **Printers**.
3. Right-click on the desired printer and click **Properties**.
4. Click the **Sharing** tab.
5. Click the **List in the directory** option.
6. Click **Apply**.

### **Hands-on Exercise**

Publish one shared printer to Active Directory. After publishing, search Active Directory to see if you can find the shared printer.

You can also use PowerShell to publish a printer in Active Directory. And, as you'll see, PowerShell has a nifty feature method for you if you have a bunch of existing printers that have not been published to Active Directory.

### **Publish a printer in AD DS using PowerShell**

Using PowerShell enables you to publish all unpublished printers with a single command! This can simplify the publishing task versus clicking through each of the printers individually to publish them.

To publish all unpublished printers in AD DS, run the following command from an elevated PowerShell prompt on the print server:

```
Get-Printer | Where Published -eq $False | Set-Printer -Published:$true
```

To confirm which printers have been published in AD DS, run the following PowerShell command:

```
Get-Printer | Where Published -eq $true
```

In this chapter, you learned how to manage Windows print services including working with print drivers, managing printer permissions, administrating print queues, setting up printer pooling, and publishing printers in Active Directory. Let's have you work through some of these tasks in the lab now.

## **Lab**

### **Installing Print Drivers**

Perform the following tasks:

- Use the Print Management console to add print driver support for a Dell 3130cn printer.
- Use PowerShell to add print driver support for a Canon iR5075 printer.

## Managing Printer Permissions

Perform the following tasks:

- Create an AD DS security group named Tailspin Toys Printer Access. Use the Print Management console to restrict print access to your print server. Only members of the Tailspin Toys Printer Access group should have printing privileges.

## Managing Printer Queues

Perform the following tasks:

- Use the Print Management console to create a print queue for a Dell 3130cn printer with an IP address of 10.0.0.15. Name the print queue Finance Printer.
- Use PowerShell to create a print queue for a second Dell 3130cn printer with an IP address of 10.0.0.16. Name the print queue Payroll Printer.

## Printer Pooling

Perform the following tasks:

- Use the Print Management console to create a printer pool. The printer pool should include both Dell 3130cn printers created in this lab.

## Publishing Printers in Active Directory

Perform the following tasks:

- Use PowerShell to publish all the unpublished printers on a print server to Active Directory.

## CHAPTER 24: WHAT NOW?

---

It's now been about a month since you began your journey with learning how to manage a Windows server. In that time, you've been introduced to a long list of technologies, services, and solutions. And, amazingly enough, we've only looked at portions of Windows Server technologies (or a portion of a technology). There is still much more to learn. This chapter will introduce you to ideas for building on your knowledge and provide you with resources to help guide you along the way. Good luck!

### Ideas for further exploration

I've always believed that you'll learn more when you are learning stuff that you find interesting. You want to enjoy learn it so much that you think you are having fun. So, pick one or two areas of exploration that stand out to you, then dive into them. Here is a high-level action plan for each topic:

- **Read about it.** You can use books, online resources, vendor information, help files, and blogs. Dive in and go into the inner workings of the technology (see below for details).
- **Install and configure it.** Use your home lab, your company's non-production environment, or an online lab (or VM) and install the solution. Configure it. Troubleshoot it. Play with it.
- **Get certified on it.** As you become well versed in a technology, consider taking a certification test to validate your knowledge and get certified. Certification will make you stand out in the IT field. I think a big chunk of what I learned in my career came from studying for certification exams!

### And here are the areas to explore

- **Initial installation and configuration.** Because this book was written with the assumption that our readers have suddenly inherited a Windows server to manage, we didn't show much information on initial installation and configuration tasks. You should dive into this area with a focus on the following tasks:
  - **Installing Windows Server.** Become familiar with how to install it and the different installation options. Explore the containers, Server Core, and Nano Server!
  - **Automating the installation.** Check out how to automate an installation with answer files and additional deployment software (such as Microsoft Deployment Toolkit (MDT) and System Center Configuration Manager).
- **Windows Server roles and features.** We touched upon some of the most common roles and features such as file services, Active Directory, and printer services. But there is a host of others that you'll find useful and helpful, including the following list (which is just a partial list):
  - **Active Directory Certificate Services (AD CS).** AD CS is the Microsoft public key infrastructure (PKI) solution to distribute and manage digital certificates to your

computers and devices. You can use certificates for authentication, two-factor authentication, securing web sites, and for securing email.

- **Hyper-V.** The Hyper-V role enables your server to host virtual machines (VMs). Virtualization is the de facto standard for new server deployments today. Virtualization ties heavily into networking and storage so be prepared to go deep into those technologies along the way.
- **Web Server (IIS).** As a server administrator, you will undoubtedly manage some web servers in your environment. IIS is often a prerequisite feature for technologies such as Exchange Server. Knowing how to install and configure a web server and even a basic web site is very good knowledge to have as a server administrator.
- **Design and planning.** Initially, you'll be doing administrative tasks to keep your servers running smoothly. But, as your knowledge expands, you will be expected to participate in design and planning work. Design and planning involves figuring out how much hardware and software you need, where you will put them, how you will configure them, how you will provide high availability and site resilience, and how you will back up and restore everything.
- **Configuration management.** One of your primary challenges will be to keep servers configured similarly which makes maintaining them easier. Configuration management can help. The following configuration management items should be on your future radar:
  - **Patch management.** Microsoft and other vendors routinely release software updates to address bugs and security vulnerabilities. As the server administrator, one of your jobs will be to ensure that those software updates are deployed to all your computers. You can use a built-in Windows Server role (Windows Server Update Services) or a configuration management product (such as System Center Configuration Manager).
  - **Server configuration.** In this book, we showed you some server configuration information such as the event logs, services, and storage. But how will you enforce your desired configuration across all servers? You can use PowerShell's Desired State Configuration (DSC), Configuration Manager, or Group Policy.
- **Windows PowerShell.** We introduced you to PowerShell in this book and we used PowerShell examples throughout the book. But PowerShell is a huge topic. Not only can you use it for your day-to-day administration, but you can build complex scripts and automation with it too.
- **The cloud.** The cloud is the next big thing in computing. But it has just started to gain traction with many companies. Administrators who invest time and energy to learn the cloud will be well positioned for the future when more organizations embrace the cloud. As a Windows Server administrator, you should start looking at Microsoft Azure and Amazon Web Services (AWS) and explore their solutions.
- **DevOps.** Development Operations (DevOps) is the fusion of two typically different jobs – a developer and an administrator (such as server administration or database administrator). As cloud computing has gained popularity, automation is a big topic. And automation requires

scripting and development. In the future, administrators will need more development skills and developers will need administrative skills.

## Other resources you'll grow to love

You can use a variety of resources to help you continue learning from here. Below is a list of some key resources that you should check out.

- **Forums.** You can post questions in a forum. And you can get answers from others that have experience with the technology. Many industry experts answer questions in forums as part of their effort to give back to the IT community. Check out the Microsoft TechNet forums at <https://social.technet.microsoft.com/forums/> - you will need to create a free account to post questions and participate in discussions.
- **Windows Server documentation.** Microsoft provides a web page that has links to various Microsoft documentation and downloads for Windows Server. It is useful, especially if you work with multiple versions of Windows Server and need to quickly find version-specific content. See <https://technet.microsoft.com/en-us/library/bb625087.aspx> for more information.
- **Online educational sites.** Some sites are Massive Online Open Course (MOOC) platforms and some are smaller and for-profit companies. You should look at [www.edx.org](http://www.edx.org), [www.lynda.com](http://www.lynda.com), and the Microsoft Virtual Academy (MVA) which has a large amount of free content. See <https://mva.microsoft.com/> for more information.