

OFFICIAL MICROSOFT LEARNING PRODUCT

6425B

Configuring and Troubleshooting Windows Server® 2008 Active Directory® Domain Services

Volume 1



Be sure to access the extended learning content on your
Course Companion CD enclosed on the back cover of the book.

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The names of manufacturers, products, or URLs are provided for informational purposes only and Microsoft makes no representations and warranties, either expressed, implied, or statutory, regarding these manufacturers or the use of the products with any Microsoft technologies. The inclusion of a manufacturer or product does not imply endorsement of Microsoft of the manufacturer or product. Links may be provided to third party sites. Such sites are not under the control of Microsoft and Microsoft is not responsible for the contents of any linked site or any link contained in a linked site, or any changes or updates to such sites. Microsoft is not responsible for webcasting or any other form of transmission received from any linked site. Microsoft is providing these links to you only as a convenience, and the inclusion of any link does not imply endorsement of Microsoft of the site or the products contained therein.

© 2009 Microsoft Corporation. All rights reserved.

Microsoft, Microsoft Press, Access, Active Directory, ActiveX, Convergence, Excel, Forefront, Hyper-V, Internet Explorer, MS, MSDN, MS-DOS, Outlook, PowerPoint, Segoe, SharePoint, SQL Server, Visio, Visual Basic, Visual Studio, Windows, Windows Live, Windows Mobile, Windows NT, Windows PowerShell, Windows Server and Windows Vista. are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

All other trademarks are property of their respective owners.

Product Number: 6425B

Part Number: X16-23526

Released: 11/2009

MICROSOFT LICENSE TERMS

OFFICIAL MICROSOFT LEARNING PRODUCTS - TRAINER

EDITION – Pre-Release and Final Release Versions

These license terms are an agreement between Microsoft Corporation and you. Please read them. They apply to the Licensed Content named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft

- updates,
- supplements,
- Internet-based services, and
- support services

for this Licensed Content, unless other terms accompany those items. If so, those terms apply.

By using the Licensed Content, you accept these terms. If you do not accept them, do not use the Licensed Content.

If you comply with these license terms, you have the rights below.

1. DEFINITIONS.

- "Academic Materials"** means the printed or electronic documentation such as manuals, workbooks, white papers, press releases, datasheets, and FAQs which may be included in the Licensed Content.
- "Authorized Learning Center(s)"** means a Microsoft Certified Partner for Learning Solutions location, an IT Academy location, or such other entity as Microsoft may designate from time to time.
- "Authorized Training Session(s)"** means those training sessions authorized by Microsoft and conducted at or through Authorized Learning Centers by a Trainer providing training to Students solely on Official Microsoft Learning Products (formerly known as Microsoft Official Curriculum or "MOC") and Microsoft Dynamics Learning Products (formerly known as Microsoft Business Solutions Courseware). Each Authorized Training Session will provide training on the subject matter of one (1) Course.
- "Course"** means one of the courses using Licensed Content offered by an Authorized Learning Center during an Authorized Training Session, each of which provides training on a particular Microsoft technology subject matter.
- "Device(s)"** means a single computer, device, workstation, terminal, or other digital electronic or analog device.
- "Licensed Content"** means the materials accompanying these license terms. The Licensed Content may include, but is not limited to, the following elements: (i) Trainer Content, (ii) Student Content, (iii) classroom setup guide, and (iv) Software. There are different and separate components of the Licensed Content for each Course.
- "Software"** means the Virtual Machines and Virtual Hard Disks, or other software applications that may be included with the Licensed Content.
- "Student(s)"** means a student duly enrolled for an Authorized Training Session at your location.

- i. **"Student Content"** means the learning materials accompanying these license terms that are for use by Students and Trainers during an Authorized Training Session. Student Content may include labs, simulations, and courseware files for a Course.
- j. **"Trainer(s)"** means a) a person who is duly certified by Microsoft as a Microsoft Certified Trainer and b) such other individual as authorized in writing by Microsoft and has been engaged by an Authorized Learning Center to teach or instruct an Authorized Training Session to Students on its behalf.
- k. **"Trainer Content"** means the materials accompanying these license terms that are for use by Trainers and Students, as applicable, solely during an Authorized Training Session. Trainer Content may include Virtual Machines, Virtual Hard Disks, Microsoft PowerPoint files, instructor notes, and demonstration guides and script files for a Course.
- l. **"Virtual Hard Disks"** means Microsoft Software that is comprised of virtualized hard disks (such as a base virtual hard disk or differencing disks) for a Virtual Machine that can be loaded onto a single computer or other device in order to allow end-users to run multiple operating systems concurrently. For the purposes of these license terms, Virtual Hard Disks will be considered "Trainer Content".
- m. **"Virtual Machine"** means a virtualized computing experience, created and accessed using Microsoft® Virtual PC or Microsoft® Virtual Server software that consists of a virtualized hardware environment, one or more Virtual Hard Disks, and a configuration file setting the parameters of the virtualized hardware environment (e.g., RAM). For the purposes of these license terms, Virtual Hard Disks will be considered "Trainer Content".
- n. **"you"** means the Authorized Learning Center or Trainer, as applicable, that has agreed to these license terms.

2. OVERVIEW.

Licensed Content. The Licensed Content includes Software, Academic Materials (online and electronic), Trainer Content, Student Content, classroom setup guide, and associated media.

License Model. The Licensed Content is licensed on a per copy per Authorized Learning Center location or per Trainer basis.

3. INSTALLATION AND USE RIGHTS.

- a. **Authorized Learning Centers and Trainers: For each Authorized Training Session, you may:**
 - i. either install individual copies of the relevant Licensed Content on classroom Devices only for use by Students enrolled in and the Trainer delivering the Authorized Training Session, provided that the number of copies in use does not exceed the number of Students enrolled in and the Trainer delivering the Authorized Training Session, **OR**
 - ii. install one copy of the relevant Licensed Content on a network server only for access by classroom Devices and only for use by Students enrolled in and the Trainer delivering the Authorized Training Session, provided that the number of Devices accessing the Licensed Content on such server does not exceed the number of Students enrolled in and the Trainer delivering the Authorized Training Session.
 - iii. and allow the Students enrolled in and the Trainer delivering the Authorized Training Session to use the Licensed Content that you install in accordance with (i) or (ii) above during such Authorized Training Session in accordance with these license terms.

- i. **Separation of Components.** The components of the Licensed Content are licensed as a single unit. You may not separate the components and install them on different Devices.
- ii. **Third Party Programs.** The Licensed Content may contain third party programs. These license terms will apply to the use of those third party programs, unless other terms accompany those programs.

b. Trainers:

- i. Trainers may Use the Licensed Content that you install or that is installed by an Authorized Learning Center on a classroom Device to deliver an Authorized Training Session.
- ii. Trainers may also Use a copy of the Licensed Content as follows:
 - A. **Licensed Device.** The licensed Device is the Device on which you Use the Licensed Content. You may install and Use one copy of the Licensed Content on the licensed Device solely for your own personal training Use and for preparation of an Authorized Training Session.
 - B. **Portable Device.** You may install another copy on a portable device solely for your own personal training Use and for preparation of an Authorized Training Session.

4. PRE-RELEASE VERSIONS. If this is a pre-release ("beta") version, in addition to the other provisions in this agreement, these terms also apply:

- a. **Pre-Release Licensed Content.** This Licensed Content is a pre-release version. It may not contain the same information and/or work the way a final version of the Licensed Content will. We may change it for the final, commercial version. We also may not release a commercial version. You will clearly and conspicuously inform any Students who participate in each Authorized Training Session of the foregoing; and, that you or Microsoft are under no obligation to provide them with any further content, including but not limited to the final released version of the Licensed Content for the Course.
- b. **Feedback.** If you agree to give feedback about the Licensed Content to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software, Licensed Content, or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.
- c. **Confidential Information.** The Licensed Content, including any viewer, user interface, features and documentation that may be included with the Licensed Content, is confidential and proprietary to Microsoft and its suppliers.
 - i. **Use.** For five years after installation of the Licensed Content or its commercial release, whichever is first, you may not disclose confidential information to third parties. You may disclose confidential information only to your employees and consultants who need to know the information. You must have written agreements with them that protect the confidential information at least as much as this agreement.
 - ii. **Survival.** Your duty to protect confidential information survives this agreement.
 - iii. **Exclusions.** You may disclose confidential information in response to a judicial or governmental order. You must first give written notice to Microsoft to allow it to seek a

protective order or otherwise protect the information. Confidential information does not include information that

- becomes publicly known through no wrongful act;
 - you received from a third party who did not breach confidentiality obligations to Microsoft or its suppliers; or
 - you developed independently.
- d. **Term.** The term of this agreement for pre-release versions is (i) the date which Microsoft informs you is the end date for using the beta version, or (ii) the commercial release of the final release version of the Licensed Content, whichever is first ("beta term").
- e. **Use.** You will cease using all copies of the beta version upon expiration or termination of the beta term, and will destroy all copies of same in the possession or under your control and/or in the possession or under the control of any Trainers who have received copies of the pre-released version.
- f. **Copies.** Microsoft will inform Authorized Learning Centers if they may make copies of the beta version (in either print and/or CD version) and distribute such copies to Students and/or Trainers. If Microsoft allows such distribution, you will follow any additional terms that Microsoft provides to you for such copies and distribution.

5. ADDITIONAL LICENSING REQUIREMENTS AND/OR USE RIGHTS.

a. Authorized Learning Centers and Trainers:

i. Software.

ii. **Virtual Hard Disks.** The Licensed Content may contain versions of Microsoft XP, Microsoft Windows Vista, Windows Server 2003, Windows Server 2008, and Windows 2000 Advanced Server and/or other Microsoft products which are provided in Virtual Hard Disks.

A. If the Virtual Hard Disks and the labs are launched through the Microsoft Learning Lab Launcher, then these terms apply:

Time-Sensitive Software. If the Software is not reset, it will stop running based upon the time indicated on the install of the Virtual Machines (between 30 and 500 days after you install it). You will not receive notice before it stops running. You may not be able to access data used or information saved with the Virtual Machines when it stops running and may be forced to reset these Virtual Machines to their original state. You must remove the Software from the Devices at the end of each Authorized Training Session and reinstall and launch it prior to the beginning of the next Authorized Training Session.

B. If the Virtual Hard Disks require a product key to launch, then these terms apply:

Microsoft will deactivate the operating system associated with each Virtual Hard Disk. Before installing any Virtual Hard Disks on classroom Devices for use during an Authorized Training Session, you will obtain from Microsoft a product key for the operating system software for the Virtual Hard Disks and will activate such Software with Microsoft using such product key.

C. These terms apply to all Virtual Machines and Virtual Hard Disks:

You may only use the Virtual Machines and Virtual Hard Disks if you comply with the terms and conditions of this agreement and the following security requirements:

- You may not install Virtual Machines and Virtual Hard Disks on portable Devices or Devices that are accessible to other networks.
 - You must remove Virtual Machines and Virtual Hard Disks from all classroom Devices at the end of each Authorized Training Session, except those held at Microsoft Certified Partners for Learning Solutions locations.
 - You must remove the differencing drive portions of the Virtual Hard Disks from all classroom Devices at the end of each Authorized Training Session at Microsoft Certified Partners for Learning Solutions locations.
 - You will ensure that the Virtual Machines and Virtual Hard Disks are not copied or downloaded from Devices on which you installed them.
 - You will strictly comply with all Microsoft instructions relating to installation, use, activation and deactivation, and security of Virtual Machines and Virtual Hard Disks.
 - You may not modify the Virtual Machines and Virtual Hard Disks or any contents thereof.
 - You may not reproduce or redistribute the Virtual Machines or Virtual Hard Disks.
- ii. Classroom Setup Guide.** You will assure any Licensed Content installed for use during an Authorized Training Session will be done in accordance with the classroom set-up guide for the Course.
- iii. Media Elements and Templates.** You may allow Trainers and Students to use images, clip art, animations, sounds, music, shapes, video clips and templates provided with the Licensed Content solely in an Authorized Training Session. If Trainers have their own copy of the Licensed Content, they may use Media Elements for their personal training use.
- iv. iv Evaluation Software.** Any Software that is included in the Student Content designated as "Evaluation Software" may be used by Students solely for their personal training outside of the Authorized Training Session.

b. Trainers Only:

- i. Use of PowerPoint Slide Deck Templates.** The Trainer Content may include Microsoft PowerPoint slide decks. Trainers may use, copy and modify the PowerPoint slide decks only for providing an Authorized Training Session. If you elect to exercise the foregoing, you will agree or ensure Trainer agrees: (a) that modification of the slide decks will not constitute creation of obscene or scandalous works, as defined by federal law at the time the work is created; and (b) to comply with all other terms and conditions of this agreement.
- ii. Use of Instructional Components in Trainer Content.** For each Authorized Training Session, Trainers may customize and reproduce, in accordance with the MCT Agreement, those portions of the Licensed Content that are logically associated with instruction of the Authorized Training Session. If you elect to exercise the foregoing rights, you agree or ensure the Trainer agrees: (a) that any of these customizations or reproductions will only be used for providing an Authorized Training Session and (b) to comply with all other terms and conditions of this agreement.

iii. Academic Materials. If the Licensed Content contains Academic Materials, you may copy and use the Academic Materials. You may not make any modifications to the Academic Materials and you may not print any book (either electronic or print version) in its entirety. If you reproduce any Academic Materials, you agree that:

- The use of the Academic Materials will be only for your personal reference or training use
- You will not republish or post the Academic Materials on any network computer or broadcast in any media;
- You will include the Academic Material's original copyright notice, or a copyright notice to Microsoft's benefit in the format provided below:

Form of Notice:

© 2009 Reprinted for personal reference use only with permission by Microsoft Corporation. All rights reserved.

Microsoft, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the US and/or other countries. Other product and company names mentioned herein may be the trademarks of their respective owners.

6. INTERNET-BASED SERVICES. Microsoft may provide Internet-based services with the Licensed Content. It may change or cancel them at any time. You may not use these services in any way that could harm them or impair anyone else's use of them. You may not use the services to try to gain unauthorized access to any service, data, account or network by any means.

7. SCOPE OF LICENSE. The Licensed Content is licensed, not sold. This agreement only gives you some rights to use the Licensed Content. Microsoft reserves all other rights. Unless applicable law gives you more rights despite this limitation, you may use the Licensed Content only as expressly permitted in this agreement. In doing so, you must comply with any technical limitations in the Licensed Content that only allow you to use it in certain ways. You may not

- install more copies of the Licensed Content on classroom Devices than the number of Students and the Trainer in the Authorized Training Session;
- allow more classroom Devices to access the server than the number of Students enrolled in and the Trainer delivering the Authorized Training Session if the Licensed Content is installed on a network server;
- copy or reproduce the Licensed Content to any server or location for further reproduction or distribution;
- disclose the results of any benchmark tests of the Licensed Content to any third party without Microsoft's prior written approval;
- work around any technical limitations in the Licensed Content;
- reverse engineer, decompile or disassemble the Licensed Content, except and only to the extent that applicable law expressly permits, despite this limitation;
- make more copies of the Licensed Content than specified in this agreement or allowed by applicable law, despite this limitation;
- publish the Licensed Content for others to copy;

- transfer the Licensed Content, in whole or in part, to a third party;
 - access or use any Licensed Content for which you (i) are not providing a Course and/or (ii) have not been authorized by Microsoft to access and use;
 - rent, lease or lend the Licensed Content; or
 - use the Licensed Content for commercial hosting services or general business purposes.
 - Rights to access the server software that may be included with the Licensed Content, including the Virtual Hard Disks does not give you any right to implement Microsoft patents or other Microsoft intellectual property in software or devices that may access the server.
- 8. EXPORT RESTRICTIONS.** The Licensed Content is subject to United States export laws and regulations. You must comply with all domestic and international export laws and regulations that apply to the Licensed Content. These laws include restrictions on destinations, end users and end use. For additional information, see www.microsoft.com/exporting.
- 9. NOT FOR RESALE SOFTWARE/LICENSED CONTENT.** You may not sell software or Licensed Content marked as "NFR" or "Not for Resale."
- 10. ACADEMIC EDITION.** You must be a "Qualified Educational User" to use Licensed Content marked as "Academic Edition" or "AE." If you do not know whether you are a Qualified Educational User, visit www.microsoft.com/education or contact the Microsoft affiliate serving your country.
- 11. TERMINATION.** Without prejudice to any other rights, Microsoft may terminate this agreement if you fail to comply with the terms and conditions of these license terms. In the event your status as an Authorized Learning Center or Trainer a) expires, b) is voluntarily terminated by you, and/or c) is terminated by Microsoft, this agreement shall automatically terminate. Upon any termination of this agreement, you must destroy all copies of the Licensed Content and all of its component parts.
- 12. ENTIRE AGREEMENT.** This agreement, and the terms for supplements, updates, Internet-based services and support services that you use, are the entire agreement for the Licensed Content and support services.
- 13. APPLICABLE LAW.**
- a. **United States.** If you acquired the Licensed Content in the United States, Washington state law governs the interpretation of this agreement and applies to claims for breach of it, regardless of conflict of laws principles. The laws of the state where you live govern all other claims, including claims under state consumer protection laws, unfair competition laws, and in tort.
 - b. **Outside the United States.** If you acquired the Licensed Content in any other country, the laws of that country apply.
- 14. LEGAL EFFECT.** This agreement describes certain legal rights. You may have other rights under the laws of your country. You may also have rights with respect to the party from whom you acquired the Licensed Content. This agreement does not change your rights under the laws of your country if the laws of your country do not permit it to do so.
- 15. DISCLAIMER OF WARRANTY.** The Licensed Content is licensed "as-is." You bear the risk of using it. Microsoft gives no express warranties, guarantees or conditions. You may have additional consumer rights under your local laws which this agreement cannot change. To the extent permitted under your local laws, Microsoft excludes the implied warranties of merchantability, fitness for a particular purpose and non-infringement.

16. LIMITATION ON AND EXCLUSION OF REMEDIES AND DAMAGES. YOU CAN RECOVER FROM MICROSOFT AND ITS SUPPLIERS ONLY DIRECT DAMAGES UP TO U.S. \$5.00. YOU CANNOT RECOVER ANY OTHER DAMAGES, INCLUDING CONSEQUENTIAL, LOST PROFITS, SPECIAL, INDIRECT OR INCIDENTAL DAMAGES.

This limitation applies to

- anything related to the Licensed Content, software, services, content (including code) on third party Internet sites, or third party programs; and
- claims for breach of contract, breach of warranty, guarantee or condition, strict liability, negligence, or other tort to the extent permitted by applicable law.

It also applies even if Microsoft knew or should have known about the possibility of the damages. The above limitation or exclusion may not apply to you because your country may not allow the exclusion or limitation of incidental, consequential or other damages.

Please note: As this Licensed Content is distributed in Quebec, Canada, some of the clauses in this agreement are provided below in French.

Remarque : Ce le contenu sous licence étant distribué au Québec, Canada, certaines des clauses dans ce contrat sont fournies ci-dessous en français.

EXONÉRATION DE GARANTIE. Le contenu sous licence visé par une licence est offert « tel quel ». Toute utilisation de ce contenu sous licence est à votre seule risque et péril. Microsoft n'accorde aucune autre garantie expresse. Vous pouvez bénéficier de droits additionnels en vertu du droit local sur la protection des consommateurs, que ce contrat ne peut modifier. La ou elles sont permises par le droit local, les garanties implicites de qualité marchande, d'adéquation à un usage particulier et d'absence de contrefaçon sont exclues.

LIMITATION DES DOMMAGES-INTÉRÊTS ET EXCLUSION DE RESPONSABILITÉ POUR LES DOMMAGES. Vous pouvez obtenir de Microsoft et de ses fournisseurs une indemnisation en cas de dommages directs uniquement à hauteur de 5,00 \$ US. Vous ne pouvez prétendre à aucune indemnisation pour les autres dommages, y compris les dommages spéciaux, indirects ou accessoires et pertes de bénéfices.

Cette limitation concerne:

- tout ce qui est relié au le contenu sous licence , aux services ou au contenu (y compris le code) figurant sur des sites Internet tiers ou dans des programmes tiers ; et
- les réclamations au titre de violation de contrat ou de garantie, ou au titre de responsabilité stricte, de négligence ou d'une autre faute dans la limite autorisée par la loi en vigueur.

Elle s'applique également, même si Microsoft connaissait ou devrait connaître l'éventualité d'un tel dommage. Si votre pays n'autorise pas l'exclusion ou la limitation de responsabilité pour les dommages indirects, accessoires ou de quelque nature que ce soit, il se peut que la limitation ou l'exclusion ci-dessus ne s'appliquera pas à votre égard.

EFFET JURIDIQUE. Le présent contrat décrit certains droits juridiques. Vous pourriez avoir d'autres droits prévus par les lois de votre pays. Le présent contrat ne modifie pas les droits que vous confèrent les lois de votre pays si celles-ci ne le permettent pas.

Welcome!

Thank you for taking our training! We've worked together with our Microsoft Certified Partners for Learning Solutions and our Microsoft IT Academies to bring you a world-class learning experience—whether you're a professional looking to advance your skills or a student preparing for a career in IT.

- **Microsoft Certified Trainers and Instructors**—Your instructor is a technical and instructional expert who meets ongoing certification requirements. And, if instructors are delivering training at one of our Certified Partners for Learning Solutions, they are also evaluated throughout the year by students and by Microsoft.
- **Certification Exam Benefits**—After training, consider taking a Microsoft Certification exam. Microsoft Certifications validate your skills on Microsoft technologies and can help differentiate you when finding a job or boosting your career. In fact, independent research by IDC concluded that 75% of managers believe certifications are important to team performance¹. Ask your instructor about Microsoft Certification exam promotions and discounts that may be available to you.
- **Customer Satisfaction Guarantee**—Our Certified Partners for Learning Solutions offer a satisfaction guarantee and we hold them accountable for it. At the end of class, please complete an evaluation of today's experience. We value your feedback!

We wish you a great learning experience and ongoing success in your career!

Sincerely,

Microsoft Learning
www.microsoft.com/learning

Microsoft | Learning

¹ IDC, Value of Certification: Team Certification and Organizational Performance, November 2006

Acknowledgement

Microsoft® Learning would like to acknowledge and thank the following for their contribution towards developing this title. Their effort at various stages in the development has ensured that you have a good classroom experience.

Dan Holme – Subject Matter Expert

A graduate of Yale University and Thunderbird, Dan has spent 15 years as a consultant and trainer, delivering solutions to tens of thousands of IT professionals from the most prestigious organizations and corporations around the world. Dan's company, Intelliem, is a boutique consulting and training firm with a *Fortune*-caliber clientele and deep expertise and experience in Windows®, Active Directory®, and Microsoft Office SharePoint®. From his base in beautiful Maui, Dan travels around the globe supporting customers and delivering Microsoft technologies training. Dan is also a contributing editor for *Windows IT Pro* magazine, a Microsoft MVP (Windows Server® Directory Services, 2007, and Office SharePoint Server, 2008-2009), and the community lead of SharePointProConnections.com. Last year, Dan published two books with Microsoft Press: the *Windows Administration Resource Kit* and the training kit for the 70-640 MCTS exam, both of which are at the top of the bestseller list for Windows books. He is currently building SharePoint solutions to support the broadcast of the 2010 Winter Olympics in Vancouver as the Microsoft Technologies Consultant for NBC Olympics, a role he played last year in Beijing and previously in Torino.

Claudia Woods – Technical Reviewer

Claudia has been a LAN Administrator, Systems Engineer and Technology Instructor for more than 10 years. As such, she has designed, implemented, and documented technology solutions for a variety of customers. Claudia has also written, edited, and presented customized technology courses for several organizations in the United States. She is a regular attendee at IT events such as TechEd, MCT Summit, and FOSE.

Originally hailing from the southeastern region of the United States, Claudia currently resides in the United Kingdom. She is a staff instructor with an international technology training firm. Her Microsoft specialties include Windows Server, Active Directory, and Exchange Messaging.

Ryan Boswell – Technical Reviewer

Ryan has worked as a Systems Engineer, IT Consultant, and Technology Instructor for more than 10 years. He holds several Microsoft certifications, including multiple levels of MCSE, MCTS, MCITP, and MCT. His specialties include Windows Server technologies, Active Directory, System Center Configuration Manager, System Center Operations Manager, and Microsoft Hyper-V™. Ryan currently resides in Denver, Colorado.

Contents

Module 1: Introducing Active Directory Domain Services (AD DS)

Lesson 1: Introducing Active Directory, Identity, and Access	1-4
Lesson 2: Active Directory Components and Concepts	1-21
Lesson 3: Install Active Directory Domain Services	1-46
Lab: Install an AD DS DC to Create a Single Domain Forest	1-56
Lesson 4: Extend IDA with Active Directory Services	1-64

Module 2: Secure and Efficient Administration of Active Directory

Lesson 1: Work with Active Directory Snap-ins	2-4
Lesson 2: Custom Consoles and Least Privilege	2-14
Lab A: Create and Run a Custom Administrative Console	2-25
Lesson 3: Find Objects in Active Directory	2-36
Lab B: Find Objects in Active Directory	2-53
Lesson 4: Use DS Commands to Administer Active Directory	2-62
Lab C: Use DS Commands to Administer Active Directory	2-81

Module 3: Manage Users

Lesson 1: Create and Administer User Accounts	3-4
Lab A: Create and Administer User Accounts	3-29
Lesson 2: Configure User Object Attributes	3-35
Lab B: Configure User Object Attributes	3-51
Lesson 3: Automate User Account Creation	3-61
Lab C: Automate User Account Creation	3-70

Module 4: Manage Groups

Lesson 1: Manage an Enterprise with Groups	4-4
Lesson 2: Administer Groups	4-45
Lab A: Administer Groups	4-66
Lesson 3: Best Practices for Group Management	4-74
Lab B: Best Practices for Group Management	4-88

Module 5: Support Computer Accounts

Lesson 1: Create Computers and Joining the Domain	5-4
Lab A: Create Computers and Joining the Domain	5-34
Lesson 2: Administer Computer Objects and Accounts	5-42
Lab B: Administer Computer Objects and Accounts	5-62

Module 6: Implement a Group Policy Infrastructure

Lesson 1: Understand Group Policy	6-4
Lesson 2: Implement GPOs	6-21
Lab A: Implement Group Policy	6-38
Lesson 3: A Deeper Look at Settings and GPOs	6-42
Lab B: Manage Settings and GPOs	6-64
Lesson 4: Manage Group Policy Scope	6-71
Lab C: Manage Group Policy Scope	6-102
Lesson 5: Group Policy Processing	6-110
Lesson 6: Troubleshoot Policy Application	6-120
Lab D: Troubleshoot Policy Application	6-132

Module 7: Manage Enterprise Security and Configuration with Group Policy Settings

Lesson 1: Delegate the Support of Computers	7-4
Lab A: Delegate the Support of Computers	7-16
Lesson 2: Manage Security Settings	7-20
Lab B: Manage Security Settings	7-48
Lesson 3: Manage Software with GPSI	7-61
Lab C: Manage Software with GPSI	7-80
Lesson 4: Auditing	7-86
Lab D: Audit File System Access	7-99

Module 8: Secure Administration

Lesson 1: Delegate Administrative Permissions	8-4
Lab A: Delegate Administration	8-25
Lesson 2: Audit Active Directory® Changes	8-33
Lab B: Audit Active Directory Changes	8-39

Module 9: Improve the Security of Authentication in an Active Directory Domain Services (AD DS) Domain

Lesson 1: Configure Password and Lockout Policies	9-4
Lab A: Configure Password and Account Lockout Policies	9-24
Lesson 2: Audit Authentication	9-30
Lab B: Audit Authentication	9-39
Lesson 3: Configure Read-Only Domain Controllers	9-43
Lab C: Configure Read-Only Domain Controllers	9-63

About This Course

This section provides you with a brief description of the course, audience, suggested prerequisites, and course objectives.

Course Description

The purpose of this 5-day course is to teach Active Directory® Technology Specialists how to configure Active Directory Domain Services (AD DS) in a distributed environment, implement Group Policy, perform backup and restore, and monitor and troubleshoot Active Directory–related issues. After completing this course, students will be able to implement and configure Active Directory Domain Services in their enterprise environment.

Audience

The primary audience for this course includes Active Directory Technology Specialists, Server Administrators, and Enterprise Administrators who want to learn how to implement Active Directory in a distributed environment; secure domains using Group Policy; perform backup and restore; and monitor and troubleshoot Active Directory configuration to ensure trouble-free operation.

Student Prerequisites

This course requires that you meet the following prerequisites:

- **Basic understanding of networking.** You should understand how TCP/IP functions and have a basic understanding of addressing, name resolution (Domain Name System [DNS]/Windows® Internet Name Service [WINS]), connection methods (wired, wireless, virtual private network [VPN]), and NET+ or equivalent knowledge.
- **Intermediate understanding of network operating systems.** You should have an intermediate understanding of operating systems such as Windows 2000, Windows XP, or Windows Server® 2003. An understanding of the Windows Vista® operating system client is nice to have.
- **An awareness of security best practices.** You should understand file system permissions, authentication methods, workstation, and server hardening methods, and so forth.
- **Basic knowledge of server hardware.** You should have an A+ or equivalent knowledge.

- **Some experience creating objects in Active Directory.**
- **Basic concepts of backup and recovery in a Windows Server Environment.** You should have basic knowledge of backup types, backup methods, backup topologies, and so forth.

Course Objectives

After completing this course, students will be able to:

- Position the strategic role of a directory service in an enterprise in relation to identity and access.
- Explain authentication and authorization processes.
- Identify the major components of AD DS.
- Understand the requirements for installing a domain controller to create a new forest.
- Identify the roles of and relationships between AD DS, Active Directory Lightweight Directory Services (AD LDS), Active Directory Rights Management Services (AD RMS), Active Directory Federation Services (AD FS), and Active Directory Certificate Services (AD CS).
- Install, locate, and describe the snap-ins used to administer AD DS.
- Perform basic administrative tasks with the Active Directory Users and Computers snap-in.
- Create a custom Microsoft® Management Console (MMC) console for administration.
- Perform administrative tasks while logged on as a user.
- Control the view of objects in the Active Directory Users and Computers snap-in.
- Locate objects in Active Directory.
- Work with saved queries.
- Identify the distinguished name (DN), relative distinguished name (RDN), and common name (CN) of an Active Directory object.
- Use the DS commands to administer Active Directory from the command line.
- Create and configure the account-related properties of a user object.

- Identify the purpose and requirements of user account attributes.
- Perform common administrative tasks to support user accounts, including password reset and account unlock.
- Enable and disable user accounts.
- Delete, move, and rename user accounts.
- View and modify hidden attributes of user objects.
- Identify the purpose and requirements of user object attributes.
- Create users from user account templates.
- Modify attributes of multiple users simultaneously.
- Export user attributes with Comma Separated Value Directory Exchange (CSVDE).
- Import users with CSVDE.
- Import users with Lightweight Directory Access Protocol (LDAP) Data Interchange Format Directory Exchange (LDIFDE).
- Create well-documented, secure, delegated groups.
- Understand group types, scope, and nesting.
- Understand the best practice for group nesting to achieve role-based management.
- Create, delete, and manage groups with DSCommands, CSVDE, and LDIFDE.
- Enumerate and copy group membership.
- Understand Default (Built In) groups.
- Understand Special Identities.
- Understand the relationship between a domain member and the domain in terms of identity and access.
- Identify the requirements for joining a computer to the domain.
- Implement best practice processes for computer joins.
- Secure AD DS to prevent the creation of unmanaged computer accounts.
- Manage computer objects and their attributes using the Windows interface and command-line tools.
- Administer computer accounts through their life cycle.

- Identify the business drivers for configuration management.
- Understand the components and technologies that comprise the Group Policy framework.
- Manage Group Policy objects (GPOs).
- Configure and understand a variety of policy setting types.
- Scope GPOs using links, security groups, WMI filters, loopback processing, and Preference targeting.
- Explain GPO storage, replication, and versioning.
- Administer a Group Policy infrastructure.
- Evaluate GPO inheritance, precedence, and Resultant Set of Policy (RSOP).
- Locate the event logs containing Group Policy–related events.
- Delegate the administration of computers.
- Use Restricted Groups policies to modify or enforce the membership of groups.
- Use Group Policy Preferences to modify the membership of groups.
- Configure security settings by using the Local Security policy.
- Create and apply security templates to manage security configuration.
- Analyze security configuration based on security templates.
- Create, edit, and apply security policies using the Security Configuration Wizard.
- Deploy security configuration with Group Policy.
- Deploy software using Group Policy Software Installation (GPSI).
- Remove software originally installed with GPSI.
- Describe the business purpose of delegation.
- Assign permissions to Active Directory objects using the security editor user interfaces and the Delegation Of Control Wizard.
- View and report permissions on Active Directory objects by using user interface and command-line tools.
- Reset the permissions on an object to the default.
- Describe the relationship between delegation and OU design.

- Configure Directory Service Changes auditing.
- Specify auditing settings on Active Directory objects.
- Identify event log entries created by Directory Access auditing and Directory Service Changes auditing.
- Implement your domain password and account lockout policy.
- Configure and assign fine-grained password policies.
- Configure auditing of authentication-related activity.
- Distinguish between account logon and logon events.
- Identify authentication-related events in the Security log.
- Identify the business requirements for Read-Only Domain Controllers (RODCs).
- Install an RODC.
- Configure password replication policy.
- Monitor the caching of credentials on an RODC.
- Understand the structure role, structure, and functionality of the Domain Name System (DNS).
- Describe client and server name resolution processes.
- Install DNS.
- Manage DNS records.
- Configure DNS server settings.
- Understand the integration between AD DS and DNS.
- Choose a DNS domain for an Active Directory domain.
- Create a zone delegation for a new Active Directory domain.
- Configure replication for Active Directory–integrated zones.
- Describe the purpose of SRV records in the domain controller location process.
- Understand read-only DNS servers.
- Understand and configure single-label name resolution.
- Configure advanced DNS server settings.

- Audit, maintain, and troubleshoot the DNS server role.
- Install a standard or read-only domain controller into new or existing domains or trees.
- Add and remove domain controllers using a variety of GUI or command-line methods.
- Configure a domain controller on Server Core.
- Understand and identify operations master roles.
- Manage the placement, transfer, and seizure of operations master roles.
- Migrate SYSVOL replication from File Replication Service (FRS) to Distributed File System Replication (DFS-R).
- Configure sites and subnets.
- Understand domain controller location and manage domain controllers in sites.
- Configure replication of the partial attribute set to global catalog servers.
- Implement universal group membership caching.
- Understand the role of application directory partitions.
- Configure replication topology with connection objects, bridgehead servers, site links, and site link bridges.
- Report, analyze, and troubleshoot replication with repadmin.exe and dcdiag.exe.
- Monitor real-time performance and events with Task Manager, Event Viewer, and Windows Reliability And Performance Monitor.
- Leverage new features of Event Viewer in Windows Server 2008, including custom views and event subscriptions.
- Monitor real-time and logged performance with Performance Monitor, data collection sets, and reports.
- Identify sources of performance and event information for AD DS domain controllers.
- Create alerts based on events and performance metrics.
- Maintain and optimize the Active Directory database.
- Back up and restore AD DS and domain controllers.

- Recover deleted objects and attributes.
- Understand domain and forest functional levels.
- Raise domain and forest functional levels.
- Identify capabilities added by each functional level.
- Design an effective domain and tree structure for AD DS.
- Identify the role of the Active Directory Migration Tool and the issues related to object migration and domain restructure.
- Understand trust relationships.
- Configure, administer, and secure trust relationships.

Course Outline

This section provides an outline of the course:

Module 1: This module explains how to install and configure Active Directory Domain Services and install and configure a read-only domain controller.

Module 2: This module explains how to work securely and efficiently in Active Directory.

Module 3: This module explains how to manage and support user accounts in Active Directory.

Module 4: This module explains how to create, modify, delete, and support group objects in Active Directory.

Module 5: This module explains how to create and configure computer accounts.

Module 6: This module explains what Group Policy is, how it works, and how best to implement Group Policy in your organization.

Module 7: This module explains how to manage security and software installation and how to audit files and folders.

Module 8: This module explains how to administer Active Directory Domain Services securely.

Module 9: This module explains the domain-side components of authentication, including the policies that specify password requirements and the auditing of authentication-related activities.

Module 10: This module explains how to implement DNS to support name resolution both within your AD DS domain and outside your domain and your intranet.

Module 11: This module explains how to administer domain controllers in a forest.

Module 12: This module explains how to create a distributed directory service that supports domain controllers in portions of your network that are separated by expensive, slow, or unreliable links.

Module 13: This module explains about the technologies and tools that are available to help ensure the health and longevity of the directory service. You will explore tools that help you monitor performance in real time, and you will learn to log performance over time so that you can keep an eye on performance trends in order to spot potential problems.

Module 14: This module explains how to raise the domain and forest functionality levels within your environment, how to design the optimal AD DS infrastructure for your enterprise, how to migrate objects between domains and forests, and how to enable authentication and resources access across multiple domains and forests.

Course Materials

The following materials are included with your kit:

- *Course Handbook*. The Course Handbook contains the material covered in class. It is meant to be used in conjunction with the Course Companion CD.
- *Course Companion CD*. The Course Companion CD contains the full course content, including expanded content for each topic page, full lab exercises and answer keys, and topical and categorized resources and Web links. It is meant to be used both inside and outside the class.



Note: To access the full course content, insert the Course Companion CD into the CD-ROM drive, and then in the root directory of the CD, double-click StartCD.exe.

- *Course evaluation*. At the end of the course, you will have the opportunity to complete an online evaluation to provide feedback on the course, training facility, and instructor.

To provide additional comments or feedback on the course, send e-mail to support@mscourseware.com. To inquire about the Microsoft Certification Program, send e-mail to mcp@microsoft.com.

Virtual Machine Environment

This section provides the information for setting up the classroom environment to support the business scenario of the course.

Virtual Machine Configuration

In this course, you will use Microsoft Virtual Server 2005 and MSL Lab Launcher to perform the labs.



Important: At the end of each lab, you must close the virtual machine and must not save any changes. To close a virtual machine without saving the changes, perform the following steps: 1. For each virtual machine that is running, close the Virtual Machine Remote Control window. 2. In the **Close** box, select **Turn Off Machine** And Discard Changes. Click **OK**.

The following table shows the role of each virtual machine that this course uses:

Virtual machine	Role
6425B-HQDC01-A	Windows Server 2008 DC in Contoso Domain
6425B-HQDC01-B	Windows Server 2008 DC in Contoso Domain
6425B-HQDC01-D	Windows Server 2008 WorkGroup Member
6425B-HQDC02-A	Windows Server 2008 Work Group Member
6425B-HQDC02-B	Windows Server 2008 DC in Contoso Domain
6425B-HQDC03-A	Windows Server 2008 Server Core in Work Group
6425B-HQDC03-B	Windows Server 2008 Server Core in Contoso Domain
6425B-DESKTOP101-A	Windows Vista Client in Contoso Domain
6425B-BRANCHDC01-A	Windows Server 2008 WorkGroup Member
6425B-BRANCHDC01-B	Windows Server 2008 Server Core DC in Contoso Domain
6425B-SERVER01-A	Windows Server 2008 DC in Contoso Domain
6425B-SERVER01-B	Windows Server 2008 WorkGroup Member
6425B-TSTD01-A	Windows Server 2008 DC in Tailspintoys Domain

Software Configuration

The following software is installed on the virtual machines:

- Windows Server 2008 Enterprise
- Windows Vista Enterprise

Classroom Setup

Each classroom computer will have the same virtual machine configured in the same way.

Course Hardware Level

To ensure a satisfactory student experience, Microsoft Learning requires a minimum equipment configuration for trainer and student computers in all Microsoft Certified Partner for Learning Solutions (CPLS) classrooms in which Official Microsoft Learning Product courseware is taught.

This course is a Hardware Level 5.5 course with additional RAM. Please see the classroom setup guide for detailed hardware spec.



Important: The Hardware Level in this course has been modified to run by default under the assumption that 4 GB RAM is available in the host machine rather than 2 GB RAM, which is the normal amount of memory required, defined by Hardware Level 5.5. So the default configuration on installation and boot up is configured to run where there is 4 GB RAM available in the host machine. For detailed steps on how to set this environment up, please follow the steps outlined in the Classroom Configuration – Hardware Level 5.5 with 4GB RAM section in the classroom setup guide.

If you do not have 4 GB RAM available in the student machines, you will need to follow alternative setup steps. An alternative LauncherSettings.config file is provided with the course, which will redefine the RAM values for each of the virtual machines to allow them to boot up and run at the normal, Hardware Level 5.5 definition, allocation of 2 GB RAM being available in the host machine. For details on how to setup the classroom where only 2 GB is available in the student machines, please see the Classroom Configuration – Hardware Level 5.5 with 4GB RAM section in the classroom setup guide.

It is also highly recommended that you read the MSL Lab Launcher Getting Started Guide, which is available in the MCT Download Center. This contains information about how to install and customize the MSL Lab Launcher in general terms and will be complementary to what is contained in this course-specific set up guide.

Each classroom computer will serve as the host for five virtual machines that will run in Virtual Server 2005R2 SP1.

Estimated Classroom setup time is approximately 140 minutes.

Module 1

Introducing Active Directory Domain Services (AD DS)

Contents:

Lesson 1: Introducing Active Directory, Identity, and Access	1-4
Lesson 2: Active Directory Components and Concepts	1-21
Lesson 3: Install Active Directory Domain Services	1-46
Lab: Install an AD DS DC to Create a Single Domain Forest	1-56
Lesson 4: Extend IDA with Active Directory Services	1-64

Module Overview

- Introducing Active Directory, Identity, and Access
- Active Directory Components and Concepts
- Install Active Directory Domain Services
- Extend IDA with Active Directory Services

Active Directory® and its related services form the foundation for enterprise networks running Windows® as, together, they act to store information about the identities of users, computers, and services; to authenticate a user or computer; and to provide a mechanism with which the user or computer can access resources in the enterprise. In this module, you will begin your exploration of Windows Server® 2008 Active Directory by installing the Active Directory Domain Services (AD DS) role and creating a domain controller (DC) in a new Active Directory forest. You will find that Windows Server 2008 continues the evolution of Active Directory by enhancing many of the concepts and features with which you are familiar from your experience with Active Directory.

This module focuses on the creation of a new Active Directory forest with a single domain in a single DC. The Lab in this module will guide you through the creation of a domain named *contoso.com* that you will use for all other labs in this course. In later modules, you will learn to implement other scenarios, including multidomain forests, upgrades of existing forests to Windows Server 2008, and advanced installation options.

Most importantly, this module sets the stage for the entire course by presenting a "big picture" view of Active Directory. You will review key concepts of authentication, authorization, and directory services, and you will take a high-level look at the major components of Active Directory and how they fit together. Whether you are highly experienced with Active Directory or newer to the platform, this module will equip you with an understanding of where you are heading in this course.

Objectives

After completing this module, you will be able to:

- Position the strategic role of a directory service in an enterprise in relation to identity and access.
- Explain authentication and authorization processes.
- Identify the major components of AD DS.
- Understand the requirements for installing a domain controller to create a new forest.
- Identify the roles of and relationships between AD DS, AD LDS, AD RMS, AD FS, and AD CS.

Lesson 1

Introducing Active Directory, Identity, and Access

- Information Protection in a Nutshell
- Identity and Access (IDA)
- Authentication and Authorization
- Authentication
- Access Tokens
- Security Descriptors, ACLs and ACEs
- Authorization
- Stand-alone (Workgroup) Authentication
- Active Directory Domains: Trusted Identity Store
- Active Directory, Identity, and Access

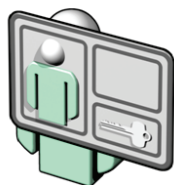
Active Directory Domain Services (AD DS) provides the functionality of an identity and access (IDA) solution for enterprise networks. The lesson reviews key concepts of IDA and Active Directory.

Objectives

After completing this lesson, you will be able to:

- Explain authentication and authorization concepts, terminology, processes, and technologies.
- Position the strategic role a directory service in an enterprise in relation to identity and access.

Information Protection in a Nutshell



- It's all about connecting users to the information they require
... SECURELY!
- IDA: Identity and Access
- AAA: Authentication, Authorization, Accounting
- CIA: Confidentiality, Integrity, Availability (& Authenticity)



Key Points

If you boil it all down, the job of an information technology (IT) professional (IT pro) is to connect users with the information they require to get *their* jobs done. That would be pretty easy, if we didn't have to worry about a little thing called "security." Because users require different levels of access to different classes of information, we must manage associating the correct users with the correct levels of access: information protection.

The industry defines several approaches to achieving information protection. Each of these "alphabet soup" frameworks is simply a different perspective on the same problem:

- IDA: Identity and Access. Users and other *security principals* (which may include computers, services, and groups) are represented as *identities* (frequently called "accounts") that are given *access* (permissions) to information, resources, or systems.

- AAA: Authentication, Authorization, and Accounting. Users provide *credentials* such as a username and password that are *authenticated* when they provide credentials that can be validated. Users are given permissions to resources (*access control*) that are used to *authorize* requests for access. Access is monitored, providing *accounting* and auditing. In some documentation, auditing is split out as a separate "A" from accounting, leading to the acronym, "AAAA."
- CIA: Confidentiality, Integrity, and Availability. Information is protected so that it is not disclosed to unauthorized individuals (*confidentiality*) and is not modified incorrectly (*integrity*) intentionally or accidentally. Information is available when needed (*availability*).

Additional Reading

- Microsoft Identity and Access Solutions - <http://go.microsoft.com/fwlink/?LinkId=168485>

Identity and Access (IDA)



- Identity: user account
- Saved in an identity store (directory database)
- Security principal
- Represented uniquely by the security identifier (SID)



- Resource: Shared Folder
- Secured with a security descriptor
- Discretionary access control list (DACL or "ACL")
- Access control entries (ACEs or "permissions")

Key Points

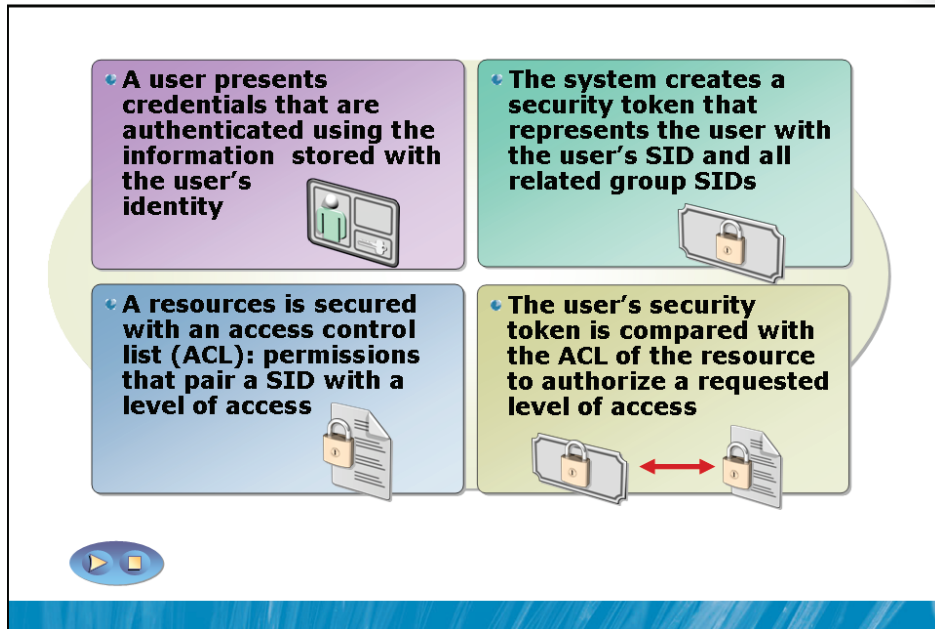
And at the core of information protection are two critical concepts: identity and access, or IDA.

Let's spend a few minutes reviewing the fundamentals, components, processes, and technologies involved with identity and access on Windows systems. Although most or all of this information should be familiar to you from your previous experience with Windows, it is important to set the stage for the role of Active Directory, and to clarify the terminology, components, and processes involved with IDA.

In a secured system, each user is represented by an identity. In the Windows systems, the identity is the user account. The accounts for one or more users are maintained in an *identity store*, also known as a directory database. An identity is called a *security principal* in Windows systems. Security principals are uniquely identified by an attribute called the *security identifier*, or SID.

On the other end of the system is the resource to which the user requires *access*. The resource is secured with permissions, and each permission specifies a pairing of a specific level of access with an identity. Many Windows resources, including and most significantly files and folders on NTFS volumes, are secured by an aptly-named *security descriptor* that contains a *discretionary access control list* (DACL) in which each permission takes the form of an *access control entry* (ACE).

Authentication and Authorization



Key Points

Between the user (security principal) and access to the resource are some important concepts and processes.

The next four slides will detail this process.

Authentication

Authentication is the process that verifies a user's identity

Credentials: at least two components required

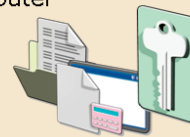
- Username
- Secret, for example, password

Two types of authentication

- Local (interactive) Logon – authentication for logon to the local computer



- Remote (network) logon – authentication for access to resources on another computer

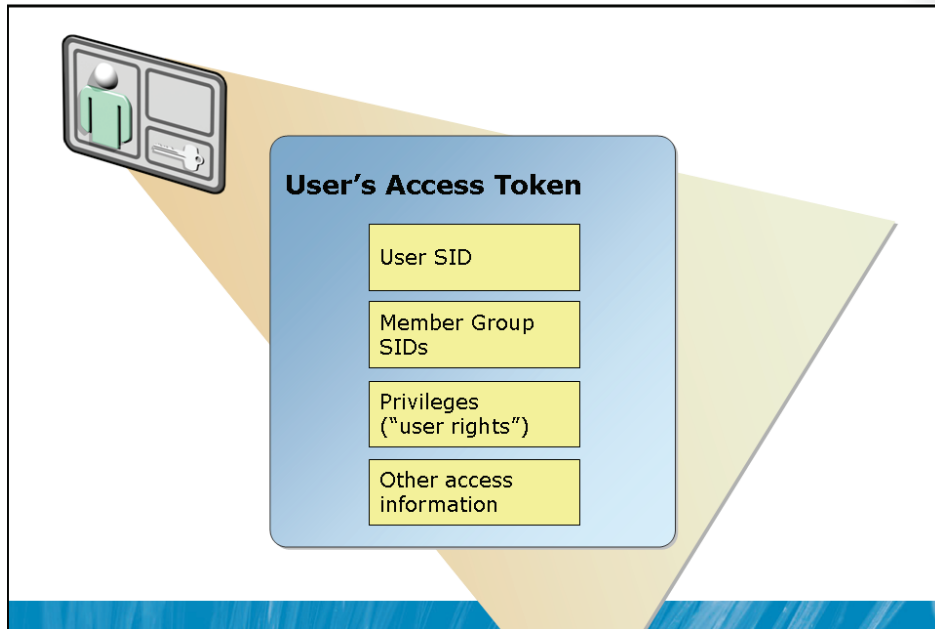


Key Points

Authentication is the process of verifying a user's identity. The user supplies *credentials* that consist of at least two components: a logon name and a secret known only to the user and the system, such as a password. The system validates the accuracy of the credentials presented by the user against those stored as part of the identity.

There are two types of authentication: local and remote. Local, or interactive, logon occurs when a user logs on to a computer directly, for example when you log on to your laptop in the morning. Remote, or network, logon occurs when you connect to another computer, such as a file server, mail server, or even a domain controller to retrieve a logon script.

Access Tokens



Key Points

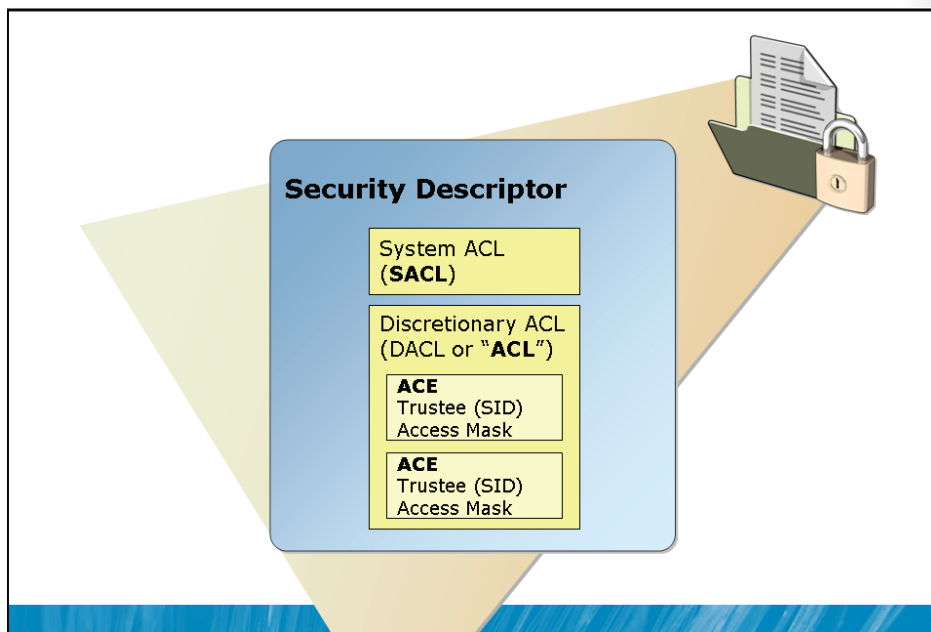
After a user has been authenticated, the Local Security Authority (LSA) generates a *security access token* (also called a *security token* or *access token*) that represents the user to the system by collecting the user's SID and the SIDs of all groups to which the user belongs. The access token also represents privileges (also called *user rights*) held by the user on the system, for example the right to shut down the system or even the right to log on to the system interactively (locally).

It is important to remember that the access token is generated and held locally, on the computer that authenticated the user. When a user logs on to his or her desktop (local or interactive logon), the desktop creates a security token and, assuming the user has the right to log on to the system interactively, proceeds to invoke the Windows Explorer process, which creates the desktop.

When a user then connects to a server to access a shared file (remote or network logon), the server authenticates the user and generates an access token on the server that represents the user with the user's SID and the SIDs of all groups to which that user belongs. The access token on the server is distinct from the access token on the user's desktop. An access token is never transmitted over the network, and the LSA of a Windows system would never accept the access token generated by another LSA.

Of course, this should be the case because a user probably belongs to different local groups on the server than on the user's desktop, and almost certainly holds different privileges (user rights) on the server than on the desktop.

Security Descriptors, ACLs, and ACEs

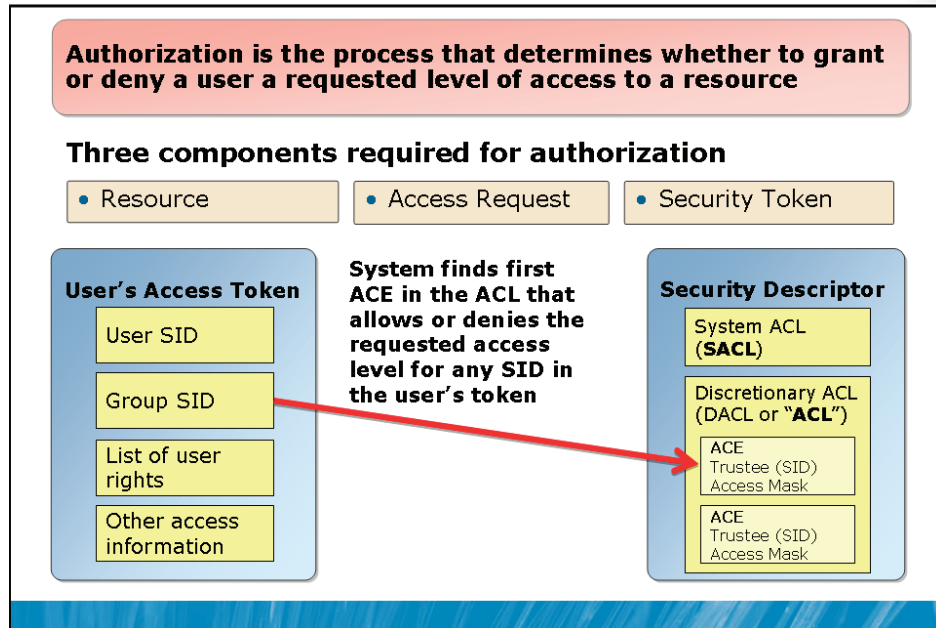


Key Points

The *security descriptor* of a secured resource, such as a file or folder on an NTFS volume, fully describes the security characteristics of the resource. The security descriptor contains the *discretionary access control list* (DACL), which contains *access control entries* (ACEs or "permissions"). Each permission is made up of a flag that indicates whether the ACE is an Allow or Deny ACE; a Trustee (the SID of a user or group); and an access mask specifying a level of access. So the ACE defines *who* (the Trustee represented by the SID) can or can't do *what* (represented by the access mask).

The security descriptor also contains the *system access control list* (SACL), which contains auditing settings and attributes such as the object's owner. Because the DACL is the focus of most day-to-day security management for a resource, the name and acronym is often shortened. Therefore, the shortened *access control list*, or ACL, while technically inaccurate, is used by many administrators and much documentation (including this course) to refer to the DACL.

Authorization



Key Points

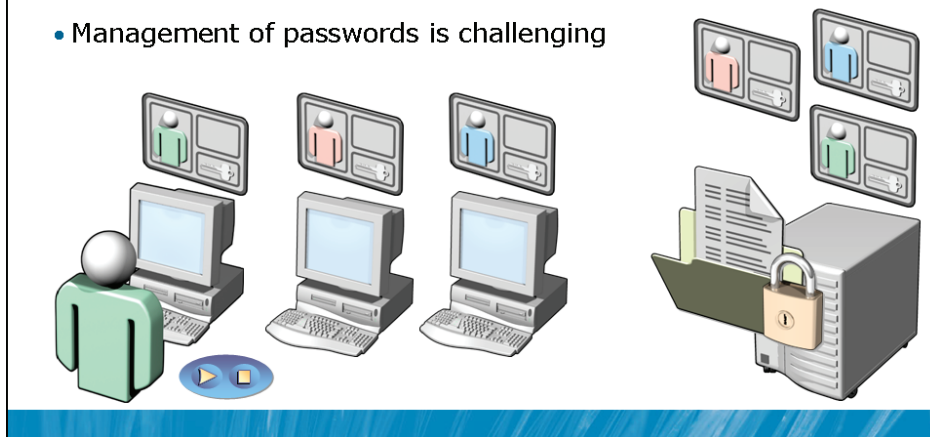
Authorization is the process that determines whether to grant or deny a user a requested level of access to a resource. An access request is made that indicates the resource, the level of access, and the security token representing the user. The security subsystem then examines the ACL of the resource, comparing the SIDs in the ACEs to the SIDs in the security token. The first ACE that matches both a SID in the token and the desired type of access determines whether the user is allowed (if the ACE is an Allow ACE) or denied (if the ACE is a Deny ACE) access to the resource. If no match is found, access is denied.

Additional Reading

- Logon and Authentication Technologies:
<http://go.microsoft.com/fwlink/?LinkId=168486>
- Authorization and Access Control Technologies:
<http://go.microsoft.com/fwlink/?LinkId=168488>

Stand-alone (Workgroup) Authentication

- The identity store is the security accounts manager (SAM) database on the Windows system
- No shared identity store
- Multiple user accounts
- Management of passwords is challenging



Key Points

In a stand-alone configuration of Windows systems, also called a *workgroup*, each computer maintains one and only one trusted identity store: a local list of users and groups stored in the registry called the *Security Accounts Manager database*, or SAM.

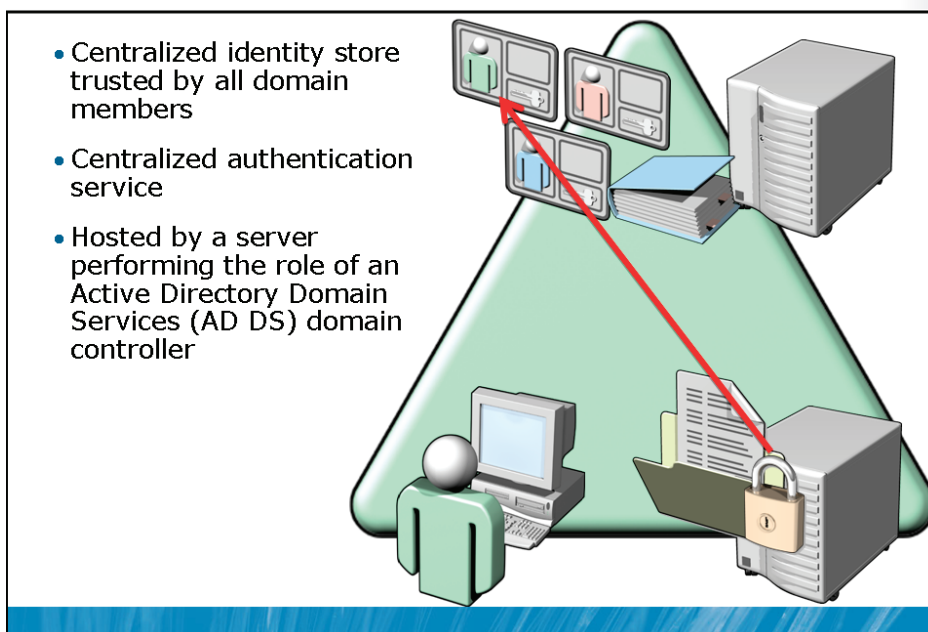
Because Windows systems are secure, a user cannot even log on to a computer without a user account on that system. The user must present credentials that are validated against the identities in the SAM. After a user has been authenticated and authorized for local logon, the Windows Explorer process is launched, which generates the familiar Windows desktop.

If the user wishes to access a shared folder on a server, there is an immediate problem: the server does not trust an identity presented to it because the identity has been authenticated by an unknown and untrusted system. The server trusts only its own identity store—its own SAM. Therefore, in order for the user to remotely log on to the server, the server must have an identity (user account) for the user in its SAM. If the logon name and password for the identity is identical to the credentials of the identity on the workstation, the authentication process that occurs is transparent to the user, but it does happen. If, however, the logon names or passwords do not match, the user will be prompted to enter credentials that are valid for the server when the user attempts to connect to the shared resource.

The ACL on a secured resource on the server cannot contain permissions that refer to untrusted identities, therefore all users that require access to the resource must have accounts on the server.

This presents immediately obvious management challenges. If the user changes his or her password on the desktop, the two accounts are no longer in sync, and the user will be prompted for credentials when connecting to the server. The problem only gets worse as you add more users, resources, and Windows systems to the environment. The management challenges of maintaining multiple identities for each user become quickly untenable.

Active Directory Domains: Trusted Identity Store



Key Points

The management and security challenges of a workgroup are solved by centralizing the identity store so that there is only one identity (user account) required for any one user—an identity store that is trusted by all computers. This unit of trusted identity is created by the introduction of an Active Directory *domain*.

An Active Directory domain provides a centralized identity store trusted by all domain members—that is, all computers that themselves maintain accounts in the domain. A domain also provides a centralized authentication service. Both the identity store (the Active Directory database) and the authentication service, along with a number of other components and services about which you will learn to wrap this course, are hosted on a server performing the role of a *domain controller*.

Active Directory, Identity, and Access

- An IDA infrastructure should
 - Store information about users, groups, computers and other identities
 - Authenticate an identity
 - Kerberos authentication used in Active Directory provides **single sign-on**. Users are authenticated only once.
 - Control access
 - Provide an audit trail
- Active Directory services
 - Active Directory Domain Services (AD DS)
 - Active Directory Lightweight Directory Services (AD LDS)
 - Active Directory Certificate Services (AD CS)
 - Active Directory Rights Management Services (AD RMS)
 - Active Directory Federation Services (AD FS)

Key Points

As mentioned in the introductions to the module and this lesson, Active Directory provides the IDA solution for enterprise networks running Windows. IDA is necessary to maintain the security of enterprise resources such as files, e-mail, applications, and databases. And IDA infrastructure should do the following:

- **Store information about users, groups, computers and other identities.** An identity is, as you've learned, a representation of an entity that will perform actions on the enterprise network. For example, a user will open documents from a shared folder on a server. You know that the document will be secured with permissions on an ACL. Access to the document is managed by the security subsystem of the server, which compares the identity of the user to the identities on ACL to determine whether the user's request for access will be granted or denied. Computers, groups, services, and other objects also perform actions on the network; they must be represented by identities. Among the information stored about an identity are properties that uniquely identify the object, such as a username or a SID, and the password for the identity. The identity store is therefore one component of an IDA infrastructure. The Active Directory data store, also known as the *directory*, is an identity store. The directory itself is hosted on and managed by a domain controller—a server performing the AD DS role.
- **Authenticate an identity.** The server will not grant the user access to the document unless the server has confidence that the identity presented in the access request is valid. To validate the identity, the user provides secrets known only to the user and the IDA infrastructure. Those secrets are compared to the information in the identity store in a process called *authentication*.

In an Active Directory domain, a protocol called Kerberos is used to authenticate identities. When a user or computer logs on to the domain, Kerberos authenticates its credentials and issues a package of information called a *ticket granting ticket* (TGT). Before the user connects to the server to request the document, a Kerberos request is sent to a domain controller along with the TGT that serves to identify the authenticated user. The domain controller issues the user another package of information called a *service ticket* that identifies the authenticated user to the server. The user presents the service ticket to the server, which accepts the service ticket as proof that the user has been authenticated.

These Kerberos transactions result in a single network logon, or *single sign-on*. After the user or computer has initially logged on and has been granted a TGT, the user is authenticated within the entire domain and can be granted service tickets that identify the user to any service. All of this ticket activity is managed by the Kerberos clients and services built into Windows and is transparent to the user.

- **Control access.** The IDA infrastructure is responsible for protecting confidential information such as the information stored in the document. Access to confidential information must be managed according to the policies of the enterprise. The ACL on the document reflects a security policy comprised of permissions that specify access levels for particular identities. The security subsystem of the server in this example is performing the access control functionality in the IDA infrastructure.
- **Provide an audit trail.** An enterprise may want to monitor changes to and activities within the IDA infrastructure, so it must provide a mechanism with which to manage auditing.

Active Directory Domain Services is the most prominent component of an IDA infrastructure, but it is not the only component of IDA that is supported by Windows Server 2008. With the release of Windows Server 2008, Microsoft has consolidated a number of previously separate components into an integrated IDA platform. Later in this module, you will learn about the following Active Directory services:

- Active Directory Lightweight Directory Services (AD LDS)
- Active Directory Certificate Services (AD CS)
- Active Directory Rights Management Services (AD RMS)
- Active Directory Federation Services (AD FS)

Each of these services plays a role in extending IDA to support more complex configurations and scenarios. Again, these details will be provided later in this module.

Lesson 2

Active Directory Components and Concepts

- Active Directory as a Database
- Demonstration: Active Directory Schema
- Organizational Units
- Policy-Based Management
- The Active Directory Data Store
- Domain Controllers
- Domain
- Replication
- Sites
- Tree
- Forest
- The Global Catalog
- Functional Level
- DNS and Application Partitions
- Trust Relationships

Modules 2–14 of this course detail the installation, configuration, management, and troubleshooting of AD DS. It is worthwhile to first gain an overview—a "big picture"—of the components, technologies, and concepts related to Active Directory.

Objectives

After completing this lesson, you will be able to:

- Identify the major components of AD DS.

Active Directory As a Database

- Active Directory is a database
 - Each "record" is an object
 - Users, groups, computers, ...
 - Each "field" is an attribute
 - Logon name, SID, password, description, membership, ...
 - Identities (security principals or "accounts")
- Services: Kerberos, DNS, replication, etc.

AD DS is, in the end, a database and the services that support or use that database
- Accessing the database
 - Windows tools, user interfaces, and components
 - APIs (.NET, VBScript, Windows PowerShell)
 - Lightweight Directory Access Protocol (LDAP)

Key Points

Active Directory is, in the end, a database of enterprise resources and configuration. A suite of services support that database and use the information in the database to provide enterprise identity and access. In database terminology, each "record" in the Active Directory database is an Active Directory object, such as a user, group, or computer. Each "field" is an attribute, also called a *property*, of an object. Attributes include the object's name, password, description, membership, or SID.

Security principals, also called "accounts," are a specific type of object in AD DS. Security principals have several unique attributes, the most important of which is the SID. The SID is used, as you learned in the previous lesson, to assign resource access to the account.

In the previous lesson, you focused on only one security principal: users. However, it is easier to manage resource access when you assign permissions to a group, and there is a class of group object, called a *security group*, that is also a security principal. Computers in a domain are also security principals. In fact, in the computer object is very similar to a user object: it has a logon name and password that the computer uses to authenticate with the domain at startup.

Finally, there is a class of objects called *inetOrgPerson*. This object class is used in very specific situations to support interoperability with a handful of third party directory services. *inetOrgPerson* is also a security principal and is, for sake of brevity, very similar to a user account.

The Active Directory database is supported and used by a number of services, including Kerberos (responsible for authentication), DNS (responsible for name resolution), and the directory replication agent (DRA), responsible for replicating the database between domain controllers.

The Active Directory database can be accessed a number of ways, using a variety of Windows components, tools, and interfaces, or programmatically through APIs, or using lightweight directory access protocol (LDAP).

Demonstration: Active Directory Schema

In this demonstration, we will

- Discover how the Schema acts as a blueprint for Active Directory by defining Attributes
 - objectSID
 - sAMAccountName
 - unicodePwd
 - member
 - Description
- and Object classes
 - User
 - Group

Key Points

In this demonstration, your instructor will introduce you to the role and structure of the schema by giving you a tour of the Active Directory Schema.

The schema is often compared to a blueprint for Active Directory. It defines the attributes and types of objects that can be stored in the directory. For example, the fact that Active Directory can have user objects, and that user objects are required to have a logon name and optionally an e-mail address is all determined by the schema.

The schema has two primary containers. The Attributes container holds definitions of every attribute supported by Active Directory. You can open the attributes for properties with which you are already familiar:

- **objectSID**: Security identifier.
- **sAMAccountName**: The pre-Windows 2000 logon name, which most administrators refer to as the "username."

- **unicodePwd:** The storage of the password. This attribute stores a password as a hash code that results from a one-way function.

You cannot read or derive the actual password from this attribute without performing some kind of brute force dictionary attack (hacking).

- **member:** The attribute that stores the membership list for a group object.

The objectClasses container defines the types of objects that can be instantiated (created) in the directory, including user and group. Object classes are associated with attributes defined in the Attributes container. These associations determine what object classes have which attributes, and which of those attributes are mandatory for a particular object class.

Demonstration Steps

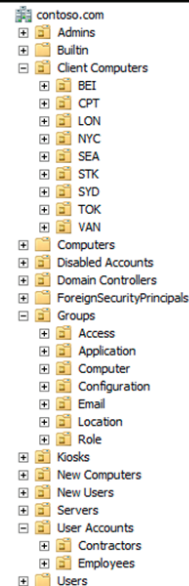
1. On the virtual machine 6425B-HQDC01-A open D:\AdminTools\ADConsole.msc. Expand the Active Directory node > Active Directory Schema [HQDC01.contoso.com] node.
2. Look at the Attributes container. Open the Properties of the following.
 - objectSID
 - sAMAccountName (what most admins call the “user name”)
 - unicodePwd
 - member
 - description
3. Open the Classes container. While scrolling through, notice familiar object classes, including user, computer, and group.

Additional Reading

- What Is the Active Directory Schema?
<http://go.microsoft.com/fwlink/?LinkId=104448>

Organizational Units

- Containers
 - Users
 - Computers
- Organizational Units
 - Containers that also support the management and configuration of objects using Group Policy
 - Create OUs to
 - Delegate administrative permissions
 - Apply Group Policy



Key Points

Active Directory is a hierarchical database. Objects in the data store can be collected in containers. One type of container is the object class called container. You have seen the default containers, including Users, Computers, and Builtin, when you open the Active Directory Users and Computers snap-in. Another type of container is the organizational unit (OU). OUs provide not only a container for objects, but also a scope with which to manage the objects. That is because OUs can have objects called Group Policy objects (GPOs) linked to them. GPOs can contain configuration settings that will then be applied automatically by users or computers in an OU.

Additional Reading

- Modules 6 and 8 examine the purpose, management, and design of organizational units.

Policy-Based Management

- Active Directory provides a single point of management for security and configuration through policies
 - Group Policy
 - Domain password and lockout policy
 - Audit policy
 - Configuration
 - Applied to users or computers by scoping a GPO containing configuration settings
 - Fine-grained password and lockout policies



Key Points

Policy-based administration eases the management burden of even the largest, most complex networks by providing a single point to configure settings that are then deployed to multiple systems.

Group Policy allows you to define security settings as well as thousands of configuration settings for one or more users or computers in your enterprise. For example, it is Group Policy that defines password and lockout policies for a domain, specifying minimum password length and password expiration policy. Group Policy can specify auditing settings, for example to monitor access to folders on the server, or to watch for changes to security sensitive groups in the Active Directory, such as Domain Admins. Group Policy can also manage configuration, for example specifying a Microsoft® Internet Explorer® home page for a group of users or preventing users from accessing registry editing tools.

The important concept of Group Policy to understand at this point in the course is that Group Policy allows you to define configuration in an object called a Group Policy object (GPO). A GPO can then be scoped (applied) to one or more users or computers.

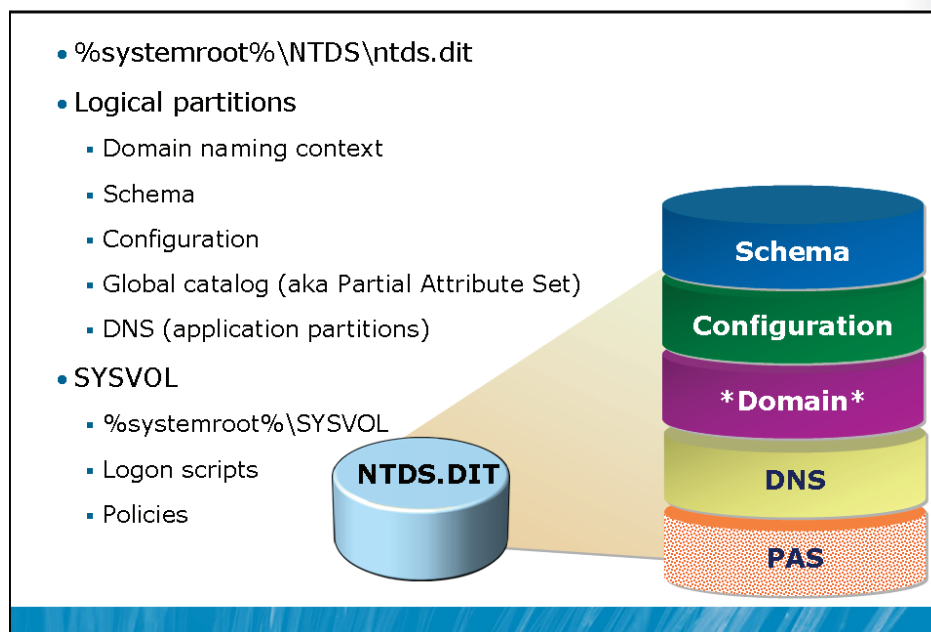
Another example of policy-based management is fine-grained password and lockout policies, a new feature of Windows Server 2008. You can now specify different password and lockout policies for different groups of users in your environment. For example, you can configure a longer minimum password length and a more frequent password change policy for members of Domain Admins than for normal users.

It is interesting and important to note that these technologies enable Active Directory to go beyond simple identity and access management, and to make a significant contribution to the broader management of your enterprise network.

Additional Reading

- Modules 6 through 9 detail policy based management.

The Active Directory Data Store



Key Points

As mentioned in the previous lesson, AD DS stores its identities in the directory—a data store hosted on domain controllers. The directory is a single file named `ntds.dit`, and is located by default in the `%systemroot%\ntds` folder on a domain controller.

The database is divided into several partitions, which will be detailed in later modules. The partitions include:

- **Schema:** Discussed in a previous topic.
- **Domain naming context (Domain NC):** A particularly important partition for day-to-day administration, because it contains the data about objects within a domain—the users, groups, and computers, for example. When you make changes to Active Directory using the Active Directory Users and Computers snap in, you are modifying the contents of the Domain NC.
- **Configuration:** Contains information about domains, services and topology.

- **DNS:** If you use Active Directory-integrated DNS, the DNS zones and resource records are stored in a partition.
- **Partial Attribute Set (PAS):** This partition is used by the Global Catalog, which is detailed in a later topic in this lesson, and in Module 12.

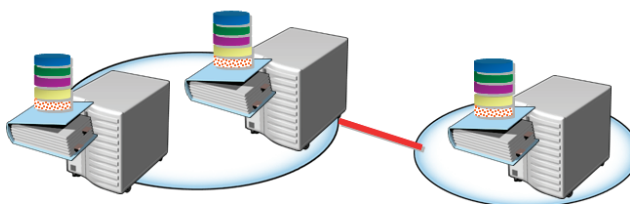
Active Directory also stores information in a folder structure called SYSVOL. By default, this folder is located in the %systemroot% folder (c:\windows). SYSVOL contains items including logon scripts and files related to Group Policy objects.

Additional Reading

- You will learn more about the partitions of Active Directory and about SYSVOL throughout this course. DNS is a focus of Module 10, and the PAS is examined in detail in Module 12. The contents of SYSVOL are explored in Module 6 and the objects stored in the Configuration are covered in Module 12. The objects in the Domain partition are covered in Modules 3-6 and database maintenance and administration tasks are detailed in Modules 9 and 13.

Domain Controllers

- Servers that perform the AD DS role
 - Host the Active Directory database (NTDS.DIT) and SYSVOL
 - Replicated between domain controllers
 - Kerberos Key Distribution Center (KDC) service: authentication
 - Other Active Directory services
- Best practices
 - Available: at least two in a domain
 - Secure: Server Core, Read-only domain controllers (RODCs)



Key Points

Domain controllers, also referred to as DCs, are servers that perform the AD DS role. As part of that role, they host and replicate the Active Directory database (NTDS.DIT) and SYSVOL.

DCs also run the Kerberos Key Distribution Center service, which performs authentication and other Active Directory services.

Because authentication is so critical to enterprise, you can imagine that the best-practice guidance is to have at least two available domain controllers, so that if clients are unable to access one, they have access to another.

In addition to availability, you must ensure that domain controllers are secure. In addition to physical security (e.g. placing DCs in secure datacenters), two options exist to improve the security of domain controllers:

- Server Core: You can install Windows Server 2008 with the Server Core installation option. This installs a minimal configuration of Windows Server 2008 that features a Command Prompt user interface, rather than Explorer. You will install a Server Core DC in the Lab for Module 11.

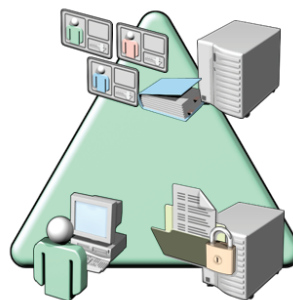
- Read-Only Domain Controllers (RODCs). RODCs give you the ability to authenticate users in less secure environments, such as branch offices, by caching credentials only for those users. Passwords for other users are not replicated to the RODC. Additionally, the RODC does not allow changes to be made to Active Directory, reducing the vulnerability of the AD DS domain to accidental or intentional damage at a less secure site. RODCs are detailed in Module 9.

Additional Reading

- Domain Controllers are discussed throughout this course, but Modules 11 and 12 are focused specifically on domain controller administration and placement. Module 9 discusses RODCs.

Domain

- Made up of one or more DCs
- All DCs replicate the Domain naming context (Domain NC)
 - The domain is the context within which Users, Groups, Computers, and so on are created
 - "Replication boundary"
- Trusted identity source: Any DC can authenticate any logon in the domain
- The domain is the *maximum* scope (boundary) for certain administrative policies
 - Password
 - Lockout



Key Points

One or more domain controllers are required to create an Active Directory domain. A domain is an administrative unit within which certain capabilities and characteristics are shared. First, all domain controllers replicate the domain's partition of the data store, which contains, among other things, the identity data for the domain's users, groups, and computers. Because all DCs maintain the same identity store, any DC can authenticate any identity in a domain.

Additionally, a domain is a scope of administrative policies such as password complexity and account lockout policies. Such policies configured in one domain affect all accounts in the domain and do not affect accounts in other domains.

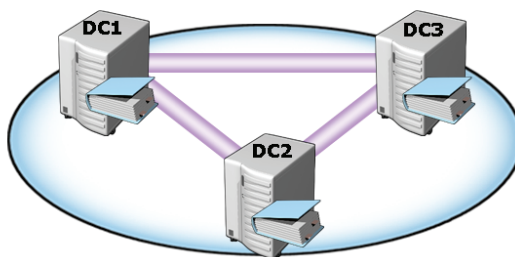
Changes can be made to objects in the Active Directory database by any domain controller, and will replicate to all other domain controllers. Therefore, in networks where replication of all data between domain controllers cannot be supported, it may be necessary to implement more than one domain in order to manage the replication of subsets of identities.

Additional Reading

- You will learn more about domains throughout this course, and Module 14 focuses on the design considerations related to how many domains you should have in your enterprise.

Replication

- **Multimaster replication**
 - Objects and attributes in the database
 - Contents of SYSVOL are replicated
- Several components work to create an efficient and robust replication topology and to replicate granular changes to AD
- The Configuration partition of the database stores information about sites, network topology, and replication



Key Points

Replication services distribute directory data across a network. This includes both the data store itself as well as data required to implement policies and configuration, including logon scripts. As you will learn in Module 12, Active Directory replication is both efficient and robust.

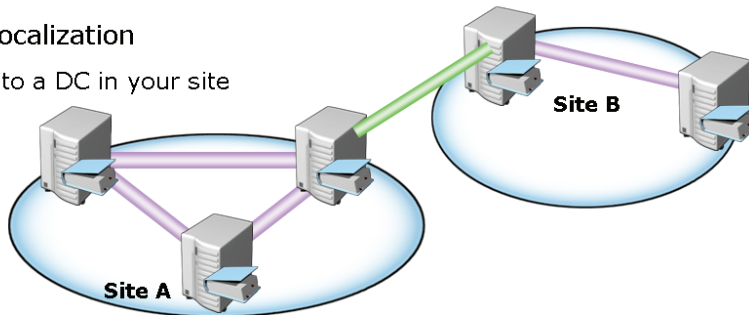
Active Directory maintains a separate partition of the data store named Configuration that maintains information about network configuration, topology, and services: the Configuration NC.

Additional Reading

- Active Directory Replication is detailed in Module 12. SYSVOL replication is discussed in Module 9.

Sites

- An Active Directory object that represents a well-connected portion of your network
 - Associated with subnet objects representing IP subnets
- Intrasite vs. intersite replication
 - Replication within a site occurs very quickly (15-45 seconds)
 - Replication between sites can be managed
- Service localization
 - Log on to a DC in your site



Key Points

When you consider the network topology of a distributed enterprise, you will certainly discuss the network's sites. Sites in Active Directory, however, have a very specific meaning because there is a specific object class called *site*.

An Active Directory site is an object that represents a portion of the enterprise within which network connectivity is good. A site creates a boundary of replication and service utilization.

Domain controllers within a site replicate changes within seconds. Changes are replicated between sites on a controlled basis with the assumption that intersite connections are slow, expensive, or unreliable compared to the connections within a site.

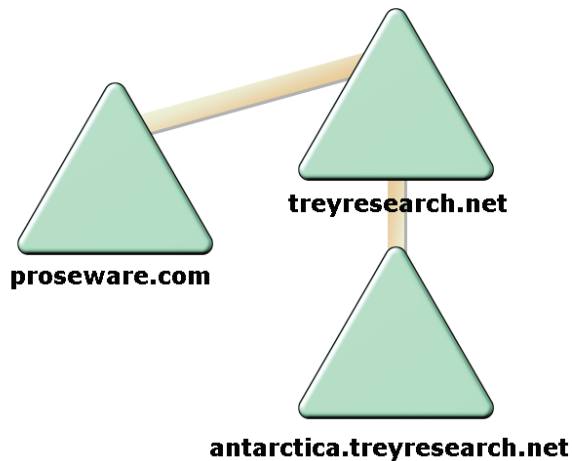
Additionally, clients will prefer to use distributed services provided by servers in their site, or the closest site. For example, when a user logs on to the domain, the Windows client first attempts to authenticate with a domain controller in its site. Only if no domain controller is available in the site will the client attempt to authenticate with a DC in another site.

Additional Reading

- Active Directory site and subnet objects are the focus of Module 12.

Tree

- One or more domains in a single instance of AD DS that share *contiguous DNS namespace*



Key Points

The domain name system (DNS) namespace of domains in a forest create trees within the forest. If a domain is a subdomain of another domain, the two domains are considered a tree. For example, if the tresearch.net forest contains two domains, tresearch.net and antarctica.tresearch.net, those domains constitute a contiguous portion of the DNS namespace, so they are a single tree. If, on the other hand, the two domains are tresearch.net and proseware.com, which are not contiguous in the DNS namespace, the forest is considered to have two trees. Trees are the direct result of the DNS names chosen for domains in the forest.

The slide illustrates an Active Directory forest for Trey Research, which maintains a small operation at a field station in Antarctica. Because the link from Antarctica to the headquarters is expensive, slow, and unreliable, Antarctica is configured as a separate domain. The DNS name of the forest is tresearch.net. The Antarctica domain is a child domain in the DNS namespace, antarctica.tresearch.net, so it is considered a child domain in the domain tree.

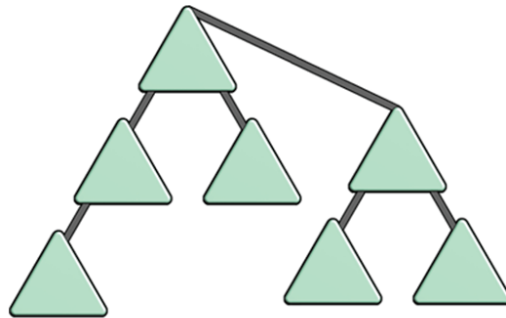
The proseware.com domain, because it does not share a contiguous DNS namespace, is another tree in the same forest.

Additional Reading

- The concepts and design of a multidomain forest are discussed in Module 14.

Forest

- A collection of one or more Active Directory domain trees
- First domain is the *forest root domain*
- Single configuration and schema replicated to *all* DCs in the forest
- A security and replication boundary



Key Points

A *forest* is a collection of one or more Active Directory domains. The first domain installed in a forest is called the *forest root domain*. A forest contains a single definition of network configuration and a single instance of the directory schema. In other words, every domain controller in a forest replicates the Configuration and Schema partitions.

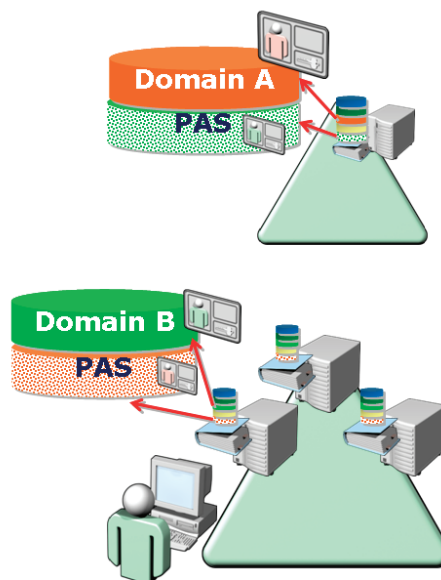
A forest is a single instance of the directory—no data is replicated by Active Directory outside the boundaries of the forest. Therefore, the forest defines both a replication and a security boundary.

Additional Reading

- The concepts and design of a multidomain forest are discussed in Module 14.

The Global Catalog

- Partial Attribute Set or Global Catalog
- Contains every object in every domain in the forest
- Contains only selected attributes
- A type of index
- Can be searched from any domain
- Very important for many applications



Key Points

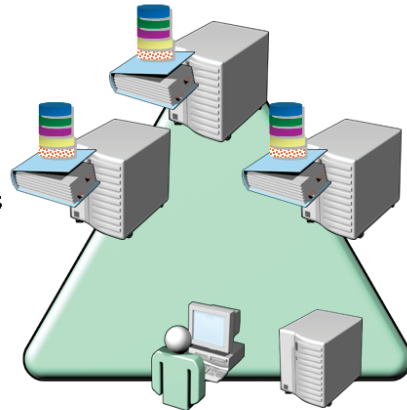
Several components and technologies enable you to query Active Directory and locate objects in the data store. A partition of the data store called the *global catalog* (also known as the *partial attribute set*) contains information about every object in the directory. It is a type of index that can be used to locate objects in the directory. This is particularly important if you are searching for objects in another domain within a forest. Because the domain controllers in your domain will not contain information about objects in other domains, you must rely on the global catalog, which has the indexed, partial attribute set for all objects in other domains.

Additional Reading

- The global catalog is explored in detail in Module 12.

Functional Level

- Domain functional levels
- Forest functional levels
- New functionality requires that *domain controllers* are running a particular version of Windows
 - Windows 2000
 - Windows Server 2003
 - Windows Server 2008
- Cannot raise functional level while DCs are running previous versions of Windows
- Cannot add DCs running previous versions of Windows after raising functional level



Key Points

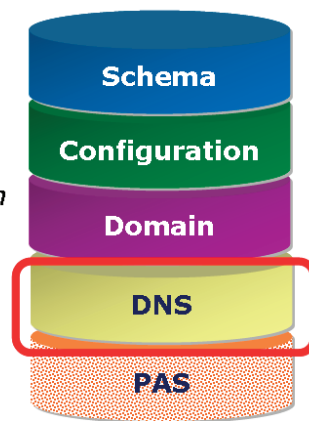
The functionality available in an Active Directory domain or forest depends on its functional level. The functional level is an AD DS setting that enables advanced domain-wide or forest-wide AD DS features. There are three domain functional levels, Windows 2000 native, Windows Server 2003 and Windows Server 2008, and two forest functional levels, Windows Server 2003 and Windows Server 2008. As you raise the functional level of a domain or forest, features provided by that version of Windows become available to AD DS. For example, when domain functional level is raised to Windows Server 2008, a new attribute becomes available that reveals the last time a user successfully logged onto a computer, the computer to which the user last logged on, and the number of failed logon attempts since the last logon. The important thing to know about functional levels is that they determine the versions of Windows permitted on domain controllers. Before you raise the domain functional level to Windows Server 2008, all domain controllers must be running Windows Server 2008.

Additional Reading

- Functional levels are detailed in Module 14.

DNS and Application Partitions

- Active Directory and DNS are tightly integrated
- One-to-one relationship between the DNS domain name and the logical domain unit of Active Directory
- Complete reliance on DNS to locate computers and services in the domain
- A domain controller acting as a DNS server can store the zone data in Active Directory itself—in an *application partition*



Key Points

Active Directory and DNS have a very close relationship. First, there is a one-to-one relationship between a DNS name and an Active Directory domain. Second, there is a complete reliance on DNS to locate computers and services within the domain. Third, it is very common to configure domain controllers to also serve as DNS servers. When you do this, you have the option to store DNS data, called a *zone*, in Active Directory itself.

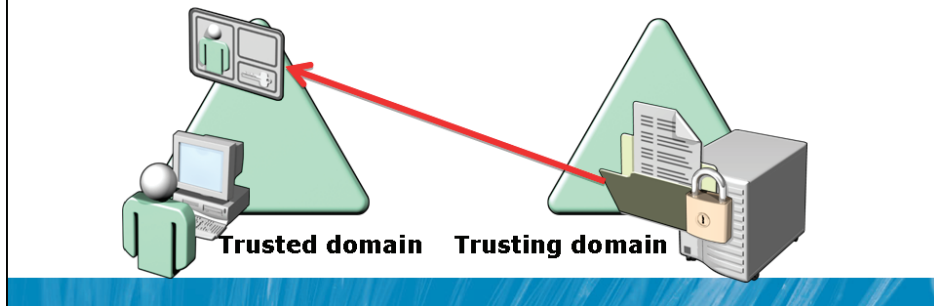
The Active Directory data store can also be used to support applications and services not directly related to AD DS. Within the database, application partitions can store data to support applications that require replicated data. The domain name system (DNS) service on a Windows Server 2008 server can store its information in a database called an *Active Directory integrated zone*, which is maintained as an application partition in AD DS and replicated using Active Directory replication services.

Additional Reading

- DNS is covered in Module 10.

Trust Relationships

- Extends concept of trusted identity store to another domain
- Trusting domain (with the resource) trusts the identity store and authentication services of the trusted domain
- A trusted user can authenticate to, and be given access to resources in, the trusting domain
- Within a forest, each domain trusts all other domains
- Trust relationships can be established with external domains



Key Points

At the beginning of this module, you considered the default, stand-alone, "workgroup," configuration of Windows Server. You then learned that, when a machine joins a domain, the Local Security Authority of the system begins to trust the identity store and authentication services provided by the domain. That allows a user account stored in the domain to be authenticated by and provide access to resources on the server.

The same concept can be extended to other domains. A domain can authenticate users from another domain and can allow those users to be assigned access to resources in the domain. This is done by establishing a domain trust relationship.

In a trust relationship, the trusting domain extends its realm of trust so that it trusts the identity store and authentication services of the trusting domain. User accounts in the trusting domain can best be authenticated, and the SIDs of user accounts in the trusted domain can be added to ACLs in the trusting domain.

Within a forest, each domain trusts every other domain. You must manually establish trust relationships between domains that are in different forests.

Additional Reading

- Trust relationships are discussed in Module 14.

Lesson 3

Install Active Directory Domain Services

- Install Windows Server 2008
- Server Manager and Role-Based Configuration of Windows Server 2008
- Prepare to Create a New Forest with Windows Server 2008
- Install and Configure a Domain Controller

This lesson discusses how to install Active Directory Domain Services and how to configure a domain controller.

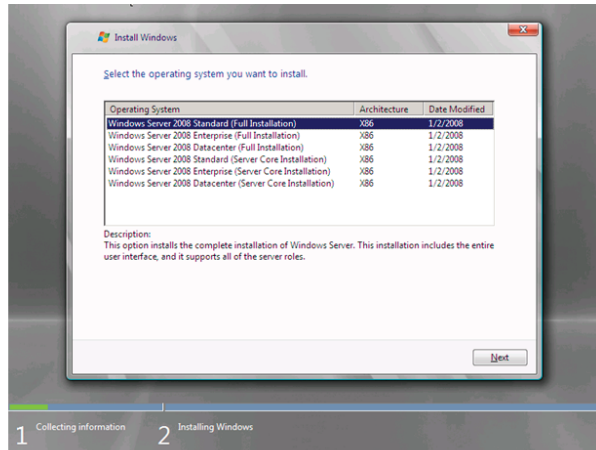
Objectives

After completing this lesson, you will be able to:

- Understand the requirements for installing a domain controller to create a new forest.
- Configure a domain controller with the AD DS role, using the Windows interface.

Install Windows Server 2008

- Boot with installation media (DVD)
- Follow prompts and select the operating system to install



Key Points

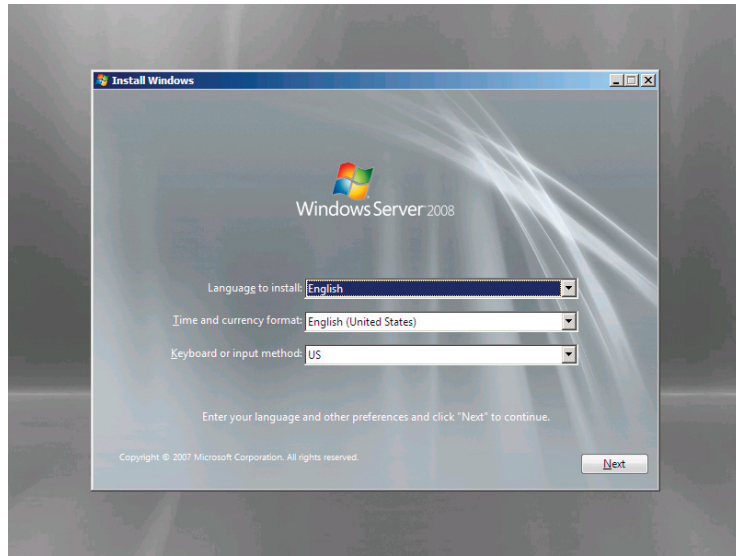
Installing Windows Server 2008 is a straightforward process:

1. Insert the Windows Server 2008 installation DVD.
2. Turn on the system.

If the system's hard disk is empty, the system should boot to the DVD. If there is information on the disk, you may be prompted to press a key to boot to the DVD.

If the system does not boot to the DVD or offer you a boot menu, go to the BIOS settings of the machine and configure the boot order to ensure the system boots to the DVD.

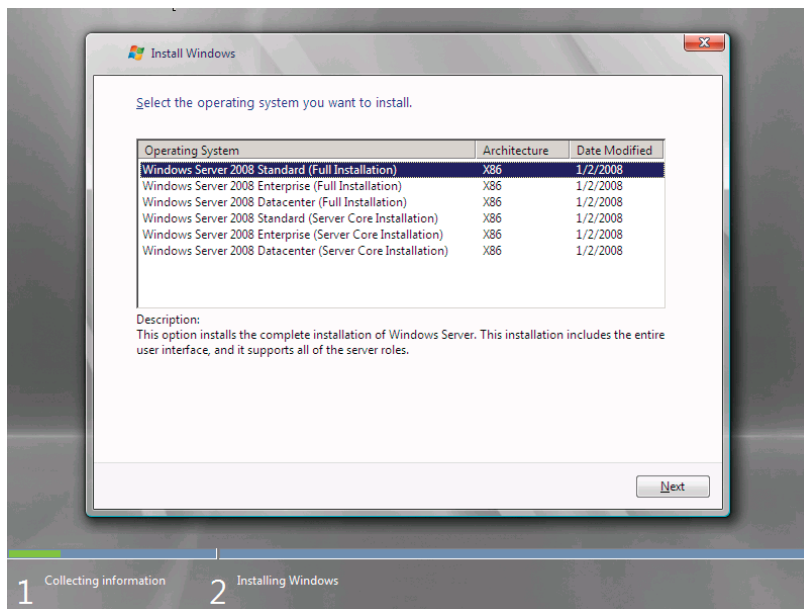
The Install Windows wizard appears, shown the following screen shot:



3. Select the language, regional setting, and keyboard layout that is correct for your system, and then click **Next**.

4. Click **Install Now**.

You are presented with a list of versions to install, as shown in the following screen shot. If you are using an x64 computer, you will be presented with x64 versions rather than x86 versions.

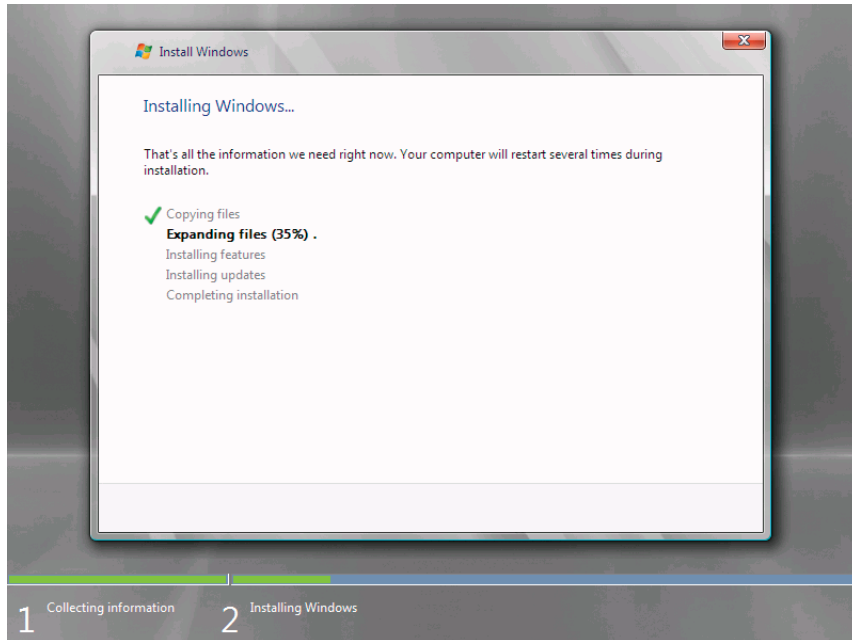


5. Select the appropriate operating system, and then click **Next**.
6. Click **I Accept The License Terms**, and then click **Next**.
7. Click **Custom (Advanced)**.
8. On the **Where Do You Want to Install Windows?** page, select the disk on which you want to install Windows Server 2008.

If you need to create, delete, extend, or format partitions, or if you need to load a custom mass storage driver in order to access the disk subsystem, click **Advanced Options**.

9. Click **Next**.

The Installing Windows dialog box appears, shown in the following screen shot. The window keeps you apprised of the progress of Windows installation.



Installation of Windows Server 2008, like Windows Vista®, is image-based. Therefore, installation is significantly faster than for previous versions of Windows even though the operating systems themselves are much larger than earlier versions. The computer will reboot one or more times during installation.

When the installation has completed, you will be informed that the user's password must be changed before logging on the first time.

10. Click **OK**.

11. Enter a password for the Administrator account in both the **New Password** and **Confirm Password** boxes, and then press ENTER.

The password must be at least seven characters long and must have at least three of four character types:

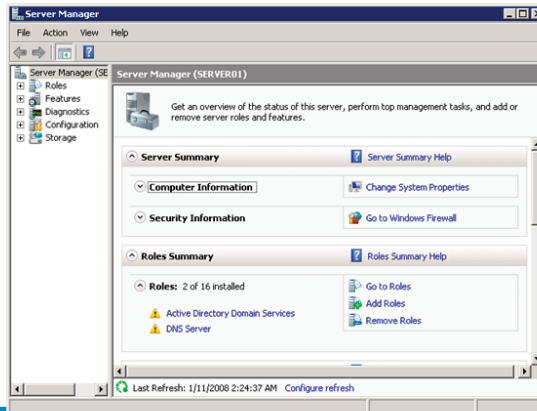
- Upper case: A–Z
- Lower case: a–z
- Numeric: 0–9
- Non-alphanumeric: symbols such as \$, #, @, and !

12. Click **OK**.

If you selected a Full Installation, the desktop for the Administrator account appears. If you installed Server Core, a command prompt appears.

Server Manager and Role-Based Configuration of Windows Server 2008

- Windows Server 2008 has minimal footprint
- Functionality is added as *roles* or *features*
- Server Manager: role and feature configuration along with the common administrative snap-ins for the server



Key Points

In order to reduce management costs as well as to reduce exposure to security vulnerabilities, Windows Server 2008 setup installs only the core operating system components. Unlike previous versions of Windows, however, the result is a minimal installation rather than an all-in-one server. Therefore, after installation of the operating system, you must add the components required for the server based on the role it will play in your enterprise. Windows Server 2008 functionality is added as roles and features. The Server Management console gives you the ability to add and remove roles. It also exposes the most common administrative snap-ins based on the server's role.

Prepare to Create a New Forest with Windows Server 2008

- Domain's DNS name (contoso.com)
- Domain's NetBIOS name (contoso)
- Whether the new forest will need to support DCs running previous versions of Windows (affects choice of functional level)
- Details about how DNS will be implemented to support AD DS
 - Default: Creating domain controller adds DNS Server role as well
- IP configuration for the DC
 - IPv4 and, optionally, IPv6
- Username and password of an account in the server's Administrators group. Account must have a password.
- Location for data store (ntds.dit) and SYSVOL
 - Default: %systemroot% (c:\windows)

Key Points

Before you install the AD DS role on a server and promote it to act as a domain controller, you should plan your Active Directory infrastructure. Among the information you will need to create a domain controller are:

- The domain's name and DNS name. A domain must have a unique DNS name, for example contoso.com, as well as a short name, for example CONTOSO, called a NetBIOS name. NetBIOS is a network protocol that has been used since the first versions of Windows NT®, and is still used by some legacy applications.
- Whether the domain will need to support domain controllers running previous versions of Windows. When you create a new Active Directory forest, you will configure the functional level. If the domain will include only Windows Server 2008 domain controllers, you can set the functional level accordingly to benefit from the enhanced features introduced by this version of Windows.

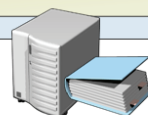
- Details for how DNS will be implemented to support Active Directory. It is a best practice to implement DNS for your Windows domain zones using Windows DNS Service, as you will learn in Module 9, however it is possible to support a Windows domain on a third-party DNS service.
- IP configuration for the domain controller. Domain controllers require static IP addresses and subnet mask values. Additionally, the domain controller must be configured with a DNS server address with which to perform name resolution. If you are creating a new forest and will run Windows DNS Service on the domain controller, you can configure the DNS address to point to the server's own IP address. After DNS is installed, the server can look to itself to resolve DNS names.
- The username and password of an account in the server's Administrators group. The account must have a password—the password cannot be blank.
- The location in which the data store (including ntds.dit) and system volume (SYSVOL) should be installed. By default, these stores are created in %systemroot%, for example c:\windows, in the NTDS and SYSVOL folders, respectively. When creating a domain controller, you can redirect these stores to other drives.

Additional Reading

- This list comprises the settings that you will be prompted to configure when creating a domain controller. There are a number of additional considerations regarding the deployment of AD DS in an enterprise setting. See the Windows Server 2008 Technical Library at <http://go.microsoft.com/fwlink/?LinkId=168483> for more information.

Install and Configure a Domain Controller

- 1 Install the Active Directory Domain Services role using the Server Manager**
- 2 Run the Active Directory Domain Services Installation Wizard**
- 3 Choose the deployment configuration**
- 4 Select the additional domain controller features**
- 5 Select the location for the database, log files, and SYSVOL folder**
- 6 Configure the Directory Services Restore Mode Administrator Password**



Key Points

To install and configure a Windows Server 2008 domain controller, you must first install the AD DS role using Server Manager. Doing so adds the files and registry components necessary for the server to later become a domain controller. But adding the role does not actually configure and enable the server as a domain controller. That's step is performed by running the Active Directory Domain Services Installation Wizard. The AD DS Installation Wizard, also known as DCPromo because the wizard can be launched using the `dcpromo.exe` command, steps you through the process of selecting the deployment configuration, adding additional domain controller features such as the DNS role, specifying the location for Active Directory files, and configuring the Directory Services Restore Mode Administrator Password, a password that is used when restoring Active Directory from a backup, as you'll learn in Module 13.

Lab: Install an AD DS DC to Create a Single Domain Forest

- Exercise 1: Perform Post-Installation Configuration Tasks
- Exercise 2: Install a New Windows Server 2008 Forest with the Windows Interface

Logon information

Virtual machine	6425B-HQDC01-D
Logon user name	Administrator
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

You have been hired to improve identity and access at Contoso, Ltd. The company currently has one server in a workgroup configuration. Employees connect to the server from their personal client computers. In anticipation of near-term growth, you have been tasked with improving the manageability and security of the company's resources. You decide to implement an AD DS domain and forest by promoting the server to a domain controller. You have just finished installing Windows Server 2008 from the installation DVD.

Exercise 1: Perform Post-Installation Configuration Tasks

In this exercise, you will prepare the server by performing post-installation configuration tasks.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Configure the display resolution.
3. Configure the time zone.
4. Change IP configuration.
5. Rename the server HQDC01.
6. Restart the server.

► Task 1: Prepare for the lab

- Start 6425B-HQDC01-D.
- Log on with username **Administrator** and password **Pa\$\$w0rd** (where the 0 is a zero).

► Task 2: Configure the display resolution

- Configure the display resolution to **1024 by 768**.

► Task 3: Configure the time zone

- Using the Initial Configuration Tasks window, change the time zone so that it is appropriate for your location.

► **Task 4: Change IP configuration**

- Using the Initial Configuration Tasks window, change the IP (IPv4) configuration to the following:
 - IP Address: **10.0.0.11**
 - Subnet Mask: **255.255.255.0**
 - Default Gateway: **10.0.0.1**
 - Preferred DNS Server: **10.0.0.11**

► **Task 5: Rename the server HQDC01**

- Using the Initial Configuration Tasks window, rename the server to **HQDC01**. Do not restart the server.

► **Task 6: Restart the server**

1. In the Initial Configuration Tasks window, note the **Add roles** and **Add features** links.

In the next exercise, you will use Server Manager to add roles and features to HQDC01. These links are another way to perform the same tasks.

By default, the Initial Configuration Tasks window will appear each time you log on to the server.

2. Select the **Do not show this window at logon** check box to prevent the window from appearing.

If you need to open the Initial Configuration Tasks window in the future, you do so by running the **Oobe.exe** command.

3. Click the **Close** button at the bottom of the window.

Server Manager appears.

Server Manager enables you to configure and administer the roles and features of a server running Windows Server 2008. You will use Server Manager in the next exercise.

At the bottom of the Server Manager window, a status message informs you, *Console cannot refresh until computer is restarted.*

4. Click the **Restart** link next to the status message.
You are prompted with the message *Do you want to restart now?*.
5. Click **Yes**.
The computer restarts.

Results: After this exercise, you will have a server named HQDC01 in the correct time zone, with display resolution of at least 1024 x 768, and with the IP configuration specified in Task 4.

Exercise 2: Install a New Windows Server 2008 Forest with the Windows Interface

Now that you have prepared the server with an appropriate name and IP configuration, you are ready to configure HQDC01 as a domain controller. In this exercise, you will add the AD DS role and create the forest and domain by promoting HQDC01 to be the first domain controller in the contoso.com forest.

The main tasks for this exercise are as follows:

1. Add the Active Directory Domain Services role to HQDC01.
2. Configure a new Windows Server 2008 forest named *contoso.com* with HQDC01 as the first domain controller.
3. Examine the default configuration of the contoso.com forest and domain.
4. Shut down the virtual machine.

► Task 1: Add the Active Directory Domain Services role to HQDC01

1. Log on to HQDC01 as **Administrator** with the password **Pa\$\$w0rd**.
2. Using Server Manager, add the role, **Active Directory Domain Services**. Accept all defaults.

► Task 2: Configure a new Windows Server 2008 forest named *contoso.com* with HQDC01 as the first domain controller

1. In Server Manager, expand the **Roles** node in the tree pane, and then select **Active Directory Domain Services**.
2. Click the **Run the Active Directory Domain Services Installation Wizard (dcpromo.exe)** link.

The Active Directory Domain Services Installation Wizard appears.

3. Click **Next**.
4. On the **Operating System Compatibility** page, review the warning about the default security settings for Windows Server 2008 domain controllers, and then click **Next**.
5. On the **Choose a Deployment Configuration** page, select **Create a new domain in a new forest**, and then click **Next**.

6. On the **Name the Forest Root Domain** page, type **contoso.com**, and then click **Next**.

The system performs a check to ensure that the DNS and NetBIOS names are not already in use on the network.

7. On the **Set Forest Functional Level** page, choose **Windows Server 2008**, and then click **Next**.

The Additional Domain Controller Options page appears.

Each of the functional levels is described in the Details box on the page. Choosing Windows Server 2008 forest functional level ensures that all domains in the forest operate at the Windows Server 2008 domain functional level, which enables several new features provided by Windows Server 2008.

In a production environment, you would choose Windows Server 2008 forest functional level when creating a new forest if you require the features provided by the Windows Server 2008 domain functional level and if you will not be adding any domain controllers running operating systems prior to Windows Server 2008.

DNS Server is selected by default. The Active Directory Domain Services Installation Wizard will create a DNS infrastructure during AD DS installation.

The first domain controller in a forest must be a global catalog server and cannot be a read-only domain controller (RODC).

8. Click **Next**.

A Static IP assignment warning appears.

Because discussion of IPv6 is beyond the scope of this training kit, you did not assign a static IPv6 address to the server in Exercise 2. You did assign a static IPv4 address in Exercise 1, and other labs in this course will use IPv4. You can therefore ignore this error in the context of the exercise.

9. Click **Yes, the computer will use a dynamically assigned IP address (not recommended)**.

A warning appears that informs you that a delegation for the DNS server cannot be created.

In the context of this exercise, you can ignore this error. Delegations of DNS domains will be discussed later in this course.

10. Click **Yes** to close the Active Directory Domain Services Installation Wizard warning message.

11. On the **Location for Database, Log Files, and SYSVOL** page, accept the default locations for the database file, the directory service log files, and the SYSVOL files, and then click **Next**.

The best practice in a production environment is to store these files on three separate volumes that do not contain applications or other files not related to AD DS. This best practice design improves performance and increases the efficiency of backup and restore.

12. On the **Directory Services Restore Mode Administrator Password** page, type **Pa\$\$w0rd** in both the **Password** and **Confirmed Password** boxes. Click **Next**.

In a production environment, you should use a very strong password for the Directory Services Restore Mode Administrator Password. Do not forget the password you assign to the Directory Services Restore Mode Administrator.

13. On the **Summary** page, review your selections.

If any settings are incorrect, click **Back** to make modifications.

14. Click **Next**.

Configuration of AD DS begins. After several minutes of configuration, the Completing the Active Directory Domain Services Installation Wizard page appears.

15. Click **Finish**.

16. Click **Restart Now**.

The computer restarts.

17. Continue with Task 3 (optional) or skip to Task 4.

► **Task 3: Examine the default configuration of the contoso.com forest and domain (OPTIONAL)**

1. Log on to HQDC01 as **Administrator** with the password **Pa\$\$w0rd**.
The Windows desktop appears and, after a moment, Server Manager opens.
2. Expand the **Roles** node in the tree pane, and expand the **Active Directory Domain Services** node.
3. Expand **Active Directory Users and Computers** and the **contoso.com** domain node.

4. Select the **Users** container in the tree.

The users and groups you see are available to any computer in the domain. For example, the domain's Administrator account can be used to log on to any computer in the domain, by default, and the Domain Users group is a member of the local Users group on each computer in the domain.

5. Select the **Builtin** container in the tree.

The groups you see are shared by and available to domain controllers, but not to member servers or workstations. For example, members of the Backup Operators group can perform backup and restore tasks on domain controllers only, and the Administrators group in the Builtin container represents the administrators of all domain controllers.

6. Select the **Computers** container in the tree.

It is empty. This is the default container for member servers and workstations.

7. Select the **Domain Controllers** organizational unit (OU) in the tree.

This is the OU into which domain controllers are placed. The computer object for HQDC01 appears in this OU.

► Task 4: Shut down the virtual machine

1. If you are not already logged on to HQDC01, log on to HQDC01 as **Administrator** with the password **Pa\$\$w0rd**.
2. Shut down HQDC01 and do not save any changes you made while doing this lab exercise.

Results: After this exercise, you will have a single-domain forest named contoso.com with a single domain controller named HQDC01.

Lab Review

After this lab you will have:

- Performed post installation tasks in naming a server HQDC01, configuring the correct time zone, with display resolution of at least 1024 x 768 and specifying its IP address information.
- Configured a single-domain forest named contoso.com with a single domain controller named HQDC01.

Lesson 4

Extend IDA with Active Directory Services

- Active Directory Lightweight Directory Services (AD LDS)
- Active Directory Certificate Services (AD CS)
- Active Directory rights Management Services (AD RMS)
- Active Directory Federation Services (AD FS)

Active Directory Domain Services is not the only component of IDA that is supported by Windows Server 2008. With the release of Windows Server 2008, Microsoft has consolidated a number of previously separate components into an integrated IDA platform. Active Directory itself now includes five different technologies, each of which play a role in extending Active Directory to support applications, identity, and information protection.

Objectives

After completing this lesson, you will be able to:

- Identify the roles of and relationships between AD DS, AD LDS, AD RMS, AD FS, and AD CS.

Active Directory Lightweight Directory Services (AD LDS)

- **Standalone version of Active Directory**
 - Used to support applications that require a directory store
 - Allow customization without impact to production Active Directory
- **Characteristics**
 - A subset of AD DS functionality, sharing the same code
 - Schema, Configuration, and Application partitions
 - Replication
 - Not dependent upon AD DS
 - Can use AD DS to authenticate Windows security principals
 - Can run multiple instances on a single server



Key Points

Active Directory Lightweight Directory Services (AD LDS) is essentially a standalone version of Active Directory, accessed by applications using Lightweight Directory Access Protocol (LDAP).

AD LDS is the replacement for Active Directory Application Mode (ADAM). The name of the previous version of the tool indicates its purpose: AD LDS is designed to provide support for directory-enabled applications. It can be used for applications that require a directory store but do not require the type of infrastructure provided by an Active Directory domain.

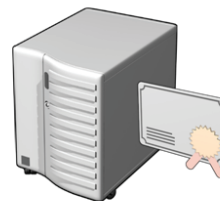
Each instance of AD LDS can have its own schema, configuration, and application partitions. This allows you to create a highly customized directory store without impacting your production IDA infrastructure, based on AD DS. While AD LDS is not dependent on AD DS, in a domain environment AD LDS is able to use AD DS authentication of Windows security principals (users, computers, and groups).

AD LDS can be configured in a stand-alone or workgroup environment, and it is even possible to run multiple instances on a single system, each with its own unique LDAP and SSL ports.

When you are adding a directory-enabled application to your environment, particularly an application that will modify the schema, you can consider using AD LDS as an alternative. Because it is a subset of AD DS functionality (AD LDS even includes the ability to replicate), most applications can work with it. AD LDS also gives you the option of extending a directory into places where you would not normally want to place your AD DS domain controllers, to support applications that are in your firewall's DMZ, for example.

Active Directory Certificate Services (AD CS)

- Extends the concept of trust
 - A certificate from a trusted certificate authority (CA) proves identity
 - Trust can be extended beyond the boundaries of your enterprise, as long as clients trust the CA of the certificates you present
- Creates a public key infrastructure (PKI)
 - Confidentiality, Integrity, Authenticity, Non-Repudiation
- *Many* uses
 - Internal-only or external
 - Secure Web sites (SSL)
 - VPN
 - Wireless authentication and encryption
 - Smart card authentication
- Integration with AD DS powerful, but not required



Key Points

Active Directory Certificate Services (AD CS) extend the concept of trust so that a user, computer, organization, or service can prove its identity outside or inside the border of your Active Directory forest.

Certificates are issued from a certificate authority (CA). When a user, computer, or service uses a certificate to prove its identity, the client in the transaction must trust the issuing CA. A list of trusted root CAs, which includes, for example, VeriSign and Thawte, is maintained by Windows, and updated as part of Windows Update.

If you think about the last time you made a purchase on an Intranet site, you will recall that it was probably performed on a site using secure sockets layer (SSL), with an HTTPS:// address. The server proves its identity to the client, your browser, representing a certificate issued by a CA that your browser trusts, such as VeriSign or Thawte.

A public key infrastructure (PKI) is based on a chain of trust. A certificate authority can create a certificate for another certificate authority. The second CA can then issue certificates to users, computers, organizations, or services that will be trusted by any client that trusts the upstream, root CA.

The certificates can be used for numerous purposes in an enterprise network, including the creation of secure channels such as the SSL example mentioned earlier and for virtual private networks (VPNs) and wireless security as well as for authentication, such as smart card logon.

AD CS gives you the technologies and tools you need to create and manage a PKI. Although AD CS can be run on a stand-alone server, it is much more common and much more powerful to run AD CS integrated with AD DS, which can act as a certificate store and can provide a framework within which to manage the lifetime of certificates: how they are obtained, renewed, and revoked.

Active Directory Rights Management Services (AD RMS)

- Ensures the integrity of information
 - Traditional model: ACL defines access. No restriction on *use*.
 - AD RMS: Ensures access is limited and defines use.
- Examples
 - Limit access to specified individuals
 - View e-mail but do not forward or print
 - View and print document but cannot change or e-mail
- Requires
 - AD RMS
 - IIS, Database (SQL Server or Windows Internal Database)
 - AD DS
 - RMS enabled applications including Microsoft Office applications, Internet Explorer



Key Points

Active Directory Rights Management Services (AD RMS) creates a framework with which you can ensure the integrity of information, both within and outside of your organization.

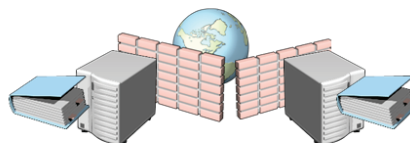
In a traditional model of information protection, access control lists (ACLs) are used to define how information can be accessed. For example, a user may be given read permission to a document. However, there is nothing to prevent that user from performing any number of actions once that document has been opened. The user can make changes to the document and save it in any location, can print the document, can forward the document via e-mail to a user who otherwise does not have read permission to the document, and so on.

AD RMS addresses these and other such scenarios by enforcing information use policies. This is all accomplished using licenses and encryption to protect information, and by having rights management-enabled applications that can consume the licenses, create usage policies, open protected content, and enforce usage policies.

AD RMS is one of the more complex Active Directory services to implement, in that it has dependencies upon AD DS as well as a number of other technologies, including IIS, a database (Microsoft SQL Server® in production, or the Windows Internal Database for testing), rights management-enabled applications, and, if information protection and usage is to be extended beyond the borders of your Active Directory forest, Active Directory Federation Services (AD FS).

Active Directory Federation Services (AD FS)

- Extends the authority of AD DS to authenticate users
- Traditional “trust”
 - Two Windows domains
 - Numerous TCP ports open in firewalls
 - “Everyone” from trusted domain is trusted
- AD FS uses Web services technologies to implement trust
 - One AD DS/LDS directory; other side can be Active Directory or other platforms
 - Port 443: transactions are secure and encrypted
 - Rules specifying which users from trusted domain are trusted
- Uses
 - Business-to-business: partnership
 - Single sign-on



Key Points

Active Directory Federation Services (AD FS) allows an organization to extend the authority of the directory service for authenticating users across multiple organizations, platforms, and network environments.

The traditional Windows domains trust relationship creates a trust in which the trusting domain allows the trusted domain to authenticate users, but the result is that all users in the trusted domain are trusted. Additionally, in order to maintain a trust, several firewall exceptions must be made that are not palatable to many organizations, and certainly not for supporting Web facing applications.

AD FS projects authenticated identities from your AD DS (or AD LDS) directory service using a Web services model that has several very important effects.

- **Cross-platform.** The Web services model allows non-Windows applications to use the identity of a user in a trusted directory.
- **Internet-facing.** Because transactions with AD FS are performed over port 443, secured and encrypted, it is much easier to support directory-enabled applications hosted in your perimeter network.

- **Rules-based.** The trusting environment has the ability to specify which identities are trusted.

AD FS is extremely useful for extending a directory's authority in business to business, partnership scenarios as well as for supporting single sign-on Web applications.

Module 2

Secure and Efficient Administration of Active Directory

Contents:

Lesson 1: Work with Active Directory Snap-ins	2-4
Lesson 2: Custom Consoles and Least Privilege	2-14
Lab A: Create and Run a Custom Administrative Console	2-25
Lesson 3: Find Objects in Active Directory	2-36
Lab B: Find Objects in Active Directory	2-53
Lesson 4: Use DS Commands to Administer Active Directory	2-62
Lab C: Use DS Commands to Administer Active Directory	2-81

Module Overview

- Work with Active Directory Snap-Ins
- Custom Consoles and Least Privilege
- Find Objects in Active Directory
- Use DS Commands to Administer Active Directory

Most administrators first experience Active Directory® by opening Active Directory Users and Computers and creating user, computer, or group objects within the organizational units (OUs) of a domain. Unfortunately, many administrators never take the time to elevate their skillsets with the Active Directory administrative tools. Whether you are a new administrator or a seasoned veteran, you need to work securely and efficiently. Therefore, this module will also share the secrets of effective administration that are often learned only after months or years of experience.

Objectives

After completing this module, you will be able to:

- Install, locate, and describe the snap-ins used to administer AD DS.
- Perform basic administrative tasks with the Active Directory Users and Computers snap-in.
- Create a custom MMC console for administration.

- Perform administrative tasks while logged on as a user.
- Control the view of objects in the Active Directory Users and Computers snap-in.
- Locate objects in Active Directory.
- Work with saved queries.
- Identify the distinguished name (DN), relative distinguished name (RDN), and common name (CN) of an Active Directory object.
- Use the DS commands to administer Active Directory from the command line.

Lesson 1

Work with Active Directory Snap-ins

- The MMC Console
- Active Directory Administration Snap-ins
- Find Active Directory Snap-ins
- Demonstration: Basic Administration with Active Directory Users and Computers

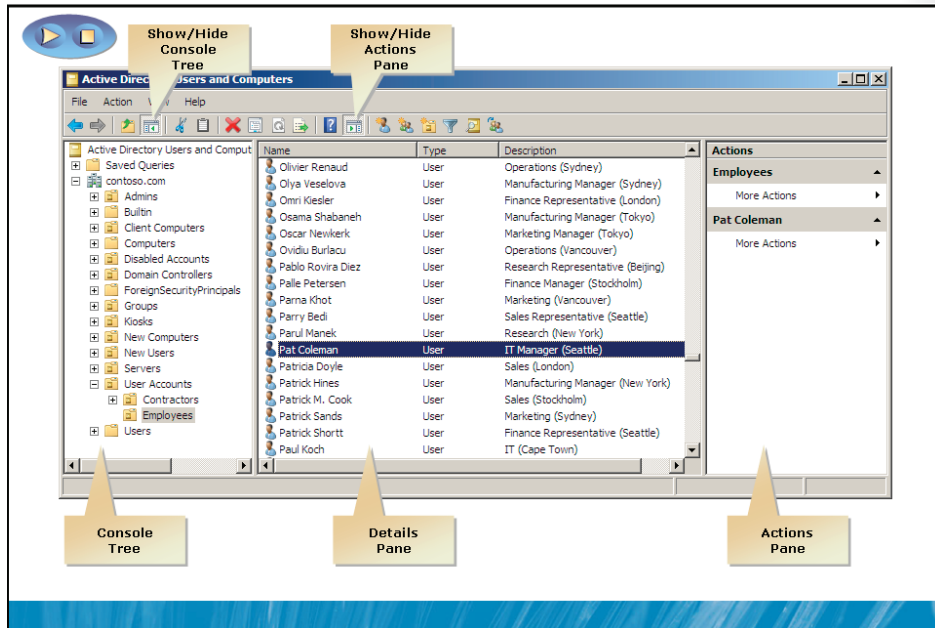
Active Directory's administrative tools, or *snap-ins*, expose the functionality you require to support the directory service. In this lesson, you will identify and locate the most important Active Directory snap-ins.

Objectives

After completing this lesson, you will be able to:

- Identify the snap-ins within Server Manager and the native consoles used to administer AD DS.
- Install the Remote Server Administration Tools (RSAT).
- Perform basic administrative tasks with the Active Directory Users and Computers snap-in.

The MMC Console



Key Points

Windows® administrative tools share a common framework called the Microsoft® Management Console (MMC). The MMC displays administrative tools, called *snap-ins*, in a customizable window with a left pane that displays the console tree (similar to the Windows® Explorer tree) and a center pane that displays details. An Actions pane on the right exposes commands, called *actions* by the MMC.

The slide above shows the major components of the MMC:

- **The console tree.** The left pane that displays the console tree; also called the scope pane
- **The Show/Hide Console Tree button.** Turns the console tree pane on and off
- **Snap-ins.** Tools that provide administrative functionality
- **The details pane.** Displays the details of the scope selected in the console tree
- **The Actions pane.** Displays commands that can be performed on the scope selected in the console tree, or the item(s) selected in the details pane

- **The Action menu.** Also displays commands that can be performed on the selected scope or items
- **The context menu (not shown).** Appears when you right-click an item in the scope or details pane; a third location from which actions can be initiated
- **The Show/Hide Action Pane button.** Turns the actions pane on and off

Question: What administrative consoles have you used that have one snap-in?

Question: What administrative consoles have you used that feature more than one snap-in?

Additional Reading

- Microsoft Management Console 3.0:
<http://go.microsoft.com/fwlink/?LinkId=168714>

Active Directory Administration Snap-ins

- **Active Directory Users and Computers**
 - Manage most common day-to-day objects, including users, groups, computers, printers, and shared folders
- **Active Directory Sites and Services**
 - Manage replication, network topology, and related services
- **Active Directory Domains and Trusts**
 - Configure and maintain trust relationships and the domain and forest functional level
- **Active Directory Schema**
 - Administer the Schema

Key Points

Most Active Directory administration is performed with the following snap-ins and consoles:

- **Active Directory Users and Computers.** This snap-in manages most common day-to-day resources, including users, groups, computers, printers, and shared folders. This is likely to be the most heavily used snap-in for an Active Directory administrator.
- **Active Directory Sites and Services.** This manages replication, network topology, and related services.
- **Active Directory Domains and Trusts.** This configures and maintains trust relationships and the domain and forest functional level.
- **Active Directory Schema.** This schema examines and modifies the definition of Active Directory attributes and object classes. It is the "blueprint" for Active Directory. It is rarely viewed and even more rarely changed. Therefore, the Active Directory Schema snap-in is not installed by default.

Additional Reading

- Active Directory Domain Services:
<http://go.microsoft.com/fwlink/?LinkId=168715>
- Managing Active Directory from MMC:
<http://go.microsoft.com/fwlink/?LinkId=168716>
- Install the Active Directory Schema snap-in:
<http://go.microsoft.com/fwlink/?LinkId=168717>

Find Active Directory Snap-ins

- Active Directory snap-ins are installed on a domain controller
 - Server Manager: Users and Computers, Sites and Services
 - Administrative Tools folder
- Install the RSAT on a member client or server
 - Windows Server® 2008
 - Server Manager → Features → Add Feature → Remote Server Administration Tools
 - Windows Vista® SP1, Windows 7
 - Download RSAT from www.microsoft.com/downloads
 - Double-click the file, then follow the instructions in the Setup Wizard.
 - Control Panel → Programs And Features → Turn Windows Features On Or Off → Remote Server Administration Tools

Key Points

Active Directory snap-ins and consoles are installed when you add the AD DS role to a server. Two commonly used Active Directory administrative tools are added to Server Manager when you install the AD DS role: the Active Directory Users and Computers snap-in and the Active Directory Sites and Services snap-in.

To administer Active Directory from a system that is not a domain controller, you must install RSAT. RSAT is a feature that can be installed from the Features node of Server Manager on Windows Server 2008.

RSAT can also be installed on Windows clients including Windows Vista Service Pack 1 (or later) and Windows 7. Simply download the RSAT installation files from www.microsoft.com/downloads. The Setup Wizard will step you through installation. After you have installed the RSAT, you must also turn on the tool or tools you wish to have visible. Use the Turn Windows Features On Or Off command in the Programs And Features application in Control Panel to do this.

Once installed and turned on, all Active Directory administrative consoles can be found in the Administrative Tools folder, which itself is found in Control Panel. In the classic view of Control Panel, you will see the Administrative Tools folder displayed. In the Control Panel Home view, administrative tools are found in System And Maintenance.

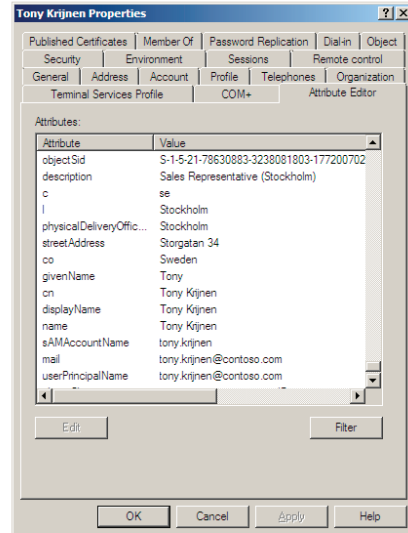
Additional Reading

- Remote Server Administration Tools Pack:
<http://go.microsoft.com/fwlink/?LinkId=168718>

Demonstration: Basic Administration with Active Directory Users and Computers

In this demonstration, you will learn:

- How to view objects in Active Directory Users and Computers
- How to refresh the view
- How to create objects
- How to configure object attributes
- How to view all object attributes



Key Points

Viewing Objects

The Active Directory Users and Computers snap-in displays the objects in the container (domain, organizational unit, or container) selected in the console tree.

Refreshing the View

The view is not refreshed automatically. If you want to see the latest changes to the view of objects, select the container in the console tree and then either click the Refresh button on the snap-in toolbar or press F5.

You must select the container in the console tree before clicking Refresh (or pressing F5)—clicking in an empty area of the details pane is not sufficient. This is a quirk of the Active Directory Users and Computers snap-in.

Creating Objects

To create an object in Active Directory Users and Computers, right-click either the domain, a container (such as Users or Computers), or an organizational unit. Then point to New and click the type of object you want to create.

When you create an object, you are prompted to configure a few of the most basic properties of the object, including the properties that are required for that type of object.

Configuring Object Attributes

After an object has been created, you can access its properties. Right-click the object and then click Properties.

The Properties dialog that appears displays many of the most common properties of the object. Properties are grouped on tabs, to make it easier to locate a specific property.

You can configure as many properties as you want, on as many tabs as you want, then click Apply or OK once to save all of the changes. The difference between Apply and OK is that the OK button closes the Properties dialog box, whereas Apply saves the changes and keeps the dialog box open so that you can make additional changes.

Viewing All Object Attributes

A user object has even more properties than are visible in its Properties dialog box. Some of the so-called hidden properties can be quite useful to your enterprise. To view these "hidden" user attributes, you must turn on the Attribute Editor, a new feature in Windows Server 2008.

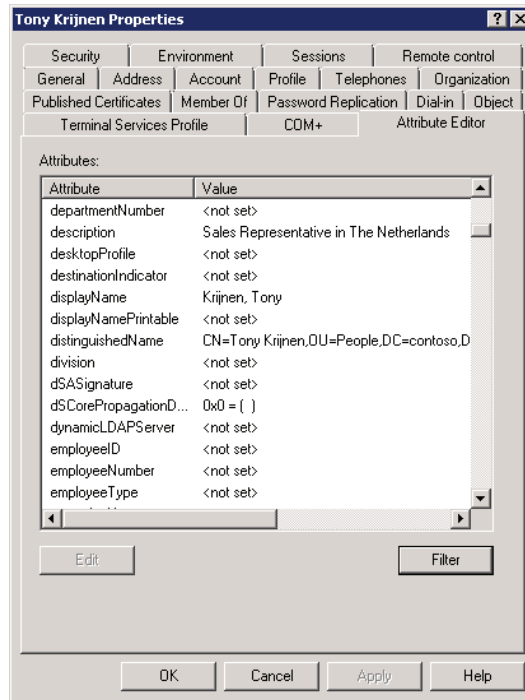
To turn on the Attribute Editor in the Active Directory Users and Computers snap-in:

- Click the **View** menu and then select the **Advanced Features** option.

To open the Attribute Editor for a specific Active Directory object:

1. Right-click the object and then click **Properties**.
2. Click the **Attribute Editor** tab.

The Attribute Editor tab of the Properties dialog box appears:



As you can see in the screen shot above, some attributes of a user object could be quite useful, including division, employeeID, employeeNumber, and employeeType. Although the attributes are not shown on the standard tabs of a user object, they are now available through the Attribute Editor.

To change the value of an attribute, double-click the value.

The attributes can also be accessed programmatically with Windows PowerShell™, Windows Visual Basic® Scripting Edition, or the Microsoft .NET Framework.

Lesson 2

Custom Consoles and Least Privilege

- **Demonstration:** Create a Custom MMC Console for Administering Active Directory
- **Secure Administration with Least Privilege, Run As Administrator, and User Account Control**
- **Demonstration:** Secure Administration with User Account Control and Run As Administrator
- **Demonstration:** "Super Consoles"

In this lesson, you will go beyond the Administrative Tools folder to work more securely and efficiently. You will learn how to build customized administrative consoles and how to work in a least privilege environment, in which you are logged on as a nonadministrative user but perform administrative tasks as an administrator.

Objectives

After completing this lesson, you will be able to:

- Create a custom MMC console for administration.
- Perform administrative tasks while logged on as a user.

Demonstration: Create a Custom MMC Console for Administering Active Directory

In this demonstration, you will learn:

- How to create a custom MMC console with multiple snap-ins
- How to register the Active Directory Schema snap-in
- Where to save a custom console

Key Points

It's easier to administer Windows when the tools you need are in one place and can be customized to meet your needs. This is achieved by creating a customized MMC administrative console that contains the snap-ins you need to perform your administrative tasks. When you create a customized MMC console, you can:

- Add multiple snap-ins so that you do not have to switch between consoles to perform your job tasks, and so that you only have to launch one console in order to perform any of your administrative tasks.
- Save the console so it can be used regularly.
- Distribute the console to other administrators.
- Save the console, and other consoles, to a shared location for unified, customized administration.

To create a customized MMC console:

1. Click **Start**. Then, in the **Start Search** box, type **mmc.exe** and press ENTER.
2. Click the **File** menu, then click **Add/Remove Snap-ins**.

The Add/Remove Snap-ins dialog box allows you to add, remove, reorder, and manage the console's snap-ins.

After you have installed the RSAT, all four Active Directory management snap-ins are installed; however the Active Directory Schema snap-in will not appear in the Add/Remove Snap-ins dialog box until after you have registered the snap-in.

To register Active Directory Schema:

1. Open a command prompt by clicking **Start**, typing **cmd.exe**, and pressing ENTER.
2. Type **regsvr32.exe schmmgmt.dll** and press ENTER.

Question: Have you built a custom MMC console?

Question: What snap-ins have you found useful?

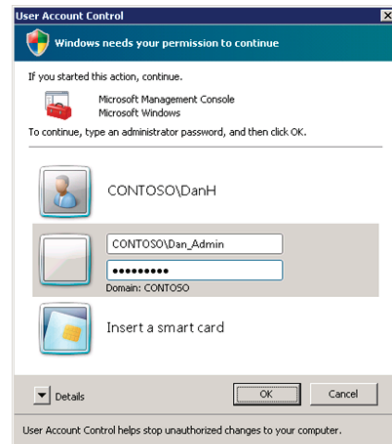
Question: Why did you build your own console?

Additional Reading

- Add, Remove, and Organize Snap-ins and Extensions in MMC 3.0:
<http://go.microsoft.com/fwlink/?LinkId=168724>

Secure Administration with Least Privilege, Run As Administrator, and User Account Control

- Maintain at least two accounts
 - A standard user account
 - An account with administrative privileges
- Log on to your computer as a standard user
 - Do not log on to your computer with administrative credentials
- Launch administrative consoles with Run As Administrator
 1. Right-click the console and click **Run As Administrator**
 2. Click **Use another account**
 3. Enter the username and password for your administrative account



Key Points

Many administrators log on to their computer using their administrative accounts. This practice is dangerous because an administrative account has more privileges and access to more of the network than a standard user account does. Therefore, malware that is launched with administrative credentials can cause significant damage.

To avoid this problem, do not log on as an administrator. Instead, log on as a standard user and use the Run As Administrator feature to launch administrative tools in the security context of an administrative account.

1. Right-click the shortcut for an executable, Control Panel applet, or MMC console that you want to launch, then click **Run as administrator**. If you do not see the command, try holding down the SHIFT key and right-clicking.

The User Account Control (UAC) dialog box appears, prompting for administrative credentials.

2. Click **Use another account**.

3. Enter the username and password of your administrative account.
4. Click **OK**.



Tip: If you will be running an application regularly as an administrator, you should create a new shortcut that preconfigures Run As Administrator. Create a shortcut and open the Properties dialog box for the shortcut. Click the Advanced button and select Run As Administrator. When you launch the shortcut, the UAC dialog box will appear.

Additional Reading

- Using Run as: <http://go.microsoft.com/fwlink/?LinkId=168725>

Demonstration: Secure Administration with User Account Control and Run As Administrator

In this demonstration, you will learn:

- How to run a custom console as an administrator
- Why it is important to save a custom console to a shared location

Key Points

When you launch a process as an administrator, the administrative account may not have access to the same locations that your user account does. Therefore, it is recommended that you save custom consoles in a location that is accessible to both your user and your administrative accounts.

To run as an administrator:

1. Right-click the shortcut for an executable, Control Panel applet, or MMC console that you want to launch, then click **Run as administrator**. If you do not see the command, try holding down the SHIFT key and right-clicking.

The User Account Control dialog box appears, prompting for administrative credentials.

2. Click **Use another account**.
3. Enter the username and password of your administrative account.
4. Click **OK**.

Additional Reading

- Using Run as: <http://go.microsoft.com/fwlink/?LinkId=168725>

Demonstration: "Super Consoles"

In this demonstration, you will learn:

- How to add a view of a file share to a custom console
 - View uses the (elevated) credentials used to launch the console
- How to create an administrative "launch pad"
 - Open external tools with the (elevated) credentials of the console

Key Points

You can extend your custom administrative MMC console to perform administrative tasks using elevated credentials that are difficult or impossible to achieve otherwise.

Scenario 1: You need to support a shared folder—to assign permissions, etc. Your administrative (secondary) account has broad administrative permissions to the shared folder. Your standard (interactive logon) account does not.

You can map a network drive using alternate credentials, but Windows prevents you from doing so if you are connected to the same server using your standard credentials. Windows Explorer does not support multiple connections to the same server using different credentials. Within the MMC console, however, the ActiveX control exposed by the Link To Web Address snap-in will connect using the credentials of the console itself.

To create a view of a shared folder:

1. Click the **File** menu, and then click **Add/Remove Snap-in**.
2. In the **Available Snap-ins** list, click **Link to Web Address**, and then click the **Add** button.

The Link to Web Address Wizard appears.

3. In **Path or URL**, type the universal naming convention (UNC) path to the shared folder, e.g. **\\ServerName\ShareName**, and then click **Next**.
4. Type a friendly name for the snap-in. This is the name that will appear in the console tree. Then click **Finish**.
5. Click **OK**.

In order to use the snap-in, the server that you are targeting must be in the Local Intranet or Trusted Sites security zone for Internet Explorer. This must be configured for the administrative credentials, because it is those credentials that are used by the mmc.exe process and by the snap-in.

1. Log on to the computer with your administrative credentials.
2. Click the **Start** button, and then click **Control Panel**.
3. Double-click **Internet Options**.
4. Click the **Security** tab.
5. Click **Local intranet** or **Trusted Sites**.
6. Click the **Sites** button.
7. Type **\\ServerName**, then click the **Add** button, then click **OK**.

There are many commands and applications that an administrator needs to run that are not MMC snap-ins. It can be tedious to launch each command or application with elevated credentials. To reduce the burden of least privilege administration, you can add these commands and applications to the MMC console. Because the MMC console is running with administrative credentials, any shell command executed from the console will automatically inherit the administrative credentials.

To create an "Administrators Launch Pad" from which you can open other tools:

1. Click the **File** menu, and then click **Add/Remove Snap-in**.
2. In the **Available Snap-ins** list, click **Folder**, then click the **Add** button, and then click **OK**.
3. In the console tree, right-click the **Folder** node you just added, and click **Rename**.
4. Type a name, e.g. **Administrators Launch Pad**, and then press ENTER.
5. Right-click the folder and click **New Taskpad View**.
The New Taskpad View Wizard appears.
6. Click **Next**.
7. On the **Taskpad Style** page, click **No List** and then click **Next**.
8. On the **Taskpad Reuse** page, click **Selected tree item** and then click **Next**.
9. On the **Name And Description** page, accept the default name and click **Next**.
10. Clear the **Add new tasks to this taskpad after the wizard closes** check box, and then click **Finish**.

To add applications and commands to the administrative launch pad:

1. Right-click the administrative launch pad and then click **Edit Taskpad View**.
2. Click the **Tasks** tab.
3. Click the **New** button.

The New Task Wizard appears.

4. Click **Next**.
5. On the **Command Type** page, click **Shell command** and then click **Next**.
6. On the **Command Line** page, enter the requested data, then click **Next**.

For example, to launch the Command Prompt, type **cmd.exe** for the Command.

If the command is not in the system path, e.g. the System32 folder, you must enter the full path to the command.

7. Type a Task name, and then click **Next**.

8. Select a **Task Icon**, and then click **Next**.

You can choose a custom icon. The following sources can provide useful icons: the command executable itself, %systemroot%\system32\shell32.dll, and %systemroot%\System32\Imageres.dll.

For example, you can use %systemroot%\system32\cmd.exe as a source for the icon for the Command Prompt.

9. Click **Next**.
10. Click **Finish** and then click **OK**.

Lab A: Create and Run a Custom Administrative Console

- Exercise 1: Perform Basic Administrative Tasks Using the Active Directory Users and Computers Snap-in
- Exercise 2: Create a Custom Active Directory Administrative Console
- Exercise 3: Perform Administrative Tasks with Least Privilege, Run As Administrator, and User Account Control
- Exercise 4 (Advanced Optional): Advanced MMC Customization and Remote Administration

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 20 minutes

Scenario

In this exercise, you are Pat Coleman, an Active Directory administrator at Contoso, Ltd. You are responsible for a variety of Active Directory support tasks, and you have found yourself constantly opening multiple consoles from the Administrative Tools folder in Control Panel. You have decided to build a single console that contains all of the snap-ins you require to do your work. Additionally, Contoso's IT security policy is changing, and you will no longer be permitted to log on to a system with credentials that have administrative privileges, unless there is an emergency. Instead, you are required to log on with nonprivileged credentials.

Exercise 1: Perform Basic Administrative Tasks Using the Active Directory Users and Computers Snap-in

In this exercise, you will perform basic administrative tasks in the Active Directory Users and Computers snap-in.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. View objects.
3. Refresh the view.
4. Create objects.
5. Configure object attributes.
6. View all object attributes.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
Pat.Coleman_Admin is a member of Domain Admins.
Server Manager opens automatically.
3. Close Server Manager.
4. Open **D:\Labfiles\Lab02a**.
5. Right-click **Lab02a_Setup.bat**, and then click **Run as administrator**.
A User Account Control dialog box appears.
6. Click **Continue**.
7. The lab setup script runs. When it is complete, press any key to continue.
8. Close the Windows Explorer window, **Lab02a**.

► **Task 2: View objects**

1. Open **Active Directory Users and Computers** from the **Administrative Tools** folder.
2. Look at the objects in the **Employees** organizational unit (OU) inside the **User Accounts** OU.

► **Task 3: Refresh the view**

- Refresh the view of the **Employees** OU.

► **Task 4: Create objects**

- Create a new OU in the root of the domain called **6425B**.

► **Task 5: Configure object attributes**

1. Open the properties of the **Pat Coleman** user object in the **Employees** OU.
2. Change the **Office** attribute on the **General** tab to **Redmond**.

► **Task 6: View all object attributes**

1. Confirm that the **Attribute Editor** tab is not visible in the **Properties** dialog box of **Pat Coleman**, and that there is no input control for the **division** property on any of the tabs.
2. Turn on the view of **Advanced Features** for the Active Directory Users and Computers snap-in.
3. View the **Attribute Editor** for **Pat Coleman**.
4. Change Pat Coleman's **division** attribute to **6425B**.
5. Close Active Directory Users and Computers.

Results: After this exercise, you will have experienced the fundamentals of administration using the Active Directory Users and Computers snap-in.

Exercise 2: Create a Custom Active Directory Administrative Console

In this exercise, you will create a single, custom administrative console that contains all of the snap-ins you need to do your work.

The main tasks for this exercise are as follows:

1. Create a custom MMC console with the Active Directory Users and Computers snap-in.
2. Add other Active Directory snap-ins to the console.
3. Add the Active Directory Schema snap-in to a custom MMC console.
4. Manage snap-ins in a custom MMC console (optional).

► Task 1: Create a custom MMC console with the Active Directory Users and Computers snap-in

1. Launch an empty MMC console and maximize it.
2. Add the **Active Directory Users and Computers** snap-in.
3. Save the console. Create a new folder called **C:\AdminTools** and save the console in that folder as **MyConsole.msc**.

► Task 2: Add other Active Directory snap-ins to the console

1. Add the **Active Directory Sites and Services** and **Active Directory Domains and Trusts** snap-ins list to your console.
2. Rename the console root **Active Directory Administrative Tools**.
3. Save the console.

► **Task 3: Add the Active Directory Schema snap-in to a custom MMC console**

1. Confirm that Active Directory Schema is not listed as an available snap-in in the **Add or Remove Snap-ins** dialog box.

The Active Directory Schema snap-in is installed with the Active Directory Domain Services role, and with the RSAT, but it is not registered, so it does not appear.

2. In the **Start** menu, right-click **Command Prompt**, and then click **Run as administrator**.

3. In the command prompt, type the command **regsvr32.exe schmmgmt.dll**.

This command registers the dynamic link library (DLL) for the Active Directory Schema snap-in. This is necessary to do one time on a system before you can add the snap-in to a console.

4. Close the Command Prompt window.
5. Add the **Active Directory Schema** snap-in to the console.
6. Save the console.

► **Task 4: (Optional): Manage snap-ins in a custom MMC console**

- Open the **Add or Remove Snap-ins** dialog box and use the **Move Up**, **Move Down**, and **Remove** buttons to rearrange your console. For future Labs, you will need the console in the condition it was in at the end of Task 3, so *do not* save your changed console. Instead, close the console without saving changes.

Results: After this exercise, you will have a custom MMC console with the Active Directory Users and Computers, Active Directory Sites and Services, Active Directory Domains and Trusts, and Active Directory Schema snap-ins.

Exercise 3: Perform Administrative Tasks with Least Privilege, Run As Administrator, and User Account Control

In this exercise, you will perform administrative tasks while logged on with standard user credentials.

The main tasks for this exercise are as follows:

1. Log on with credentials that do not have administrative privileges.
2. Run Server Manager as an administrator.
3. Examine the credentials used by running processes.
4. Run the command prompt as an administrator.
5. Run Administrative Tools as an administrator.
6. Run a custom administrative console as an administrator.

► **Task 1: Log on with credentials that do not have administrative privileges**

1. Log off of HQDC01.
2. Log on to HQDC01 as **Pat.Coleman** with the password, **Pa\$\$w0rd**.
Pat.Coleman is a member of Domain Users and has no administrative privileges.

► **Task 2: Run Server Manager as an administrator**

1. Click the **Server Manager** icon in the **Quick Launch**, next to the **Start** button.

A User Account Control dialog box appears.

Because your user account is not a member of Administrators, the dialog box requires you to enter administrative credentials: a username and a password.

If you do not see the User Name and Password boxes, make sure that you are logged on as Pat.Coleman and *not* as Pat.Coleman_Admin.

2. In the **User name** box, type **Pat.Coleman_Admin**.
3. In the **Password** box, type **Pa\$\$w0rd** and press ENTER.

Server Manager opens.

► **Task 3: Examine the credentials used by running processes**

1. Right-click the taskbar and click **Task Manager**.
2. Click the **Processes** tab.
3. Click **Show processes from all users** and, in the **User Account Control** dialog box, authenticate as **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**

Task Manager can run without administrative credentials, but it will show only those processes running under the current user account. Therefore, the User Account Control dialog box includes an option to authenticate using the same credentials with which you are logged on: Pat.Coleman.

4. Click the **Processes** tab and sort by **User Name**.
5. Locate the processes being run as **Pat.Coleman** and **Pat.Coleman_Admin**.

Question: Which processes are running as Pat.Coleman_Admin? What applications do the processes represent?

► **Task 4: Run the command prompt as an administrator**

1. Click **Start**, then right-click **Command Prompt**, and then click **Run as administrator**.
2. In the **User Account Control** dialog box, authenticate as **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

The Administrator: Command Prompt window appears.

3. Close the Command Prompt window.
4. Click **Start**, and in the **Start Search** box, type **cmd.exe**, and then press CTRL+SHIFT+ENTER.

In the Start Search box, the keyboard shortcut CTRL+SHIFT+ENTER runs the specified command as an administrator.

5. In the **User Account Control** dialog box, authenticate as **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

The Administrator: Command Prompt window appears.

► **Task 5: Run administrative tools as an administrator**

1. Click the **Show Desktop** icon in the **Quick Launch**, next to the **Start** button.
2. Click **Start**, then point to **Administrative Tools**, then right-click **Active Directory Users and Computers**, and then click **Run as administrator**.
3. In the **User Account Control** dialog box, authenticate as **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

► **Task 6: Run a custom administrative console as an administrator**

You are beginning to see that it can become tedious to run as an administrator each and every administrative tool that you require. One advantage of a custom administrative console is that you can launch the console, containing multiple snap-ins, with a single Run As Administrator command.

1. Close all open windows on your desktop.
2. Run **C:\AdminTools\MyConsole** with administrative credentials. In the **User Account Control** dialog box, authenticate as **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
3. Log off of HQDC01. Do not shut down or reset the virtual machine.

Results: After this exercise, you will have learned that by having a single, custom administrative console, you make it easier for yourself to work securely. You can log on to your computer with user (nonadministrative) credentials and run that single console as an administrator.

Exercise 4 (Advanced Optional): Advanced MMC Customization and Remote Administration

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

The main tasks for this exercise are as follows:

1. Start SERVER01-A.
2. Start DESKTOP101-A.
3. Use the procedures described in this lesson to further customize your MMC console (C:\AdminTools\MyConsole.msc). Add a snap-in that provides administrative access to the Data share on SERVER01 (\\SERVER01\Data).
4. Using the procedures described in this lesson, create an Administrators Launch Pad with a task that opens the Command Prompt.
5. Add a task to the Administrators Launch Pad that allows you to shut down a computer remotely. There are no procedures listed for this task: You are on your own! Tip: Shutdown.exe.
6. Copy your console to D:\AdminTools. The D:\AdminTools folder is shared as \\HQDC01\AdminTools.
7. Log on to DESKTOP101 as Pat.Coleman with the password Pa\$\$w0rd, and create a shortcut to \\HQDC01\AdminTools\MyConsole.msc.
8. Configure the properties of the shortcut so that it always prompts for administrative credentials. There are no procedures listed for this task: You are on your own!
9. Run your custom console as an administrator.
10. Log off of DESKTOP101 and HQDC01. Do not shut down or reset the virtual machines.



Note: Do not shut down the virtual machines once you are finished with this lab as the settings you have configured here will be used in the Lab B.

Lab Review Questions

Question: Which snap-in are you most likely to use on a day-to-day basis to administer Active Directory?

Question: When you build a custom MMC console for administration in your enterprise, what snap-ins will you add?

Lesson 3

Find Objects in Active Directory

- Find Objects in Active Directory
- Demonstration: Use the Select users, Contacts, Computers, or Groups Dialog Box
- Options for Locating Objects in Active Directory Users and Computers
- Demonstration: Control the View of Objects in Active Directory Users and Computers
- Demonstration: Use the Find Command
- Determine Where an Object Is Located
- Demonstration: Use Saved Queries

As your Active Directory becomes populated with user, group, computer, and other objects, it may become difficult to find a specific object or objects that you wish to modify. In this lesson, you will learn several ways to locate objects in Active Directory.

Objectives

After completing this lesson, you will be able to:

- Control the view of objects in the Active Directory Users and Computers snap-in.
- Locate objects in Active Directory.
- Work with saved queries.

Find Objects in Active Directory

- When you assign permissions to a folder or file
 - Select the group or user to which permissions are assigned
- When you add members to a group
 - Select the user or group that will be added as a member
- When you configure a linked attribute such as Managed By
 - Select the user or group that will be displayed on the Managed By tab
- When you need to administer a user, group, or computer
 - Perform a search to locate the object in Active Directory, instead of browsing for the object

You have learned how to create objects in Active Directory. But what good is information in a directory service if you can't get it out of the directory as well? There are many occasions on which you will need to locate objects in Active Directory:

- **Granting permissions.** When you configure permissions for a file or folder, you must select the group (or user) to which permissions should be assigned.
- **Adding members to groups.** A group's membership can consist of users, computers, groups, or any combination of the three. When you add an object as a member of a group, you must select the object.
- **Creating links.** Linked properties are properties of one object that refer to another object. Group membership is, in fact, a linked property. There are other linked properties, such as the Managed By attribute, that are also links. When you specify the Managed By name, you must select the appropriate user or group.
- **Looking up an object.** You can search for any object in your Active Directory domain.

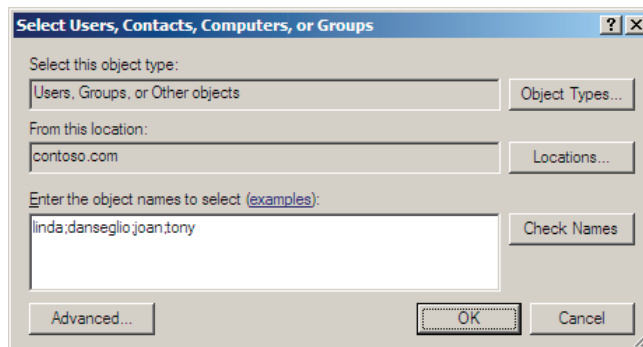
There are many other situations that will require searching Active Directory. There are several user interfaces that you will encounter. In this lesson, you'll learn some tricks for working with each.

Demonstration: Use the Select Users, Contacts, Computers, or Groups Dialog Box

In this demonstration, you will learn:

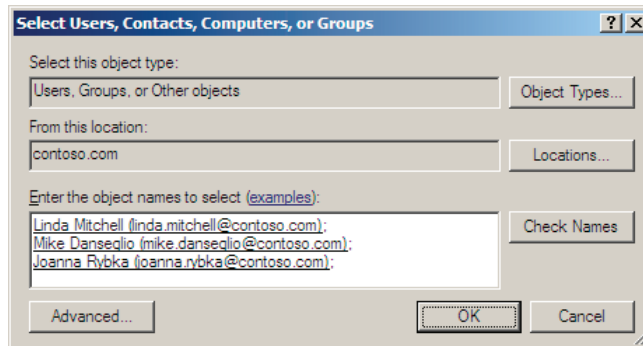
- How to select users with the Select dialog box

When you add a member to a group, assign a permission, or create a linked property, you are presented with the Select Users, Contacts, Computers, or Groups dialog box shown here.



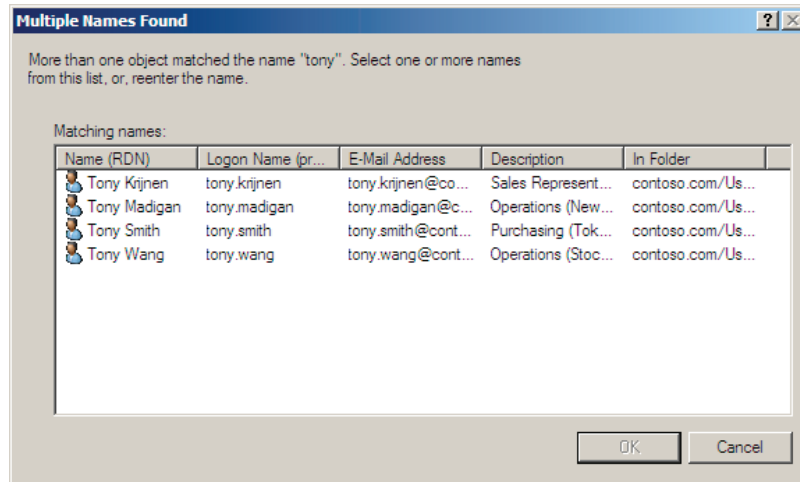
If you know the names of the objects you need, you can type them directly into the large text box. Multiple names can be entered, separated by semicolons, as shown above.

When you click OK, Windows looks up each item in the list and converts it into a link to the object, then closes the dialog box. The Check Names button also converts each name to a link, but leaves the dialog box open, as shown here:



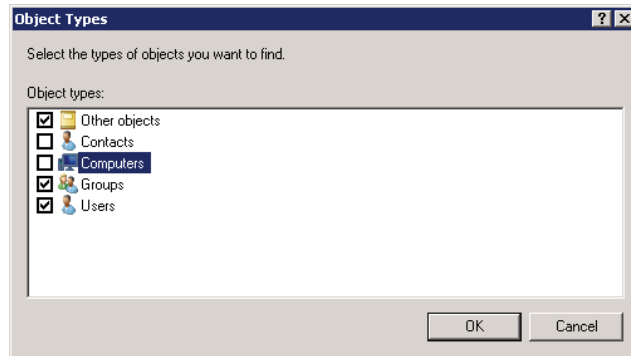
You do not need to enter the full name—you can enter either the user's first or last name or even just part of the first or last name. For example, the first screen shot above shows the first name *linda*, the last name *danseglio*, the partial name *joan*, and the name *tony*. When you click OK or Check Names, Windows will attempt to convert your partial name to the correct object. If there is only one matching object, the names will be resolved as shown in the second screenshot above.

If there are multiple matches, such as the name *Tony*, you will be presented with the Multiple Names Found box shown below. Select the correct name(s) and click OK.

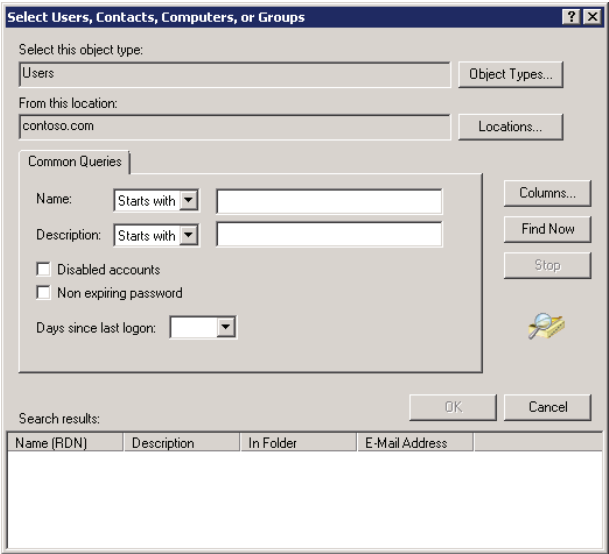


By default, the Select dialog box searches the entire domain. If you are getting too many results and wish to narrow down the scope of your search, or if you need to search another domain or the local users and groups on a domain member, click Locations.

Additionally, the Select dialog box—despite its full name, Select Users, Contacts, Computers or Groups—rarely searches all four object types. When you add members to a group, for example, computers are not searched by default. If you enter a computer name, it will not be resolved correctly. When you specify the name on the Managed By tab, groups are not searched by default. You must make sure that the Select dialog box is scoped to resolve the types of objects you want to select. Click the Object Types button and use the Object Types dialog box shown below to select the correct types, and then click OK.

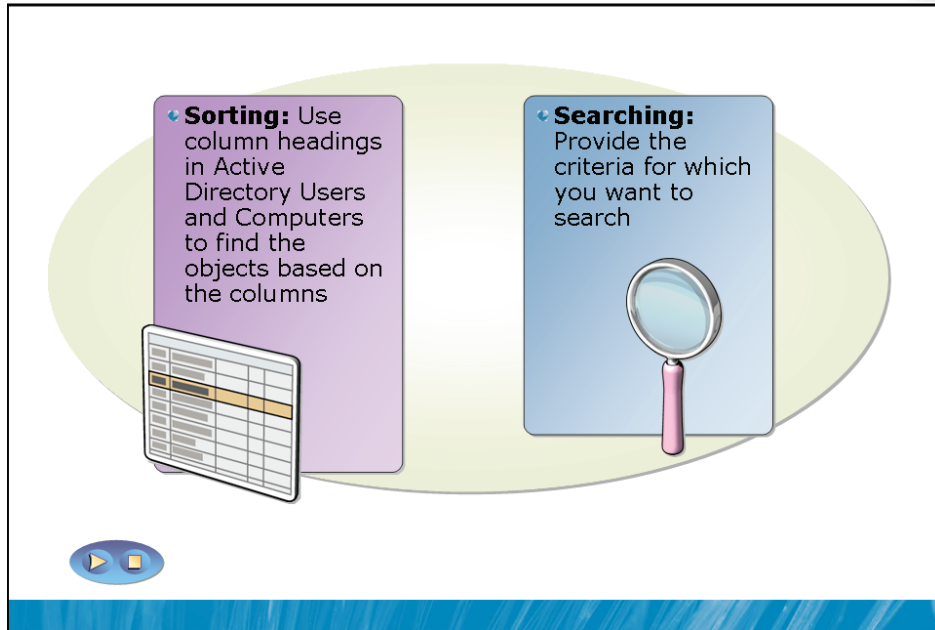


If you are having trouble locating the objects you want, click the Advanced button on the Select dialog box. The advanced view, shown below, allows you to search both name and description fields, as well as disabled accounts, non-expiring passwords, and “stale” accounts that have not logged on for a specific period of time.



Some of the fields on the Common Queries tab may be disabled, depending on the object type you are searching for. Click the Object Types button to specify exactly the type of object you want.

Options for Locating Objects in Active Directory Users and Computers



Key Points

Although you can navigate through Active Directory, browsing for an object, you will often locate the object you need more quickly by sorting or searching.

Additional Reading

- Search Active Directory: <http://go.microsoft.com/fwlink/?LinkId=168729>

Demonstration: Control the View of Objects in Active Directory Users and Computers

In this demonstration, you will learn:

- How to add or remove columns in the details pane
- How to sort objects based on columns in the details pane

Key Points

The details pane of the Active Directory Users and Computers snap-in can be customized to help you work effectively with the objects in your directory. Use the Add/Remove Columns command on the View menu to add columns to the details pane. Not every attribute is available to be displayed as a column, but you are certain to find columns that will be useful to display, such as User Logon Name. You might also find columns that are unnecessary. If your OUs have only one type of object (user or computer, for example), the Type column may not be helpful.

When a column is visible, you can change the order of columns by dragging the column headings to the left or right. You can also sort the view in the details pane by clicking the column—the first click will sort in ascending order, the second in descending order, just like in Windows Explorer.

A common customization is to add the Last Name column to a view of users, so that they can be sorted by last name. It is generally easier to find users by last name than by the Name column, which is the common name (CN) and is generally first name-last name.

To add the Last Name column to the details pane:

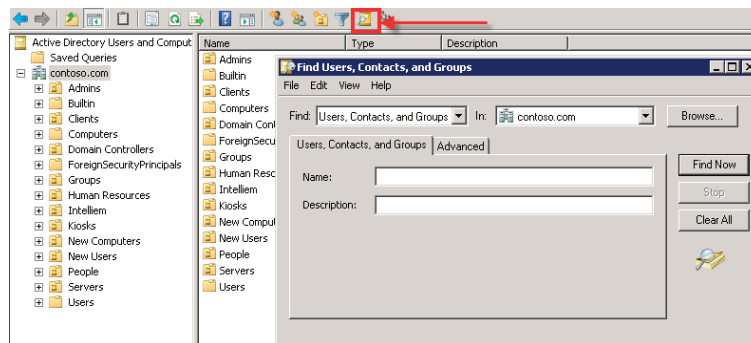
1. Click the **View** menu, and then click **Add/Remove Columns**.
2. In the **Available Columns** list, click **Last Name**.
3. Click the **Add** button.
4. In the **Displayed columns** list, click **Last Name** and click **Move Up** two times.
5. In the **Displayed columns** list, click **Type** and click **Remove**.
6. Click **OK**.
7. In the details pane, click the **Last Name** column header to sort alphabetically by last name.

Demonstration: Use the Find Command

In this demonstration, you will learn:

- How to search for objects in Active Directory using the Find command

Windows systems also provide the Active Directory query tool, called the *Find box* by many administrators. One way to launch the Find box is to click the Find Objects In Active Directory Domain Services button in the Active Directory Users and Computers snap-in. The button and the resulting Find box are shown below.



Use the Find drop-down list to specify the type(s) of objects you want to query, or select Common Queries or Custom Search. The In drop-down list specifies the scope of the search. It is recommended that, whenever possible, you narrow the scope of the search to avoid the performance impacts of a large, domain-wide search. Together, the Find and the In lists define the scope of the search.

Next, configure the search criteria. Commonly used fields are available as criteria based on the type of query you are performing. When you have specified your search scope and criteria, click Find Now. The results will appear.

You can then right-click any item in the results list and choose administrative commands such as Move, Delete, and Properties.

Determine Where an Object Is Located

1. Ensure that **Advanced Features** is selected in the **View** menu of the MMC console
 2. Find the object
 3. Open its **Properties** dialog box
 4. Click the **Object** tab
 5. View the **Canonical name of object**
- or*
- In the Find dialog box, **View** → **Choose Columns** and add the **Published At** column

Key Points

Sometimes you want to find an object using the Find command, because you don't actually know where the object is.

To determine where an object is located:

1. Click the **View** menu, and then select **Advanced Features**.
2. Click the **Find** button, and then perform a search for the object.
3. Right-click the object, then click **Properties**, and then click the **Object** tab.
4. The **Canonical name of object** shows you the path to the object, starting at the domain.

Alternately, in the Find dialog box, you can display the Published At column.

1. In the **Find** dialog box, click the **View** menu, and then click **Choose Columns**.
2. In the **Columns Available** list, click **Published At** and then click **Add**.
3. Click **OK**.

Demonstration: Use Saved Queries

In this demonstration, you will learn:

- How to create a saved query
- How to distribute a saved query
- Why saved queries are an efficient and effective tool for administration

Key Points

Windows Server 2003 introduced the Saved Queries node of the Active Directory Users and Computers snap-in. This powerful function allows you to create rule-driven views of your domain, displaying objects across one or more OUs.

To create a saved query:

1. Open the **Active Directory Users and Computers** snap-in.
Saved queries are not available in the Active Directory Users and Computers snap-in that is part of Server Manager. You must use the Active Directory Users and Computers console or a custom console with the snap-in.
2. Right-click **Saved Queries**, point to **New**, then click **Query**.
3. Enter a name for the query.
4. Optionally, enter a description.

5. Click **Browse** to locate the root for the query.
The search will be limited to the domain or OU you select. It is recommended that you narrow your search as much as possible, to improve search performance.
6. Click **Define Query** to define your query.
7. In the **Find** dialog box, select the type of object you want to query.
The tabs in the dialog box and the input controls on each tab change to provide options that are appropriate for the selected query.
8. Configure the criteria for your query.
9. Click **OK**.

After your query is created, it is saved within the instance of the Active Directory Users and Computers snap-in. So if you opened the Active Directory Users and Computers console (dsa.msc), your query will be available the next time you open the console. If you created the saved query in a custom console, it will be available in that custom console. To transfer saved queries to other consoles or users, you can export the saved query as an XML file, and then import it to the target snap-in.

The view of the saved query in the details pane can be customized as described earlier, with specific columns and sorting. A very important benefit of saved queries is that the customized view is specific to each saved query. When you add the Last Name column to the normal view of an OU, the Last Name column is actually added to the view of every OU, so you will see an empty Last Name column even for an OU of computers or groups. With saved queries, you can add the Last Name column to a query for user objects, and other columns for other saved queries.

Saved queries are a powerful way to virtualize the view of your directory and to monitor for issues such as disabled or locked accounts. Learning to create and manage saved queries is a worthwhile use of your time.

Lab B: Find Objects in Active Directory

- Exercise 1: Find Objects in Active Directory
- Exercise 2: Use Saved Queries
- Exercise 3 (Advanced Optional): Explore Saved Queries

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 20 minutes

Scenario

Contoso now spans five geographic sites around the world, with over 1000 employees. As your domain has become populated with so many objects, it has become more difficult to locate objects by browsing. You are tasked with defining best practices for locating objects in Active Directory for the rest of the team of administrators. You are also asked to monitor the health of certain types of accounts.

Exercise 1: Find Objects in Active Directory

In this exercise, you will use several tools and interfaces that make it easier for you to find an object in Active Directory.

The main tasks for this exercise are as follows:

1. Explore the behavior of the Select dialog box.
2. Control the view of objects in the Active Directory Users and Computers snap-in.
3. Use the Find command.
4. Determine where an object is located.

► Task 1: Explore the behavior of the Select dialog box



Important Note: The steps in this task guide you through using several important Active Directory Users and Computers interfaces. You can think of this task as a "tour" of the interfaces and their features. The specific changes you are making are less important than the experience you gain with the nuances of these interfaces. **Follow the exact steps listed** and don't worry about *what* you are doing; instead **focus on how you are doing it** and how the user interfaces behave.

The virtual machine should already be started and available after completing Lab A. However, if it is not, you should launch it complete the exercises in Lab A before continuing.

1. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd** and run your custom console, **C:\AdminTools\MyConsole.msc** as an administrator with username **Pat.Coleman_Admin** and password **Pa\$\$w0rd**. Alternately, run the pre-created console, **D:\AdminTools\ADConsole.msc** as an administrator.
2. In the console tree, expand the **Active Directory Users and Computers** snap-in, the **contoso.com** domain, and the **User Accounts** OU, and then click the **Employees** OU.
3. Right-click **Pat Coleman** and then click **Properties**.
4. Click the **Member Of** tab.

5. Click **Add**.
6. In the **Select** dialog box, type the name **Special**.
7. Click **OK**. The name is resolved to **Special Project**.
8. Click **OK** again to close the **Properties** dialog box.
9. In the console tree, expand the **Groups** OU, and then click the **Role** OU.
10. In the details pane, right-click the **Special Project** group and then click **Properties**.
11. Click the **Members** tab.

12. Click **Add**.

The Select Users, Contacts, Computers, or Groups dialog box appears.

13. Type **linda;joan**, and then click the **Check Names** button.

The Select dialog box resolves the names to Linda Mitchell and Joanna Rybka and underlines the names to indicate visually that the names are resolved.

14. Click **OK**.

15. Click **Add**.

16. Type **carole**, and then click **OK**.

The Select dialog box resolves the name to Carole Poland and closes. You see Carole Poland on the Members list.

When you click the OK button, a “Check Names” operation is performed prior to closing the dialog box. It is not necessary to click the Check Names button unless you want to check names and remain in the Select dialog box.

17. Click **Add**.

18. Type **tony;jeff**, and then click **OK**.

Because there are multiple users matching “tony,” the Multiple Names Found box appears.

19. Click **Tony Krijnen** and click **OK**.

Because there are multiple users matching “jeff,” the Multiple Names Found box appears.

20. Click **Jeff Ford** and click **OK**. Click **OK** to close the **Special Project Properties** dialog box.

Whenever there is more than one object that matches the information you enter, the check names operation will give you the opportunity to choose the correct object.

21. In the console tree, click the **Application** OU under the **Groups** OU.
22. In the details pane, right-click the **APP_Office** group and then click **Properties**.
23. Click the **Members** tab.
24. Click **Add**.
25. In the **Select** dialog box, type **DESKTOP101**.
26. Click **Check Names**.

A Name Not Found dialog box appears, indicating that the object you specified could not be resolved.

27. Click **Cancel** to close the **Name Not Found** box.
28. In the **Select** box, click **Object Types**.
29. Select the check box next to **Computers** and click **OK**.
30. Click **Check Names**.

The name will resolve now that the Select box is including computers in its resolution.

31. Click **OK**.
32. Click **OK** to close the **APP_Office Properties** dialog box.

► **Task 2: Control the view of objects in the Active Directory Users and Computers snap-in**

1. In the console tree, expand the **contoso.com** domain and the **User Accounts** OU, and then click the **Employees** OU.
2. Click the **View** menu, and then click **Add/Remove Columns**.
3. In the **Available Columns** list, click **Last Name**.

4. Click the **Add** button.
5. In the **Displayed columns** list, click **Last Name** and click **Move Up** two times.
6. In the **Displayed columns** list, click **Type** and click **Remove**.
7. Click **OK**.
8. In the console tree, expand the **contoso.com** domain and the **User Accounts** OU, and then click the **Employees** OU.
9. In the details pane, click the **Last Name** column header to sort alphabetically by last name.
10. Click the **View** menu, and then click **Add/Remove Columns**.
11. In the **Available Columns** list, click **Pre-Windows 2000 Logon**.
12. Click the **Add** button.
13. In the **Displayed columns** list, click **Pre-Windows 2000 Logon** and click **Move Up**.
14. Click **OK**.
15. In the console tree, expand the **contoso.com** domain and the **User Accounts** OU, and then click the **Employees** OU.

► **Task 3: Use the Find command**

1. In the console tree, expand the **contoso.com** domain and the **User Accounts** OU, and then click the **Employees** OU.
2. Click the **Find** button in the toolbar.
3. In the **Name** box, type **Dan**, and then click **Find Now**.
4. How many items were found? Look at the status bar, at the bottom of the Find Users, Contacts, and Groups window.
5. Click the **In** drop-down list, and then click **Entire Directory**.
6. Click **Find Now**.

7. How many items were found? Look at the status bar, at the bottom of the Find Users, Contacts, and Groups window.
8. Close the **Find Users, Contacts, and Groups** dialog box.

► **Task 4: Determine where an object is located**

1. Turn on the view of **Advanced Features** for the **Active Directory Users and Computers** snap-in.
2. Use the **Find** command to locate users in domain whose names begin with **Pat.Coleman**. You should see two results.
3. Use the properties of **Pat Coleman (Admin)** to determine where the user is located in Active Directory.

Results: After this exercise, you will have learned that there are several interfaces with which you perform searches against Active Directory, and you know how to control the view in the Active Directory Users and Computers snap-in.

Exercise 2: Use Saved Queries

In this exercise, you will create saved queries, with which administrative tasks can be more efficiently performed.

The main tasks for this exercise are as follows:

1. Create a saved query that displays all domain user accounts.
2. Create a saved query that shows all user accounts with non-expiring passwords.
3. Transfer a query to another computer.

► Task 1: Create a saved query that displays all domain user accounts

- Create a saved query called **All User Objects** that shows all users in the domain.

► Task 2: Create a saved query that shows all user accounts with non-expiring passwords

- Create a saved query called **Non-Expiring Passwords** that shows all users in the domain whose passwords do not expire.

Note that, for the purposes of maintaining a simple, single password for all users in this course, *all* user accounts are configured so that passwords do not expire. In a production environment, user accounts should not be configured with non-expiring passwords.

► **Task 3: Transfer a query to another computer**

1. Export the **Non-Expiring Passwords** query to C:\AdminTools\Query_NonExpPW.xml.
2. Delete the **Non-Expiring Passwords** query.
3. Import the C:\AdminTools\Query_NonExpPW.xml query.
4. Log off of HQDC01.

Results: After this exercise, you will have two saved queries. One, **All User Objects**, demonstrates that a saved query can create a virtualized view of your domain, allowing you to see objects that meet a set of criteria, regardless of which OU those objects are in. The second query, **Non-Expiring Passwords**, demonstrates that you can use saved queries to monitor the health of your environment.

Exercise 3 (Advanced Optional): Explore Saved Queries

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

The main tasks for this exercise are as follows:

1. Run the pre-created console, D:\AdminTools\ADConsole.msc as an administrator and explore the console.
2. Examine the queries used in the Saved Queries node of the Active Directory Users and Computers snap-in.

Notice that administrators using this tool will rarely, if ever, need to "dive in" to the organizational unit structure underneath the contoso.com domain in the console tree. Almost all day-to-day administrative tasks can be performed with the views in Saved Queries.



Note: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in the Lab C.

Lab Review Questions

Question: In your work, what scenarios require you to search Active Directory?

Question: What types of saved queries could you create to help you perform your administrative tasks more efficiently?

Lesson 4

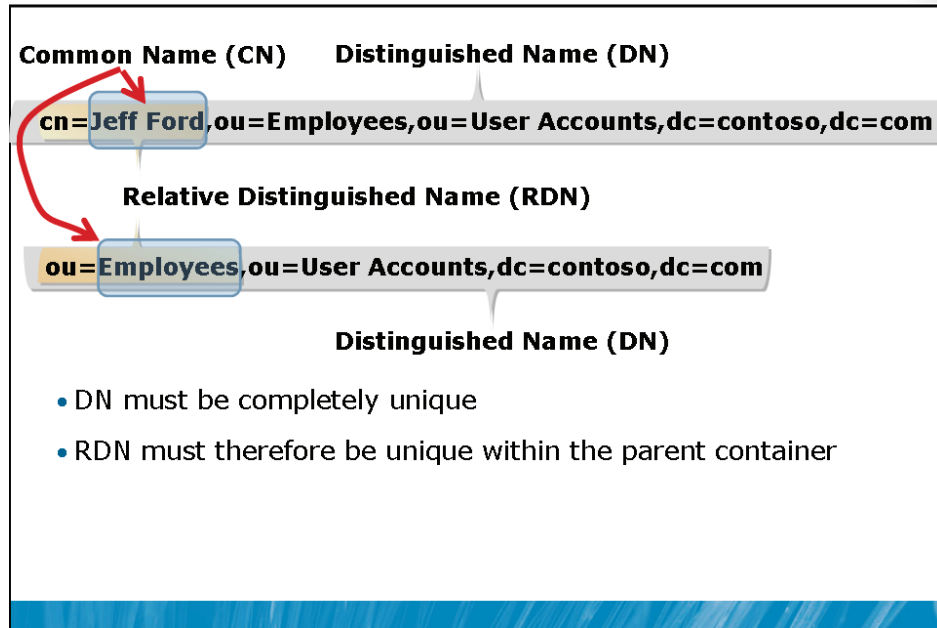
Use DS Commands to Administer Active Directory

- DNs, RDNs, and CNs
- The DS Commands
- Find Objects with DSQuery
- Retrieve Object Attributes with DSGet
- Pipe NDs to Other DS Commands
- Modify Object Attributes with DSMod
- Delete an Object with DSRm
- Move an Object with DSMove
- Add an Object with DSAdd
- Administration without the GUI

After completing this lesson, you will be able to:

- Identify the distinguished name (DN), relative distinguished name (RDN), and common name (CN) of an Active Directory object.
- Use the DS commands to administer Active Directory from the command line.

DNs, RDNs, and CNs



Distinguished names (DNs) are a kind of path to an object in Active Directory. Each object in Active Directory has a completely unique DN. Our user, Jeff Ford, has the following DN:

`CN=Jeff Ford,OU=Employees,,OU=User Accounts,DC=contoso,DC=com`

You can see what is happening: the DN is a path, starting at the object and working up to the top level domain in the contoso.com DNS namespace. CN means common name. You learned about this property earlier: When you create a user, the Full Name box is used to create the CN of the user object. As you know, OU means organizational unit. And DC means domain component.

The portion of the DN prior to the first OU or container is called the relative distinguished name, or RDN. In the case of Jeff Ford, the RDN of the object is `CN=Jeff Ford`. Not every RDN is a CN. The DN of the Employees OU is `OU=Employees,OU=User Accounts,DC=contoso,DC=com`. The RDN of the Employees OU is therefore `OU=Employees`.

Because the DN of an object must be unique within the directory service, the RDN of an object must be unique within its container. That's why if you hire a second Jeff Ford, and if both user objects need to be in the same OU, you will have to give that user a different CN. The same logic applies as to files in a folder: you cannot have two files with identical names in a single folder.

You will encounter DNs regularly as you work with Active Directory, just as you encounter file paths regularly if you work with files and folders. It's very important to be able to read them and interpret them.

The DS Commands

- **DSQuery.** Performs a query based on parameters provided at the command line and returns a list of matching objects
- **DSGet.** Returns specified attributes of an object
- **DSMod.** Modifies specified attributes of an object
- **DSMove.** Moves an object to a new container or OU
- **DSAdd.** Creates an object in the directory
- **DSRm.** Removes an object, all objects in the subtree beneath a container object, or both
- **DScommand /?**
For example: dsquery /?

Key Points

Windows provides command-line utilities that perform functionality similar to that of the Active Directory Users and Computers snap-in. Many of those commands begin with the letters *DS*, so they are often referred to as *the DS commands*.

The following DS commands are supported in Windows Server 2008:

- **DSQuery.** Performs a query based on parameters provided at the command line and returns a list of matching objects
- **DSGet.** Returns specified attributes of an object
- **DSMod.** Modifies specified attributes of an object
- **DSMove.** Moves an object to a new container or OU
- **DSAdd.** Creates an object in the directory
- **DSRm.** Removes an object, all objects in the subtree beneath a container object, or both

Each command is well documented. Type the command name followed by `/?`—for example, `dsquery /?`—for help with the command.

Find Objects with DSQuery

- **dsquery *objectType***
 - *objectType*: user, computer, group, ou
 - By default, search scope is the entire domain
 - **-limit** switch to specify number of results
 - 100 is default
 - 0 means "return all results"
- **dsquery *objectType* -attribute "*criteria*"**
 - *attribute* is *objectType* specific: **dsquery *objectType* /?**
 - Examples for user: -name, -samid, -office, -desc
 - *criteria* in quotes if there is a space. Wildcards (*) allowed
- **dsquery *objectType* *BaseDN***
-scope {subtree|onelevel|base}
 - Specify search start and scope

Key Points

DSQuery can locate objects in Active Directory.

Type `dsquery.exe /?` to learn its syntax and usage.

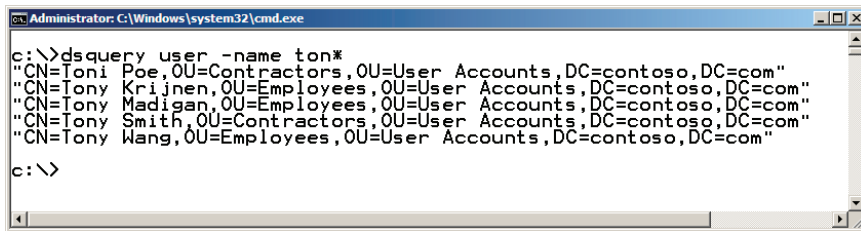
You use most DS commands by specifying the object type you want the command to work against. For example, `dsquery user` would be entered to look for a user, whereas `dsquery computer`, `dsquery group`, and `dsquery ou` would query for their respective object types.

If you use the `dsquery objectType` command by itself, it will return the distinguished names of all users in the domain.

To prevent a runaway query, DSQuery limits itself to 100 results. You can use the `-limit` switch to specify how many results you want returned. Use `-limit 0` to return all objects.

Following the *objectType* specifier, you can use switches to indicate the criteria for the query. For example, each object can be located by its name with the *-name* switch. Most objects can be queried based on the description (*-desc*). Security principals can be located based on their pre-Windows 2000 logon name before Windows 2000 (*-samid*). To learn which properties may be queried, type *dsquery objectType /?*; for example, *dsquery user /?*.

For example, if you want to locate all users whose names begin with *Ton*, you would enter this command: *dsquery user -name ton**. After the property switch, *-name* in this case, you can enter the criteria, which are not case sensitive and can include wildcards such as the asterisk, which represents zero or more characters. The DSQuery command returns matching objects with their DNs by default, as you can see below:



```
c:\>dsquery user -name ton*
"CN=Tony Poe,OU=Contractors,OU=User Accounts,DC=contoso,DC=com"
"CN=Tony Krijnen,OU=Employees,OU=User Accounts,DC=contoso,DC=com"
"CN=Tony Madigan,OU=Employees,OU=User Accounts,DC=contoso,DC=com"
"CN=Tony Smith,OU=Contractors,OU=User Accounts,DC=contoso,DC=com"
"CN=Tony Wang,OU=Employees,OU=User Accounts,DC=contoso,DC=com"
c:\>
```

If DNs are not the results you would like to see, add the *-o* switch to the DSQuery command. You can add *-o samid*, for example, to return the results with logon names before Windows 2000, or *-o upn* to return the list as user logon names, known as user principal names (UPNs).

DSQuery can perform searches using wildcards, such as the asterisk (*), which represents zero or more characters. The following command retrieves all users whose names start with *Dan*:

```
dsquery user -name "Dan*"
```

Remember that DSQuery criteria are not case sensitive.

Finally, you can limit the scope of the search performed by DSQuery by adding the DN of an OU or container after the *objectType* element of the command. For example, the following command searches for users whose names begin with *Dan*, but only in the Admins OU:

```
dsquery user "ou=Admins,dc=contoso,dc=com" -name "Dan*"
```

By default, the search includes all sub-OUs of the base. You can use the *-base* parameter to limit the search further—for example, to only the specified OU without its sub-OUs.

Retrieve Object Attributes with DSGet

- **`dsget objectType objectDN -attribute`**
 - **Common syntax for many DS commands**
- **`dsget user "cn=Jeff Ford,ou=Employees,ou=User Accounts,dc=contoso,dc=com" -email`**
- What is the difference between DSGet and DSQuery?

Key Points

The DSGet command returns attributes of one or more objects. For example, the following command returns the e-mail address of Jeff Ford:

```
dsget user "cn=Jeff Ford,ou=Employees,ou=User  
Accounts,dc=contoso,dc=com" -email
```

This command illustrates a common theme for most of the DS commands. Most of the DS commands take two modifiers after the command itself: the object type and the object's DN. In the example shown above, the object type, `user`, immediately follows the command. After the object type is the object's DN. When the object's DN includes a space, surround the DN with quotes.

The DSGet command can bring back different attributes for each object type. And, unfortunately, the parameter name of a DS command that represents an attribute is not always the same name as either the Active Directory Users and Computers interface or the actual attribute name in the Schema. For example, DSGet uses the -samid parameter to return the user's pre-Windows 2000 logon name, which is the *sAMAccountName* attribute in the schema. Finally, DSGet can return only a subset of available attributes for any type of object.

To learn which attributes DSGet can retrieve for an object type, type:

```
dsget objectType /?.
```

Question: Can you explain the difference between the DSQuery command and the DSGet command?

Pipe DNs to Other DS Commands

- Typing DNs is difficult!
 - `dsget user "cn=Jeff Ford,ou=Employees,ou=User Accounts,dc=contoso,dc=com" -email`
- DSQuery returns DNs
 - **`dsquery user -name "Jeff Ford"`**
> "cn=Jeff Ford,ou=Employees,ou=User Accounts,dc=contoso,dc=com"
- Pipe (send) the DNs from DSQuery to DSGet with |
 - **`dsquery user -name "Jeff Ford" | dsget user -email`**
 - Or multiple results:
`dsquery user -name "Dan*" | dsget user -email`
- DSGet can return DNs from some attributes as well
 - **`dsget group groupDN -members | dsget user -samid`**

Key Points

In the previous topic, you learned that the following command returns the e-mail address of Jeff Ford:

```
dsget user "cn=Jeff Ford,ou=Employees,ou=User  
Accounts,dc=contoso,dc=com" -email
```

You can imagine that typing the DN of an object at the command line is time consuming and prone to error. Luckily, there is an easier way. You can "pipe" DNs from DSQuery to the other DS commands.

Remember that DSQuery locates objects based on search criteria and returns their DNs as a kind of "output." Other DS commands, such as DSGet, require the DN of the object or objects they will operate upon—they take DNs as "input." You can chain two commands together so that the "output" of DSQuery becomes the "input" of DSGet or another DS command. This is called "piping" because you are sending the output from one command through a "pipe" into another command.

Piping is achieved using the pipe character (|).

Examine the following command:

```
dsquery user -name "Jeff Ford" | dsget user -email
```

You know that the first part of the command will return the DN of any object whose name attribute (CN) is equal to "Jeff Ford."

You know that the second part of the command returns the e-mail address attribute of a user. But notice that the DN is missing from the DSGet command. That creates the "input" end of the pipe. The "output" of DSQUery is sent through the pipe rather than returned to the console as text output.

So that command will also return Jeff Ford's e-mail address, without requiring you to identify and accurately type the DN for his user object.

The next command retrieves the e-mail addresses of all users whose names start with *Dan*:

```
dsquery user -name "Dan*" | dsget user -email
```



Important: DSGet can return one or more DNs when it retrieves certain attributes, including the member and memberof attributes. When a DSGet command will produce DNs, you can pipe the results to another DS command.

Modify Object Attributes with DSMod

- `dsmod objectType "objectDN" -attribute "new value"`
- `dsmod user "cn=Jeff Ford,ou=Employees,ou=User Accounts,dc=contoso,dc=com" -dept "Information Technology"`
- `dsquery user "ou=Admins,dc=contoso,dc=com" | dsmod user -department "Information Technology"`

Key Points

The DSMod command modifies specified attributes of one or more objects. The basic syntax is:

```
dsmod objectType "objectDN" -attribute "new value"
```

For example, this command changes the department attribute of Jeff Ford to Information Technology:

```
dsmod user "cn=Jeff Ford,ou=Employees,ou=User Accounts,dc=contoso,dc=com" -dept "Information Technology"
```

You can specify multiple attributes to modify on a single command line. To find out which attributes can be modified for a type of object, type `dsmod objectType /?`; for example, `dsmod user /?`.

The results of DSQuery can be piped to DSMod. You can change the department attribute of all users in the Admins OU with this command:

```
dsquery user "ou=Admins,dc=contoso,dc=com" | dsmod user -department  
"Information Technology"
```

Delete an Object with DSRm

- `dsrm objectDN`
 - Note that DSRm does not take an *objectType*
- `dsrm "cn=DESKTOP234,ou=Client Computers,dc=contoso,dc=com"`
- `dsquery computer -stalepwd 90 | dsrm`

Key Points

The DSRm command removes (deletes) an object from Active Directory. Because DSRm deletes objects without confirmation prompts, be very careful when you enter the command. The syntax of the command is simple:

```
dsrm objectDN
```

For example, this command deletes the computer named DESKTOP234:

```
dsrm "cn=DESKTOP234,ou=Client Computers,dc=contoso,dc=com"
```

If you specify the DN of a container or organizational unit, DSRm will, by default, delete the container and all of its subcontainers or sub-OU's.

The results of DSQuery can be piped to DSRm. You can delete all computers that have not logged on in more than 90 days with this command:

```
dsquery computer -stalepwd 90 | dsrm
```

Move an Object with DSMove

- **`dsmove objectDN -newparent targetOUDN`**
 - *objectDN*: object to be moved
 - *targetOUDN*: target (destination) OU
- **`dsmove objectDN -newname newName`**
 - *objectDN*: object to be moved
 - *newName*: new name for object (used in the RDN)

Key Points

The DSMove command moves an object to a new container or OU. The syntax is very straightforward:

```
dsmove objectDN -newparent targetOUDN
```

In this syntax, *objectDN* is the DN of the object you want to move, and *targetOUDN* is the DN of the OU to which the object will be moved.

DSMove is also used to rename the RDN of an object, with the `-newname` parameter:

```
dsmove objectDN -newname newName
```

Add an Object with DSAdd

- `dsadd objectType objectDN -attribute "value"`
 - *objectType*: class of object to add
 - *objectDN*: OU in which to create object
 - *-attribute "value"*: attributes to populate
 - Each object class has required attributes
- `dsadd ou "ou=Lab,dc=contoso,dc=com"`

Key Points

DSAdd creates an object in the directory. The basic syntax is:

```
dsadd objectType objectDN
```

In this syntax, *objectType* is the class of object to create: either user, group, computer or OU, and *objectDN* is the DN for the new object. The *objectDN* parameter is required, as it represents both the DN of the OU or container in which the object will be created and the RDN of the object itself.

For example, this command creates a top-level OU called Lab:

```
dsadd ou "ou=Lab,dc=contoso,dc=com"
```


Most object classes have parameters that must be configured when you are creating an object. For example, user accounts require a pre-Windows 2000 logon name (*sAMAccountName* attribute). Other modules will detail how you can use DS commands to create, find, modify, and delete users, groups, and computers. In those modules, you will learn how to use DSAdd with the correct parameters for that type of object.

Administration Without the GUI

- Command Prompt
 - DS commands
 - csvde.exe and ldifde.exe
- LDAP
 - ldp.exe
- Windows PowerShell
- Scripting
 - Windows PowerShell scripts
 - VBScript

Key Points

There are other tools that you can use to administer AD DS without the GUI administrative tools. You will experience many of these in later modules.

Lab C: Use DS Commands to Administer Active Directory

- Exercise 1: Use DS Commands to Administer Active Directory

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

Contoso is growing, and changes need to be made to objects in Active Directory. You are an administrator of AD DS, and you know that it can be easier to create, delete, and modify objects using the command prompt than using Active Directory Users and Computers.

Exercise 1: Use DS Commands to Administer Active Directory

In this exercise, you will use DS commands to perform basic administrative tasks. Some of these tasks would be difficult or impossible to perform in the user interface of Active Directory Users and Computers.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Find objects with DSQuery.
3. Retrieve object attributes with DSGet.
4. Pipe DNs from DSQuery to other DS commands.
5. Pipe DNs from DSGet to DSMod (advanced, optional).

Remember that you can always type the command, followed by a `/?`, for help with the command. When a command works with a particular type of object, type *command objectType /?* for even more help.

► Task 1: Prepare for the lab

The virtual machine should already be started and available after completing Labs A and B. However, if it is not, you should launch it and complete the exercises Labs A and B before continuing.

1. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
2. Open **D:\Labfiles\Lab02c**.
3. Run **Lab02c_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
4. The lab setup script runs. When it is complete, press any key to continue.
5. Close the Windows Explorer window, **Lab02c**.

► Task 2: Find objects with DSQuery

1. Open Command Prompt with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. Use **DSQuery** to find all users whose last names are **Mitchell**.

► **Task 3: Retrieve object attributes with DSGet**

1. From the command prompt, get the e-mail address of **Tony Krijnen**.

The distinguished name of Tony's user account is:

cn=Tony Krijnen,ou=Employees,ou=User Accounts,dc=contoso,dc=com

2. From the command prompt, list the members of the **Finance Managers** group.

The distinguished name of the Finance Managers group is:

cn=Finance Managers,ou=Role,ou=Groups,dc=contoso,dc=com

► **Task 4: Pipe DNs from DSQuery to other DS commands**

Scott and Linda Mitchell are joining the Special Project team. They are the only two employees with the last name Mitchell who work at Contoso. They work in the Vancouver office.

1. Using a single command, add the Mitchells to the Special Project group.

Perform this step without typing the DN of the Mitchells' user accounts.

The DN of the Special Project group is "**cn= Special Project,ou=Role,ou=Groups,dc=contoso,dc=com**"

If you receive an error that says "The specified name is already a member of the group," use Active Directory Users and Computers to remove Scott Mitchell and Linda Mitchell from the Special Project group, then try again.

You may receive an Access Denied error. What is causing this error, and what can you do to work around it?

2. Using a single command, retrieve the e-mail address of all users in the Vancouver office.

Users in the Vancouver office have the word Vancouver in the Description field.

If you receive a warning that your DSQuery has reached its limit, what can you do to ensure all results are returned?

3. Using a single command, change the **office** attribute of the Mitchells to **Vancouver**.

► **Task 5 (Advanced Optional): Pipe DNs from DSGet to DSMod**

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

Contoso is relocating and centralizing the executive leadership from regional offices to the Seattle office.

- Using a single command, change the office attribute of all members of the Executives group to **Headquarters**. Do this without typing the DN of the Executives group.

► **Task 6 (Advanced Optional): DSQuery ***

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

- Type **dsquery /?** and examine the syntax of **dsquery.exe** *. Notice that you can use this form of DSQuery to display an arbitrary set of attributes using the Schema-defined name for the attribute.

Results: After this exercise, the Mitchells will belong to the Special Project group. The **office** attribute of the Mitchells is set to Vancouver, and the members of the Executives group have their **office** attribute set to **Headquarters**. You've learned how to administer Active Directory from the command line with DS commands!



Note: After completing this exercise, turn off all virtual machines and discard undo disks.

Lab Review Questions

Question: What can you do to avoid typing DNs of users, groups, or computers into DSGet, and other DS commands?

Question: How are wildcard searches with DSQuery different than searches performed with the Find command in Active Directory Users and Computers? In other words, what kind of search have you performed in this lab that would not have been possible using the basic interface of the Find command?

Module 3

Manage Users

Contents:

Lesson 1: Create and Administer User Accounts	3-4
Lab A: Create and Administer User Accounts	3-29
Lesson 2: Configure User Object Attributes	3-35
Lab B: Configure User Object Attributes	3-51
Lesson 3: Automate User Account Creation	3-61
Lab C: Automate User Account Creation	3-70

Module Overview

- Create and Administer User Accounts
- Configure User Object Attributes
- Automate User Account Creation

In this module, you will learn to create and support user accounts. User accounts stored in the directory are the fundamental component of identity. Because of their importance, knowledge of user accounts and the tasks related to supporting them is critical to the success of an administrator in a Windows® enterprise.

Each day in an enterprise network brings with it a unique set of challenges related to user management. Employees are hired, moved, married, and divorced, and most eventually leave the organization. As human beings, they make mistakes such as forgetting passwords or locking out their accounts by logging on incorrectly.

Administrators must respond to all these changes, and your ability to work effectively with user accounts can make a big difference in your overall productivity. This module begins with a discussion of options for creating user accounts by using the Active Directory Users and Computers snap-in and the DSAdd command. These skills, which are effective for creating a single user account or a small number of user accounts, can become clumsy and inefficient when you are working with large numbers of accounts, so the module also introduces several options for automating the creation of users.

Of course, creating a user is only the first step in the lifecycle of a user in a domain. After creating the user, you must configure attributes that define both the properties of the security principal (the “account”) and properties that define and manage the user. You must also know how and when to administer the account—to perform password resets and to unlock the account, for example. You must be able to move the user between organizational units (OUs) and, eventually, deprovision the account by disabling or deleting it. This module will cover the procedures used to support a user object through its lifecycle—procedures you can perform using both the Windows interface and the command-line or automation tools.

Objectives

After completing this module, you will be able to:

- Create and configure the account-related properties of a user object.
- Identify the purpose and requirements of user account attributes.
- Perform common administrative tasks to support user accounts, including password reset and account unlock.
- Enable and disable user accounts.
- Delete, move, and rename user accounts.
- View and modify hidden attributes of user objects.
- Identify the purpose and requirements of user object attributes.
- Create users from user account templates.
- Modify attributes of multiple users simultaneously.
- Export user attributes with CSVDE.
- Import users with CSVDE.
- Import users with LDIFDE.

Lesson 1

Create and Administer User Accounts

- User Account
- Demonstration: Create a User Object
- Create Users with DSAdd
- Name Attributes
- Rename a User Account
- Account Attributes
- Reset a User's Password
- Unlock a User Account
- Disable and Enable a User Account
- Delete a User Account
- Move a User Account

A user account is the cornerstone of identity and access (IDA) in Active Directory® Domain Services. Consistent, efficient, and secure processes regarding the administration of user accounts is therefore the cornerstone of enterprise security management.

Objectives

After completing this lesson, you will be able to:

- Create and configure the account-related properties of a user object.
- Identify the purpose and requirements of user account attributes.
- Perform common administrative tasks to support user accounts, including password reset and account unlock.
- Enable and disable user accounts.
- Delete, move, and rename user accounts.

User Account

- A user account is an object that
 - Enables authentication of a user with attributes, including a user logon name and password
 - Is a security principal with a security identifier (SID) that can be assigned permissions to resources
- A user account can be stored
 - In Active Directory®, where it enables logon to the domain and can be assigned permissions to resources anywhere in the domain
 - Domain user accounts are administered with Active Directory snap-ins and commands
 - In the local security accounts manager (SAM) database of a member computer, where it enables logon to the local computer and can be assigned permissions to local resources
 - Local user accounts are administered with the Local Users and Groups snap-in and the net local user command

Key Points

User objects are often referred to as *user accounts*. But when you look closely, what you think of as an “account” (the username, password, and perhaps the security identifier or SID) is just a subset of attributes of a user object. Active Directory user objects include numerous attributes that are either only indirectly related to the account (such as the profile path property) or are attributes of the human being whom the account represents (such as the email address, phone number, and manager properties).

User accounts—the actual “account” attributes of the user object—do two things. They enable authentication—the logon process during which the identity of the user is validated by comparing the user’s logon name and password. And then, once the user has logged on, the account SID is compared with permissions on resources that the user attempts to access. Module 1 described the logon process, the generation of the security token that includes the user’s SID, and the mechanism through which permissions in an ACL are compared to the SIDs in the token to determine the level of access to a resource.

A user account can be created and stored in Active Directory. A domain user account enables logon to any machine in the domain, and access to resources throughout the domain. Of course, both sets of activities are subject to the logon rights, privileges, and permissions assigned to the account.

And although Active Directory accounts are certainly the focus of this course, accounts can also be stored in the local security accounts manager (SAM) database, enabling local logon and access to local resources. Local user accounts are, for the most part, beyond the scope of this course.

Demonstration: Create a User Object

In this demonstration, you will learn:

- How to create a user
- How to configure the properties of a user object

Key Points

A user object, often referred to as a *user account*, includes the username and password, which serve as the logon credentials for a user. A user object also includes several other attributes that describe and manage the user.

To create a user object:

1. Right-click the OU or container in which you want to create the user, point to **New**, and then click **User**.
2. In **First name**, type the user's first name.
3. In **Initials**, type the user's middle initial(s).

Note that this property is, in fact, meant for the initials of a user's middle name, not the initials of the user's first and last name.

4. In **Last name**, type the user's last name.

5. The **Full name** field is populated automatically. Make modifications to it if necessary.

The Full name field is used to create several attributes of a user object, most notably the common name (CN) and display name properties. The CN of a user is the Name displayed in the details pane of the snap-in. It must be unique within the container or OU. Therefore, if you are creating a user object for a person with the same name as an existing user in the same OU or container, you will need to enter a unique name in the Full name field.

6. In **User logon name**, type the name that the user will log on with and, from the drop-down list, select the **UPN Suffix** that will be appended to the user logon name following the @ symbol.

Usernames in Active Directory can contain some special characters (including periods, hyphens, and apostrophes), which let you generate accurate usernames such as O'Hare and Smith-Bates. However, certain applications may have other restrictions, so it is recommended to use only standard letters and numerals until you have fully tested the applications in your enterprise for compatibility with special characters in logon names.

The list of available UPN suffixes can be managed by using the Active Directory Domains and Trusts snap-in. Right-click the root of the snap-in, Active Directory Domains and Trusts, click Properties, and use the UPN Suffixes tab to add or remove suffixes. The DNS name of your Active Directory domain will always be available as a suffix and cannot be removed.

7. In the **User logon name (pre-Windows 2000)** box, enter the pre-Windows 2000 logon name, often called the "downlevel" logon name. In the Active Directory database, the name for this attribute is sAMAccountName.
8. Click **Next**.
9. Enter an initial password for the user in the **Password** and **Confirm password** boxes.
10. Select **User must change password at next logon**.

It is recommended that you always select this option so that the user can create a new password unknown to the IT staff. Appropriate support staff can always reset the user's password at a future date if they need to log on as the user or access the user's resources. But only users should know their passwords on a day-to-day basis.
11. Click **Next**.
12. Review the summary and then click **Finish**.

The New Object – User interface allows you to configure a limited number of account-related properties, such as name and password settings. However, a user object in Active Directory supports dozens of additional properties. These can be configured after the object has been created.

1. Right-click the user object you created and then click **Properties**.
2. Configure user properties.
3. Click **OK**.

Additional Reading

- Active Directory Users and Computers Help: Managing Users:
<http://go.microsoft.com/fwlink/?LinkId=168742>
- Create a New User Account:
<http://go.microsoft.com/fwlink/?LinkId=168743>

Create Users with DSAdd

- `dsadd user "UserDN" -samid pre-Windows 2000 logon name -pwd { password | * } -mustchpwd yes`
 - **UserDN.** Distinguished name of user to create
 - **-samid.** Required for new account
 - **Pre-Windows 2000 logon name.** The "downlevel" logon name that can be used in the logon format domain\username and that becomes %username%
 - **-pwd password.** The desired initial password
 - Asterisk (*) will prompt you to enter it at the command prompt, so that the plain text password is not entered with the command
 - **-mustchpwd { yes | no }.** User must change password at next logon
 - Lots of optional attributes, including
 - -email
 - -hmdir & -profile

Can use \$username\$ token to represent the value of -samid; for example,
 -profile \\server01\users\\$username\$\profile

Key Points

Use the DSAdd command to create objects in Active Directory. The DSAdd User command creates a user object and accepts parameters that specify properties of the user. The following command shows the basic parameters required to create a user account:

```
dsadd user "UserDN" -samid pre-Windows 2000 logon name
-pwd {Password | *} -mustchpwd yes
```

The `-pwd` parameter specifies the password. If it is set to an asterisk (*), you are prompted for a user password. The `-mustchpwd` parameter specifies that the user must change the password at next logon.

DSAdd User accepts a number of parameters that specify properties of the user object.

The following command creates a user with some of the more important fields populated:

```
dsadd user "cn=Amy Strande,ou=Employees,ou=User  
Accounts,dc=contoso,dc=com" -samid Amy.Strande -fn Amy -ln Strande -  
display "Strande, Amy" -pwd Pa$$w0rd -desc "Vice President, IT"
```

Most parameter names are self-explanatory: *-email*, *-profile*, and *-company*, for example. Type **dsadd user /?** or search the Windows Server® 2008 Help And Support Center for thorough documentation of the DSAdd User parameters.

The special token `$username$` represents the user logon name (pre-Windows 2000)—the `sAMAccountName` attribute—in the value of the *-email*, *-hmdir*, *-profile*, and *-webpg* parameters. For example, to configure a home folder for a user when creating the user with the DSAdd User command shown earlier, add the following parameter:

```
-hmdir \\server01\users\$username$\documents
```

Additional Reading

- DSAdd: <http://go.microsoft.com/fwlink/?LinkId=168744>

Name Attributes

- **User logon name (pre-Windows 2000): sAMAccountName**
 - Unique in domain **CONTOSO\Tony.Krijnen**
 - 20-character limit
- **User logon name: userPrincipalName (UPN)**
 - Name + @ + UPN suffix **Tony.Krijnen@contoso.com**
 - Unique in forest
- **Name or Full Name: cn (common name) Tony Krijnen**
 - Unique in OU so that the relative distinguished name (RDN) is unique in the OU, so that, in turn, the object's distinguished name (distinguishedName attribute) is unique in the forest
- **Display name: displayName Krijnen, Tony**
 - Exchange global address list (GAL)
 - Best if unique, but not technically required to be unique

Key Points

There are several attributes related to the name of a user object and an account. It is important to understand the distinctions between them.

- A user's **User logon name (pre-Windows 2000)** is, behind the scenes, the sAMAccountName attribute. It's also sometimes called the samid. It must be unique for the entire domain.
- The **User logon name** is the userPrincipalName attribute, abbreviated as UPN. The UPN consists of a logon name and a UPN suffix which is, by default, the DNS name of the domain in which you create the object. The UPN must be unique for the entire forest. E-mail addresses, which must be unique for the whole world, certainly meet that requirement. Consider using e-mail addresses as UPNs. If your Active Directory domain name is not the same as your e-mail domain name, you must add the e-mail domain name as an available UPN suffix. To do this, open the Active Directory Domains and Trusts snap-in, right-click the root of the snap-in, and then click Properties.

- The **Name** of a user, which is shown in the first column in the details pane of the Active Directory Users and Computers snap-in, and also presented as Full Name in some interfaces, including the New Object–User dialog box. It must be unique in the OU. The Name field is actually the common name (CN), stored as the cn attribute. The cn must be unique in the OU because it is the first element of the distinguished name (DN), the distinguishedName attribute, which must be unique within the forest.
- The **display name** is the displayName attribute that appears in the Microsoft® Exchange global address list (GAL). It can be easier to locate users in the GAL if they are sorted by last name, so you can create a naming convention for your organization that specifies that the displayName attribute takes the LastName, FirstName syntax. There is no requirement for uniqueness of the displayName attribute, though it is certainly easier to locate users in the GAL if each has a unique display name!

Question: What do you do in your organization to ensure the uniqueness of name attributes, and what naming conventions do you use?

Additional Reading

- Object Names: <http://go.microsoft.com/fwlink/?LinkId=168745>

Rename a User Account

- In Active Directory Users and Computers:
 1. Right-click the user, and then click **Rename**.
 2. Type the new common name (CN), and press **Enter**.
 3. Type the **Full Name** (which maps to cn and name)
 4. Type the **First Name** and **Last Name**.
 5. Type the **Display Name**.
 6. **User Logon Name** and **User Logon Name (Pre-Windows 2000)**.
- `dsmod user UserDN [-upn UPN] [-fn FirstName] [-mi Initial] [-ln LastName] [-dn DisplayName] [-email EmailAddress]`
 - You cannot change the user logon names or CN with DSMod
- `dsmove user UserDN -newname "New CN"`

Key Points

When a user account needs to be renamed, there can be one or more attributes you must change.

To rename a user in the Active Directory Users and Computers snap-in:

1. Right-click the user, and then click **Rename**.
2. Type the new common name (CN) for the user, and press ENTER.
The Rename User dialog box appears and prompts you to enter additional name attributes.
3. Type the **Full Name** (which maps to the cn and name attributes)
4. Type the **First Name** and **Last Name**.
5. Type the **Display Name**.
6. Type the **User Logon Name** and **User Logon Name (Pre-Windows 2000)**.

From a command prompt, you can use the DSMod command with the following syntax:

```
dsmod user UserDN [-upn UPN][-fn FirstName][-mi Initial][-ln LastName]  
[-dn DisplayName][-email EmailAddress]
```

where *UserDN* is the distinguished name (DN) of the user object. Each parameter, for example *-dn*, is preceded by a dash and followed by the value to which the corresponding attribute will be configured.

You cannot change the *samAccountName* attribute by using DSMod, and you cannot change the CN of the object using DSMod.

You can use the DSMove command with the *-newname* parameter to change the CN of the object.

Account Attributes

- Logon Hours: logonHours
- Log On To: userWorkstations
- User must change password at next logon
- User cannot change password
- Password never expires
- Account is disabled
- Store password using reversible encryption
- Smart Card is required for interactive logon
- Account is trusted for delegation
- Account expires

Key Points

On the Account tab of a user's Properties dialog box, you can find the attributes that are directly related to the fact that a user is a security principal, meaning that it is an identity to which permissions and rights can be assigned.

The table below summarizes the account attributes.

Property	Description
Logon Hours	Click Logon Hours to configure the hours during which a user is allowed to log on to the network.
Log On To	Click Log On To if you want to limit the workstations to which the user can log on. This is called Computer Restrictions in other parts of the user interface and maps to the <i>userWorkstations</i> attribute. You must have NetBIOS over TCP/IP enabled to use this feature, because it uses the computer name rather than the Media Access Control (MAC) address of its network card to restrict logon.
User Must Change Password At Next Logon	Select this check box if you want the user to change the password you have entered the first time he or she logs on. You cannot select this option if you have selected Password Never Expires. Selecting this option will automatically clear the mutually exclusive User Cannot Change Password option.
User Cannot Change Password	Select this check box if you have more than one person using the same domain user account (such as Guest) or to maintain control over user account passwords. This option is commonly used to manage service account passwords. You cannot select this option if you have selected User Must Change Password At Next Logon.
Password Never Expires	Select this check box if you never want the password to expire. This option will automatically clear the User Must Change Password At Next Logon setting, because the two are mutually exclusive. This option is commonly used to manage service account passwords.
Account Is Disabled	Select this check box to disable the user account--for example, when creating an object for a newly hired employee who does not yet need access to the network.

(continued)

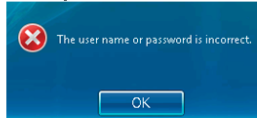
Property	Description
Store Password Using Reversible Encryption	This option, which stores the password in Active Directory without using Active Directory's powerful, nonreversible encryption hashing algorithm, exists to support applications that require knowledge of the user password. If it is not absolutely required, do not enable this option because it weakens password security significantly. Passwords stored using reversible encryption are similar to those stored as plaintext.
Smart Card Is Required For Interactive Logon	Smart cards are portable, tamper-resistant hardware devices that store unique identification information for a user. They are attached to, or inserted into, a system and provide an additional, physical identification component to the authentication process.
Account Is Trusted For Delegation	This option enables a service account to impersonate a user to access network resources on behalf of a user. This option is not typically selected, certainly not for a user object representing a human being. It is used more often for service accounts in three-tier (or multitier) application infrastructures.
Account Expires	Use the Account Expires controls to specify when an account expires.

Additional Reading

- User Properties - Account Tab:
<http://go.microsoft.com/fwlink/?LinkId=168746>

Reset a User's Password

- A user forgets his or her password and attempts to log on



- In Active Directory Users and Computers, right-click the user object and click **Reset Password**



- Best practices
 - Assign a temporary, unique, strong password to the user
 - Select **User must change password at next login**
 - Communicate the password to the user in a secure manner
- `dsmod user UserDN -pwd NewPassword -mustchpwd yes`

Key Points

If the user forgets his or her password and attempts to log on, he or she will receive a logon message.

Before the user can log on successfully, you will have to reset that password. You do not need to know the user's old password to do so.

To reset a user's password in the Active Directory Users and Computers snap-in:

1. Right-click the user object, and then click **Reset Password**.
The Reset Password dialog box appears.
2. Enter the new password in both the **New Password** and **Confirm Password** boxes.

It is a best practice to assign a temporary, unique, strong password for the user.

3. Select the **User Must Change Password At Next Logon** check box.

It is a best practice to force the user to change the password at the next logon, so that the user ends up with a password known only by the user.

4. Click **OK**.
5. Communicate the temporary password to the user in a secure manner.

You can also use the DSMod command to reset a user's password and, optionally, to force the user to change that password at the next logon. Type the following command:

```
dsmod user UserDN -pwd NewPassword -mustchpwd yes
```

where UserDN is the distinguished name (DN) of the user object and NewPassword is the new password. The -mustchpwd yes parameter forces the user to change the password at the next logon.



Tip: Configure highly complex passwords for service accounts. Services require credentials with which to access system resources. Many services require a domain user account with which to authenticate, and it is common to specify that the account password never expires. In such situations, be sure you use a long, complex password. If the service account is used by services on a limited number of systems, you can increase the security of the account by configuring the Log On To property with the list of systems using the service account.

Question: What are the security implications of administrators having the right to reset user passwords?

Question: Who should be able to reset the password for standard users? For accounts with administrative privileges? For service accounts?

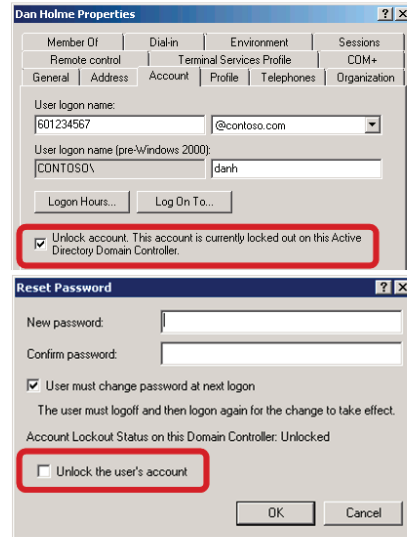
Question: What business practices for password reset are in place at your organization?

Additional Reading

- Reset a User Password: <http://go.microsoft.com/fwlink/?LinkId=168747>

Unlock a User Account

- In Active Directory Users and Computers, right-click the user object and click **Properties**. Click the **Account** tab, and then select **Unlock Account**.
- In the **Reset Password** dialog box, select **Unlock the user's account**.
- Watch out for drives mapped with alternate credentials. A leading cause of account lockout is when the alternate credentials' password changes.



Key Points

An Active Directory domain supports account lockout policies. A lockout policy is designed to prevent an intruder from attempting to penetrate the enterprise network by logging on repeatedly with various passwords until he or she finds a correct password. When a user attempts to log on with an incorrect password, a logon failure is generated. When too many logon failures occur within a specified period of time, defined by the lockout policy, the account is locked out. The next time the user attempts to log on, a notification clearly states the account lockout. You will learn to configure account lockout policies in Module 9.

Your lockout policy can define a period of time after which a lockout account is automatically unlocked. But when a user is trying to log on and discovers that he or she is locked out, it is likely he or she will contact the help desk for support.

To unlock a user account in the Active Directory Users and Computers snap-in:

1. Right-click the user object, and then click **Properties**.
2. Click the **Account** tab.
3. Select the **Unlock Account** check box.

Windows Server 2008 also adds the option to unlock a user's account when you choose the Reset Password command.

To unlock a user account while resetting the user's password:

- In the **Reset Password** dialog box, select the **Unlock the user's account** check box.

This method is particularly handy when a user's account has become locked out because the user did, in fact, forget the password. You can now assign a new password, specify that the user must change the password at next logon, and unlock the user's account in one dialog box.

Watch for drives mapped with alternate credentials: A common cause of account lockout is a drive mapped with alternate credentials. If the alternate credentials' password is changed, and the Windows client attempts repeatedly to connect to the drive, that account will be locked out.

Question: Other than forgotten passwords, have you experienced other scenarios that lead to account lockout?

Additional Reading

- Module 9 covers account lockout policies in detail.

Disable and Enable User Accounts

- In Active Directory Users and Computers, right-click the user object and click **Disable Account** or **Enable Account**
- `dsmod user UserDN -disabled {yes|no}`

Key Points

User accounts are security principals—identities that can be given access to network resources. Because each user is a member of Domain Users and of the Authenticated Users special identity, each user account has at least read access to a vast amount of information in Active Directory and on your file systems unless you have been disciplined and unusually successful at locking down access control lists (ACLs).

Therefore, it is important not to leave user accounts open. That means you should configure password policies and auditing—both discussed in other modules—and procedures to ensure that accounts are being used appropriately.

If a user account is provisioned before it is needed, or if an employee will be absent for an extended period of time, disable the account.

To disable an account in the Active Directory Users and Computers snap-in:

- Right-click a user and then click **Disable Account**.

If an account is already disabled, the Enable Account command will appear when you right-click the user.

From the command prompt, you can use the DSMod command, as in the following example:

```
dsmod user UserDN -disabled yes
```

Enabling an account is just a matter of changing *yes* to *no* for the DSMod command:

```
dsmod user UserDN -disabled no
```

In each command, UserDN is the distinguished name (DN) of the user object, and the *-disabled {yes|no}* parameter disables or enables the account.

Question: What business practices for disabling and enabling accounts are in place in your organization?

Question: What are the security implications of someone having the right to disable or enable user accounts?

Question: Under what circumstances would you disable a user account rather than delete it?

Additional Reading

- Disable or Enable a User Account:
<http://go.microsoft.com/fwlink/?LinkId=168748>

Delete a User Account

- In Active Directory Users and Computers, select the user and press **Delete** or right-click the user object and click **Delete**
 - `dsrm UserDN`
- When you delete an account, you lose
 - The group memberships
 - The security identifier (SID)
- Common practice
 - Disable the account and move it to an OU for disabled objects
 - After a period of time, delete the account

Key Points

When an account is no longer necessary, you can delete it from your directory.

To delete a user account in Active Directory Users and Computers:

1. Select the user and press **Delete** or right-click the user and then click **Delete**.
You are prompted to confirm your choice because of the significant implications of deleting a security principal.
2. Confirm the prompt.

You can delete objects from Active Directory by using the DSRm command, another of the DS commands. DSRm uses a simple syntax:

```
dsrm UserDN
```

where *UserDN* is the distinguished name (DN) of the user object. Notice that, unlike other DS commands, DSRm is not followed by the user object class.

It is critical to consider that once the account has been deleted, it is eventually purged entirely from the directory. You cannot simply re-create a new account with the same name as a deleted account and hope it has the same group memberships and access to resources; it will not. The loss of the user's SID and of its group memberships can cause significant problems if, later, you realize you need the account.

Therefore, many organizations choose to decommission a user account in stages. First, the account is disabled. After a period of time, it is deleted. Active Directory actually maintains a subset of the account's properties—most notably its SID—for a period of time called the *tombstone lifetime*, which is 180 days by default. After that time, the account's record is removed from the directory.

You can also consider recycling a user account. If a user leaves your organization, it's possible you will eventually hire a replacement who will need very similar resource access, group memberships, and user rights as the previous user. You can disable the account until a replacement is found, then rename the account to match the new user's name. The previous user's SID, group memberships, and resource access are thereby transferred to the replacement.

Question: What are the business practices related to decommissioning a user account in your organization?

Additional Reading

- Delete a User Account: <http://go.microsoft.com/fwlink/?LinkId=168749>

Move a User Account

- In Active Directory Users and Computers, right-click the user and then click **Move**
or
drag the user object and drop it onto the destination OU
- `dsmove UserDN -newparent TargetOUDN`

Key Points

To move a user object in the Active Directory Users and Computers snap-in:

1. Right-click the user and then click **Move**.
2. Click the folder to which you want to move the user account. Then click **OK**.

Alternately, you can

- Drag the user object and drop it onto the destination OU.

To move a user with a command-line tool, use DSMove. DSMove uses the following syntax:

```
dsmove UserDN -newparent TargetOUDN
```

The DSMove command does not specify the user object class. Instead, it simply indicates the distinguished name (DN) of the user to move and, in the *TargetOUDN* placeholder, the DN of the OU to which the user will be moved.

Consider that when you move a user, you might change the Group Policy objects (GPOs) that apply to that user. GPOs are discussed in a later module.

You can use the `DSMove` command with the *-newname* switch to change the common name (CN) of the object.

Additional Reading

- Move a User Account: <http://go.microsoft.com/fwlink/?LinkId=168750>

Lab A: Create and Administer User Accounts

- Exercise 1: Create User Accounts
- Exercise 2: Administer User Accounts
- Exercise 3 (Advanced Optional): Explore User Account Name Attributes

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

You are the administrator of Contoso, Ltd., an online university for adult education. Two new employees have been hired: Chris Mayo and Amy Strande. You must create accounts for these users. As time passes, Chris Mayo leaves the organization, and his account must be administered according to the company policy for user account lifecycle management.

Exercise 1: Create User Accounts

In this exercise, you will create user accounts with both the Active Directory Users and Computers snap-in and the command prompt.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Create a user account with Active Directory Users and Computers.
3. Create a user account with the DSAdd command.

► Task 1: Prepare for the lab

- Start 6425B-HQDC01-A.
- Log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
- Run **D:\Labfiles\Lab03b\Lab03a_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

► Task 2: Create a user account with Active Directory Users and Computers

- Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
- Create a user account for Chris Mayo in the **Employees** OU.
 - First Name: **Chris**
 - Last Name: **Mayo**
 - User Logon Name: **Chris.Mayo**
 - User Logon Name (Pre-Windows 2000): **Chris.Mayo**
 - Password: **Pa\$\$w0rd**
 - Specify that he must change the password at the next logon

► **Task 3: Create a user account with the DSAdd command**

- Run Command Prompt with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
- At the command prompt, create a user account for Amy Strande in the **Employees OU**.
 - First Name: **Amy**
 - Last Name: **Strande**
 - User Principal Name: **Amy.Strande@contoso.com**
 - User Logon Name (Pre-Windows 2000): **Amy.Strande**
 - Display Name: **Strande, Amy**
 - Description: **Vice President, IT**
- In **Active Directory Users and Computers**, open the properties of the user account you just created and confirm that the attributes were set correctly.

Results: After this exercise, you will have user accounts named Chris Mayo and Amy Strande in the Employees OU.

Exercise 2: Administer User Accounts

In this exercise, you will perform common tasks that support user accounts through their lifecycle in Active Directory.

The main tasks for this exercise are as follows:

1. Administer a user account.
2. Administer the lifecycle of a user account.

► Task 1: Administer a user account

The user account for Amy Strande is currently disabled, because no password was specified using the DSAdd command.

1. What parameter could you have used with the DSAdd command to specify a password?
2. In **Active Directory Users and Computers**, reset the password for **Amy Strande** to **Pa\$\$w0rd**, and specify that she must change the password at the next logon.
3. In **Active Directory Users and Computers**, enable Amy Strande's user account.
4. What command could have been used at the command prompt to reset the password, specify that the password must be changed at the next logon, and enable the account? Write the command, including all of the parameters.

Results: After this exercise, Amy Strande's account will be enabled.

► Task 2: Administer the lifecycle of a user account

1. Contoso's policy for user account lifecycle management states the following:
 - When a user leaves the organization for any reason, including leave of absence, the user's account must be disabled immediately and moved to the Disabled Accounts OU.
 - Sixty days after the termination of a user, the user's account must be deleted.
2. Chris Mayo has left Contoso, Ltd. Disable his account and move it to the Disabled Accounts OU.

3. It has been 60 days since you disabled Chris Mayo and company procedures specify that after 60 days, a disabled user account must be deleted. Delete the user account for Chris Mayo.
4. Log off of HQDC01.

Results: After this exercise, Chris Mayo's account will have been deleted.



Note: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in the Lab B.

Exercise 3 (Advanced Optional): Explore User Account Name Attributes

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

The main tasks for this exercise are as follows:

1. Create a sample user account. In the **Full Name** box, type the user's name using the format *LastName, FirstName*.
2. Look at the display of the user in the Active Directory Users and Computers details pane. You should see the user listed as *LastName, FirstName*.
3. In the properties of the user object, click the **Attribute Editor** tab and examine the actual value of the cn attribute.
4. Use the Active Directory Schema snap-in to examine the sAMAccountName attribute. What is its schema-defined length limit?
5. Attempt to create a user with a 30-character name in the **User logon name (pre-Windows 2000)** box. Experiment to determine the maximum length of the sAMAccountName attribute. Active Directory restricts the sAMAccountName attribute for user objects to a length that is significantly shorter than the schema-defined length.

Lab Review Questions

Question: In this lab, which attribute(s) can be modified when you are creating a user account with the command prompt that cannot be modified when creating a user account with Active Directory Users and Computers?

Question: What happens when you create a user account that has a password that does not meet the requirements of the domain?

Lesson 2

Configure User Object Attributes

- Demonstration: A Tour of User Attributes
- View All Attributes
- Modify Attributes of Multiple Users
- Modify User Attributes with DSMod and DSGet
- Demonstration: Create Users with Templates
- Create Users with Templates

Key Points

A user object in Active Directory is far more than just a handful of properties related to the user's security identity, or account. A user object includes attributes that describe the individual and his or her relationship with the organization, as well as contact information and configuration of the user's experience on his or her computer. In this lesson, you will explore many of the more useful attributes of user objects, and you will learn how to administer these attributes for one or more users.

Objectives

After completing this lesson, you will be able to:

- View and modify hidden attributes of user objects.
- Identify the purpose and requirements of user object attributes.

- Modify attributes of multiple users simultaneously.
- Manage user attributes from the command prompt.
- Create users from user account templates.

Demonstration: A Tour of User Attributes

In this demonstration, you will learn:

- How to access the properties of a user
- The role of each tab in the user Properties dialog box

Key Points

When you create a user with the Active Directory Users and Computers snap-in's New Object – User Wizard, you are prompted for some common properties, including logon names, passwords, and user first and last names. A user object in Active Directory, however, supports dozens of additional properties that you can configure at any time with the Active Directory Users and Computers snap-in.

To read and modify the attributes of a user object, right-click the user and then click Properties.

Tony Krijnen Properties

Member Of | Dial-in | Environment | Sessions
 Remote control | Terminal Services Profile | COM+
General | Address | Account | Profile | Telephones | Organization

Tony Krijnen

First name: Tony Initials:

Last name: Krijnen

Display name: Krijnen, Tony

Description: Sales Representative in The Netherlands

Office: Amsterdam

Telephone number: 12.345.6789 Other...

E-mail: tony.krijnen@contoso.com

Web page: www.contoso.com Other...

OK Cancel Apply Help

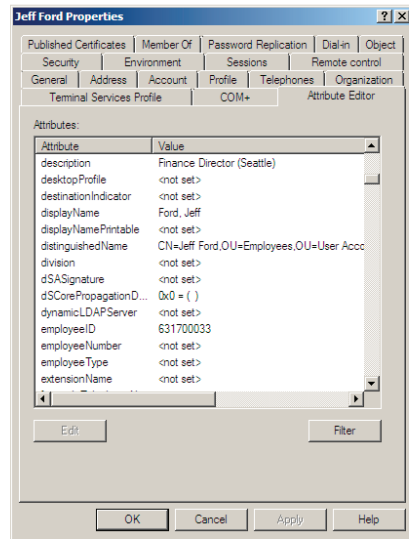
Attributes of a user object fall into several broad categories that appear on tabs of the dialog box.

- **Account attributes: The Account tab.** These properties include logon names, passwords, and account flags. Many of these attributes can be configured when you create a new user with the Active Directory Users and Computers snap-in. The Account Properties section details account attributes.

- **Personal information: The General, Address, Telephones, and Organization tabs.** The General tab exposes the name properties that are configured when you create a user object, as well as basic description and contact information. The Address and Telephones tabs provide detailed contact information. The Telephones tab is also where Microsoft chose to put the Notes field, which maps to the info attribute and is a very useful general-purpose text field that is underused by many enterprises. The Organization tab shows job title, department, company, and organizational relationships.
- **User configuration management: The Profile tab.** Here you can configure the user's profile path, logon script, and home folder.
- **Group membership: The Member Of tab.** You can add the user to and remove the user from groups and change the user's primary group. Group memberships and the primary group will be discussed in another module.
- **Terminal services: The Terminal Services Profile, Environment, Remote Control, and Sessions tabs.** These four tabs enable you to configure and manage the user's experience when the user is connected to a Terminal Services session.
- **Remote access: The Dial-in tab.** You can enable and configure remote access permission for a user on the Dial-in tab.
- **Applications: The COM+ tab.** This tab enables you to assign the user to an Active Directory COM+ partition set. This feature facilitates the management of distributed applications.

View All Attributes

- The Attribute Editor
- In Active Directory Users and Computers, click the View menu, then select Advanced Features



Key Points

The Attribute Editor allows you to view and edit all attributes of a user object. The Attribute Editor tab is not visible until you enable Advanced Features from the View menu of the Microsoft Management Console (MMC).

The Attribute Editor displays all the system attributes of the selected object. The Filter button enables you to choose to see even more attributes, including backlinks and constructed attributes.

Backlinks are attributes that result from references to the object from other objects. The easiest way to understand backlinks is to look at an example: the memberOf attribute. When a user is added to a group, it is the group's member attribute that is changed: The distinguished name of the user is added to this multivalued attribute. Therefore, the member attribute of a group is called a forward link attribute. A user's memberOf attribute is updated automatically by Active Directory when the user is referred to by a group's member attribute. You do not ever write directly to the user's memberOf attribute—it is dynamically maintained by Active Directory.

A constructed attribute is one of the results from a calculation performed by Active Directory. An example is the tokenGroups attribute. This attribute is a list of the security identifiers (SIDs) of all the groups to which the user belongs, including nested groups. To determine the value of tokenGroups, Active Directory must calculate the effective membership of the user, which takes a few processor cycles. Therefore, the attribute is not stored as part of the user object or dynamically maintained. Instead, it is calculated when needed. Because of the processing required to produce constructed attributes, the Attribute Editor does not display them by default. They also cannot be used in Lightweight Directory Access Protocol (LDAP) queries.

Question: Are you using any of the hidden attributes in your organization? If so, how do you interact with those attributes (read them and modify them)?

Modify Attributes of Multiple Users

- How to do it
 - Select multiple users (for example, by using CTRL+click)
 - Right-click any one of the selected users, and then click **Properties**
- Attributes that can be modified
 - **General:** Description, Office, Telephone Number, Fax, Web Page, E-mail
 - **Account:** UPN Suffix, Logon Hours, Computer Restrictions (logon workstations), all Account Options, Account Expires
 - **Address:** Street, P.O. Box, City, State/Province, ZIP/Postal Code, Country/Region
 - **Profile:** Profile Path, Logon Script, Home Folder
 - **Organization:** Title, Department, Company, Manager

Key Points

The Active Directory Users and Computers snap-in enables you to modify the properties of multiple user objects simultaneously.

To modify attributes of multiple users in the Active Directory Users and Computers snap-in:

1. Select several user objects by holding the CTRL key as you click each user, or by using any other multiselection technique.
Be certain that you select only objects of one class, such as users.
2. After you have multiselected the objects, right-click any one of them and then click **Properties**.

When you have multiselected the user objects, a subset of properties is available for modification.

- General: Description, Office, Telephone Number, Fax, Web Page, E-mail
- Account: UPN Suffix, Logon Hours, Computer Restrictions (logon workstations), all Account Options, Account Expires
- Address: Street, P.O. Box, City, State/Province, ZIP/Postal Code, Country/Region
- Profile: Profile Path, Logon Script, Home Folder
- Organization: Title, Department, Company, Manager

Manage User Attributes with DSMod and DSGet

- DSMod *modifies* the attributes of object(s)


```
dsmod user UserDN... [-parameter value ...]
```

 - **UserDN** distinguishedName of the user(s) to modify
 - **Parameter.** Attribute to modify. **dsmod user /?**
 - Often does not map to the same name as LDAP (dsmod dept vs. LDAP department)
- DSGet *gets* (returns) the value of attributes of object(s)


```
dsget user UserDN... [-parameter ...]
```

 - dsget user /?
- DSQuery can return objects based on search criteria and *pipe* those objects to DSGet and DSMod


```
dsquery user -desc "Marketing Task Force" | dsget user -email
```

Key Points

The DSMod and DSGet commands are two Active Directory command-line tools, which are also called DS commands.

DSMod

DSMod modifies the attributes of one or more existing objects. The basic syntax of DSMod is:

```
dsmod user UserDN... [-parameter value...]
```

The *UserDN* parameter specifies the distinguished name of the user that will be modified. The remaining parameters indicate the attribute to change and the new value. For example, the following command changes the office attribute of Tony Krijnen:

```
dsmod user "cn=Tony Krijnen,ou=Employees,OU=User  
Accounts,dc=contoso,dc=com" -office "Stockholm"
```

The attribute parameters do not map directly to the names of LDAP attributes of a user object. For example, the *-dept* parameter of the DSMod User command modifies the department attribute of a user object. Additionally, DSMod User can modify only a subset of user attributes. Type **dsmod user /?** for usage information and a list of supported parameters.

Piping Multiple DN's to DSMod

The *UserDN* parameter of the DSMod command does not have to be entered directly into the command prompt. There are two ways to pipe DN's to it. The first is to enter the DN's into the console. Let's assume that you need to change the office attribute of two users, Linda Mitchell and Scott Mitchell, to reflect their relocation to the Sydney office. At the command prompt, type the following command:

```
dsmod user -office "Sydney"
```

The *UserDN* parameter is missing. The console (the command prompt) waits for you to enter DN's of users. Enter one per line, surrounded with quotes, pressing ENTER at the end of each DN. After entering the last DN and pressing ENTER, press CTRL+Z at the beginning of the next line and then press ENTER, to indicate that you are finished. The command will then execute against each of the DN's you have entered.

A more sophisticated way to send DN's to the DSMod command is by piping the results of a DSQuery command. DSQuery searches Active Directory for specified criteria and returns the DN's of matching objects. For example, to change the office attribute of Linda and Scott Mitchell's accounts to Sydney, use the following command:

```
dsquery user -name "* Mitchell" | dsmod user -office "Sydney"
```

The DSQuery User command searches Active Directory for users whose names end with Mitchell. The resulting objects' DN's are then piped to DSMod User, which changes the office attribute to Sydney.

As another example, assume you want to assign all users a home folder on SERVER01. The following command changes the homeDirectory and homeDrive attributes of user objects in the User Accounts OU:

```
dsquery user "ou= User Accounts,dc=contoso,dc=com" | dsmod user -hmdir "\\server01\users\%username%\documents" -hmdrv "U:"
```

The special `$username$` token can be used to represent the `sAMAccountName` of user objects when using DS commands to configure the value of the `-email`, `-hmdir`, `-profile`, and `-webpg` parameters.

DSGet

The `DSGet` command gets and outputs selected attributes of one or more objects. Its syntax, like that of `DSMod`, is:

```
dsget user UserDN... [-parameter...]
```

You can supply the DNs of one or more user objects by specifying them in the command, separated by spaces, by entering them in the console, or by piping the results of a `DSQuery User` command. Unlike `DSMod`, `DSGet` takes only a parameter and not an associated value. For example, `DSGet` takes the `-samid` parameter like `DSMod` does, but it does not take a value. Instead, it reports the current value of the attribute. For example, to display the pre-Windows 2000 logon name of Jeff Ford in the Employees OU, use the following command:

```
dsget user "cn=Jeff Ford,ou= Employees,ou=User  
Accounts,dc=contoso,dc=com" -samid
```

To display the email addresses of all users whose description attribute indicates that they are in the Sydney office, use this command:

```
dsquery user -desc "*Sydney*" | dsget user -email
```

Demonstration: Create Users with Templates

In this demonstration you will learn:

- What a template user account is, and why it is useful
- How to create a template user account
- How to copy a template user account

Key Points

Users in a domain often share many similar properties. For example, all sales representatives can belong to the same security groups, log on to the network during similar hours, and have home folders and roaming profiles stored on the same server. When you create a new user, you can simply copy an existing user account rather than creating a blank account and populating each property.

Since the days of Windows NT 4.0, Windows has supported the concept of user account templates. A user account template is a generic user account prepopulated with common properties. For example, you can create a template account for sales representatives that is preconfigured with group memberships, logon hours, a home folder, and roaming profile path.

To create a user account template:

1. Create a user account and prepopulate appropriate attributes.



Tip: Use a naming standard that makes templates easy to find. For example, set the full name to begin with an underscore (_), as in _Sales User. The underscore will cause all templates to appear at the top of the list of users in an OU.

2. Disable the template user account.

The template account itself should not be used to log on to the network, so be sure to disable the account.

To create a user based on the template:

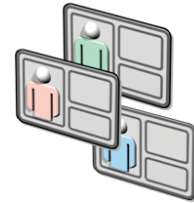
1. Right-click the template user account and then click **Copy**.
The Copy Object – User Wizard appears.
2. In **First name**, type the user's first name.
3. In **Last name**, type the user's last name.
4. Modify the **Full name** value if necessary.
5. In **User logon name**, type the user logon name, then select the appropriate user principal name (UPN) suffix in the drop-down list.
6. In **User logon name (pre-Windows 2000)**, type the user's pre-Windows 2000 username.
7. Click **Next**.
8. In **Password** and **Confirm password**, type the user's password.
9. Select the appropriate password options.
10. If the user account from which the new user account was copied was disabled, clear **Account is disabled** to enable the new account.

Additional Reading

- Copy a User Account: <http://go.microsoft.com/fwlink/?LinkId=168751>

Create Users with Templates

- **General tab.** No properties are copied
- **Address tab.** P.O. box, city, state or province, ZIP or postal code, and country or region are copied
 - Note that the street address itself is not copied
- **Account tab.** Logon hours, logon workstations, account options, and account expiration
- **Profile tab.** Profile path, logon script, home drive, and home folder path
- **Organization tab.** Department, company, and manager
- **Member Of tab.** Group membership and primary group



Key Points

It's important to realize that not all attributes are copied. The list below summarizes the attributes that are copied. It is not useful to configure any other attributes in the template, as they will not be copied.

- **General tab.** No properties are copied from the General tab.
- **Address tab.** P.O. box, city, state or province, ZIP or postal code, and country or region are copied. Note that the street address itself is not copied.
- **Account tab.** Logon hours, logon workstations, account options, and account expiration are copied.
- **Profile tab.** Profile path, logon script, home drive, and home folder path are copied.
- **Organization tab.** Department, company, and manager are copied.
- **Member Of tab.** Group membership and primary group are copied.



Note: There are other attributes that are copied that are not even visible in the user Properties dialog box. These attributes include assistant, division, and employee type.

Question: What other methods do you use to create new user accounts with common attributes?

Lab B: Configure User Object Attributes

- Exercise 1: Examine User Object Attributes
- Exercise 2: Manage User Object Attributes
- Exercise 3: Create Users from a Template
- Exercise 4 (Advanced Optional): Create Users with a Batch File

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

You are the administrator of Contoso, Ltd., an online university for adult education. Changes in the Sales department require you to modify attributes of Sales users. Additionally, you decide to make it easier to create new accounts for salespeople by preparing a user account template.

Exercise 1: Examine User Object Attributes

In this exercise, you will examine the attributes of a user object.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Explore the properties of an Active Directory user object.
3. Explore all attributes of an Active Directory user object.
4. Analyze the naming and display of user object attributes.

► Task 1: Prepare for the lab

The virtual machine should already be started and available after completing Lab A. However, if it is not, you should launch it complete the exercises in Lab A before continuing.

- Start 6425B-HQDC01-A.
- Log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
- Run **D:\Labfiles\Lab03b\Lab03b_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

► Task 2: Explore the properties of an Active Directory user object

- Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
- Open the properties of **Tony Krijnen** in the **Employees** OU.
- In this sample contoso.com domain, attributes have been configured on the **General**, **Address**, **Account** and **Organization** tabs. Examine each of these tabs, then close the **Properties** dialog box.

► Task 3: Explore all attributes of an Active Directory user object

- Enable the **Advanced Features** view of the **Active Directory Users and Computers** snap-in.
- Examine the **Attribute Editor** tab of Tony Krijnen's **Properties** dialog box.

► **Task 4: Analyze the naming and display of user object attributes**

- For each of the following attributes in the **Tony Krijnen Properties** dialog box, identify the corresponding attribute name on the **Attribute Editor** tab:

Properties dialog box tab	Property name	Attribute name as shown on the Attribute Editor tab
General	First name	
General	Last name	
General	Display name	
General	Description	
General	Office	
General	Telephone number	
General	E-mail	
Address	Street	
Address	City	
Address	ZIP/Postal Code	
Address	Country	
Organization	Job Title	
Organization	Department	
Organization	Company	

Questions:

1. Use the Attribute Editor tab to answer the following questions.
 - Does the employeeID attribute, shown on the Attribute Editor tab, show up on a normal tab of the Properties dialog box? If so, which one? What about carLicense?
 - Looking at the Attribute Editor tab, what is the distinguished name (DN) of Tony Krijnen's object?
 - Looking at the Attribute Editor tab, what is Tony's user principal name (UPN)? On which other tab does the attribute appear, and how is it labeled and displayed?
2. Thought questions: Try to answer the following questions. However, it is possible that you may not come up with an answer. That is OK. Once you've tried to think of an answer, you can look at the Lab Answer Key.
 - Why might the sn attribute be named sn?
 - What is the use of the c attribute?

Exercise 2: Manage User Object Attributes

In this exercise, you will manage the attributes of user objects.

The main tasks for this exercise are as follows:

1. Modify the attributes of multiple user objects.
2. Manage user attributes from the command prompt.

► Task 1: Modify the attributes of multiple user objects

A special Marketing Task Force has been established by Ariane Berthier, the Vice President of Marketing. Members of the task force are being relocated to Headquarters and will report directly to Ariane.

- Select the following users in the **Employees** OU: **Adam Barr**, **Adrian Lannin**, **Ajay Manchepalli**, **Ajay Solanki**, **Allan Guinot**, **Anav Silverman** and **András Tóth**.
- Configure the following properties for the users:
 - Office: **Headquarters**.
 - Description: **Marketing Task Force**.
 - Manager: **Ariane Berthier**.
- After changing the attributes, open the properties of Adam Barr and examine the attributes you just changed.
- The **Manager** attribute is a linked attribute. The other side of the link is the **Direct Reports** attribute. Open the properties of Ariane Berthier and examine the **Direct Reports**.

► Task 2: Manage user attributes from the command prompt

- Open the Command Prompt with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
- Use the DS commands to list the e-mail addresses of all users in the Marketing Task Force.



Tip: Users in the Marketing Task Force share a common Description property.

- Use the DS commands to configure the home folder for all users in the Marketing Task Force, so that they each have a U: drive that maps to `\\FILE01\\TaskForceUsers\\username`, where *username* is each user's unique logon name.
- In **Active Directory Users and Computers**, confirm that the changes you made were applied correctly by examining the properties of **Adam Barr**.

Exercise 3: Create Users from a Template

In this exercise, you will create a user account template and then generate a new user account based on that template.

The main tasks for this exercise are as follows:

1. Create a user account template for Sales.
2. Create a new user account based on a template.

► Task 1: Create a user account template for Sales

- In the **Employees** OU, create a template account for new sales people with the following properties:
 - First Name and Last Name: blank.
 - Full Name: **_Sales User** (note the underscore at the beginning of the name).
 - User Logon Name: **Template.Sales**.
 - Password: **Pa\$\$w0rd**.
 - User must change password at next logon.
 - Account is disabled.
 - Member of: Sales.
 - Department: Sales.
 - Company: Contoso, Ltd.
 - Manager: **Anibal Sousa**.
 - Account Expires: **last day of the current year**.

► Task 2: Create a new user account based on a template

- In the **Employees** OU, create an account for a new sales person based on the **_Sales User** template. The account should have the following properties:
 - First Name: **Rob**.
 - Last Name: **Young**.
 - User logon name: **Rob.Young**.

- Password: **Pa\$\$w0rd**.
- Account is enabled.

Results: After this exercise, you will have a user account named Rob Young in the Employees OU. The account will have all of the attributes you configured for the _Sales User template.

Exercise 4 (Advanced Optional): Create Users with a Batch File

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

The main tasks for this exercise are as follows:

1. Create a script called `User_Provision.bat` that uses `DSAdd` to provision a user in the Employees OU. The goal is to be able to run the script with two parameters: first name and last name. The batch file should take these two parameters and create a user account with the following attributes:
 - The first name and last name as defined in the parameters in the command line.
 - The name in the format *FirstName LastName*.
 - The `sAMAccountName` in the format *FirstName.LastName*.
 - The `userPrincipalName` in the format *FirstName.LastName@contoso.com*.
 - The e-mail address in the same format as the UPN.
 - The `displayName` in the format *LastName, FirstName*.
 - An initial password of **Pa\$\$w0rd** that the user must change at first login.



Tip: when you run a batch script with parameters, the batch script can refer to the first parameter as `%1` and the second parameter as `%2`.

2. Run the Command Prompt with administrative credentials and test the script to create a sample user account. Confirm that the user is created successfully and all attributes are populated according to the specifications shown above.
3. Compare your results to `D:\Labfiles\Lab03b\User_Provision.bat`.



Note: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in the Lab C.

Lab Review Questions

Question: What options have you learned for modifying attributes of new and existing users?

Question: What are the advantages and disadvantages of each?

Lesson 3

Automate User Account Creation

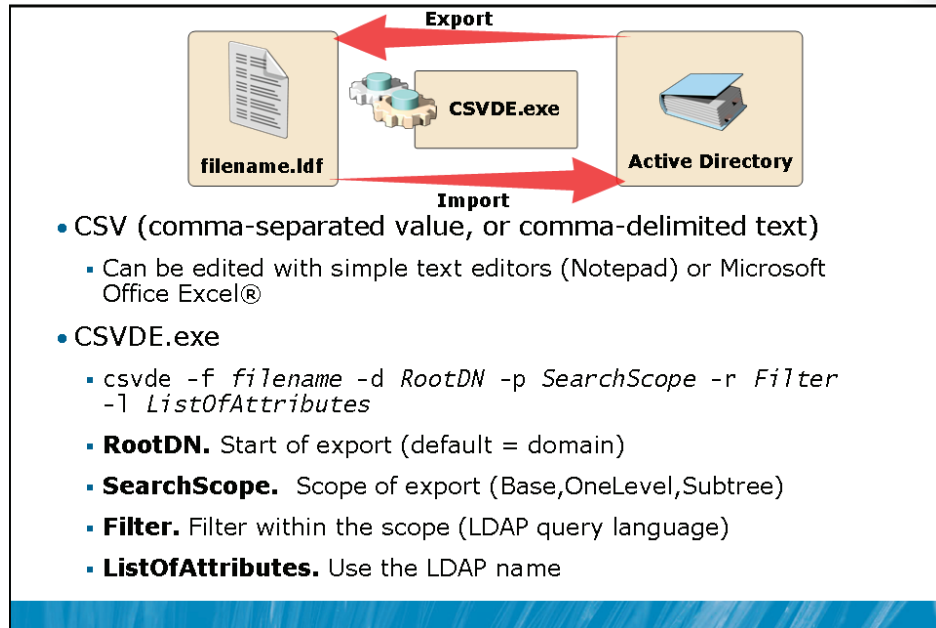
- Export Users with CSVDE
- Import Users with CSVDE
- Import Users with LDIFDE

Although the procedures discussed in Lessons 1 and 2 can be applied to create a small number of users, you will need more advanced techniques to automate the creation of user accounts when a large number of users must be added to the domain. In this lesson, you will learn several of these techniques.

After completing this lesson, you will be able to:

- Export user attributes with CSVDE
- Import users with CSVDE
- Import users with LDIFDE

Export Users with CSVDE



Key Points

CSVDE is a command-line tool that exports or imports Active Directory objects to or from a comma-delimited text file (also known as a comma-separated value text file, or .csv file). Comma-delimited files can be created, modified, and opened with tools as familiar as Notepad and Microsoft Office Excel®.

The basic syntax of the CSVDE command for export is:

```
csvde -f filename
```

However, that command will export all objects in your Active Directory domain. You will want to limit the scope of the export, which you can do with the following four parameters:

- **-d RootDN.** Specifies the distinguished name of the container from which the export will begin. The default is the domain itself.

- **-p SearchScope.** Specifies the scope of the search relative to the container specified by *-d*. *SearchScope* can be either *base* (this object only), *onelevel* (objects within this container), or *subtree* (this container and all subcontainers). The default is *subtree*.
- **-r Filter.** Filters the objects returned within the scope configured by *-d* and *-p*. Filter is a Lightweight Directory Access Protocol (LDAP) query syntax. You will work with a filter in the lab for this lesson. LDAP query syntax is beyond the scope of this course. See <http://go.microsoft.com/fwlink/?LinkId=168752> for more information.
- **-l ListOfAttributes.** Specifies the attributes that will be exported. Use the LDAP name for each attribute, separated by a comma, as in
-l DN,objectClass,sAMAccountName,sn,givenName,userPrincipalName

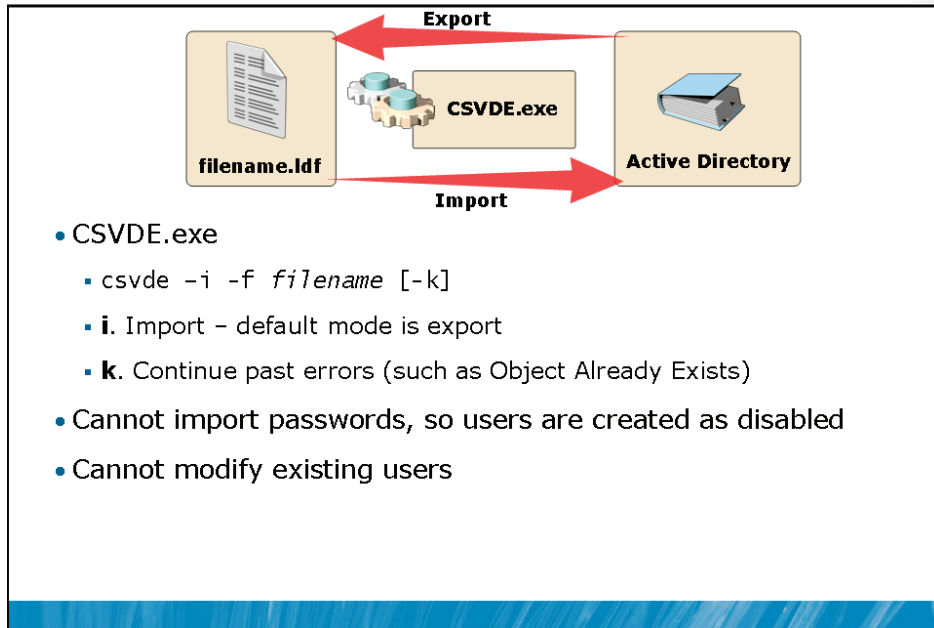
The output of a CSVDE export lists the LDAP attribute names on the first line. Each object follows, one per line, and must contain exactly the attributes listed on the first line. Here's a sample file:

```
DN,objectClass,sn,givenName,sAMAccountName,userPrincipalName

"CN=David Jones,OU=Employees,OU=User
Accounts,DC=contoso,DC=com",user,Jones,David,david.jones,david.jones@contoso.com

"CN=Lisa Andrews,OU=Employees,OU=User
Accounts,DC=contoso,DC=com",user,Andrews,Lisa,lisa.andrews,lisa.andrews@contoso.com
```

Import Users with CSVDE



Key Points

CSVDE can also create user accounts by importing a .csv file. If you have user information in existing Excel or Microsoft Office Access® databases, you will find that CSVDE is a powerful way to take advantage of that information to automate user account creation.

The basic syntax of the CSVDE command for import is:

```
csvde -i -f filename -k
```

The `-i` parameter specifies import mode; without it, the default mode of CSVDE is export. The `-f` parameter identifies the file name to import from or export to. The `-k` parameter is useful during import operations because it instructs CSVDE to ignore errors including Object Already Exists

The import file itself is a comma-delimited text file (.csv or .txt) in which the first line defines the imported attributes by their LDAP attribute names. Each object follows, one per line, and must contain exactly the attributes listed on the first line. Here's a sample file:

```
DN,objectClass,sn,givenName,sAMAccountName,userPrincipalName

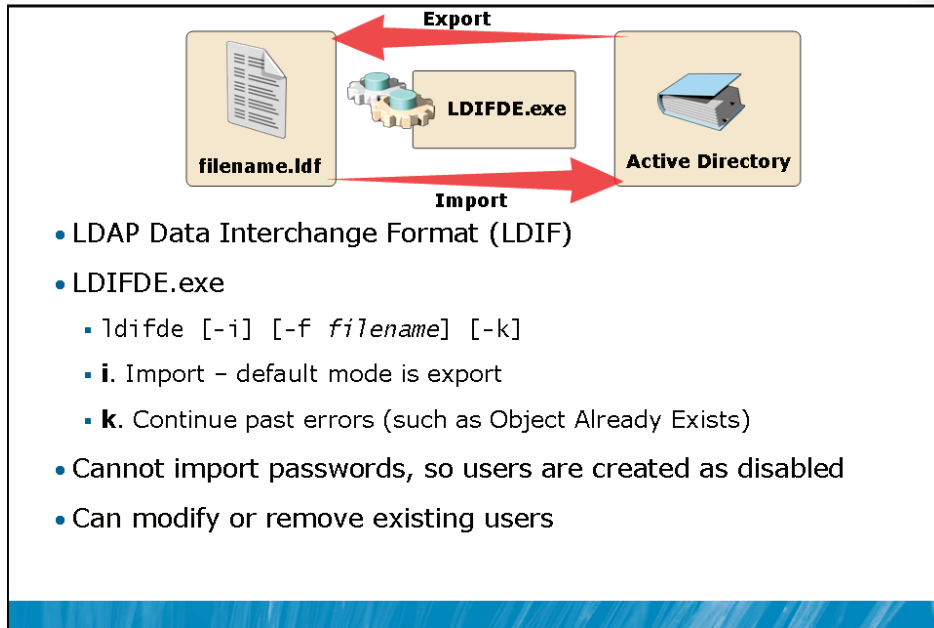
"CN=David Jones,OU=Employees,OU=User
Accounts,DC=contoso,DC=com",user,Jones,David,david.jones,david.jones@contoso.com

"CN=Lisa Andrews,OU=Employees,OU=User
Accounts,DC=contoso,DC=com",user,Andrews,Lisa,lisa.andrews,lisa.andrews@contoso.com
```

This file, when imported by the CSVDE command, will create a user object for Lisa Andrews in the Employees OU. The user logon names, last name and first name, are configured by the file. You cannot use the CSVDE to import passwords, and without a password, the user account will be disabled initially. After you have reset the password, you can enable the object.

For more information about CSVDE, including details regarding its parameters and usage to export directory objects, type **csvde /?** or search the Windows Server 2008 Help and Support Center.

Import Users with LDIFDE



Key Points

You can also use LDIFDE.exe to import or export Active Directory objects, including users. LDIF is a draft Internet standard for file format that can be used to perform batch operations against directories that conform to the LDAP standards. LDIF supports both import and export operations as well as batch operations that modify objects in the directory. The LDIFDE command implements these batch operations by using LDIF files.

The LDIF file format consists of a block of lines that, together, constitute a single operation. Multiple operations in a single file are separated by a blank line. Each line comprising an operation consists of an attribute name followed by a colon and the value of the attribute. For example, suppose you wanted to import user objects for two sales representatives named Bonnie Kearney and Bobby Moore. The contents of the LDIF file would look similar to the following example:

```
dn: CN=Bonnie Kearney,OU=Employees,OU=User Accounts,DC=contoso,DC=com
changetype: add
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: user
cn: Bonnie Kearney
sn: Kearney
title: Operations
description: Operations (London)
givenName: Bonnie
displayName: Kearney, Bonnie
company: Contoso, Ltd.
sAMAccountName: bonnie.kearney
userPrincipalName: bonnie.kearney@contoso.com
mail: bonnie.kearney@contoso.com

dn: CN=Bobby Moore,OU=Employees,OU=User Accounts,DC=contoso,DC=com
changetype: add
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: user
cn: Bobby Moore
sn: Moore
title: Legal
description: Legal (New York)
givenName: Bobby
displayName: Moore, Bobby
company: Contoso, Ltd.
sAMAccountName: bobby.moore
userPrincipalName: bobby.moore@contoso.com
mail: bobby.moore@contoso.com
```

Each operation begins with the DN attribute of the object that is the target of the operation. The next line, `changetype`, specifies the type of operation: add, modify, or delete.

As you can see, the LDIF file format is not as intuitive or familiar as the comma-separated text format. However, because the LDIF format is also a standard, many directory services and databases can export LDIF files.

After creating or obtaining an LDIF file, you can perform the operations specified by the file by using the LDIFDE command. From a command prompt, type **ldifde** /? for usage information. The two most important switches for the LDIFDE command are:

- **-i**. Turns on import mode. Without this parameter, LDIFDE exports information.
- **-f filename**. The file from which to import, or to which to export.

For example, the following command will import objects from the file named Newusers.ldf:

```
ldifde -i -f newusers.ldf
```

The command accepts a variety of modifications using parameters. The most useful parameters are summarized below:

Command	Usage
General parameters	
-i	Import mode (Default is export mode)
-f <i>filename</i>	Import or export filename
-s <i>servername</i>	The domain controller to bind to for the query
-c <i>FromDN ToDN</i>	Convert occurrences of <i>FromDN</i> to <i>ToDN</i> . This is useful when importing objects from another domain, for example.
-v	Turn on verbose mode
-j <i>path</i>	Log file location
-?	Help
Export-specific parameters	
-d <i>RootDN</i>	The root of the LDAP search. The default is the root of the domain.
-r <i>Filter</i>	LDAP search filter. The default is (objectClass=*), meaning all objects.
-p <i>SearchScope</i>	The scope, or depth, of the search. Can be <i>subtree</i> (the container and all child containers), <i>base</i> (the immediate child objects of the container only), or <i>onelevel</i> (the container and its immediate child containers).
-l <i>list</i>	Comma-separated list of attributes to include in export for resulting objects. Useful if you want to export a limited number of attributes.
-o <i>list</i>	List of attributes (comma-separated) to omit from export for resulting objects. Useful if you want to export all but a few attributes.
Import-specific parameters	
-k	Ignore errors and continue processing if Constraint Violation or Object Already Exists errors appear.

Lab C: Automate User Account Creation

- Exercise 1: Export and Import Users with CSVDE
- Exercise 2: Import Users with LDIFDE

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

You are the administrator of Contoso, Ltd., an online university for adult education. You are hiring several new employees. The Human Resources department has provided you with extracts from their database, in both comma-delimited text format and in LDIF format. You want to import those data files to create user accounts for the new hires.

Exercise 1: Export and Import Users with CSVDE

In this exercise, you will use the CSVDE command to export user attributes and to create new user accounts from a comma-delimited text file.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Export users with CSVDE.
3. Import users with CSVDE.

► Task 1: Prepare for the lab

The virtual machine should already be started and available after completing Labs A and B. However, if it is not, you should launch it complete the exercises in Labs A and B before continuing.

- Start 6425B-HQDC01-A.
- Log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
- Run **D:\Labfiles\Lab03c\Lab03c_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

► Task 2: Export users with CSVDE

- Open the Command Prompt with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
- Type the following command:

```
csvde -f D:\Labfiles\Lab03c\UsersNamedApril.csv -r "(name=April*)"
-1 DN,objectClass,sAMAccountName,sn,givenName,userPrincipalName
```

and then press ENTER.

- Open **D:\Labfiles\Lab03c\UsersNamedApril.csv** in Notepad.
- Examine the file, and then close it.

► Task 3: Import users with CSVDE

- Open **D:\Labfiles\Lab03c\NewUsers.csv** with Notepad. Examine the information about the users listed in the file.
- Type the following command:

```
csvde -i -f D:\Labfiles\Lab03c\NewUsers.csv -k
```

and then press ENTER.

The two users are imported.

- Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**. Confirm that the users were created successfully.
 - If you have had the Active Directory Users and Computers snap-in open during this exercise, you might have to refresh your view to see the newly created accounts.
- Examine the accounts to confirm that first name, last name, user principal name, and pre-Windows 2000 logon name are populated according to the instructions in NewUsers.txt.
- Reset the passwords of the two accounts to **Pa\$\$w0rd**.
- Enable the two accounts.
- Close NewUsers.csv.

Exercise 2: Import Users with LDIFDE

Like CSVDE, LDIFDE can be used to import users. The LDIF file format, however, is not a typical delimited text file. In this exercise, you will use LDIFDE to import two users.

The main tasks for this exercise are as follows:

- Import users with LDIFDE.

► Task 1: Import users with LDIFDE

- Open `D:\Labfiles\Lab03c\NewUsers.ldf` with Notepad. Examine the information about the users listed in the file.
- Type the following command:

```
ldifde -i -f D:\Labfiles\Lab03c\NewUsers.ldf -k
```

then press ENTER.

The two users are imported.

- In **Active Directory Users and Computers**, confirm that the users were created successfully.
 - If you have had the Active Directory Users and Computers snap-in open during this exercise, you might have to refresh your view to see the newly created accounts.
- Examine the accounts to confirm that user properties are populated according to the instructions in `NewUsers.ldf`.
- Reset the passwords of the two accounts to **Pa\$\$w0rd**.
- Enable the two accounts.
- Close `NewUsers.ldf`.
- Log off HQDC01.

Results: After this exercise, you will have imported accounts for Lisa Andres, David Jones, Bobby Moore and Bonnie Kearney.



Note: After completing this exercise, turn off all virtual machines and discard undo disks.

Lab Review Question

Question: What scenarios lend themselves to importing users with CSVDE and LDIFDE?

Module 4

Manage Groups

Contents:

Lesson 1: Manage an Enterprise with Groups	4-4
Lesson 2: Administer Groups	4-45
Lab A: Administer Groups	4-66
Lesson 3: Best Practices for Group Management	4-74
Lab B: Best Practices for Group Management	4-88

Module Overview

- Manage an Enterprise with Groups
- Administer Groups
- Best Practices for Group Management

Although users and computers, and even services, change over time, business roles and rules tend to remain more stable. Your business probably has a finance role, which requires certain capabilities in the enterprise. The user or users who perform that role will change, but the role will remain. For that reason, it is not practical to manage an enterprise by assigning rights and permissions to individual user, computer, or service identities. Management tasks should be associated with groups. In this course, you will use groups to identify administrative and user roles, to filter Group Policy, to assign unique password policies, to assign rights and permissions, and more. To prepare for those tasks, in this module you will learn how to create, modify, delete, and support group objects in an Active Directory® domain.

Objectives

After completing this module, you will be able to:

- Understand the role of groups in managing an enterprise.
- Create well-documented, secure, delegated groups.

- Understand group types, scope, and nesting.
- Understand the best practice for group nesting to achieve role-based management.
- Create, delete, and manage groups with CSVDE and LDIFDE.
- Enumerate and copy group membership.
- Understand default (Builtin) groups.
- Understand special identities.

Lesson 1

Manage an Enterprise with Groups

- Demonstration: Create a Group Object
- Access Management Without Groups
- Groups Add Manageability
- Groups Add Scalability
- One Type of Group Is Not Enough
- Role-Based Management: Role Groups and Rule Groups
- Define Group Naming Conventions
- Group Type
- Group Scope
- Local Groups
- Global Groups
- Universal Groups
- Group Scope Possibilities Summarized
- Manage Group Membership
- Develop a Group Management Strategy (IGDLA)
- Role-Based Management and Windows Group Management Strategy

You are certainly familiar with the purpose of groups: to collect items and manage them as a single entity. The implementation of group management in Active Directory is not intuitive; Active Directory is designed to support large, distributed environments, so it includes seven different types of groups: two types of domain groups with three scopes each, plus local security groups. In this lesson, you will learn the purpose that each of these groups plays, and you'll learn to align your business requirements with the potentially complex options that Active Directory provides.

Objectives

After completing this lesson, you will be able to:

- Understand the role of groups in managing an enterprise.
- Define group naming conventions.
- Understand group types.

- Understand group scope.
- Identify group membership and nesting possibilities.
- Understand the best practice for group nesting to achieve role-based management.

Demonstration: Create a Group Object

In this demonstration, you will learn

- How to create a group
- How to configure the properties of a group object

Key Points

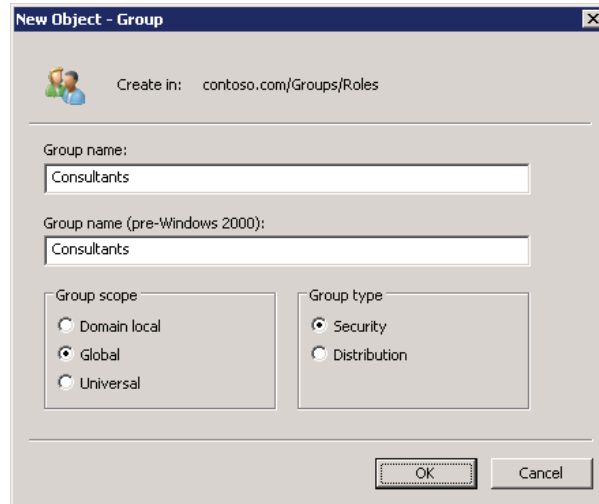
Groups are an important class of object, because they are used to collect users, computers, and other groups in order to create a “single point of management.” The most straightforward and common use of a group is to grant permissions to a shared folder. If a group has been given Read access to a folder, for example, any of the group’s members will be able to read the folder. You do not have to grant Read access directly to each individual member—you can manage access to the folder simply by adding and removing members of the group.

To create a group:

1. Open the **Active Directory Users and Computers** snap-in.
2. In the console tree, expand the node that represents your domain (for example, contoso.com) and navigate to the organizational unit (OU) or container (such as Users) in which you want to create the group.

3. Right-click the OU or container, point to **New**, and then click **Group**.

The New Object - Group dialog box appears



4. Type the name of the new group in the **Group name** box.
Most organizations have naming conventions that specify how group names should be created. Be sure to follow the guidelines of your organization.
By default, the name you type is also entered as the Group name (pre-Windows® 2000). It is very highly recommended that you keep the two names the same.
5. Do not change the name in the **Group name (pre-Windows 2000)** box.
6. Choose the **Group type**.
 - A **Security** group is a group that can be given permissions to resources. It can also be configured as an e-mail distribution list.
 - A **Distribution** group is an e-mail-enabled group that cannot be given permissions to resources and is therefore used only when a group is an e-mail distribution list that has no possible requirement for access to resources.

Group type will be discussed in more detail later in this module.

7. Select the **Group scope**.

- A **Global** group is typically used to identify users based on criteria such as job function, location, etc.
- A **Domain local** group is used to collect users and groups who share similar resource access needs, such as all users who need to be able to modify a project report.
- A **Universal** group is typically used to collect users and groups from multiple domains.

Group scope will be discussed in more detail later in this module.

8. Click **OK**.

Group objects have a number of properties that are useful to configure. These can be specified after the object has been created.

To specify properties for a group:

1. Right-click the group, and then click **Properties**.
2. Enter the properties for the group.
 - Be sure to follow the naming conventions and other standards of your organization.
 - The group's **Members** and **Member Of** tabs specify who belongs to the group and what groups the group itself belongs to.
 - The group's **Description** field, because it is easily visible in the details pane of the Active Directory Users and Computers snap-in, is a good place to summarize the purpose of the group and the contact information for the individual(s) responsible for deciding who is and is not a member of the group.
 - The group's **Notes** field can be used to provide more detail about the group.

- The **Managed By** tab can be used to link to the user or group that is responsible for the group. The contact information on the **Managed By** tab is populated from the account specified in the **Name** box. The **Managed By** tab is typically used for contact information so that if a user wants to join the group, you can decide who in the business should be contacted to authorize the new member. However, if you select the **Manager can update membership list option**, the account specified in the **Name** box will be given permission to add and remove members of the group. This is one method to delegate administrative control over the group.

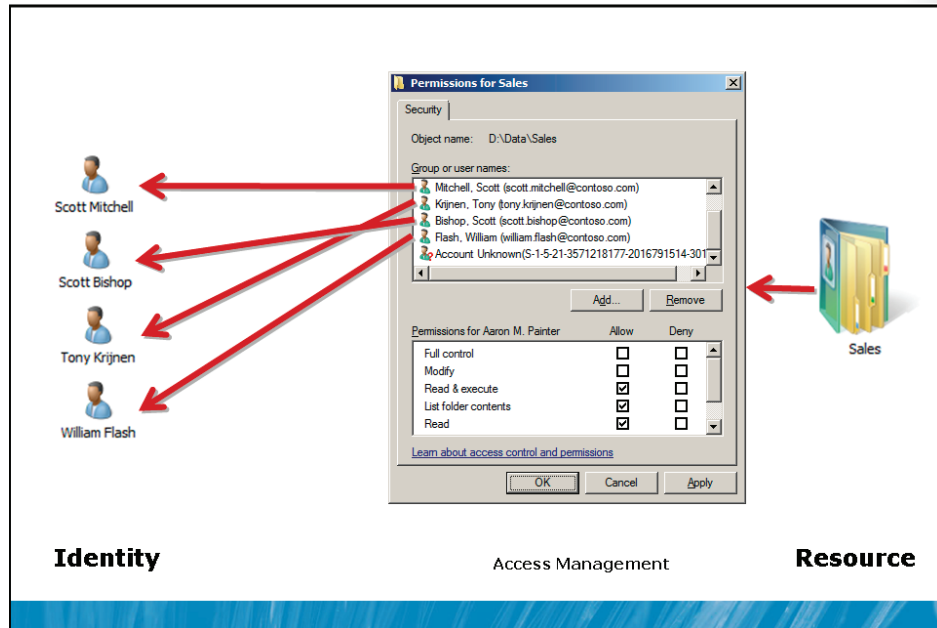
To change the user or group that is referred to on the **Managed By** tab, click the **Change** button underneath the **Name** box. By default, the **Select User, Contact, or Group** dialog box that appears does not, despite its name, search for groups. To search for groups, you must first click the **Object Types** button and select **Groups**.

3. Click **OK**.

Additional Reading

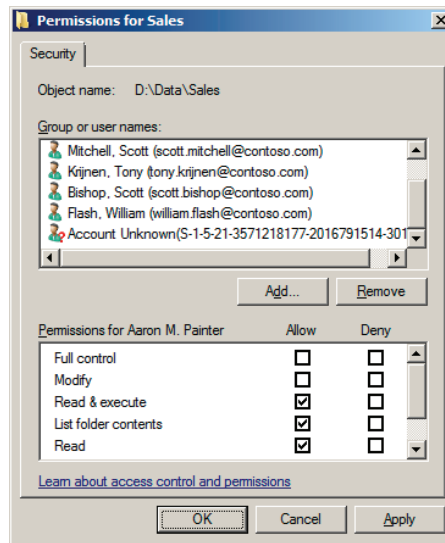
- Create a New Group: <http://go.microsoft.com/fwlink/?LinkId=168757>

Access Management Without Groups



Key Points

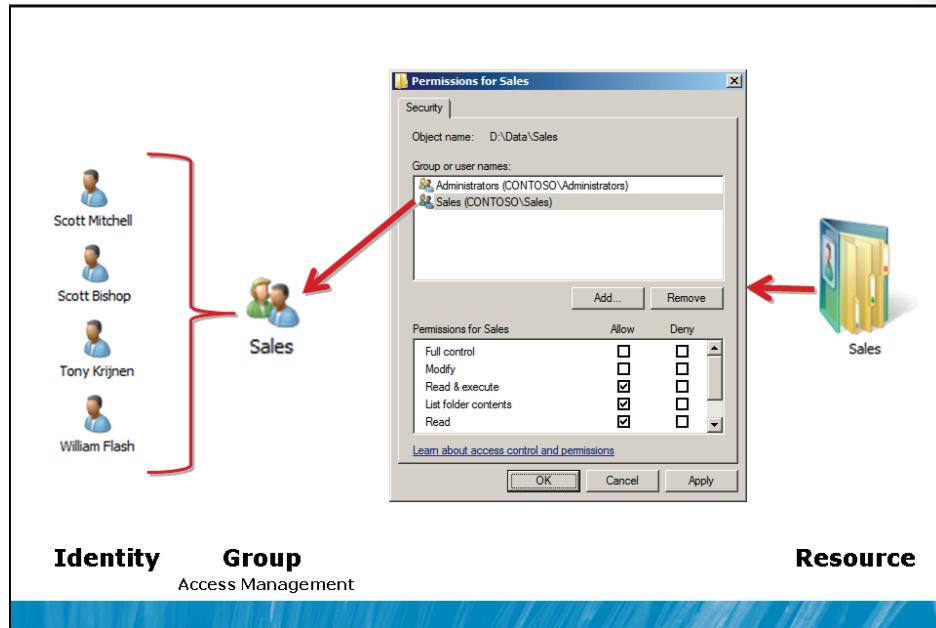
Imagine that all of the 100 users in the sales department require Read-level access to a shared folder called Sales on a server. It is not manageable to assign permissions to each user individually. When new salespeople are hired, you will have to add the new accounts to the access control list (ACL) of the folder. When accounts are deleted, you will have to remove the permissions from the ACL, or you will be left with a missing account entry on the ACL, as shown below, which results from a SID on the ACL that refers to an account that cannot be resolved.



Imagine now that all of the 100 users in the sales department require Read access to three shared folders on three different servers. The management headaches just increased significantly. How many permissions would you have to apply just to configure access to three folders on three different servers for 100 users? 300!

When you manage permissions by adding and removing identities to and from an ACL, it becomes difficult to answer the question, “Who can read the Sales folder?” In order to answer the question, you must reverse engineer the ACL. And, in the broader example, if the Sales folders are distributed across three servers, you would have to evaluate three separate ACLs to answer the question.

Groups Add Manageability



Key Points

The example presented in the previous topic may seem extreme, because you have no doubt learned that although assigning permissions to a resource for an individual identity—user or computer—is possible, the best practice is to assign a single permission to a group and then to manage access to the resource simply by changing the membership of the group.

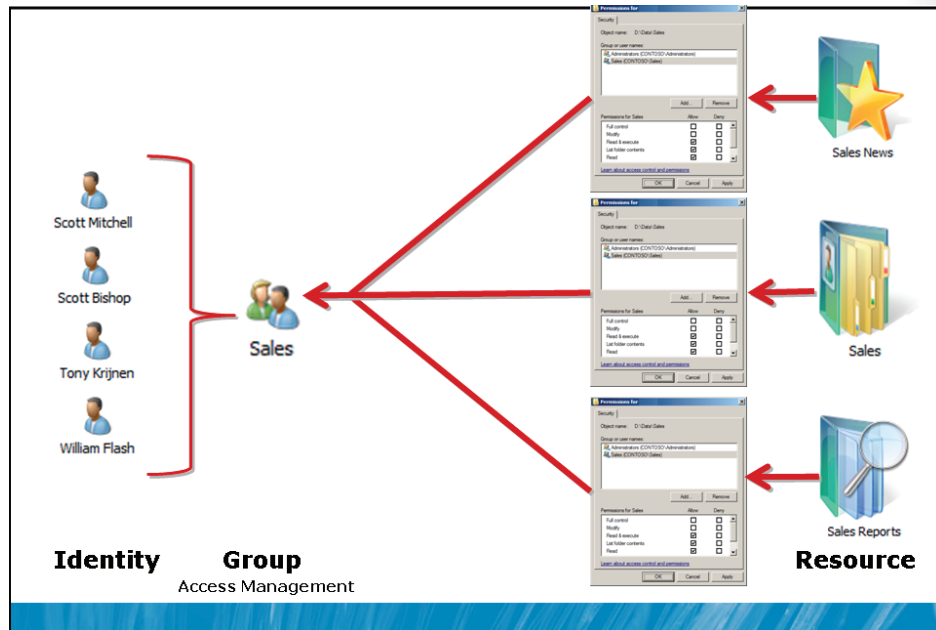
So, to continue the example, you could create a group called Sales and assign the group Allow Read permission on the Sales folder. You now have a single point of management. The Sales group effectively manages access to the shared folder. You can add new sales users to the group, and they will gain access to the shared folder. When you delete an account, it is automatically deleted from the group, so you will not have unresolvable SIDs on your ACL.

It is also easier to answer the question, "Who can read the Sales folder?" You can simply enumerate the membership of the Sales group.

The Sales group has become the focus of access management tasks.

There's an extra benefit: Because your ACL will remain stable, with the Sales group having Allow Read permission, your backups will be easier. When you change the ACL of a folder, the ACL propagates to all child files and folders, setting the Archive flag and thereby requiring a backup of all files, even if the contents of the files have not changed.

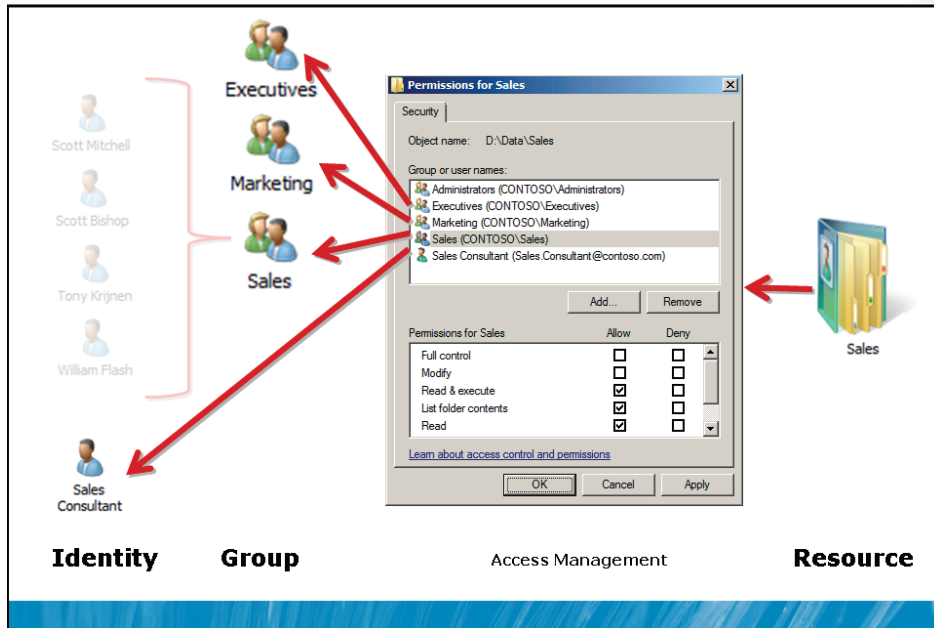
Groups Add Scalability



Key Points

If the sales users require Read access to three folders on three separate servers, you could assign the Sales group Allow Read permission on each of the three folders. After you assign the three permissions, the Sales group provides a single point of management for resource access. The Sales group effectively manages access to all three shared folders. You can add new sales users to the group, and they will gain access to the three shared folders on the three servers. When you delete an account, it is automatically deleted from the group, so you will not have unresolvable SIDs on your ACLs.

One Type of Group Is Not Enough



Key Points

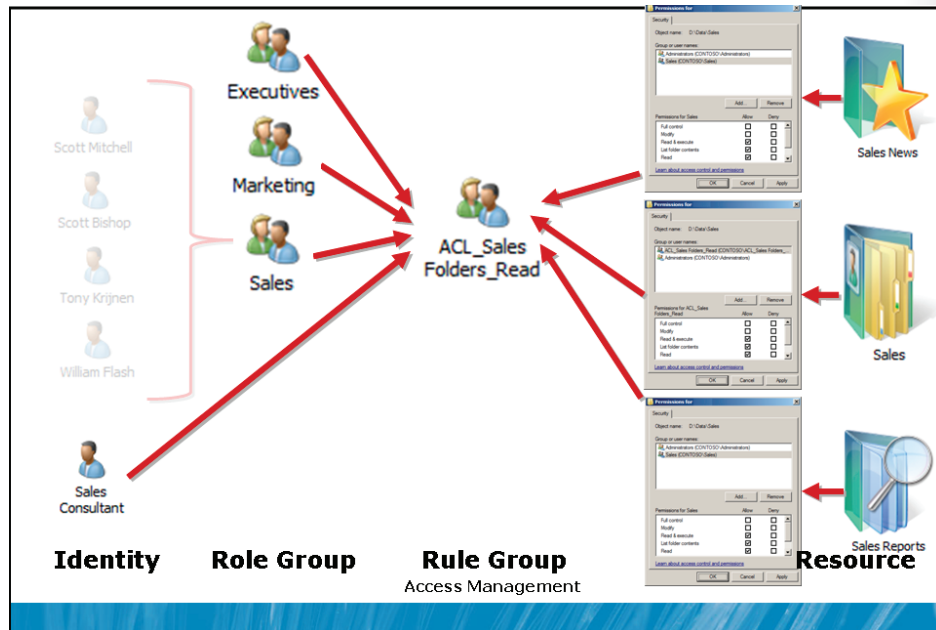
Imagine next that it is not only salespeople who require Read access to the folders. The Executives, Marketing department employees and the sales consultant hired by your organization also require Read permission to the same folders.

You could add those groups to the ACL of the folders, granting each of them Allow Read permission, but soon you will end up with an ACL with multiple permissions, this time assigning the Allow Read permission to multiple groups instead of multiple users. To give the three groups and one user permission to the three folders on the three servers, you will have to add twelve permissions! The next group that requires access will require three more changes to grant permissions to the ACLs of the three shared folders.

What if eight users who are not salespeople, marketing employees, or executives, have a business need for Read access to the three folders? Do you add their individual user accounts to the ACLs? If so, that's 24 more permissions to add and manage!

You can see that using only one type of group—a role group that defines the business roles of users—quickly becomes an ineffective way of enabling management of access to the three folders. If the management rule suggests that three roles and nine additional users require access to the resource, you are assigning a total of 36 permissions on ACLs. It becomes very difficult to maintain compliance and to audit. Even simple questions such as, "Can you tell me every user who can read the Sales folders?" become difficult to answer.

Role-Based Management: Role Groups and Rule Groups



Key Points

The solution is to recognize that there are two types of management that must take place to effectively manage this scenario. You must manage the users as collections, based upon their business roles. And, separately, you must manage access to the three folders.

The three folders are also a collection of items: They are a single resource—a collection of Sales folders—that just happens to be distributed across three folders on three servers. And you are trying to manage Read access to that resource. You need a single point of management with which to manage access to the resource.

This requires another group—a group that represents Read access to the three folders on the three servers. Imagine that you create a group called ACL_Sales Folders_Read. This group will be assigned the Allow Read permission on the three folders. The Sales, Marketing, and Executives groups, along with the individual users, will all be members of the ACL_Sales Folders_Read group. You assign only three permissions: one on each folder, granting Read access to the ACL_Sales Folders_Read group.

The ACL_Sales Folders_Read group becomes the focus of access management. As additional groups or users require access to the folders, they will be added to that group. It also becomes much easier to report who has access to the folders. Instead of having to examine the ACLs on each of the ten folders, you simply examine the membership of the ACL_Sales Folders_Read group.

In order to effectively manage even a slightly complex enterprise, you will need two "types" of groups that perform two distinct purposes:

- **Groups that define roles.** These groups, referred to as *role groups*, contain users, computers, and other role groups based on common business characteristics such as location, job type, etc.
- **Groups that define management rules.** These groups, referred to as *rule groups*, define how an enterprise resource is being managed.

This approach to managing the enterprise with groups is called *role-based management*. You define roles of users based on business characteristics—for example, department or division affiliation such as sales, marketing, and executives, and you define management rules—for example, the rule that manages which roles and individuals can access the three folders.

You can achieve both management tasks using groups in a directory. Roles are represented by groups that contain users, computers, and other roles. That's right—roles can include other roles, for example a Managers role might include the Sales Managers, Finance Managers, and Production Managers roles. Management rules, such as the rule that defines and manages Read access to the three folders, are represented by groups as well. Rule groups contain roles and, occasionally, individual users or computers such as the sales consultant and eight other users in the example.

The key takeaway is that there are two "types" of groups: one that defines the role, and the other that defines how a resource is managed.

Additional Reading

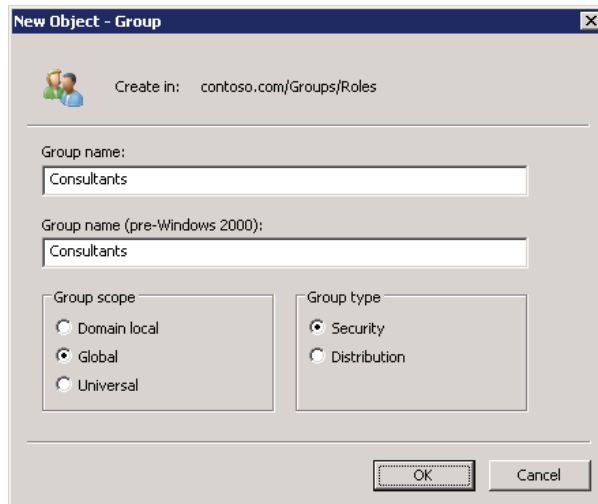
- For more information about role-based management, see *Windows Administration Resource Kit: Productivity Solutions for IT Professionals* by Dan Holme (Microsoft Press, 2008).

Define Group Naming Conventions

- Name properties
 - **Group name.** cn and name of group -- unique within OU
 - **Group name (pre-Windows 2000).** sAMAccountName of group -- unique in domain
 - Use the same name (unique in the domain) for *both* properties
- Naming conventions
 - **Role groups.** Simple, unique name, such as Sales or Consultants
 - **Management groups.** For example, ACL_Sales Folders_Read
 - **Prefix.** Management purpose of group, such as ACL
 - **Resource identifier.** What is managed, such as Sales Folders
 - **Suffix.** Access level, such as Read
 - **Delimiter.** Separates name components, such as underscore (_)

Key Points

Earlier in this lesson, you learned how to create a group using the Active Directory Users and Computers snap-in by right-clicking the OU in which you want to create a group, pointing to New, and then clicking Group. The New Object - Group dialog box, shown below, allows you to specify fundamental properties of the new group.



The following name properties can be configured here:

- **Group name.** cn and name of group object, must be unique only within OU
- **Group name (pre-Windows 2000).** sAMAccountName of group, unique in domain



Important best practice: Use the same name (unique in the domain) for *both* properties.

The first properties you must configure are the group's names. A group, like a user or computer, has several names. The first, shown in the Group Name box above, is used by Windows 2000 and later systems to identify the object—it becomes the cn, and name attributes of the object. The second, the pre-Windows 2000 name, is the sAMAccountName attribute, used to identify the group to computers running Windows NT 4.0 and to some devices, such as network attached storage (NAS) devices running non-Microsoft operating systems. The cn and name attributes must be unique only within the container—the OU—in which the group exists. The sAMAccountName must be unique in the entire domain. Technically, the sAMAccountName could be a different value than the cn and name, but it is highly discouraged to make these different. *Pick a name that is unique in the domain, and use it in both name fields in the New Object - Group dialog box.*

The following naming conventions are recommended:

- **Role groups.** Simple, unique name, such as Sales or Consultants
- **Management groups.** For example, ACL_Sales Folders_Read
 - **Prefix.** This identifies the management purpose of group, such as ACL for groups managing access permissions to shared resources.
 - **Resource identifier.** This is a unique identifier for what is being managed.
 - **Suffix.** For resource access groups, this is the type of access the group manages.
 - **Delimiter.** This should be a consistently used marker separating prefix, identifier, and suffix, such as an underscore (_). Do not use the delimiter elsewhere in the name—use it only as a delimiter.

The name you choose should help you manage the group and manage your enterprise on a day-to-day basis. It is recommended to follow a naming convention that identifies the type of group and the purpose of the group.

The example in the previous topic used a group name, ACL_Sales Folders_Read.

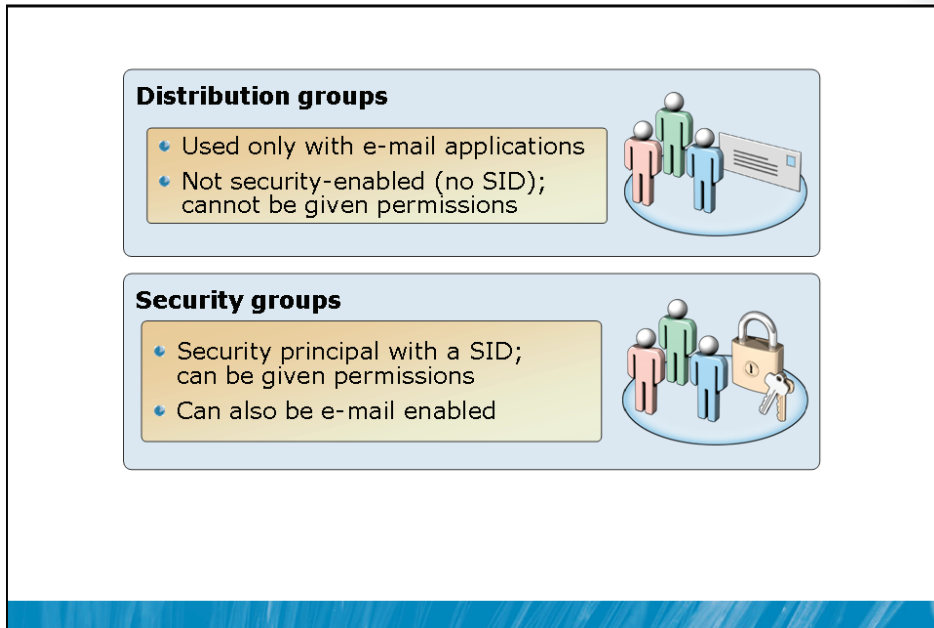
- **Prefix.** The prefix identifies the management purpose of the group. In this case, it is a group used to manage access permissions to a folder. It is used on access control lists, so the prefix ACL is used.
- **Resource identifier.** The main part of the name uniquely identifies the resource that is being managed with the group—in this example, Sales Folders.
- **Suffix.** The suffix further defines what is being managed by the group. In the case of resource access management groups, the suffix defines the level of access provided to members of the group. In our example, that is Read.
- **Delimiter.** A delimiter—in this case, an underscore—is used to separate parts of the name. Note that the delimiter is not used between the words Sales and Folder. Spaces are acceptable in group names—you will just need to enclose group names in quotes when you refer to them in commands or in scripts. You can create scripts that use the delimiter to deconstruct group names to facilitate auditing and reporting.

Keep in mind that role groups that define user roles will often be used by non-technical users. For example, you might e-mail enable the Sales group so that it can be used as an e-mail distribution list. Therefore, it is recommended that your naming convention for role groups is to keep them simple and straightforward. In other words, your naming convention for role groups is *not* to use prefixes or suffixes or delimiters—just a user-friendly, descriptive name.

Additional Reading

- For more information about managing groups effectively, see *Windows Administration Resource Kit: Productivity Solutions for IT Professionals* by Dan Holme (Microsoft Press, 2008).

Group Type



Key Points

There are two types of groups: security and distribution. When you create a group, you make the selection of the group type in the New Object – Group dialog box.

Distribution groups are used primarily by e-mail applications. These groups are not security enabled—they do not have SIDs—so they cannot be given permission to resources. Sending a message to a distribution group sends the message to all members of the group.

Security groups are security principals with SIDs. These groups can therefore be used in permission entries in ACLs to control security for resource access. Security groups can also be used as distribution groups by e-mail applications. If a group will be used to manage security, it must be a security group.

Because security groups can be used for both resource access and e-mail distribution, many organizations use only security groups. However, it is recommended that if a group will be used only for e-mail distribution, you should create the group as a distribution group. Otherwise, the group is assigned a SID, and the SID is added to the user's security access token, which can lead to unnecessary bloat of the security token.

Group Scope

- Four group scopes
 - Local
 - Global
 - Domain Local
 - Universal
- Characteristics that distinguish each scope
 - **Replication.** Where are the group and its membership stored?
 - **Membership.** What types of objects, from which domains, can be members of the group?
 - **Availability (Scope).** Where can the group be used? In what scopes of groups can the group *be* a member? Can the group be added to an ACL?

Key Points

Groups have members: users, computer, and other groups; groups can be members of other groups; and groups can be referred to by ACLs, Group Policy object (GPO) filters, and other management components. Group scope impacts each of these characteristics of a group: what it can contain, what it can belong to, and where it can be used. There are four group scopes: global, domain local, local, and universal.

The characteristics that define each scope fall into these categories:

- **Replication.** Where is the group defined, and to what systems is the group replicated?

- **Membership.** What types of security principals can the group contain as members? Can the group include security principals from trusted domains?

In a Module 14, you will learn about trust relationships, or trusts. A trust allows a domain to refer to another domain for user authentication, to include security principals from the other domain as group members, and to assign permissions to security principals in the other domain. The terminology used can be confusing. If Domain A trusts Domain B, then Domain A is the trusting domain and Domain B is the trusted domain. Domain A accepts the credentials of users in Domain B. It forwards requests by Domain B users to authenticate to a domain controller in Domain B, because it trusts the identity store and authentication service of Domain B. Domain A can add Domain B's security principals to groups and ACLs in Domain A.

- **Availability.** Where can the group be used? Is the group available to add to another group? Is the group available to add to an ACL?

Keep these broad characteristics in mind as you explore the details of each group scope.

Local Groups

- **Replication**
 - Defined in the security accounts manager (SAM) of a domain member or workgroup computer
 - Membership not replicated to any other system
- **Membership: Local group can include as members**
 - Any security principals from the domain: users (U), computers (C), global groups (GG), or domain local groups (DLG)
 - U, C, GG from any domain in the forest
 - U, C, GG from any trusted domain
 - Universal groups (UG) defined in any domain in the forest
- **Availability/scope**
 - Limited to the machine on which the group is defined; can be used for ACLs on the local machine only
 - Cannot be a member of any other group

Key Points

Local groups are truly local—defined on and available to a single computer. Local groups are created in the security accounts manager (SAM) database of a domain member computer—both workstations and servers have local groups. Local groups have the following characteristics:

- **Replication.** A local group is defined only in the local SAM database of a domain member. The group and its membership is not replicated to any other system.
- **Membership.** A local group can include as members:
 - Any security principals from the domain: users, computers, global groups, or domain local groups
 - Users, computers, and global groups from any domain in the forest
 - Users, computers, and global groups from any trusted domain
 - Universal groups defined in any domain in the forest

- **Availability.** A local group has only machine-wide scope. It can be used in ACLs on the local machine only. A local group cannot be a member of any other group.

Best Practice

In a workgroup, you use local groups to manage security of resources on a system. In a domain, however, managing the local groups of individual machines becomes unwieldy, and is for the most part unnecessary. It is not recommended to create custom local groups on domain members. There are very few scenarios in a domain environment that are addressed by using local groups. In most cases, the Users and Administrators local groups are the only local groups that you should be concerned with managing in a domain environment.

Domain Local Groups

- **Replication**
 - Defined in the domain naming context
 - Group and membership replicated to every DC in domain
- **Membership: Domain local group can include as members**
 - Any security principals from the domain: U, C, GG, DLG
 - U, C, GG from any domain in the forest
 - U, C, GG from any trusted domain
 - UG defined in any domain in the forest
- **Availability/scope**
 - Can be on ACLs on any resource on any domain member
 - Can be a member of other domain local groups or of machine local groups
- **Well suited for defining business management rules**

Key Points

Domain local groups are used primarily to manage permissions to resources. For example, the ACL_Sales Folders_Read group discussed earlier in the lesson would be created as a domain local group. Domain local groups have the following characteristics:

- **Replication.** A domain local group is defined in the domain naming context. The group object and its membership (the member attribute) are replicated to every domain controller in the domain.
- **Membership.** A domain local group can include as members:
 - Any security principals from the domain: users, computers, global groups, or other domain local groups.
 - Users, computers, and global groups from any domain in the forest.
 - Users, computers, and global groups from any trusted domain.
 - Universal groups defined in any domain in the forest.

- **Availability.** A domain local group can be added to ACLs on any resource on any domain member. Additionally, a domain local group can be a member of other domain local groups, or even machine local groups.

The membership capabilities of a domain local group (the groups to which a domain local group can belong) are identical to those of local groups, but the replication and availability of the domain local group make it useful across the entire domain.

Best Practice

Domain local groups are well suited for defining business management rules, such as resource access rules, because the group can be applied anywhere in the domain, and it can include members of any type within the domain as well as members from trusted domains.

For example, a domain local security group named ACL_Sales Folders_Read might be used to manage Read access to a collection of folders that contain sales information on one or more servers.

Global Groups

- **Replication**
 - Defined in the domain naming context
 - Group and membership is replicated to every DC in domain
- **Membership: Global group can include as members**
 - *Only* security principals from the same domain: U, C, GG, DLG
- **Availability/scope**
 - Available for use by all domain members, all other domains in the forest, and all trusting external domains
 - Can be on ACLs on any resource on any computer in any of those domains
 - Can be a member of any DLG or UG in the forest, and of any DLG in a trusting external domain
- **Well suited for defining roles**

Key Points

Global groups are used primarily to define collections of domain objects based on business roles. Role groups, such as the Sales and Marketing groups mentioned earlier, as well as roles of computers such as a Sales Laptops group, will be created as global groups. Global groups have the following characteristics:

- **Replication.** A global group is defined in the domain naming context. The group object, including the member attribute, is replicated to all domain controllers in the domain.
- **Membership.** A global group can include as members only those users, computers, and other global groups in the same domain.
- **Availability.** A global group is available for use by all domain members, as well as by all other domains in the forest and all trusting external domains. A global group can be a member of any domain local or universal group in the domain or in the forest. It can also be a member of any domain local group in a trusting domain. Finally, a global group can be added to ACLs in the domain, in the forest, or in trusting domains.

As you can see, global groups have the most limited membership (only users, computers, and global groups from the same domain) but the broadest availability across the domain, the forest, and trusting domains.

Bets Practice

Global groups are well suited to defining roles, because roles are generally collections of objects from the same directory.

For example, global security groups named Consultants and Sales might be used to define users who are consultants and salespeople, respectively.

Universal Groups

- **Replication**
 - Defined in a single domain in the forest
 - Replicated to the global catalog (forestwide)
- **Membership: Universal group can include as members**
 - U, C, GG, and UG from any domain in the forest
- **Availability/scope**
 - Available to every domain and domain member in the forest
 - Can be on ACLs on any resource on any system in the forest
 - Can be a member of other UGs or DLGs anywhere in the forest
- **Useful in multidomain forests**
 - Defining roles that include members from multiple domains
 - Defining business management rules that manage resources in multiple domains in the forest

Key Points

Universal groups have the following characteristics:

- **Replication.** A universal group is defined in a single domain in the forest but is replicated to the global catalog. You will learn more about the global catalog in Module 12. Objects in the global catalog will be readily accessible across the forest.
- **Membership.** A universal group can include as members users, global groups, and other universal groups from any domain in the forest.
- **Availability.** A universal group can be a member of a universal group or domain local group anywhere in the forest. Additionally, a universal group can be used to manage resources—for example, to assign permissions—anywhere in the forest.

Universal groups are useful in multidomain forests. They allow you to define roles or to manage resources that span more than one domain. The best way to understand universal groups is through an example: Trey Research has a forest with three domains: Americas, Asia, and Europe. Each domain has user accounts and a global group called Regional Managers that includes the managers of that region. Remember that global groups can contain only users from the same domain. A universal group called Trey Research Regional Managers is created, and the three Regional Managers groups are added as members. The Trey Research Regional Managers group therefore defines a role for the entire forest. As users are added to any one of the Regional Managers groups, they will, through group nesting, be members of the Trey Research Regional Managers.

Trey Research is planning to release a new product that requires collaboration across its regions. Resources related to the project are stored on file servers in each domain. In order to define who has the ability to modify files related to the new product, a universal group is created called ACL_New Product_Modify. That group is assigned the Allow Modify permission to the shared folders on each of the file servers in each of the domains. The Trey Research Regional Managers group is made a member of the ACL_New Product_Modify group, as are various global groups and a handful of users from each of the regions.

Best Practice

As you can see from this example, universal groups can help you to represent and consolidate roles that span domains in a forest, and to define rules that can be applied across the forest.

Group Scope Possibilities Summarized

Group Scope	Members from Same Domain	Members from Domain in Same Forest	Members from Trusted External Domain	Can be Assigned Permissions to Resources
Local	U, C, GG, DLG, UG and local users	U, C, GG, UG	U, C, GG	On the local computer only
Domain Local	U, C, GG, DLG, UG	U, C, GG, UG	U, C, GG	Anywhere in the domain
Universal	U, C, GG, UG	U, C, GG, UG	N/A	Anywhere in the forest
Global	U, C, GG	N/A	N/A	Anywhere in the domain or a trusted domain

U	User
C	Computer
GG	Global Group
DLG	Domain Local Group
UG	Universal Group

Key Points

Both on the certification examinations and in day-to-day administration, it is important that you are completely familiar with the membership characteristics of each group scope.

The table below summarizes the objects that can be members of each group scope.

Group Scope	Members from the Same Domain	Members from Another Domain in the Same Forest	Members from a Trusted External Domain
Local	Users Computers Global groups Universal groups Domain local groups Also local users defined on the same computer as the local group	Users Computers Global groups Universal groups	Users Computers Global groups
Domain Local	Users Computers Global groups Domain local groups Universal groups	Users Computers Global groups Universal groups	Users Computers Global groups
Universal	Users Computers Global groups Universal groups	Users Computers Global groups Universal groups	N/A
Global	Users global groups	N/A	N/A

Question: What types of objects can be members of a global group in a domain?

Manage Group Membership

- **Methods**
 - The group's Members tab (Add/Remove)
 - The member's Member Of tab (Add/Remove)
 - The member's Add to a group command (Add)
- **You are always changing the member attribute**
 - memberOf is a backlink attribute updated by Active Directory
- **Changes to membership do not take effect immediately**
 - Requires logon (for a user) or startup (for a computer)
 - Token built with SIDs of member groups at those times
 - Account for replication of membership change to the user or computer's domain controller
 - Tip: Change group membership on a DC in the user's site

Key Points

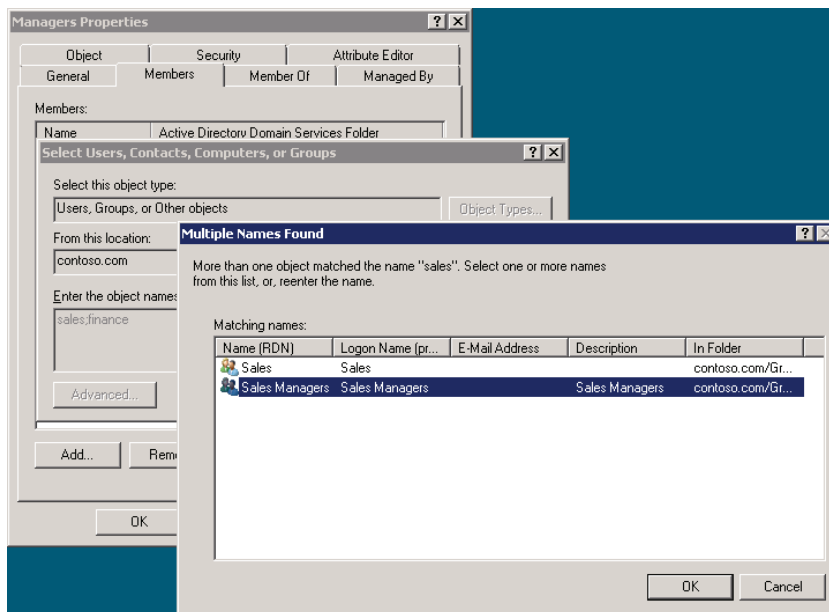
When you need to add or remove members of a group, you have several methods to do so.

The Members Tab

To manage group membership using the group's Members tab:

1. Open the group's **Properties** dialog box.
2. Click the **Members** tab.
3. To remove a member, simply select the member and click **Remove**.

4. To add a member, click the **Add** button. The **Select Users, Computers, or Groups** dialog box appears, as shown below:



There are several tips worth mentioning about this process:

- In the Select dialog box, in the Enter The Object Names box, you can type multiple accounts separated by semicolons. For example, in the screenshot shown above, both sales and finance were entered. They are separated by a semicolon.
- You can type partial names of accounts—you do not need to type the full name. Windows searches Active Directory for accounts that begin with the name you entered. If there is only one match, Windows selects it automatically. If there are multiple accounts that match, the Multiple Names Found dialog box appears, allowing you to select the specific object you want. This shortcut—typing partial names—can save time when you are adding members to groups and can help when you don't remember the exact name of a member.
- By default, Windows searches only for users and groups that match the names you enter in the Select dialog box. If you want to add computers to a group, you must click the Options button and select Computers.

- By default, Windows searches only domain groups. If you want to add local accounts, click the Locations button on the Select dialog box.
- If you cannot find the member you want to add, click the Advanced button on the Select dialog box. A more powerful query window will appear, giving you more options for searching Active Directory.

The Member Of Tab

To manage group membership using the member object's Member Of tab:

1. Open the properties of the member object, and then click its **Member Of** tab.
2. To remove the object from a group, select the group and then click the **Remove** button.
3. To add the object to a group, click the **Add** button and select the group.

The Add to a group Command

To manage group membership using the Add to a group command:

1. Right-click one or more selected objects in the Active Directory Users and Computers details pane.
2. Click the **Add to a group** command.
3. Use the **Select** dialog box to specify the group.

The Member and MemberOf Attributes

When you add a member to a group, you change the group's member attribute. The member attribute is a multivalued attribute: Each member is a value represented by the DN of the member. If the member is moved or renamed, Active Directory automatically updates the member attributes of groups that include the member.

When you add a member to a group, the member's memberOf attribute is also updated, indirectly. The memberOf attribute is a special type of attribute called a backlink. It is updated by Active Directory when a forward link attribute, such as member, refers to the object. When you add a member to a group, you are always changing the member attribute. Therefore, when you use the Member Of tab of an object to add to a group, you are actually changing the group's member attribute. Active Directory updates the memberOf attribute automatically.

Helping Membership Changes Take Effect Quickly

When you add a user to a group, the membership does not take effect immediately. Group membership is evaluated at logon for a user (at startup for a computer). Therefore, a user will have to log off and log on before the membership change becomes a part of the user's token.

Additionally, there may be a delay while the group membership change replicates. (Replication will be discussed in Module 12.) This is particularly true if your enterprise has more than one Active Directory site. You can facilitate the speed with which a change impacts a user by making the change on a domain controller in the user's site. Right-click the domain in the Active Directory Users and Computers snap-in, and then click Change Domain Controller.

Develop a Group Management Strategy (IGDLA)

- **I**dentities (users or computers) are members of

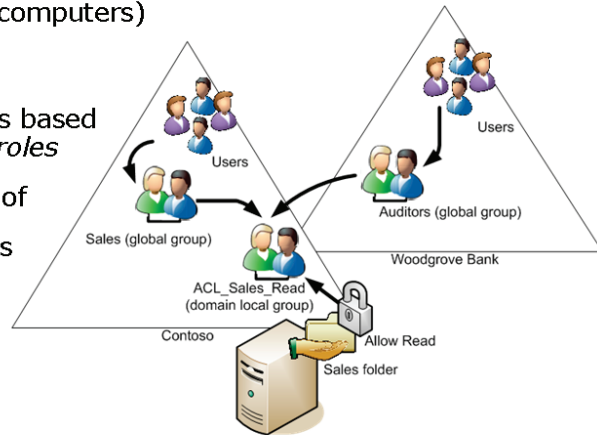
- **G**lobal groups that collect members based on those members' *roles*

which are members of

- **D**omain Local groups that provide *management* of some kind, such as management of resource access

which are

- Assigned **A**ccess to a resource (for example, on an **A**CL)
- Multi-domain forest: **IGUDLA**



Key Points

Adding groups to other groups—a process called nesting—can create a hierarchy of groups that support your business roles and management rules. Now that you have learned the business purposes and technical characteristics of groups, it is time to align the two in a strategy for group management.

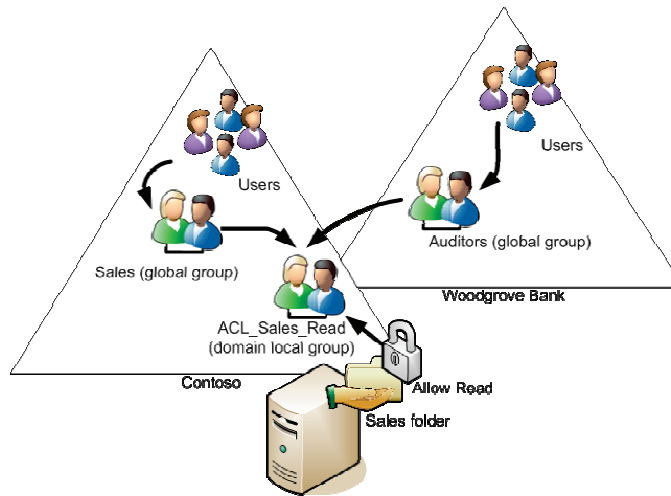
Earlier in this lesson, you learned what types of objects can be members of each group scope. Now it is time to identify what types of objects *should* be members of each group scope. This leads to the best practice for group nesting, known as IGDLA:

- **I**dentities (user and computer accounts) are members of.
- **G**lobal groups that represent business roles. Those role groups (global groups) are members of.

- **Domain Local groups** that represent management rules—determining who has Read permission to a specific collection of folders, for example. These rule groups (domain local groups) are granted.
- **Access to resources.** In the case of a shared folder, access is granted by adding the domain local group to the folder's access control list (ACL), with a permission that provides the appropriate level of access.

In a multidomain forest, there are universal groups as well, which fit in between global and domain local groups. Global groups from multiple domains are members of a single universal group. That universal group is a member of domain local groups in multiple domains. You can remember the nesting as *IGUDLA*.

This best practice for implementing group nesting translates well even in multidomain scenarios. Consider the figure below:



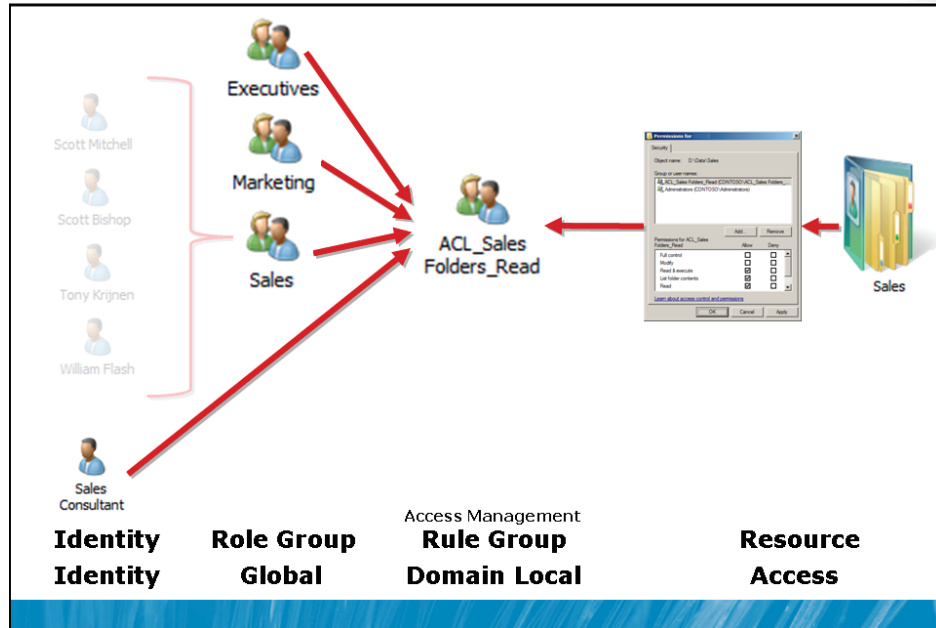
This figure represents a group implementation that reflects not only the technical view of group management best practices (IGDLA) but also the business view of role-based, rule-based management.

Consider the following scenario: the sales force at Contoso, Ltd. has just completed its fiscal year. Sales files from the previous year are in a folder called Sales. The sales force needs Read access to the Sales folders. Additionally, a team of auditors from Woodgrove Bank, a potential investor, require Read access to the Sales folders to perform the audit. The steps to implement the security required by this scenario are as follows:

1. Assign users with common job responsibilities or other business characteristics to role groups implemented as global security groups. This happens separately in each domain. Sales people at Contoso are added to a Sales role group. Auditors at Woodgrove Bank are added to an Auditors role group.
2. Create a group to manage access the Sales folders with Read permission. This is implemented in the domain containing the resource that is being managed. In this case, it is the Contoso domain in which the Sales folders reside. The resource access management rule group is created as a domain local group, ACL_Sales Folders_Read.
3. Add the role groups to the resource access management rule group to represent the management rule. These groups can come from any domain in the forest or from a trusted domain such as Woodgrove Bank. Global groups from trusted external domains, or from any domain in the same forest, can be members of a domain local group.
4. Assign the permission that implements the required level of access. In this case, grant the Allow Read permission to the domain local group.

This strategy results in single points of management, reducing the management burden. There is one point of management that defines who is in Sales, or who is an Auditor. Those roles, of course, are likely to have access to a variety of resources beyond simply the Sales folders. There is another single point of management to determine who has Read access to the Sales folders. And, of course, the Sales folders may not just be a single folder on a single server: It could be a collection of folders across multiple servers, each of which assigns Allow Read permission to the single domain local group.

Role-Based Management and Windows Group Management Strategy



Key Points

Role-based management is a concept used throughout information technology and information protection, and it can be attained with out-of-the-box capabilities of Active Directory. IGDLA is the implementation of role-based management using Active Directory groups.

Lesson 2

Administer Groups

- Create Groups with DSAdd
- Import Groups with CSVDE
- Import Groups with LDIFDE
- Convert Group Type and Scope
- Modify Group Membership with DSMod
- Modify Group Membership with LDIFDE
- Retrieve Group Membership with DSGet
- Copy Group Membership
- Move and Rename Groups
- Delete Groups

After completing this lesson, you will be able to:

- Create groups with DSADD, CSVDE, and LDIFDE.
- Manage and convert group type and scope.
- Manage group membership with DSMOD and LDIFDE.
- Enumerate group membership with DSGET.
- Delete a group with DSRM.
- Copy group membership.

Create Groups with DSAdd

- `dsadd group GroupDN -secgrp {yes|no} -scope {g | l | u}`
 - **GroupDN.** Distinguished name of group to create
 - **-secgrp.** Security-enabled (yes=security; no=distribution)
 - **-scope.** Scope (**g**lobal, domain **l**ocal, **u**niversal)
 - **-samid.** sAMAccountName (not necessary; defaults to cn)
 - **-desc Description.** description attribute
 - **-member MemberDN** Space-separated list of members to add when creating the group
 - **-memberof GroupDN** Space-separated list of groups to add this group to

```
dsadd group "CN=Marketing,OU=Role,OU=Groups,
DC=contoso,DC=com"
-samid Marketing -secgrp yes -scope g
```

Key Points

The DSAdd command allows you to add objects to Active Directory.

To add a group, type the command:

```
dsadd group GroupDN
```

where *GroupDN* is the distinguished name (DN) of the group, such as "CN=Finance Managers,OU=Role,OU=Groups,DC=contoso,DC=com." Be certain to surround the DN with quotes if the DN includes spaces.

For example, to create a new global security group named Marketing, the command would be:

```
dsadd group "CN=Marketing,OU=Role,OU=Groups,DC=contoso,DC=com"
-samid Marketing -secgrp yes -scope g
```

You can also provide the *GroupDN* parameter one of the following ways:

- Provide the parameter by piping a list of DNs from another command, such as *DSQuery*.
- Provide it by typing each DN at the command prompt, separated by spaces.
- Provide it by leaving the DN parameter empty, at which point you can type the DNs, one at a time, at the keyboard console of the command prompt. Press ENTER after each DN. Press CTRL+Z and ENTER after the last DN.

Any of these three options allows you to generate multiple groups at once with *DSAdd*.

The *DSAdd* command can also configure group attributes of the groups you create with the following optional parameters:

- **-secgrp { yes | no }**. This parameter specifies group type: security (yes) or distribution (no).
- **-scope { l | g | u }**. This determines the group scope: domain local (l), global (g), or universal (u).
- **-samid Name**. This parameter specifies the *sAMAccountName* of the group. If not specified, the name of the group from its DN is used. It is recommended that the *sAMAccountName* and the group *name* be the same, so you do not need to include this parameter when using *DSAdd*.
- **-desc Description**. This configures the group's description.
- **-members MemberDN** This parameter adds members to the group. Members are specified by their DNs in a space-separated list.
- **-memberof GroupDN** This makes the new group a member of one or more existing groups. The groups are specified by their DNs in a space-separated list.

Import Groups with CSVDE

- Comma-separated values (csv) file format

Comma-separated list of attributes
Groups to create, one per line, with all attributes listed
on the first line

- Example

```
objectClass,sAMAccountName,DN,member
group,Marketing,"CN=Marketing,OU=Role,OU=Groups,
DC=contoso,DC=com",
"CN=Linda Mitchell,OU=Employees,OU=User Accounts,
DC=contoso,DC=com;CN=Scott Mitchell,OU=Employees,
OU=User Accounts,DC=contoso,DC=com"
```

- `csvde -i -f "filename" [-k]`

- **-i.** Import; default mode is export
- **-f.** File name
- **-k.** Continue on error, such as object already exists

- CSVDE can create groups, not modify existing groups

Key Points

CSVDE imports data from comma-separated values (.csv) files. It is also able to export data to a .csv file. The following example shows a .csv file that will create a group, Marketing, and populate the group with two initial members: Linda Mitchell and Scott Mitchell.

```
objectClass,sAMAccountName,DN,member
group,Marketing,"CN=Marketing,OU=Role,OU=Groups,
DC=contoso,DC=com","CN=Linda Mitchell,OU=Employees,
OU=User Accounts,DC=contoso,DC=com;
CN=Scott Mitchell,OU=Employees,OU=User Accounts,
DC=contoso,DC=com"
```

The objects listed in the member attribute must already exist in the directory service. Their distinguished names (DNs) are separated by semicolons within the member column.

Take note of the use of quotation marks in the example shown above. Quotation marks are required when an attribute includes a comma, otherwise the comma would be interpreted as a delimiter. The DN of the group includes commas, and so must be surrounded by a comma. In the case of a multivalued attribute such as member, each value is separated by a semicolon—there are two values in member in the example above. The entire member attribute is surrounded by quotation marks, not each individual value of the member attribute.

You can import this file into Active Directory using the command:

```
csvde -i -f "filename" [-k]
```

The `-i` parameter specifies import mode. Without it, CSVDE uses export mode. The `-f` parameter precedes the file name, and the `-k` parameter ensures that processing continues even if errors are encountered, such as the object already exists, or the member cannot be found.

CSVDE can be used to create objects, not to modify existing objects. You cannot use CSVDE to import members to existing groups.

Import Groups with LDIFDE

- Lightweight Directory Access Protocol Data Interchange Format (LDIF) file

```
DN: CN=Finance,OU=Role,OU=Groups,DC=contoso,DC=com
changeType: add
CN: Finance
description: Finance Users
objectClass: group
sAMAccountName: Finance
```

```
DN: CN=Research,OU=Role,OU=Groups,DC=contoso,DC=com
changeType: add
CN: Research
description: Research Users
objectClass: group
sAMAccountName: Research
```

- `Ldifde -i -f "filename" [-k]`
 - **-i.** Import (default mode is export)
 - **-f.** File name
 - **-k.** Continue on error, such as object already exists

Key Points

LDIFDE is a tool that imports and exports files in the Lightweight Directory Access Protocol Data Interchange Format (LDIF) format. LDIF files are text files within which operations are specified by a block of lines separated by a blank line. Each operation begins with the DN attribute of the object that is the target of the operation. The next line, `changeType`, specifies the type of operation: add, modify, or delete.

The following LDIF file creates two groups, Finance and Research:

```
DN: CN=Finance,OU=Role,OU=Groups,DC=contoso,DC=com
changeType: add
CN: Finance
description: Finance Users
objectClass: group
sAMAccountName: Finance

DN: CN=Research,OU=Role,OU=Groups,DC=contoso,DC=com
changeType: add
CN: Research
description: Research Users
objectClass: group
sAMAccountName: Research
```

Convention would suggest saving the file with an .ldf extension; for example, groups.ldf.

To import the groups into the directory, issue the ldifde.exe command as shown here:

```
ldifde -i -f groups.ldf
```

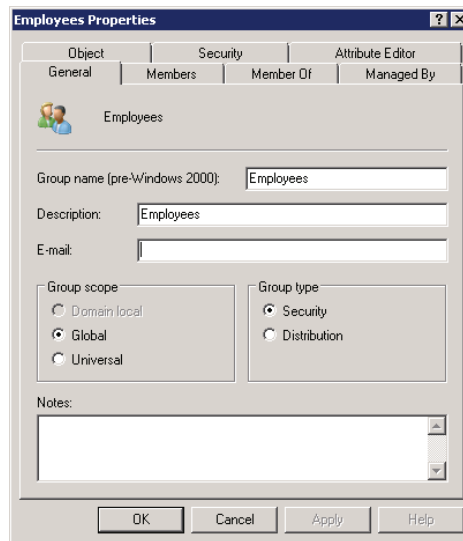
The `-i` parameter specifies import mode. Without it, LDIFDE uses export mode. The `-f` parameter precedes the file name, and the `-k` parameter ensures that processing continues even if errors are encountered, such as the object already exists.

Convert Group Type and Scope

- In Active Directory Users and Computers, you can change group type:
 - Security to distribution (* lose permissions assigned to group)
 - Distribution to security
- In Active Directory Users and Computers, you can change the group scope:
 - Global to universal
 - Domain local to universal
 - Universal to global
 - Universal to domain local
 - You cannot change DL → G or G → DL directly, but you *can* change DL → U → G or G → U → DL.
 - Change prevented if memberships are invalid—fix, then retry
- `dsmod group GroupDN -secgrp { yes | no }`
`-scope { l | g | u }`

Key Points

If, after creating a group, you determine that you need to modify the group's scope or type, you can do so. Open the Properties of an existing group and, on the General tab, shown below, you will see the existing scope and type. At least one more scope and type are available to be selected.



You can convert the group type at any time by changing the selection in the Group Type section of the General tab. Be cautious, however. When you convert a group from security to distribution, any resources to which the group had been assigned permission will no longer be accessible in the same way. After the group becomes a distribution group, users who log on to the domain will no longer include the group's SID in their security access tokens.

You can change the group scope in one of the following ways:

- Global to universal
- Domain local to universal
- Universal to global
- Universal to domain local

The only scope changes that you cannot make directly are from global to domain local or domain local to global. However, you can make these changes indirectly by first converting to universal scope, then converting to the desired scope. So all scope changes are possible.

Remember, however, that a group's scope determines the types of objects that can be members of the group. If a group already contains members, or is a member of another group, you will be prevented from changing scope. For example, if a global group is a member of another global group, you cannot change the first group to universal scope, because a universal group cannot be a member of a global group. You will be given an explanatory error message, such as that shown below. You must correct the membership conflicts before you can change the group's scope.



The DSMod command can be used to change group type and scope using the following syntax:

```
dsmod group GroupDN -secgrp { yes | no } -scope { l | g | u }
```

The *GroupDN* is the distinguished name of the group to modify. The following two parameters affect group scope and type:

- **-secgrp { yes | no }**. Specifies group type: security (*yes*) or distribution (*no*)
- **-scope { l | g | u }**. Determines the group scope: domain local (*l*), global (*g*), or universal (*u*)

Modify Group Membership with DSMod

```
· dsmod group "GroupDN" [options]
```

- -addmbr "Member DN"
- -rmmbr "Member DN"

```
dsmod group "CN=Research,OU=Role,OU=Groups,  
DC=contoso,DC=com" -addmbr "CN=Mike Danseglio,  
OU=Employees,OU=User Accounts,DC=contoso,DC=com"
```

Key Points

The DSMod command's basic syntax is:

```
dsmod group "GroupDN" [options]
```

You can use options such as *-samid* and *-desc* to modify the sAMAccountName and description attributes of the group. Most useful, however, are the options that let you modify a group's membership:

```
-addmbr "Member DN"
```

This adds members to the group.

```
-rmmbr "Member DN"
```

This removes members from the group.

As with all DS commands, the Member DN is the distinguished name of another Active Directory object, surrounded by quotes if the DN includes spaces. Multiple Member DN entries can be included, separated by spaces. For example, to add Mike Danseglio to the Research group, the DSMod command would be:

```
dsmod group "CN=Research,OU=Role,OU=Groups,DC=contoso,DC=com"  
-addmbr "CN=Mike Danseglio,OU=Employees,OU=User Accounts,  
DC=contoso,DC=com"
```


Modify Group Membership with LDIFDE

- LDIF file

```
dn: CN=Finance,OU=Role,OU=Groups,DC=contoso,DC=com
changetype: modify
add: member
member: CN=April Stewart,OU=Employees,OU=User Accounts,
      dc=contoso,dc=com
member: CN=Mike Fitzmaurice,OU=Employees,OU=User Accounts,
      dc=contoso,dc=com
-
```

- changetype: modify
- Third line: What type of change? Add a value to member
 - To delete a member, just change to delete: member
- Change operation is terminated with line containing only -

Key Points

LDIFDE can also be used to modify existing objects in Active Directory using LDIF operations with a changeType of modify. To add two members to the Finance group, the LDIF file would be:

```
dn: CN=Finance,OU=Role,OU=Groups,DC=contoso,DC=com
changetype: modify
add: member
member: CN=April Stewart,OU=Employees,OU=User
Accounts,dc=contoso,dc=com
member: CN=Mike Fitzmaurice,OU=Employees,OU=User
Accounts,dc=contoso,dc=com
-
```

The changeType is set to modify, and then the change operation is specified: add objects to the member attribute. Each new member is then listed on a separate line that begins with the attribute name, member. The change operation is terminated with a line containing a single dash. Changing the third line to the following would remove the two specified members from the group:

```
delete: member
```

Retrieve Group Membership with DSGet

- No option to show *fully enumerated* group memberships in Active Directory Users and Computers
- DSGet allows full enumeration (including nested members)
 - `dsget group "GroupDN" -members [-expand]`
 - Shows members of group (*GroupDN*), optionally including nested members (*-expand*)
 - `dsget {user|computer} "ObjectDN" -memberof [-expand]`
 - Shows membership of user or computer (*ObjectDN*), optionally including nested group memberships (*-expand*)

Key Points

The DSGet and DSMod commands are particularly helpful for managing the membership of groups. As you might know, there is no option in the Active Directory Users and Computers snap-in to list all the members of a group, including nested members. You can only see direct members of a group on the group's Members tab. Similarly, there is no way to list all the groups to which a user or computer belongs, including nested groups. You can only see direct membership on the user's or computer's Member Of tab.

The DSGet command allows you to retrieve a complete list of a group's membership, including nested members, with the following syntax:

```
dsget group "GroupDN" -members [-expand]
```

The *-expand* option performs the magic of expanding nested groups' members.

Similarly, the DSGet command can be used to retrieve a complete list of groups to which a user or computer belongs, again by using the expand option in the following commands:

```
dsget user "UserDN" -memberof [-expand]  
dsget computer "ComputerDN" -memberof [-expand]
```

The memberof option returns the value of the user's or computer's memberOf attribute, showing the groups to which the object directly belongs. By adding the expand option, those groups are searched recursively, producing an exhaustive list of all groups to which object user belongs in the domain.

Copy Group Membership

- Copy members from one group to another

```
dsget group "CN=Sales,OU=Role,OU=Groups,DC=contoso,DC=com" -members |
dsmod group "CN=Marketing,OU=Role,OU=Groups,DC=contoso,DC=com" -addmbr
```

- Copy memberships of one user to another

```
dsget user "SourceUserDN" -memberof |
dsmod group -addmbr "TargetUserDN"
```

Key Points

You can use DSGet in combination with DSMod to copy group membership. In the following example, the DSGet command is used to get information about all the members of the Sales group and then, by piping that list to DSMod, to add those users to the Marketing group:

```
dsget group "CN=Sales,OU=Role,OU=Groups,DC=contoso,DC=com" -members |
dsmod group "CN=Marketing,OU=Role,OU=Groups,DC=contoso,DC=com" -addmbr
```

Notice the use of *piping*. The "output" of DSGet (distinguished names of members of the first group) is piped, using the pipe symbol ("|"), to act as the "input" for the DNs that are missing from the -addmbr switch.

Similarly, the DSGet and DSMod commands can work together to copy the group membership of one object, such as a user, to another object:

```
dsget user "SourceUserDN" -memberof |
dsmod group -addmbr "TargetUserDN"
```

Move and Rename Groups

- Active Directory Users and Computers
 - Right-click group, then click Move or Rename
- DSMove command
 - `dsmove ObjectDN [-newname NewName] [-newparent TargetOUDN]`
 - ObjectDN is the DN of the group
 - **-newparent** *TargetOUDN* moves the group to a new OU
 - **-newname** *NewName* changes the cn of the group
 - Must use DSMod Group to change the sAMAccountName

```
dsmove "CN=Public Relations,OU=Role,OU=Groups,DC=contoso,DC=com"
-newparent "OU=Marketing,DC=contoso,DC=com"

dsmove "CN=Marketing,OU=Role,OU=Groups,DC=contoso,DC=com"
-newname "Public Relations"
dsmod group "CN=Public Relations,OU=Role,OU=Groups,DC=contoso,DC=com"
-samid "Public Relations"
```

Key Points

You can move and rename groups in Active Directory Users and Computers by right-clicking the group and then clicking the Move or the Rename command.

The DSMove command allows you to move or rename an object within a domain. You cannot use it to move objects between domains. Its basic syntax is:

```
dsmove ObjectDN [-newname NewName] [-newparent TargetOUDN]
```

The object is specified using its distinguished name in the *ObjectDN* parameter. To rename the object, specify its new common name as the value of the *-newname* parameter. To move an object to a new location, specify the distinguished name of the target container as the value of the *-newparent* parameter.

For example, to change the name of the Marketing group to Public Relations, type:

```
dsmove "CN=Marketing,OU=Role,OU=Groups,DC=contoso,DC=com"
-newname "Public Relations"
```

To then move that group to the Marketing OU, type:

```
dsmove "CN=Public Relations,OU=Role,OU=Groups,DC=contoso,DC=com"  
-newparent "OU=Marketing,DC=contoso,DC=com"
```

Delete Groups

- Active Directory Users and Computers: Right-click, Delete

- DSRm command

- `dsrm ObjectDN ... [-subtree [-exclude]] [-noprompt] [-c]`

- `-noprompt` prevents prompting to confirm each deletion
 - `-c` continues if an error occurs (such as access denied)
 - `-subtree` deletes the object and all child objects
 - `-subtree -exclude` deletes all child objects but not the object itself

```
dsrm "CN=Public Relations,OU=Role,OU=Groups,DC=contoso,DC=com"
```

- Deleting a security group has significant impact

- SID is lost and cannot be re-established by re-creating group
 - Tip: First, record all members and delete all members for a test period, to evaluate any unintended side effects

Key Points

You can delete a group in the Active Directory Users and Computers snap-in by right-clicking the group and choosing the Delete command.

Also, DSRm can be used to delete a group, or any other Active Directory object. The basic syntax of DSRm is:

```
dsrm ObjectDN ... [-subtree [-exclude]] [-noprompt] [-c]
```

The object is specified by its distinguished name in the *ObjectDN* parameter. You will be prompted to confirm the deletion of each object unless you specify the *-noprompt* option. The *-c* switch puts DSRm into continuous operation mode, in which errors are reported but the command keeps processing additional objects. Without the *-c* switch, processing halts on the first error.

The *-subtree* option causes DSRm to delete the object and all child objects. The *-subtree -exclude* option will delete all child objects, but not the object itself.

To delete the Public Relations group, type:

```
dsrm "CN=Public Relations,OU=Role,OU=Groups,DC=contoso,DC=com"
```

Know the Impact Before Deleting a Group

When you delete a group, you are removing a point of management in your organization. Be certain you have evaluated the environment to know that there are no permissions or other resources that rely on the group. Deleting a group is a serious action with potentially significant consequences. When you delete a group, you remove its SID. Recreating the group with the same name does not restore permissions, because the new group's SID is different than that of the original group.

It is recommended that, before you delete a group, you record its membership and *remove all members for a period of time, to determine whether the members lose access to any resources*. If anything goes wrong, simply re-add the members. If the test succeeds, then delete the group.

Lab A: Administer Groups

- Exercise 1: Implement Role-Based Management Using Groups
- Exercise 2: Manage Group Membership from the Command Prompt
- Exercise 3 (Advanced Optional): Explore Group Membership Reporting Tools
- Exercise 4 (Advanced Optional): Understand "Account Unknown" Permissions

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

In order to improve the manageability of resource access at Contoso, Ltd., you have decided to implement role-based management. The first application of role-based management will be to manage who can access the folders containing Sales information. You must create groups that manage access to that sensitive information. Business rules are that Sales and Marketing employees, as well as a team of Consultants, should be able to read the Sales folders. Additionally, Bobby Moore requires Read access. Finally, you have been asked to discover a way to produce a list of group members, including those who are in nested groups; and a list of a user's group membership, including indirect or nested membership.

Exercise 1: Implement Role-Based Management Using Groups

In this exercise, you will implement role-based management using groups and the best practice group nesting strategy, IGDLA. You will create different scopes and types using both the Active Directory Users and Computers snap-in and command-line tools.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Create role groups with Active Directory Users and Computers.
3. Create role groups with DSAdd.
4. Add users to the role group.
5. Implement a role hierarchy in which Sales Managers are also part of the Sales role.
6. Create a resource access management group.
7. Assign permissions to the resource access management group.
8. Define which roles and users have access to a resource.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Run **D:\Labfiles\Lab04a\Lab04a_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

► Task 2: Create role groups with Active Directory Users and Computers

1. Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. Create global security groups called **Sales** and **Consultants** in the **Groups\Role** OU.

► **Task 3: Create a group with DSAdd**

1. Run Command Prompt with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. Using the **DSAdd** command, create a global security group named **Auditors** in the **Groups\Role** OU.
3. In **Active Directory Users and Computers**, confirm that the object has been created.

► **Task 4: Add users to the role group**

1. Add **Tony Krijnen** to the **Sales** group using the **Members** tab of the **Sales** group.
2. Add **Linda Mitchell** to the **Sales** group by right-clicking **Linda Mitchell** and choosing **Add to a group**.

► **Task 5: Implement a role hierarchy in which Sales Managers are also part of the Sales role**

- Add the **Sales Managers** group as a member of the **Sales** group by using the **Member Of** tab of the **Sales Managers** group.

► **Task 6: Create a resource access management group**

- Create a domain local security group named **ACL_Sales Folders_Read** in the **Groups\Access** OU.

► **Task 7: Assign permissions to the resource access management group**

1. Create a folder in **D:\Data** named **Sales**.
2. Right-click the **Sales** folder, then click **Properties**, and then click the **Security** tab.
3. Click **Edit**, and then click **Add**.
4. Type **ACL_** and press ENTER.

Notice that when you use a prefix for group names, such as the **ACL_** prefix for resource access groups, you can find them quickly.

5. Click **ACL_Sales Folders_Read**, and then click **OK**.
6. Confirm that the group has been given Read & Execute permission.
7. Click **OK** to close each open dialog box.

► **Task 8: Define which roles and users have access to a resource**

- Add **Sales, Consultants, Auditors, and Bobby Moore** to the **ACL_Sales Folders_Read** group.

Results: After this exercise, you will have a simple role-based management implementation to manage Read access to the Sales folder.

Exercise 2: Manage Group Membership from the Command Prompt

In this exercise, you will manage group membership from the command prompt using commands such as DSGet and DSMod.

The main tasks for this exercise are as follows:

1. Modify group membership with DSMod.
2. Retrieve group membership with DSGet.

► Task 1: Modify group membership with DSMod

1. Switch to the command prompt. Type the following command on one line, and then press ENTER.

```
dsmod group "CN=Auditors,OU=Role,OU=Groups,DC=contoso,DC=com" -  
addmbr "CN=Mike Danseglio,OU=Employees,OU=User Accounts,  
DC=contoso,DC=com" "CN=Finance Managers,OU=Role,  
OU=Groups,DC=contoso,DC=com"
```

2. In **Active Directory Users and Computers**, confirm that the membership of the **Auditors** group includes **Mike Danseglio** and the **Finance Managers** group.

► Task 2: Retrieve group membership with DSGet

1. Switch to the command prompt.
2. List the direct members of the **Auditors** group by typing the following command, and then pressing ENTER:

```
dsget group "CN=Auditors,OU=Role,OU=Groups,DC=contoso,DC=com"  
-members
```

3. List the full list of members of the **Auditors** group by typing the following command, and then pressing ENTER:

```
dsget group "CN=Auditors,OU=Role,OU=Groups,DC=contoso,DC=com"  
-members -expand
```

4. List the full list of members of the **ACL_Sales Folders_Read** group by typing the following command, and then pressing ENTER:

```
dsget group "CN=ACL_Sales Folders_Read,OU=Access,  
OU=Groups,DC=contoso,DC=com" -members -expand
```

5. List the direct group membership of **Mike Danseglio** by typing the following command, and then pressing ENTER:

```
dsget user "CN=Mike Danseglio,OU=Employees,OU=User Accounts,  
DC=contoso,DC=com" -memberof
```

6. List the full group membership of **Mike Danseglio** by typing the following command on one line, and then pressing ENTER:

```
dsget user "CN=Mike Danseglio,OU=Employees,OU=User Accounts,  
DC=contoso,DC=com" -memberof -expand
```

Results: After this exercise, you will have a simple role-based management implementation to manage Read access to the Sales folder.

Exercise 3 (Advanced Optional): Explore Group Membership Reporting Tools

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

The main tasks for this exercise are as follows:

1. Open **D:\AdminTools\Members_Report.hta**. Enter the name of a group, and then click **Report**.
2. Open **D:\AdminTools\MemberOf_Report.hta**. Enter the name of a user, computer, or group, and then click **Report**.

Exercise 4 (Advanced Optional): Understand "Account Unknown" Permissions

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

The main tasks for this exercise are as follows:

1. In the **Role** OU, create a global security group named **Test**.
2. Give the group **Read & Execute** permission to the **D:\Data\Sales** folder.
3. Delete the group named **Test**.
4. Examine the **Security** tab of the Sales folder's properties dialog box. If you still see the Test group listed, Windows Explorer may be caching the mapping of the SID to the group name. Log off, log on, and check again.



Note: Do not shut down the virtual machines once you are finished with this lab as the settings you have configured here will be used in the Lab B.

Lab Review Questions

Question: Describe the purpose of global groups in terms of role-based management.

Question: What types of objects can be members of global groups?

Question: Describe the purpose of domain local groups in terms of role-based management of resource access.

Question: What types of objects can be members of domain local groups?

Question: If you have implemented role-based management and are asked to report who can read the Sales folders, what command would you use to do so?

Lesson 3

Best Practices for Group Management

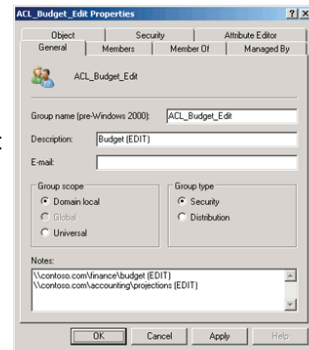
- Best Practices for Group Documentation
- Protect Groups from Accidental Deletion
- Delegate Membership Management with the Managed By Tab
- Default Groups
- Special Identities

After completing this lesson, you will be able to:

- Document the purpose of a group using the group's attributes.
- Protect a group from accidental deletion.
- Delegate group membership management using the Managed By tab.
- Understand shadow groups.
- Understand default (Builtin) groups.
- Understand special identities.

Best Practices for Group Documentation

- Why document groups?
 - Easier to find them when you need them
 - Easier to understand how and when to use a group
- Establish and adhere to a strict naming convention
 - Prefix, for example, helps distinguish APP_Budget from ACL_Budget_Edit
 - Prefix helps you *find* the group in the Select dialog box
- Summarize a group's purpose with its description
 - Appears in Active Directory Users and Computers details pane
- Detail a group's purpose in its Notes field

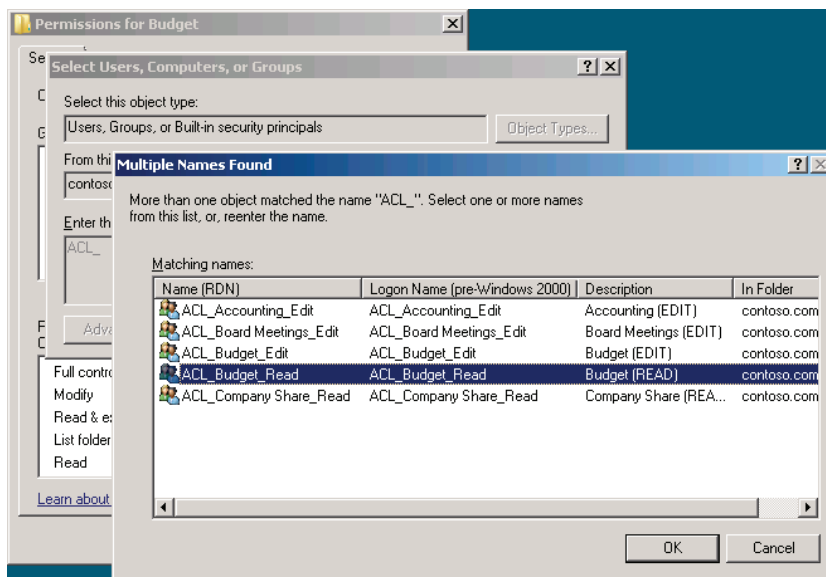


Key Points

Creating a group in Active Directory is easy. It is not so easy to make sure that the group is used correctly over time. You can facilitate the correct management and use of a group by documenting its purpose, to help administrators understand how and when to use the group. There are several best practices which, although unlikely to be addressed by the certification exam, will prove immensely useful to your enterprise group administration.

Establish and Adhere To a Strict Naming Convention

An earlier lesson addressed a suggested naming convention. In the context of ongoing group administration, establishing and following group naming standards increases administrative productivity. Using prefixes to indicate the purpose of a group, and using a consistent delimiter between the prefix and the descriptive part of the group names, can help users locate the correct group for a particular purpose. For example, the prefix APP can be used to designate groups that are used to manage applications, and the prefix ACL can be used for groups that are assigned permissions on access control lists (ACLs). With such prefixes, it becomes easier to locate and interpret the purpose of groups named, for example, APP_Accounting versus ACL_Accounting_Read—the former is used to manage the deployment of the accounting software, and the latter to provide Read access to the accounting folder. Prefixes also help to group the names of groups in the user interface. The screen shot below shows an example:



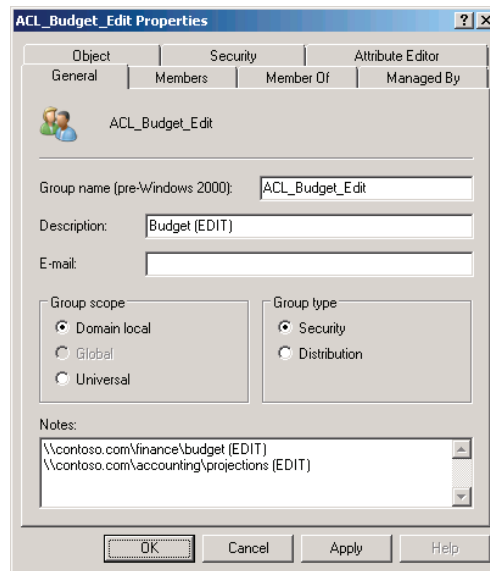
When attempting to locate a group to use in assigning permissions to a folder, you can type the prefix ACL_ in the Select dialog box and click OK. A Multiple Items Found dialog box appears showing only the ACL_ groups in the directory, thereby ensuring that permissions will be assigned to a group that is designed to manage resource access.

Summarize a Group's Purpose with its Description Attribute

Use the description attribute of a group to summarize the group's purpose. Because the Description column is enabled by default in the details pane of the Active Directory Users and Computers snap-in, the group's purpose can be highly visible to administrators.

Detail a Group's Purpose in its Notes

When you open a group's Properties dialog box, the Notes field is visible at the bottom of the General tab. This field can be used to document the group's purpose. For example, you can list the folders to which a group has been given permission, as shown below:



Protect Groups from Accidental Deletion

1. In the Active Directory Users and Computers snap-in, click the **View** menu and make sure that **Advanced Features** is selected.
2. Open the **Properties** dialog box for a group.
3. On the **Object** tab, select the **Protect Object From Accidental Deletion** check box.
4. Click **OK**.

Key Points

Protect yourself from the potentially devastating results of deleting a group by protecting each group you create from deletion. Windows Server® 2008 makes it easy to protect any object from accidental deletion.

To protect an object, follow these steps:

1. In the **Active Directory Users and Computers** snap-in, click the **View** menu and make sure that **Advanced Features** is selected.
2. Open the **Properties** dialog box for a group.
3. On the **Object** tab, select the **Protect Object From Accidental Deletion** check box.
4. Click **OK**.

This is one of the few places in Windows in which you actually have to click OK. Clicking Apply does not modify the ACL based on your selection.

The Protect Object From Accidental Deletion option applies an access control entry (ACE) to the ACL of the object that explicitly denies the Everyone group both the Delete permission and the Delete Subtree permission. If you really do want to delete the group, you can return to the Object tab of the Properties dialog box and clear the Protect Object From Accidental Deletion check box.

Deleting a group has a high impact on administrators and, potentially, on security. Consider a group that has been used to manage access to resources. If the group is deleted, access to that resource is changed. Either users who should be able to access the resource are suddenly prevented from access, creating a denial-of-service scenario, or if you had used the group to deny access to a resource with a Deny permission, inappropriate access to the resource becomes possible.

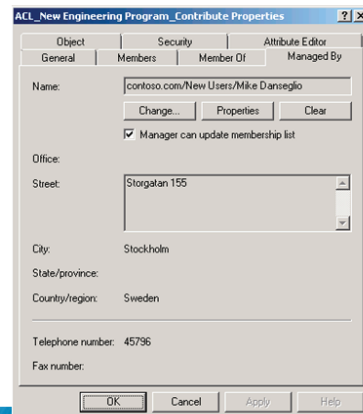
Additionally, if you re-create the group, the new group object will have a new security identifier (SID), which will not match the SIDs on ACLs of resources. So you must instead perform object recovery to reanimate the deleted group before the tombstone interval is reached. When a group has been deleted for the tombstone interval—60 days by default—the group and its SID are permanently deleted from Active Directory. When you reanimate a tombstoned object, you must re-create most of its attributes including, importantly, the member attribute of group objects. That means you must rebuild the group membership after restoring the deleted object. Alternatively, you can perform an authoritative restore or, in Windows Server 2008, turn to your Active Directory snapshots to recover both the group and its membership. Authoritative restore and snapshots are discussed in Module 13.

You can learn more about recovering deleted groups and their memberships in Knowledge Base article 840001, which you can find at <http://go.microsoft.com/fwlink/?LinkId=168758>.

In any event, it is safe to say that recovering a deleted group is a skill you should hope to use only in disaster recovery fire drills, not in a production environment.

Delegate Membership Management with the Managed By Tab

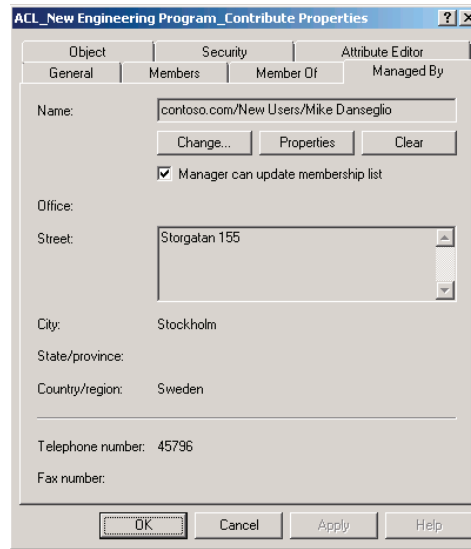
- The Managed By tab serves two purposes:
 - Provide contact information for who manages the group
 - Allow specified user (or group) to modify group membership if Manager Can Update Membership List is selected
- Tips
 - Must click OK (not just Apply) to change the ACL on the group
 - To set a group in the Name box, click Change, then click Object Types, and then click Groups



Key Points

After a group has been created, you might want to delegate the management of the group's membership to a team or an individual who has the business responsibility for the resource that the group manages. For example, let's assume that your finance manager is responsible for creating next year's budget. You create a shared folder for the budget and assign Write permission to a group named ACL_Budget_Edit. If someone needs access to the budget folder, he or she contacts the help desk to enter a request, the help desk contacts the finance manager for business approval, and then the help desk adds the user to the ACL_Budget_Edit group. You can improve the responsiveness and accountability of the process by allowing the finance manager to change the group's membership. Then users needing access can request access directly from the finance manager, who can make the change, removing the intermediate step of contacting the help desk. To delegate the management of a group's membership, you must assign to the finance manager the Allow Write Member permission for the group. The member attribute is the multivalued attribute that is the group's membership.

The easiest way to delegate membership management of a single group is to use the Managed By tab. The Managed By tab of a group object's Properties dialog box is shown here:



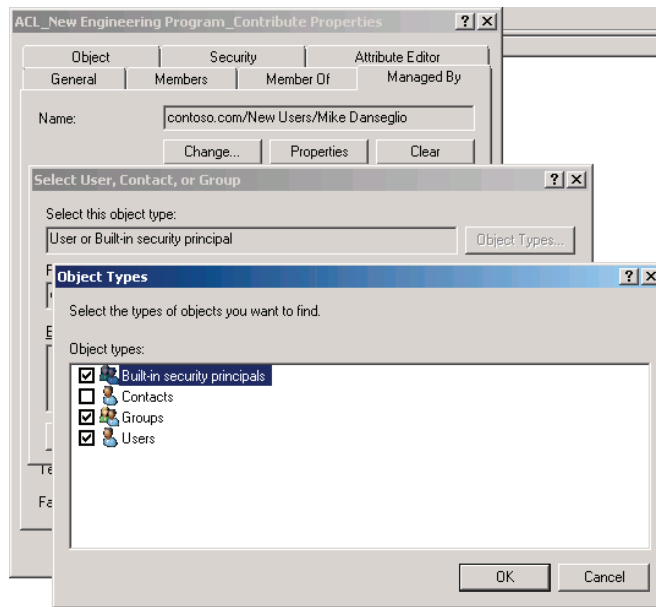
The Managed By tab serves two purposes. First, it provides contact information related to the manager of a group. You can use this information to contact the business owner of a group to obtain approval prior to adding a user to the group.

The second purpose served by the Managed By tab is to manage the delegation of the member attribute. Note the check box shown above. It is labeled **Manager can update membership list**. When selected, the user or group shown in the Name box is given the Allow Write Member permission. If you change or clear the manager, the appropriate change is made to the group's ACL.



Tip: You must actually click OK to implement the change. Clicking Apply does not change the ACL on the group.

It is not quite so easy to insert a group into the Managed By tab of another group. When you click the Change button, the Select User, Contact, Or Group dialog box appears. If you enter the name of a group and click OK, an error occurs. That's because this dialog box is not configured to accept groups as valid object types, even though "Group" is in the name of the dialog box itself. To work around this odd limitation, click the Object Types button, and then select the check box next to Groups. Click OK to close both the Object Types and Select dialog boxes. Be sure to select the Manager Can Update Membership List check box if you want to assign the Allow Write Member permission to the group. When a group is used on the Managed By tab, no contact information is visible, as groups do not maintain contact-related attributes.



After you have delegated group membership management, a user does not require Active Directory Users and Computers to modify the membership of the group. A user can simply use the **Search Active Directory** capability of Windows clients to find the group, then change its membership.

To find a group:

1. Click **Start**, then click **Network**.
2. Click the **Search Active Directory** button in the toolbar.
3. Type the name of the group and click **Find Now**.

Default Groups

- Default local groups in the BUILTIN and Users containers
 - Enterprise Admins, Schema Admins, Administrators, Domain Admins, Server Operators, Account Operators, Backup Operators, Print Operators
- Reference to their rights and privileges in Student Manual
 - Know these rights for the certification exams
- Problems with these groups
 - Highly overdelegated
 - Account Operators, for example, can log on to a DC
 - Protected
 - Users who are members of these groups become protected and are not un-protected when removed
- Best practice: Keep these groups empty and create custom groups with the rights and privileges you require

Key Points

There are a number of groups that are created automatically on a Windows Server 2008 server. These are called *default local groups*, and they include well-known groups such as Administrators, Backup Operators, and Remote Desktop Users. There are additional groups that are created in a domain, both in the BUILTIN and Users containers, including Domain Admins, Enterprise Admins, and Schema Admins. The following list provides a summary of capabilities of the subset of default groups that have significant permissions and user rights related to the management of Active Directory.

Enterprise Admins (Users Container of the Forest Root Domain)

This group is a member of the Administrators group in every domain in the forest, giving it complete access to the configuration of all domain controllers. It also owns the Configuration partition of the directory and has full control of the domain naming context in all forest domains.

Schema Admins (Users Container of the Forest Root Domain)

This group owns and has full control of the Active Directory schema.

Administrators (BUILTIN Container of Each Domain)

This group has complete control over all domain controllers and data in the domain naming context. It can change the membership of all other administrative groups in the domain, and the Administrators group in the forest root domain can change the membership of Enterprise Admins, Schema Admins, and Domain Admins. The Administrators group in the forest root domain is arguably the most powerful service administration group in the forest.

Domain Admins (Users Container of Each Domain)

This group is added to the Administrators group of its domain. It therefore inherits all of the capabilities of the Administrators group. It is also, by default, added to the local Administrators group of each domain member computer, giving Domain Admins ownership of all domain computers.

Server Operators (BUILTIN Container of Each Domain)

This group can perform maintenance tasks on domain controllers. It has the right to log on locally, start and stop services, perform backup and restore operations, format disks, create or delete shares, and shut down domain controllers. By default, this group has no members.

Account Operators (BUILTIN Container of Each Domain)

This group can create, modify, and delete accounts for users, groups, and computers located in any organizational unit in the domain (except the Domain Controllers OU), as well as in the Users and Computers container. Account Operators cannot modify accounts that are members of the Administrators or Domain Admins groups, nor can they modify those groups. Account Operators can also log on locally to domain controllers. By default, this group has no members.

Backup Operators (BUILTIN Container of Each Domain)

This group can perform backup and restore operations on domain controllers, as well as log on locally and shut down domain controllers. By default, this group has no members.

Print Operators (BUILTIN Container of Each Domain)

This group can maintain print queues on domain controllers. It can also log on locally and shut down domain controllers.

The default groups that provide administrative privileges should be managed carefully, because they typically have broader privileges than are necessary for most delegated environments; and because they often apply protection to their members.

The Account Operators group is a perfect example. If you examine its capabilities in the list above, you will see that its rights are very broad indeed. It can even log on locally to a domain controller. In very small enterprises, such rights would probably be appropriate for one or two individuals who would probably be domain administrators anyway. In larger enterprises, the rights and permissions granted to Account Operators are usually far too broad.

Additionally, Account Operators is, like the other administrative groups listed above, a protected group.

Protected groups are defined by the operating system and cannot be un-protected. Members of a protected group become protected. The result of protection is that the permissions (ACLs) of members are modified so that they no longer inherit permissions from their OU, but rather receive a copy of an ACL that is quite restrictive. For example, if Jeff Ford is added to the Account Operators group, his account becomes protected, and the help desk, which can reset all other user passwords in the Employees OU, cannot reset Jeff Ford's password.

For more information about protected accounts, see Knowledge Base article 817433 at <http://go.microsoft.com/fwlink/?LinkId=168759> and Knowledge Base article 840001 at <http://go.microsoft.com/fwlink/?LinkId=168760>. If you want to search the Internet for resources, use the keyword adminSDHolder.

For these reasons—overdelegation and protection—you should strive to *avoid adding users to the groups listed above that do not have members by default: Account Operators, Backup Operators, Server Operators, and Print Operators. Instead, create custom groups* to which you assign permissions and user rights that achieve your business and administrative requirements.

For example, if Scott Mitchell should be able to perform backup operations on a domain controller, but should not be able to perform restore operations that could lead to database rollback or corruption, and should not be able to shut down a domain controller, don't put Scott in the Backup Operators group. Instead, create a group and assign it only the Backup Files And Directories user right, then add Scott as a member.

There is an exhaustive reference to the default groups in a domain and to the default local groups on Microsoft TechNet. If you are not familiar with the default groups and their capabilities, you should prepare for the examination by reading them. The default domain groups reference is at <http://go.microsoft.com/fwlink/?LinkId=168761> and the default local groups reference is at <http://go.microsoft.com/fwlink/?LinkId=168762>.

Special Identities

- **Membership is controlled by Windows:**
 - Cannot be viewed, edited, or added to other groups
 - Can be used on ACLs
- **Examples**
 - **Anonymous Logon.** Represents connections to a computer without a username and password
 - **Authenticated Users.** Represents identities that have been authenticated, but does not include the Guest identity
 - **Everyone.** Includes Authenticated Users and Guest (but *not* Anonymous Logon by default in Windows Server 2003/2008)
 - **Interactive.** Users logged on locally or with Remote Desktop
 - **Network.** Users accessing a resource over the network

Key Points

Windows and Active Directory also support special identities, groups for which membership is controlled by the operating system. You cannot view the groups in any list (in the Active Directory Users and Computers snap-in, for example), you cannot view or modify the membership of these special identities, and you cannot add them to other groups. You can, however, use these groups to assign rights and permissions. The most important special identities, often referred to as *groups*, for convenience, are described in the following list:

- **Anonymous Logon.** This identity represents connections to a computer and its resources that are made without supplying a username and password. Prior to Windows Server 2003, this group was a member of the Everyone group. Beginning in Windows Server 2003, this group is no longer a default member of the Everyone group.
- **Authenticated Users.** This represents identities that have been authenticated. This group does not include Guest, even if the Guest account has a password.

- **Everyone.** This identity includes Authenticated Users and the Guest account. On computers running versions of Windows earlier than Windows Server 2003, this group includes Anonymous Logon.
- **Interactive.** This represents users accessing a resource while logged on locally to the computer that is hosting the resource, as opposed to accessing the resource over the network. When a user accesses any given resource on a computer to which the user is logged on locally, the user is automatically added to the Interactive group for that resource. Interactive also includes users logged on through a Remote Desktop connection.
- **Network.** This represents users accessing a resource over the network, as opposed to users who are logged on locally at the computer that is hosting the resource. When a user accesses any given resource over the network, the user is automatically added to the Network group for that resource.

The importance of these special identities is that they allow you to provide access to resources based on the type of authentication or connection, rather than the user account. For example, you could create a folder on a system that allows users to view its contents when they are logged on locally to the system, but that does not allow the same users to view the contents from a mapped drive over the network. This would be achieved by assigning permissions to the Interactive special identity.

Lab B: Best Practices for Group Management

- Exercise 1: Implement best practices for group management

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

Your implementation of role-based management at Contoso has been highly successful. As the number of groups in the domain has increased, you've come to realize that it is important to document groups thoroughly and to prevent administrators from accidentally deleting a group. Finally, you want to allow the business owners of resources to manage access to those resources by delegating to those owners the right to modify the membership of appropriate groups.

Exercise 1: Implement Best Practices for Group Management

In this exercise, you will perform the following tasks to document, delegate, and secure groups:

1. Prepare for the lab.
2. Create a well-documented group.
3. Protect a group from accidental deletion.
4. Delegate group membership management.
5. Validate the delegation of group membership management.

► Task 1: Prepare for the lab

The virtual machine should already be started and available after completing Lab A. However, if it is not, you should launch it and complete the exercises in Lab A before continuing.

1. Start 6425B-HQDC01-A.
2. Log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.

► Task 2: Create a well-documented group

1. Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. In the properties of the **ACL_Sales Folders_Read** group, configure the following:
 - A **Description** that summarizes the resource management rule represented by the group: **Sales Folders (READ)**
 - In the **Notes** box, type the following paths to represent the folders that have permissions assigned to this group:
\\contoso\teams\Sales (READ)
\\file02\data\Sales (READ)
\\file03\news\Sales (READ)

► **Task 3: Protect a group from accidental deletion**

1. Enable the **Advanced Features** view of the **Active Directory Users and Computers** snap-in.
2. Protect the **ACL_Sales Folders_Read** group from being accidentally deleted.
3. Attempt to delete the group. Confirm that the attempt to delete the group is denied.

► **Task 4: Delegate group membership management**

- Configure the **Managed By** attribute of **Auditors** to refer to **Mike Danseglio**.

► **Task 5: Validate the delegation of group membership management**

1. Log off of HQDC01, then log on with username **Mike.Danseglio** and password **Pa\$\$w0rd**.
2. Open the **Network** window and use **Search Active Directory** to locate the **Auditors** group.
3. Add the **Executives** group to the **Auditors** group.
4. Log off HQDC01.



Note: After completing this exercise, turn off all virtual machines and discard undo disks.

Exercise 2 (Advanced Optional): Maintain Shadow Group Membership

Advanced Optional exercises provide additional challenges for students who are able to complete lab exercises quickly. There are no answers in the Lab Answer Key.

The main tasks for this exercise are as follows:

1. Confirm that a global security group called **Administrative Identities** exists in the **Groups\Role** OU.

This will be a shadow group that contains accounts in the Admin Identities OU.

2. Create a batch script that uses DSQuery, DSGet, and DSMod to synchronize the group's membership with the users in the Admin Identities OU.

The script will be run at regular intervals to keep the group's membership synchronized with the Admin Identities OU. The script must account for new users added to the OU, and for users removed from the OU.

3. Test the script by performing the following steps using administrative credentials:
 - a. Run the script.
 - b. Confirm that the membership of the Administrative Identities group is the same as the contents of the Admin Identities OU.
 - c. Create a new user account in the Admin Identities OU.
 - d. Run the script.
 - e. Confirm that the group contains the newly added user.
 - f. Disable the new user account and move the user into the Disabled Accounts OU.
 - g. Run the script.
 - h. Confirm that the group no longer contains the user.
4. Time permitting, create a Scheduled Task that runs the script once every minute. Repeat the test sequence, but instead of running the script, allow the scheduled task to run.

You can compare your batch script with
D:\Labfiles\Lab04b\Shadow_Group.bat.

Note that the DS Commands achieve the goal of maintaining a shadow group, but they do not do so particularly efficiently. Removing all members and re-adding all members generates a larger-than-necessary amount of replication traffic, particularly for a large shadow group with hundreds or thousands of members. You can use a scripting language, such as VBScript or Windows PowerShell, to create a more efficient script that updates, rather than replaces, group membership to account for changes. See the *Windows Administration Resource Kit: Productivity Solutions for IT Professionals* by Dan Holme (Microsoft Press, 2008) for a sample script.

Lab Review Questions

Question: What are some benefits of using the Description and Notes fields of a group?

Question: What are the advantages and disadvantages of delegating group membership?

Module 5

Support Computer Accounts

Contents:

Lesson 1: Create Computers and Joining the Domain	5-4
Lab A: Create Computers and Joining the Domain	5-34
Lesson 2: Administer Computer Objects and Accounts	5-42
Lab B: Administer Computer Objects and Accounts	5-62

Module Overview

- Create Computers and Join the Domain
- Administer Computer Objects and Accounts

Computers in a domain are security principals, like users. They have an account with a logon name and password that Windows® changes automatically every 30 days or so. They authenticate with the domain. They can belong to groups, have access to resources, and be configured by Group Policy. And, like users, computers sometimes lose track of their passwords, requiring a reset, or have accounts that need to be disabled or enabled.

Managing computers—both the objects in Active Directory and the physical devices—is one of the day-to-day tasks of most IT professionals. New systems are added to your organization, computers are taken offline for repairs, machines are exchanged between users or roles, and older equipment is retired or upgraded leading to the access of replacement systems. Each of these activities requires managing the identity of the computer represented by its object, or account, and Active Directory.

Unfortunately, most enterprises do not invest the same kind of care and process in the creation and management of computer accounts as they do for user accounts, even though both are security principals. In this chapter, you will learn how to create computer objects, which include attributes that are required for the objects to be accounts. You will learn how to support computer accounts through their life cycle, including configuration, troubleshooting, repairing, and de-provisioning computer objects. You will also deepen your understanding of the process through which a computer joins a domain, so that you can identify and avoid potential points of failure.

Objectives

After completing this module, you will be able to:

- Understand the relationship between a domain member and the domain, in terms of identity and access.
- Identify the requirements for joining a computer to the domain.
- Implement best-practice processes for computer joins.
- Secure AD DS to prevent the creation of unmanaged computer accounts.
- Manage computer objects and their attributes using the Windows Interface and command-line tools.
- Administer computer accounts through their life cycle.

Lesson 1

Create Computers and Join the Domain

- Workgroups, Domains, and Trusts
- Requirements for Joining a Computer to the Domain
- The Computer's Container and Organizational Units (OUs)
- Prestage a Computer Account
- Join a Computer to the Domain
- Secure Computer Creation and Joins
- Automate Computer Account Creation
- Import Computers with CSVDE
- Import Computers with LDIFDE
- Create Computers with DSAdd
- Create and Join Computers with NetDom

The default configuration of Windows Server® 2008—as well as all other versions of Windows server and client operating systems—is that the computer belongs to a workgroup. Before you can log on to a computer with a domain account, that computer must belong to the domain. To join the domain, the computer must have an account in the domain which, like a user account, includes a logon name (the sAMAccountName attribute), a password, and a security identifier (SID) that uniquely represents the computer as a security principal in the domain. Those credentials allow the computer to authenticate against the domain and to create a secure relationship that then allows users to log on to the system with domain accounts. In this lesson, you will learn the steps to prepare the domain for a new computer account, and you will explore the process through which a computer joins the domain.

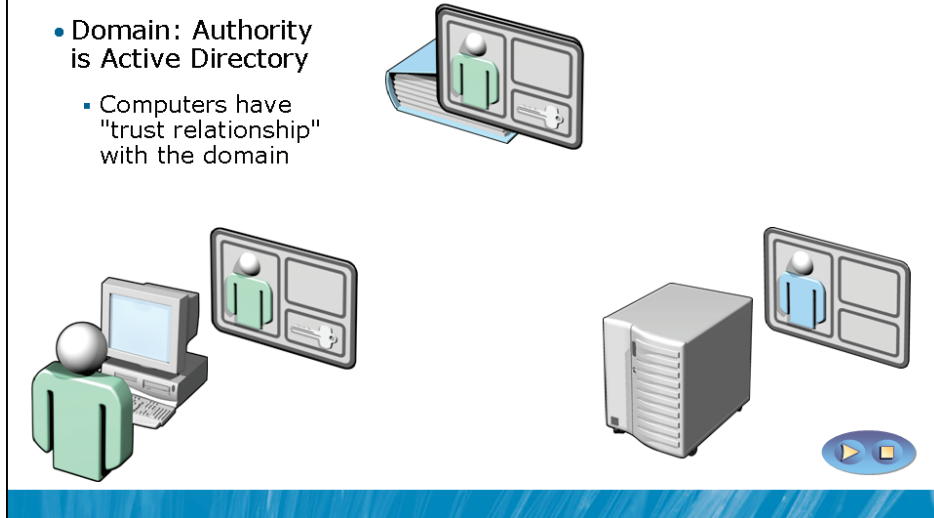
Objectives

After completing this lesson, you will be able to:

- Understand the relationship between a domain member and the domain, in terms of identity and access.
- Identify the requirements for joining a computer to the domain.
- Prestage a computer account.
- Join a computer to the domain.
- Redirect the default computer container.
- Prevent nonadministrative users from creating computers and joining the domain.
- Use command-line tools to import, create, and join computers.

Workgroups, Domains, and Trusts

- **Workgroup:** Authority is Security Accounts Manager (SAM)
 - Identity is local to each computer
- **Domain:** Authority is Active Directory
 - Computers have "trust relationship" with the domain



Key Points

In a workgroup, each system maintains an identity store of user and group accounts against which users can be authenticated and access can begin. The local identity store on each computer is called the Security Accounts Manager (SAM) database. If a user logs on to a workgroup machine the system authenticates the user against its local SAM database. If a user connects to another system, to access a file for example, the user is reauthenticated against the identity store of the remote system. From a security perspective, a workgroup computer is, for all intents and purposes, a stand-alone system.

When a computer joins a domain, it delegates the task of authenticating users to the domain. Although the computer continues to maintain its SAM database to support local user and group accounts, user accounts will typically be created in the central domain directory. When a user logs on to the computer with a domain account, the user is now authenticated by a domain controller, rather than by the SAM. Said another way, the computer now trusts another authority to validate a user's identity. Trust relationships are generally discussed in the context of two domains, as you will learn in another module, but there is also a trust between each domain member computer and its domain that is established when the computer joins the domain.

Requirements for Joining a Computer to the Domain

- A computer object must exist in the directory service
- You must have permissions to the computer object that allow you to join a computer to it
- You must be a member of the local Administrators group on the computer to change its domain or workgroup membership

Key Points

Three things are required for you to join a computer to an Active Directory domain:

- A computer object must be created in the directory service.
- You must have appropriate permissions to the computer object. The permissions allow you to join a computer with the same name as the object to the domain.
- You must be a member of the local Administrators group on the computer to change its domain or workgroup membership.

The remainder of this lesson examines each of these requirements.

The Computer's Container and Organizational Units (OUs)

- The default Computer's container is a *container*, not an *organizationalUnit* object
 - Cannot link Group Policy objects (GPOs) to a container
 - Cannot create sub-OUs in a container
- Best practice is to create OUs for computer objects
 - Servers
 - Typically subdivided by server role
 - Client computers
- Divide OUs based first on administration, then to facilitate configuration with Group Policy

Key Points

Before you create a computer object in the directory service—the first of the three requirements for joining a computer to the domain—you must have a place to put it.

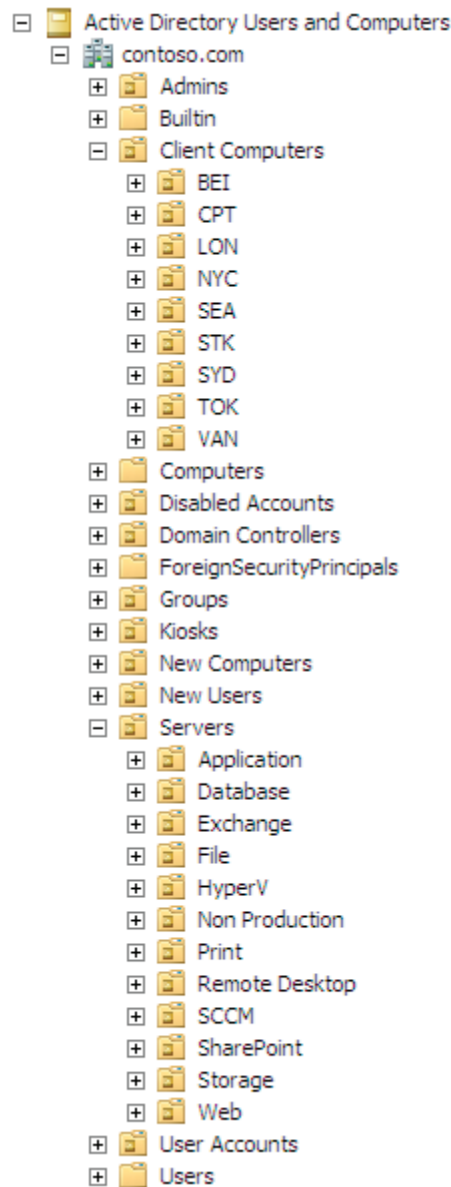
The Default Computers Container

When you create a domain, the Computers container is created by default (CN=Computers). This container is not an organizational unit (OU), it is an object of class container. There are subtle but important differences between a container and an OU: You cannot create an OU within a container, so you cannot subdivide the Computers OU; and you cannot link a Group Policy object to a container. Therefore, it is very highly recommended that you create custom OUs to host computer objects instead of using the Computers container.

OUs for Computers

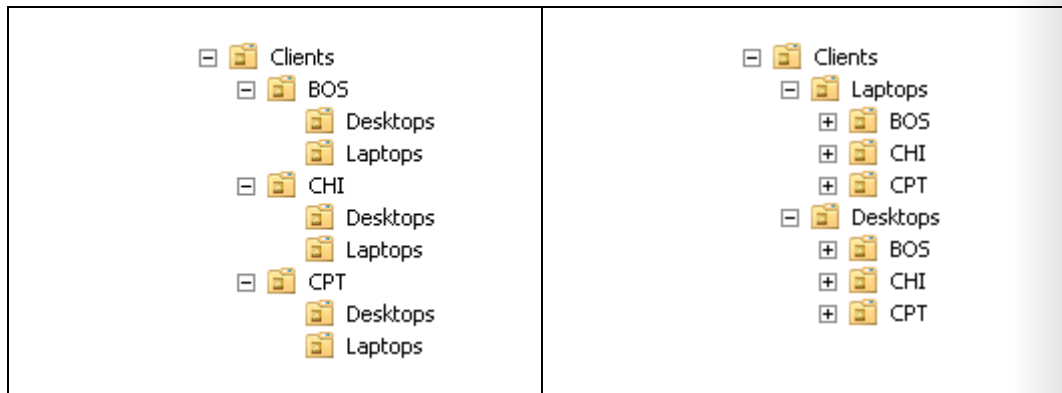
Most organizations create at least two OUs for computer objects: one to host computer accounts for client computers—desktops, laptops, and other user systems—and another for servers. These two OUs are in addition to the Domain Controllers OU created by default during the installation of Active Directory. In each of these OUs, computer objects are created. There is no technical difference between a computer object in a client's OU and a computer object in a server's or domain controller's OU: computer objects are computer objects. But separate OUs are typically created to provide unique scopes of management, so that you can delegate management of client objects to one team and management of server objects to another.

Your administrative model might necessitate further dividing your client and server OUs. Many organizations create sub-OUs beneath a server OU to collect and manage specific types of servers—for example, an OU for file and print servers and an OU for database servers. By doing so, the team of administrators for each type of server can be delegated permissions to manage computer objects in the appropriate OU. Similarly, geographically distributed organizations with local desktop support teams often divide a parent OU for clients into sub-OUs for each site. This approach enables each site's support team to create computer objects in the site for client computers and join computers to the domain using those computer objects. This is an example only: What is most important is that your OU structure reflects your administrative model so that your OUs provide single points of management for the delegation of administration.



Additionally, separate OUs allow you to create different baseline configurations using different GPOs linked to the client and the server OUs. Group Policy, discussed in detail in another module, allows you to specify configuration for collections of computers by linking GPOs that contain configuration instructions to OUs. It is common for organizations to separate clients into desktop and laptop OUs. GPOs specifying desktop or laptop configuration can then be linked to appropriate OUs.

If your organization has decentralized, site-based administration and wants to manage unique configurations for desktops and laptops, you face a design dilemma: Should you divide your clients OU based on administration and then subdivide desktops and laptops, or should you divide your clients OU into desktop and laptop OUs and then subdivide based on administration? The options are illustrated below.



Because the primary design driver for Active Directory OUs is the efficient delegation of administration through the inheritance of access control lists (ACLs) on OUs, the design on the left would be recommended.

Delegating Permission to Create Computers

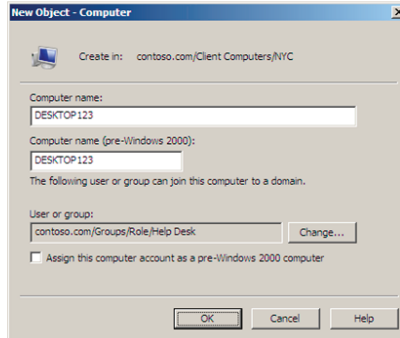
By default, the Enterprise Admins, Domain Admins, Administrators, and Account Operators groups have permission to create computer objects in any new OU. However, as discussed in the module about groups, it is recommended that you tightly restrict membership in the first three groups, and that you do not add administrators to the Account Operators group.

Instead, you should delegate the permission to create computer objects to appropriate administrators or support personnel. The permission required to create a computer object is Create Computer Objects. This permission, assigned to a group for an OU, allows members of the group to create computer objects in that OU. For example, you might allow your desktop support team to create computer objects in the clients OU, and allow your file server administrators to create computer objects in the file servers OU.

The permissions required to perform computer management tasks are listed in the topic, "Secure Computer Creation and Joins." Module 8 details the process of delegation.

Prestage a Computer Account

- Prestage (pre-create) a computer in the correct OU
- Right-click the OU and choose New → Computer

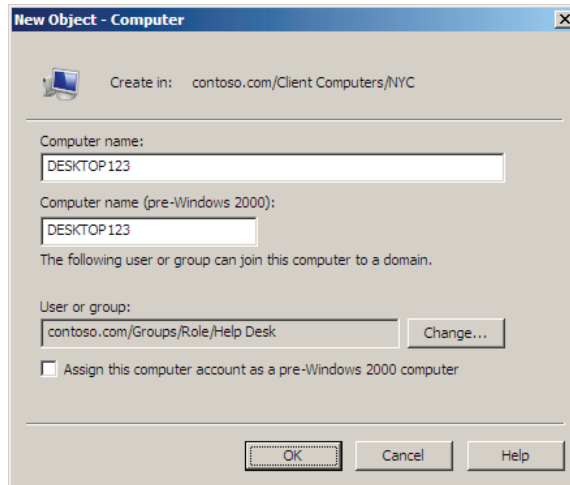


- Computer Name and Computer Name (Pre-Windows 2000) should be the same
- User Or Group box delegates permissions to the specified account to join the computer to the domain

Key Points

You can and should create a computer account in the correct OU before joining the computer to the domain. This process of creating a computer account in advance is called *prestaging* a computer.

After you have been given permission to create computer objects, you can do so by right-clicking the OU and choosing Computer from the New menu. The New Object – Computer dialog box, shown below, appears:



Enter the computer name, following the naming convention of your enterprise, and select the user or group that will be allowed to join the computer to the domain with this account. The two computer names—Computer Name and Computer Name (Pre-Windows 2000)—should be the same: There is very rarely if ever a justification for configuring them separately.



Note: The permissions that are applied to the user or group you select in the wizard are more than are necessary simply to join a computer to the domain. The selected user or group is also given the ability to modify the computer object in other ways. For guidance regarding a least privilege approach to delegating permission to join a computer to the domain, see *Windows Administration Resource Kit: Productivity Solutions for IT Professionals* by Dan Holme (Microsoft® Press, 2008).

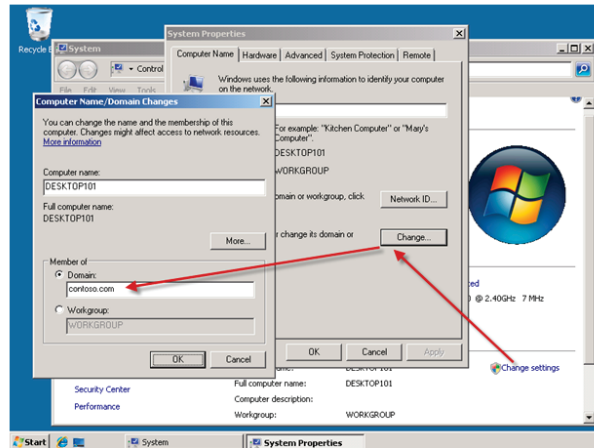
The process you complete to create a computer account before joining the computer to the domain is called *prestaging* the account.

There are two major advantages of prestaging a computer:

- The account is in the correct OU and is therefore delegated according to the security policy defined by the ACL of the OU.
- The computer is within the scope of GPOs linked to the OU, before the computer joins the domain.

Join a Computer to the Domain

- From the System Properties dialog box or window



- Prompted for domain credentials
- Requires restart

Key Points

By prestaging the computer object, you fulfill the first two requirements for joining a computer to a domain: the computer object exists, and you have specified who has permissions to join a computer with the same name to the domain. Now, a local administrator of the computer can change the computer's domain membership and enter the specified domain credentials to successfully complete the process.

To join a computer to the domain, follow these steps:

1. Log on to the computer with credentials that belong to the local Administrators group on the computer.

Only local administrators can alter the domain or workgroup membership of a computer.

2. Open the **System Properties** dialog box using one of the following methods:
Windows XP, Windows Server 2003:

- Open the **System properties** dialog box by doing one of the following:
 - Right-click **My Computer**, and then click **Properties**.
 - Press **Windows Logo+Pause**.

Windows Vista®, Windows Server 2008:

- a. Open the **System properties** dialog box by doing one of the following:
 - Right-click **Computer**, and then click **Properties**.
 - Press **Windows Logo+Pause**.
 - b. In the **Computer name, domain, and workgroup settings** section, click **Change Settings**.
 - c. If prompted by **User Account Control**, click **Continue** or enter administrative credentials as appropriate.
3. Click the **Computer Name** tab.
 4. Click **Change**.
 5. Under **Member Of**, click **Domain**.
 6. Type the name of the domain you want to join.



Note: Use the full DNS name of the domain. Not only is this more accurate and more likely to succeed, but if it does not succeed it indicates that there could be a problem with DNS name resolution that should be rectified before joining the machine to the domain.

7. Click **OK**.
8. Windows prompts for the credentials of your user account in the domain.

The domain checks to see if a computer object already exists with the name of the computer. One of the following three things happens:

- If the object exists and a computer with that name has already joined the domain, an error is returned, and you cannot join the computer to the domain.

- If the object exists and it is prestaged—a computer with the same name has not joined the domain—the domain confirms that the domain credentials you entered have permission to join the domain using that account. These permissions were discussed in the section, “Prestaging a Computer Account.”
- If the computer account is not prestaged, Windows checks to see if you have permissions to create a new computer object in the default computer container. If you do have permissions to create a new computer object in the default computer container, the object is created with the name of the computer. This method of joining a domain is supported for backwards compatibility but is not recommended. It is recommended to stage the account as indicated earlier, and as detailed in the next section, “Secure Computer Creation and Joins.”

The computer then joins the domain by assuming the identity of its Active Directory object. It configures its SID to match the domain computer account's SID and sets an initial password with the domain. The computer then performs other tasks related to joining the domain. It adds the Domain Admins group to the local Administrators group and the Domain Users group to the local Users group.

9. You are prompted to restart the computer. Click **OK** to close this message box.
10. Click **Close** (in Windows Vista) or **OK** (in Windows XP) to close the **System Properties** dialog box.
11. You are prompted, again, to restart the computer, after which the system is fully a member of the domain, and you can log on using domain credentials.

Secure Computer Creation and Joins

- Prestage computer objects in the correct OUs
 - Computer is in correct OU and does not require moving
 - Group Policy applies to the computer immediately after joining the domain
 - Tighter security of computer OU and Computers container
- Configure the default computer container
 - `redircmp "DN of OU for new computer objects"`
- Restrict the ability of users to create computers
 - By default, *any* user can join 10 machines to the domain
 - Requires no prestaging
 - Change the *ms-DS-MachineAccountQuota* value to 0
- Delegate to appropriate groups the permission to create computer objects in the appropriate OUs

Key Points

Prestage Computer Objects

The best practice is to prestage a computer account prior to joining the machine to the domain. Unfortunately, Windows allows you to join a computer to a domain without following best practice. You can log on to a workgroup machine as a local administrator and change the machine's membership to the domain. On demand, Windows creates a computer object in the default computer container, gives you permission to join a computer to that object, and then proceeds to join the system to the domain.

There are three problems with this behavior of Windows:

- First, the computer account created automatically by Windows is placed in the default computer container, which is *not where the computer object belongs* in most enterprises.
- Second, *you must move the computer* from the default computer container into the correct OU, which is an extra step that is *often forgotten*.

- Third, any domain user can also do this—no domain-level administrative permissions are required. *Any user can join any computer to the domain if you don't manage and secure the process.* Because a computer object is a security principal, and because the creator of a computer object owns the object and can change its attributes, this exposes a potential security vulnerability. The next sections detail these disadvantages.

Configuring the Default Computer Container

When you join a computer to the domain and the computer object does not already exist in Active Directory, Windows automatically creates a computer account in the default computer container, which is called Computers (CN=Computers,DC=domain, by default). The problem with this relates to the discussion of OU design earlier in the lesson. If you have implemented the best practices described there, you have delegated permissions to administer computer objects in specific OUs for clients and servers. Additionally, you might have linked GPOs to those OUs to manage the configuration of these computer objects. If a new computer object is created outside of those OUs, in the default computer container, the permissions and configuration it inherits from its parent container will be different than what it should have received. You will then need to remember to move the computer from the default container to the correct OU after joining the domain.

There are two recommended steps to reduce the likelihood of this problem. First, you should endeavor to always prestage computer accounts. If an account is prestaged for a computer in the correct OU, then when the computer joins the domain it will use the existing account and will be subject to the correct delegation and configuration.

Second, to reduce the impact of systems being joined to the domain without a prestaged account, you should change the default computer container so that it is not the Computers container itself, but instead is an OU that is subject to appropriate delegation and configuration. For example, if you have an OU called Clients, you can instruct Windows to use that OU as the default computer container, so that if computers are joined to the domain without prestaged accounts, the objects are created in the Clients OU.

The `redircmp.exe` command is used to redirect the default computer container with the following syntax:

```
redircmp "DN of OU for new computer objects"
```

Now, if a computer joins the domain without a prestaged computer account, Windows creates the computer object in the specified organizational unit.



Note: Redirecting the Default User Container. The same concepts apply to the creation of user accounts. By default, if a user account is created using a legacy practice that does not specify the OU for the account, the object is created in the default user container (CN=Users,DC=domain, by default). The `redirusr.exe` command can be used to redirect the default container to an actual OU that is delegated and configured appropriately. `Redirusr`, like `redircmp`, takes a single option: the distinguished name (DN) of the OU that will become the default user container.

Restricting the Ability of Users to Create Computers

When a computer account is prestaged, the permissions on the account determine who is allowed to join that computer to the domain. When an account is not prestaged, Windows will, by default, allow any authenticated user to create a computer object in the default computer container. In fact, Windows will allow any authenticated user to create 10 computer objects in the default computer container. The creator of a computer object, by default, has permission to join that computer to the domain. It is through this mechanism that any authenticated user can join 10 computers to the domain without any explicit permissions to do so.

The 10-computer quota is configured by the `ms-DS-MachineAccountQuota` attribute of the domain. It allows any authenticated user to join a machine to the domain, no questions asked. This is problematic from a security perspective because computers are security principals. And the creator of a security principal has permission to manage that computer's properties. In a way, the quota is like allowing any domain user to create 10 user accounts, without any controls.

It is highly recommended that you close this loophole, so that nonadministrative users cannot join machines to the domain. To change the `ms-DS-MachineAccountQuota` attribute, follow these steps:

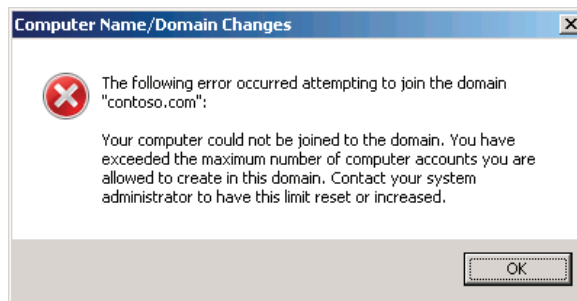
1. Open the ADSI Edit MMC console from the **Administrative Tools** folder.
2. Right-click **ADSI Edit**, and then click **Connect To**.
3. In the **Connection Point** section, click **Select A Well Known Naming Context**, and then select **Default Naming Context** from the drop-down list.
4. Click **OK**.
5. In the console tree, expand **Default Naming Context**.
6. Right-click the domain folder—"dc=contoso,dc=com", for example—and then click **Properties**.

7. Click **ms-DS-MachineAccountQuota** and click **Edit**.
8. Type **0**.
9. Click **OK**.

The Authenticated Users group also is assigned the user right to add workstations to the domain, but you do not have to modify this right if you have changed the default value of the ms-DS-MachineAccountQuota attribute.

After you have changed the ms-DS-MachineAccountQuota attribute to 0, you can be assured that the only users who can join computers to the domain are those who have been specifically delegated permission to join prestaged computer objects or to create new computer objects.

Once you've eliminated this loophole, you must make sure you have given appropriate administrators explicit permission to create computer objects in the correct OUs, as described in the "Delegating Permission to Create Computers" section, otherwise the error message shown here will appear.



Delegating Computer Management

The fourth task to improve the security of computer accounts is to delegate computer management tasks at the OU level. Delegation is discussed in Module 8. The following DSACLs commands can be used to delegate computer management tasks:

- Create a computer:

```
dsac1s "DN of OU" /I:T /G "DOMAIN\group":CC;computer
```

- Delete a computer:

```
dsac1s "DN of OU" /I:T /G "DOMAIN\group":DC;computer
```

- Join a computer to the domain:

```
dsacl "DN of OU" /I:S /G "DOMAIN\group":  
"Validated write to DNS host name";computer  
dsacl "DN of OU" /I:S /G "DOMAIN\group":  
"Validated write to service principal name";computer  
dsacl "DN of OU" /I:S /G "DOMAIN\group":  
CA;Reset Password;computer  
dsacl "DN of OU" /I:S /G "DOMAIN\group":  
WP;Account Restrictions;computer
```

The four commands listed above should be entered at the command prompt with no space after the colon.

- Move a computer

Requires permissions to delete computers in the source OU and create computers in the destination OU. Even though a move does not actually delete/create the account, that is the permission that is used by the Access Check.

Question: What two things determine whether you can join a computer account to the domain?

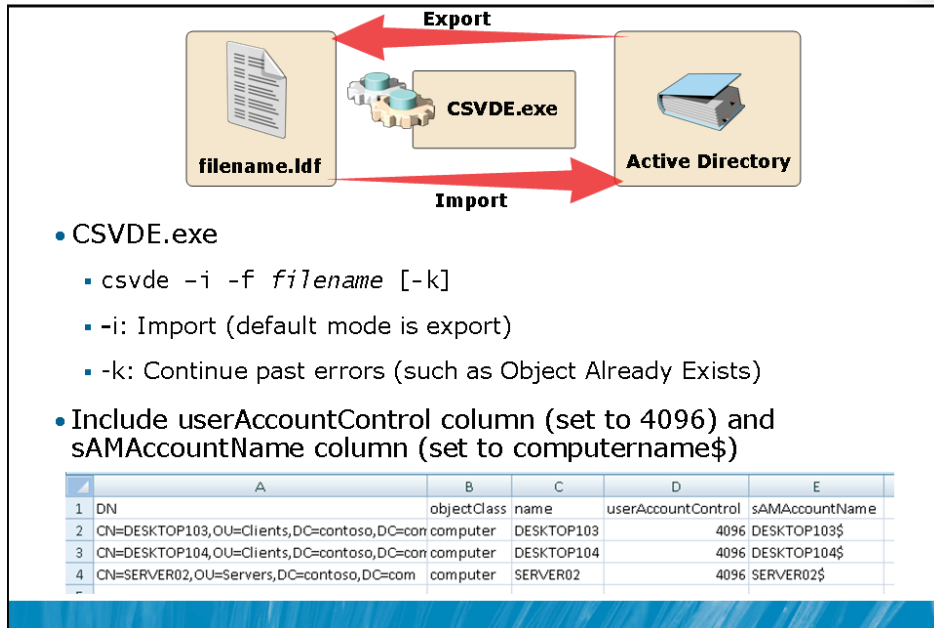
Automate Computer Account Creation

- Comma Separated Value Directory Exchange (CSVDE)
 - Import (create) or export computer accounts
- Lightweight Directory Access Protocol (LDAP) Data Interchange Format Directory Exchange (LDIFDE)
 - Import (create), modify, or export computer accounts
- DSAdd
 - Create computer accounts and set initial properties
- NetDom
 - Create computer accounts
 - Join machines to domain

Key Points

The steps you have learned for creating a computer account become burdensome if you are tasked with creating dozens or even hundreds of computer accounts at the same time. Commands such as CSVDE, LDIFDE, and DSAdd can import and automate the creation of computer objects. Scripts can also allow you to provision computer objects—that is, to perform business logic such as the enforcement of computer naming conventions.

Import Computers with CSVDE



Key Points

CSVDE is a command-line tool that imports or exports Active Directory objects from or to a comma-delimited text file (also known as a comma-separated value text file, or .csv file). The basic syntax of the CSVDE command is:

```
csvde [-i] [-f "Filename"] [-k]
```

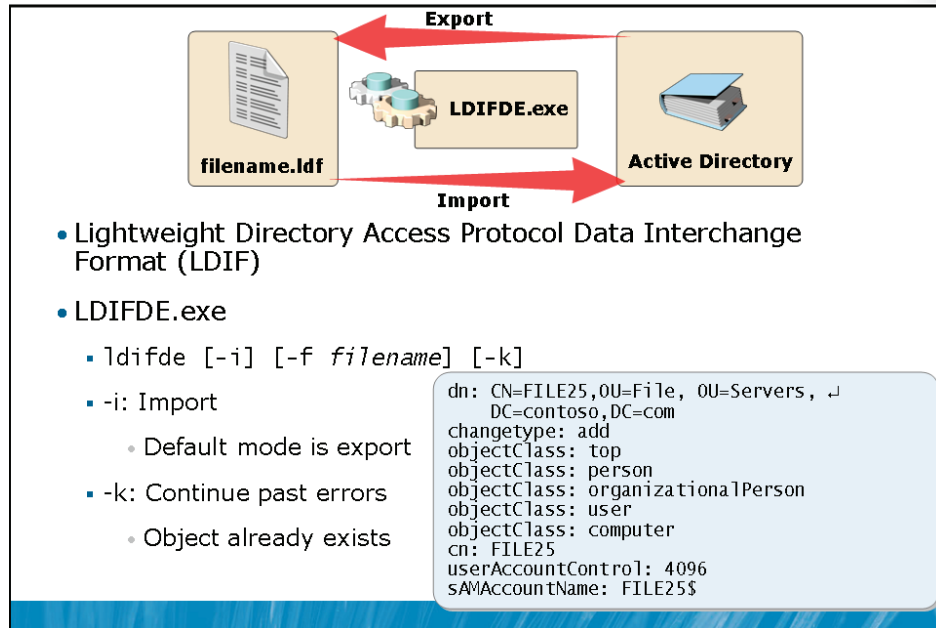
The -i option specifies import mode—without it, the default mode of CSVDE is export. The -f option identifies the file name to import from or export to. The -k option is useful during import operations, as it instructs CSVDE to ignore errors, including “object already exists,” “constraint violation,” and “attribute or value already exists.”

Comma-delimited files can be created, modified, and opened with tools as familiar as Notepad and Microsoft Office Excel®. The first line of the file defines the attributes by their Lightweight Directory Access Protocol (LDAP) attribute names. Each object follows, one per line, and must contain exactly the attributes listed on the first line. A sample file is shown in Excel below.

	A	B	C	D	E
1	DN	objectClass	name	userAccountControl	sAMAccountName
2	CN=DESKTOP103,OU=Clients,DC=contoso,DC=com	computer	DESKTOP103	4096	DESKTOP103\$
3	CN=DESKTOP104,OU=Clients,DC=contoso,DC=com	computer	DESKTOP104	4096	DESKTOP104\$
4	CN=SERVER02,OU=Servers,DC=contoso,DC=com	computer	SERVER02	4096	SERVER02\$

When importing computers, be sure to include the userAccountControl attribute, and set it to 4096. This attribute ensures that the computer will be able to join the account. Also include the pre-Windows 2000 logon name of the computer, the sAMAccountName attribute, which is the name of the computer followed by a dollar sign (\$), as shown above.

Import Computers with LDIFDE



Key Points

LDIFDE.exe imports data from files in the Lightweight Directory Access Protocol Data Interchange Format (LDIF) format. LDIF files are text files within which operations are specified by a block of lines separated by a blank line. Each operation begins with the DN attribute of the object that is the target of the operation. The next line, `changetype`, specifies the type of operation: add, modify, or delete.

The listing below is an LDIF file that will create a computer account in the Servers OU:

```
dn: CN=FILE25,OU=File,OU=Servers,DC=contoso,DC=com
changetype: add
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: user
objectClass: computer
cn: FILE25
userAccountControl: 4096
sAMAccountName: FILE25$
```

The basic syntax of the LDIFDE command is similar to that of the CSVDE command:

```
ldifde [-i] [-f "Filename"] [-k]
```

By default, LDIFDE is in export mode. The -i option specifies import mode. You must specify -f to identify the file you are using for import or export. LDIFDE will stop when it encounters errors unless you specify the -k option, in which case LDIFDE continues processing.

Create Computers with DSAdd

- DSAdd creates objects in Active Directory

```
dsadd computer ComputerDN
```

- ComputerDN: The distinguished name (DN) of the computer

Multiple values can be provided by:

- Separating ComputerDN ComputerDN... with a space
- Leaving ComputerDN empty, then entering DNs one at a time followed by ENTER, with CTRL+Z and then ENTER after the last DN
- Piping a list of DNs from another command, such as DSQuery

- Optional options

- -samid ComputerName
- -desc Description
- -loc Location

Key Points

The DSAdd command is used to create objects in Active Directory. To create computer objects, simply type:

```
dsadd computer ComputerDN
```

where ComputerDN is the distinguished name (DN) of the computer, such as CN=DESKTOP123,OU=NYC,OU=Client Computers,DC=contoso,DC=com.

If the computer's DN includes a space, surround the entire DN with quotation marks. The ComputerDN option can include more than one distinguished name for new computer objects, making DSAdd Computer a handy way to generate multiple objects at once. The option can be entered in one of the following ways:

- By typing each DN at the command prompt, separated by spaces.

- By leaving the DN option empty, at which point you can type the DNs, one at a time, at the keyboard console of the command prompt. Press ENTER after each DN. Press CTRL+Z and ENTER after the last DN.
- By piping a list of DNs from another command, such as DSQuery.

The DSAdd Computer command can take the following optional options after the DN option:

- -samid ComputerName
- -desc Description
- -loc Location

Create and Join Computers with NetDom

- Create account
 - `netdom add ComputerName /domain:DomainName [/ou:"OUDN"] [/UserD:DomainUsername /PasswordD:DomainPassword]`
- Join the domain (and, if necessary, create account)
 - `netdom join MachineName /Domain:DomainName [/OU:"OUDN"] [/UserD:DomainUsername] [/PasswordD:{DomainPassword}*}] [/User0:LocalUsername] [/Password0:{LocalPassword}*}] [/SecurePasswordPrompt] [/REBoot[:TimeInSeconds]]`

Key Points

The NetDom command is also able to perform a variety of domain account and security tasks from the command prompt. You can also use NetDom to create a computer account, by typing the following command:

```
netdom add ComputerName /domain:DomainName [/ou:"OUDN"] [/UserD:DomainUsername /PasswordD:DomainPassword]
```

This command creates the computer account for ComputerName in the domain indicated by the /domain option, using the credentials specified by /UserD and /PasswordD. The /ou option causes the object to be created in the OU specified by the organizational unit distinguished name (OUDN) distinguished name following the option. If no OUDN is supplied, the computer account is created in the default computer container. The user credentials must, of course, have permissions to create computer objects.

Using NetDom.exe

The NetDom.exe command allows you to join a computer to the domain from the command prompt. The basic syntax of the command is:

```
netdom join MachineName /Domain:DomainName [/OU:"OUDN"]
           [/UserD:DomainUsername] [/PasswordD:{DomainPassword|*} ]
           [/UserO:LocalUsername] [/PasswordO:{LocalPassword|*} ]
           [/SecurePasswordPrompt]
           [/REBoot[:TimeInSeconds]]
```

It can be useful to join a machine to a domain from the command prompt. The first reason this is useful is because the join can be included in a script that performs other actions. For example, you could create a batch file that creates the computer account using NetDom or DSAdd—the latter of which allows you to specify other attributes, including description—and then joins the machine to that account using NetDom. Second, NetDom.exe can be used to remotely join a machine to the domain. Third, NetDom.exe allows you to specify the OU for the computer object. The command's options are, for the most part, self explanatory. /UserO and /PasswordO are credentials that are members of the workgroup computer's local Administrators group. Specifying * for the password causes NetDom.exe to prompt for the password at the command prompt. /UserD and /PasswordD are domain credentials with permission to create a computer object, if the account is not prestaged, or to join a computer to a prestaged account. The /REBoot option causes the system to reboot after joining the domain. The default timeout is 30 seconds. The /SecurePasswordPrompt option displays a popup for credentials when * is specified for either /PasswordO or /PasswordD.



Note: If you want to be able to use NetDom remotely, the Windows Firewall configuration on the computer that will be joined to the domain must allow Network Discovery and Remote Administration.

Lab A: Create Computers and Join the Domain

- Exercise 1: Join a Computer to the Domain with the Windows Interface
- Exercise 2: Secure Computer Joins
- Exercise 3: Manage Computer Account Creation with Best Practices

Logon information

Virtual machine	6425B-HQDC01-A	6425B-SERVER01-B
Logon user name	Pat.Coleman	
Administrative user name	Pat.Coleman_Admin	Administrator
Password	Pa\$\$w0rd	Pa\$\$w0rd

Estimated time: 20 minutes

Scenario

You are an administrator for Contoso, Ltd. During a security audit, it was identified that there is no control over the creation of new computer accounts: both clients and servers are being added to the domain with no assurance that process is being followed. In fact, a number of computer accounts were discovered in the Computers container. These computer objects were for active computer accounts, but the computers had not been created in or moved to the correct OUs within the Client Computers or Servers OUs according to standard procedures. You've been tasked with improving the procedures.

Exercise 1: Join a Computer to the Domain with the Windows® Interface

In this exercise, you will join a computer to the domain using the Windows interface, and then you will remove the machine from the domain.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Identify and correct a DNS configuration error.
3. Join SERVER01 to the domain.
4. Verify the location of the SERVER01 account.
5. Remove SERVER01 from the domain.
6. Delete the SERVER01 account.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A and 6425B-SERVER01-B.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Open **D:\Labfiles\Lab05a**.
4. Run **Lab05a_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
5. The lab setup script runs. When it is complete, press any key to continue.
6. Close the Windows Explorer window, **Lab05a**.
7. Start 6425B-SERVER01-B.

► Task 2: Identify and correct a DNS configuration error

1. Log on to SERVER01 as **Administrator** with the password **Pa\$\$w0rd**.
2. Open **System Properties** using one of the following methods:
 - Click **Start**, then right-click **Computer**, and then click **Properties**.
 - Open **System** from **Control Panel**.
 - Press the WINDOWS LOGO key and the PAUSE key.

3. Attempt to join the computer to the domain **contoso.com**, being sure to use the *fully qualified domain name (contoso.com)* rather than the NetBIOS name for the domain (contoso).

Doing so tests that DNS is configured correctly on the client for locating the domain.

4. Change the DNS Server configuration on the client to **10.0.0.11**.

Question: Why might the join have succeeded if you had used the domain name **contoso** instead of **contoso.com**? What might go wrong after the domain was successfully joined with DNS but incorrectly configured?

► **Task 3: Join SERVER01 to the domain**

1. Join SERVER01 to the domain. When prompted for domain credentials, enter the username **Aaron.Painter** and the password **Pa\$\$w0rd**.

Note that Aaron.Painter is a standard user in the contoso.com domain. He has no special rights or permissions, and yet he is able to join a computer to the domain. He does have to be logged on to the computer with an account that is a member of the computer's Administrators group.

2. Allow the system to restart.

► **Task 4: Verify the location of the SERVER01 account**

1. On HQDC01, run Active Directory Users and Computers as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Locate the SERVER01 account.

Question: In which OU or container does the account exist?

► **Task 5: Remove SERVER01 from the domain**

1. Log on to SERVER01 as Administrator with the password **Pa\$\$w0rd**.
2. Change SERVER01's domain/workgroup membership to a workgroup named **WORKGROUP**.
3. Restart the server.

► **Task 6: Delete the SERVER01 account**

Question: On HQDC01, refresh the view of the Computers container and examine the SERVER01 account. What is its status?

Question: You were not prompted for domain credentials in Task 5, and yet a change was made to the domain: the computer account was reset and disabled. What credentials were used to do this? What credentials were used to change the workgroup/domain membership of SERVER01?

- Delete SERVER01's account.

Results: After this exercise, you will be familiar with typical legacy practices used to join computers to a domain.

Exercise 2: Secure Computer Joins

In this exercise, you will implement best practices to secure the joining of machines to the domain.

The main tasks for this exercise are as follows:

1. Redirect the default computer container.
2. Restrict unmanaged domain joins.
3. Validate the effectiveness of ms-DS-MachineAccountQuota.

► Task 1: Redirect the default computer container

1. Run a command prompt as an administrator with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Use the **RedirCmp** command to redirect the default computers container to the **New Computers** OU in the **contoso.com** domain.

► Task 2: Restrict unmanaged domain joins

1. Run the ADSI Edit console as an administrator with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Connect to the domain and, in the properties of the domain, change the **ms-DS-MachineAccountQuota** to zero (0).

► Task 3: Validate the effectiveness of ms-DS-MachineAccountQuota

- Log on to SERVER01 as **Administrator** and attempt to join **SERVER01** to the **contoso.com** domain just as you did in Exercise 1. When prompted for domain credentials, enter the username **Aaron.Painter** and the password **Pa\$\$w0rd**.

In the Exercise 1, Aaron Painter was able to join the domain. Now, he is unable to join the domain.

Question: What message do you receive when a user is no longer able to create a computer object because of the ms-DS-MachineAccountQuota?

Results: After this exercise, the container for creating computer accounts will be redirected to the New Computers OU, and users will be restricted from joining computers to the domain without explicit permissions to do so.

Exercise 3: Manage Computer Account Creation with Best Practices

In this exercise, you will implement several best practices for creating computer accounts and joining machines to the domain.

The main tasks for this exercise are as follows:

1. Prestage a computer account.
2. Join a computer remotely to a prestaged account using NetDom.

► Task 1: Prestage a computer account

1. Run **Active Directory Users and Computers** as an administrator with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Servers\File** OU, create a new computer object for **SERVER01** and give the **AD_Server_Deploy** group permission to join the computer to the domain.

► Task 2: Join a computer remotely to a prestaged account using NetDom

In this task, you will join **SERVER01** to the domain *remotely*, using credentials that are in the local Administrators group of **SERVER01** and domain credentials that are in the **AD_Server_Deploy** group.

1. Run the command prompt as an administrator, with the username **Aaron.Painter_Admin** and the password **Pa\$\$word**.

Note that **Aaron.Painter_Admin** is not an administrator, *per se*. The Run as an administrator command allows you to launch a process with any credentials, as long as those credentials have sufficient privilege to launch the process itself.
2. Type the command **whoami /groups** to list the group memberships of the current account (**Aaron.Painter_Admin**). Note that the user is a member of **AD_Server_Deploy** and is not a member of any other administrative group.
3. Using the **NetDom** command, join **SERVER01** to the domain. Use the local Administrator account credentials for **SERVER01** and the domain credentials for **Aaron.Painter_Admin**, who is a member of **AD_Server_Deploy** and therefore has permission to join the computer to the domain. Configure the server to reboot automatically in 5 seconds.

Type the following command, and then press ENTER:

```
netdom join SERVER01 /domain:contoso.com  
/UserO:Administrator /Password0:*  
/UserD:CONTOSO\Aaron.Painter_Admin /PasswordD:*  
/REBoot:5
```



Note: SERVER01 has firewall exceptions configured ports 135, 139, and for Network Discovery (NB-Name-In). These exceptions allow NetDom Join to be used to remotely join SERVER01 to the domain.

4. The server restarts.

Results: After this exercise, SERVER01 will be joined to the domain with an account in the Servers\File OU.



Important: Do not shut down the virtual machines once you are finished with this lab as the settings you have configured here will be used in the Lab B.

Lab Review Questions

Question: What did you learn about the pros and cons of various approaches to creating computer accounts in an AD DS domain?

Question: What are the two credentials that are necessary for any computer to join a domain?

Lesson 2

Administer Computer Objects and Accounts

- Configure Computer Attributes
- Move a Computer
- Computer Accounts and Secure Channel
- Recognize Computer Account Problems
- Reset a Computer Account
- Rename a Computer
- Disable and Enable a Computer
- Delete and Recycle Computer Accounts

A computer account begins its life cycle when it is created and when the computer joins the domain. Day-to-day administrative tasks include configuring computer properties; moving the computer between OUs; managing the computer itself; and renaming, resetting, disabling and enabling, and eventually deleting the computer object. This lesson looks closely at the computer properties and procedures involved with these tasks, and will equip you to administer computers in a domain.

Objectives

After completing this lesson, you will be able to:

- Configure computer account properties.
- Move a computer between OUs.
- Recognize computer account problems.

- Reset a computer account.
- Rename a computer.
- Disable and enable a computer.

Configure Computer Attributes

- Useful attributes
 - Description
 - Location
 - US\WA\SEA\HQ\Building33\Floor3\Q04\1531
 - Used by location-aware applications such as Search For Printers
 - Managed By
 - Link to user who is the primary user of the computer
 - Link to group that is responsible for the computer (servers)
 - Member Of
 - Groups: Group Policy filtering, software deployment
 - `dsmod computer "ComputerDN" [-desc "Description"] [-loc "Location"]`

Key Points

When you create a computer object using Active Directory Users and Computers, you are prompted to configure only the most fundamental attributes, including the computer name and the delegation to join the computer to the domain. Computers have several properties that are not visible when you are creating the computer object; you should configure these properties as part of the process of staging the computer account.

Open a computer object's Properties dialog box to set its location and description, configure its group memberships and dial-in permissions, and link it to the user object of the user to whom the computer is assigned. The Operating System tab is read-only. The information will be blank until a computer has joined the domain using that account, at which time the client publishes the information to its account.

Several object classes in Active Directory support the `managedBy` attribute that is shown on the Managed By tab. This linked attribute creates a cross-reference to a user object. All other properties—the addresses and telephone numbers—are displayed directly from the user object. They are not stored as part of the computer object itself. Some organizations use the Managed By tab to link the computer to the primary user of the computer. Alternatively, you might choose to link the computer to a group that is responsible for the support of a computer—an option that might be attractive for computer accounts that represent servers, for example.

On the Member Of tab of a computer's Properties dialog box, you can add the computer to groups. The ability to manage computers in groups is an important and often underutilized feature of Active Directory. A group to which computers belong can be used to assign resource access permissions to the computer, to filter the application of a Group Policy object (GPO), or as a collection for a software management tool, such as Microsoft System Center Configuration Manager 2007.

As with users and groups, it is possible to select more than one computer object and subsequently manage or modify properties of all selected computers simultaneously.

Configuring Computer Attributes with DSMod

The DSMod command is able to modify the description and the location attributes of a computer object. It uses the following syntax:

```
dsmod computer "ComputerDN" [-desc "Description"] [-loc "Location"]
```

Move a Computer

- Using Active Directory Users and Computers
 - Drag and drop
 - Right-click the computer, and then click Move
- `dsmove ObjectDN [-newname NewName] [-newparent ParentDN]`
 - `-newname NewName`: Used to rename a computer
 - `-newparent ParentDN`: Used to move a computer to the OU specified by ParentDN

Key Points

Many organizations have multiple OUs for computer objects. Some domains, for example, have computer OUs based on geographic sites, as shown earlier in this module. If you have more than one OU for computers, it is likely that someday you will need to move a computer between OUs.

To move a computer using the Active Directory Users and Computers snap-in:

- Drag and drop *or*
- Right-click the computer, and then click **Move**.

The DSMove command allows you to move a computer object or any other object. The syntax of DSMove is:

```
dsmove ObjectDN [-newname NewName] [-newparent ParentDN]
```

The -newname option allows you to rename an object. The -newparent option allows you to move an object. To move a computer named DESKTOP153 from the Computers container to the NYC OU, you would type the following:

```
dsmove "CN=DESKTOP153,CN=Computers,DC=contoso,DC=com" -newparent  
"OU=NYC,OU=Client Computers,DC=contoso,DC=com"
```

Computer Account and Secure Channel

- **Computers have accounts**
 - sAMAccountName and password
 - Used to create a secure channel between the computer and a domain controller
- **Secure channel can be broken**
 - Reinstalling computer, even with same name, generates new SID and password
 - Restoring a computer from an old backup, or rolling back a computer to an old snapshot
 - Computer and domain disagree about what the password is

Key Points

Every member computer in an Active Directory domain maintains a computer account with a username (sAMAccountName) and password, just like a user account does. The computer stores its password in the form of a local security authority (LSA) secret and changes its password with the domain every 30 days or so. The NetLogon service uses the credentials to log on to the domain, which establishes the secure channel with a domain controller.

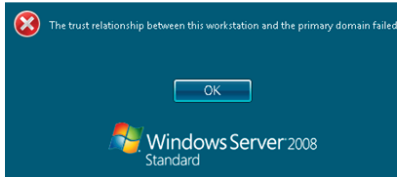
Computer accounts and the secure relationships between computers and their domain are robust. However, certain scenarios might arise in which a computer is no longer able to authenticate with the domain. Examples of such scenarios include the following:

- After reinstalling the operating system on a workstation, the workstation is unable to authenticate, even though the technician used the same computer name. Because the new installation generated a new SID and because the new computer does not know the computer account password in the domain, it does not belong to the domain and cannot authenticate to the domain.

- A computer is completely restored from backup and is unable to authenticate. It is likely that the computer changed its password with the domain after the backup operation. Computers change their passwords every 30 days, and Active Directory remembers the current and previous password. If the restore operation restored the computer with a significantly outdated password, the computer will not be able to authenticate.
- A computer's LSA secret gets out of synch with the password known by the domain. You can think of this as the computer forgetting its password; although it did not forget its password, it just disagrees with the domain over what the password really is. When this happens, the computer cannot authenticate and the secure channel cannot be created.

Recognize Computer Account Problems

- Logon messages

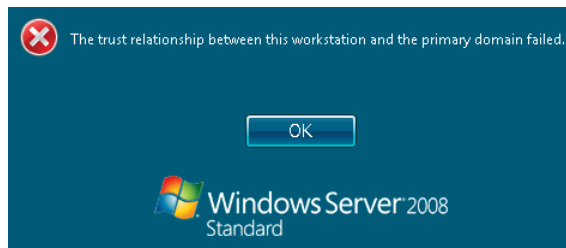


- Event log errors, including key words such as
 - Password
 - Trust
 - Secure channel
 - Relationships with the domain or domain controllers
- A computer account is missing in Active Directory

Key Points

The most common signs of computer account problems are the following:

- Messages at logon indicate that a domain controller cannot be contacted, that the computer account might be missing, that the password on the computer account is incorrect, or that the trust relationship (another way of saying “the secure relationship”) between the computer and the domain has been lost. An example is shown here.



- Error messages or events in the event log indicate similar problems or suggest that passwords, trusts, secure channels, or relationships with the domain or a domain controller have failed. One such error is NETLOGON Event ID 3210: Failed To Authenticate, which appears in the computer's event log.
- A computer account is missing in Active Directory.

Reset a Computer Account

- Do not simply remove computer from domain and rejoin
 - Creates new account: new SID, lost group memberships
- Reset the secure channel
 - Active Directory Users and Computers*
 - Right-click the computer, and then click Reset Account
 - DSMod*
 - `dsmmod computer "ComputerDN" -reset`
 - NetDom
 - `netdom reset MachineName /domain:DomainName /User:User0 /Password:Password0 {Password | *}`
 - NLTest
 - `nlttest /server:ServerName /sc_reset:DOMAIN\DomainController`
 - * = requires rejoining domain and rebooting

Key Points

When the secure channel fails, you must reset the secure channel. Many administrators do so by removing the computer from the domain, putting it in a workgroup, and then rejoining the domain. This is not a good practice, because it has the potential to delete the computer account altogether, which loses the computer's SID and, more importantly, its group memberships. When you rejoin the domain, even though the computer has the same name, the account has a new SID, and all the group memberships of the previous computer object must be re-created.

Do not remove a computer from the domain and rejoin it

If the trust with the domain has been lost, do not remove a computer from the domain and rejoin it. Instead, reset the secure channel.

To reset the secure channel between a domain member and the domain, use the Active Directory Users and Computers snap-in, DSMod.exe, NetDom.exe, or NLTest.exe. If you reset the account, the computer's SID remains the same and it maintains its group memberships.

To reset the secure channel using the Active Directory Users and Computers snap-in:

1. Right-click a computer, and then click **Reset Account**.
2. Click **Yes** to confirm your choice.
3. Rejoin the computer to the domain, and then reboot the computer.

To reset the secure channel using DSMod:

1. Type the following command:

```
dsmod computer "ComputerDN" -reset.
```

2. Rejoin the computer to the domain, and then reboot the computer.

To reset the secure channel using NetDom:

- Type the following command:

```
netdom reset MachineName /domain DomainName /User0 UserName  
/Password0 {Password | *}
```

where the credentials belong to the local Administrators group of the computer.

This command resets the secure channel by attempting to reset the password on both the computer and the domain, so it does not require rejoining or rebooting.

To reset the secure channel using NLTest, on the computer that has lost its trust, type the command:

```
NLTEST /SERVER:SERVERNAME /SC_RESET:DOMAIN\DOMAINCONTROLLER
```

For example:

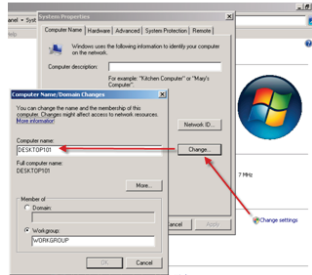
```
nltest /server:SERVER02 /sc_reset:CONTOSO\SERVER01
```

This command, like NetDom, attempts to reset the secure channel by resetting the password on both the computer and in the domain, so it does not require rejoining or rebooting.

Because NLTest and NetDom reset the secure channel without requiring a reboot, you should try those commands first. Only if those are not successful should you use the Reset Account command or DSMod to reset the computer account.

Rename a Computer

- Use System Properties of computer itself to rename computer *and* its account correctly



- NetDom
 - `netdom renamecomputer MachineName /NewName:NewName [/User0:LocalUsername] [/Password0:{LocalPassword}*] [/UserD:DomainUsername] [/PasswordD:{DomainPassword}*] [/SecurePasswordPrompt] [/REBoot[:TimeInSeconds]]`
- Be cautious of impact that rename can have on services and on certificates associated with computer's name

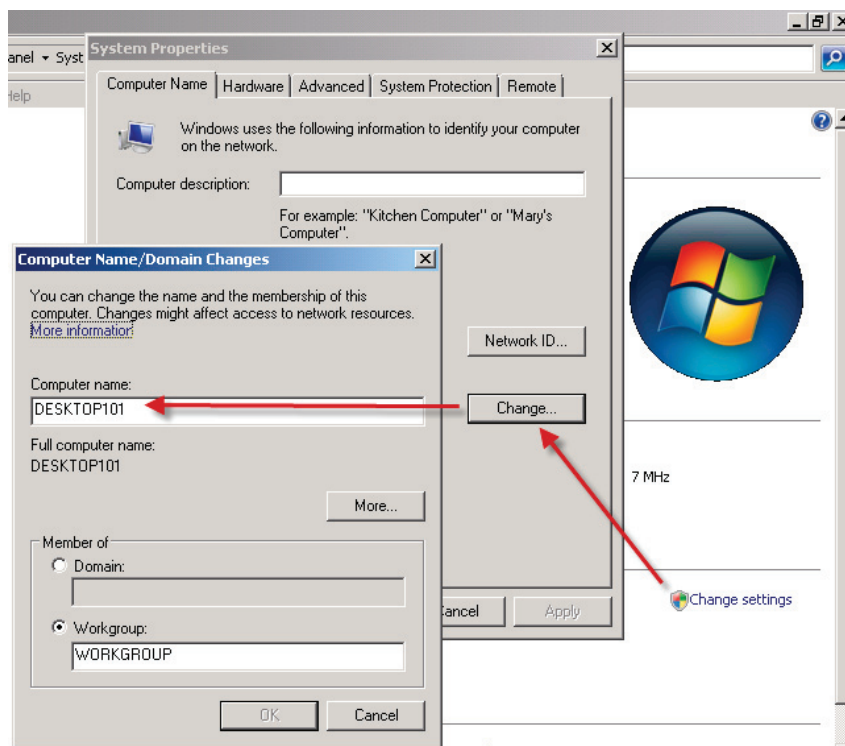
Key Points

When you rename a computer, you must be careful to do it correctly. Remember that the computer uses its name to authenticate with the domain, so if you rename only the domain object, or only the computer itself, they will be out of synch. You must rename the computer in such a way that both the computer and the domain object are changed.

You can rename a computer correctly by logging on to the computer itself, either locally or with a remote desktop session.

1. Open **System Properties** from Control Panel.
2. In the **Computer name, domain, and workgroup settings** section, click **Change Settings**.
3. If you are prompted by **User Account Control**, click **Continue**.

4. Click the **Computer Name** tab.
5. Click the **Change** button.



6. Type the new name and click **OK** twice to close the dialog boxes.
7. Restart the computer to allow the change to take effect.

From the command prompt, you can use the NetDom command, with the following syntax:

```
netdom renamecomputer MachineName /NewName:NewName
[/User0:LocalUsername] [/Password0:{LocalPassword}*} ]
[/UserD:DomainUsername] [/PasswordD:{DomainPassword}*} ]
[/SecurePasswordPrompt] [/REBoot[:TimeInSeconds]]
```

In addition to specifying the machine to rename (MachineName) and the desired new name (NewName), you must have credentials that are a member of the local Administrators group on the computer and credentials that have permission to rename the domain computer object. By default, NetDom will use the credentials with which the command is executed. You can specify credentials using /UserO and /PasswordO for the credentials in the computer's local Administrators group, and /UserD and /PasswordD for the domain credentials with permission to rename the computer object. Specifying * for the password causes NetDom.exe to prompt for the password at the command prompt. The /SecurePasswordPrompt option displays a popup for credentials when * is specified for either /PasswordO or /PasswordD. After you rename a computer, you must reboot the computer. The /REBoot option causes the system to reboot after 30 seconds, unless otherwise specified by TimeInSeconds.

When you rename a computer, you can adversely affect services running on the computer. For example, Active Directory Certificate Services (AD CS) relies on the server's name. Be certain to consider the impact of renaming a computer before doing so. Do not use these methods to rename a domain controller.

Disable and Enable a Computer

- Disable computer if it will be offline for extended time
 - Similar to disabling a user who is on a leave of absence
 - Prevents secure channel from being established, so users who do not have cached credentials on the computer cannot log on
- Active Directory Users and Computers
 - Right-click computer, and then click Enable Account or Disable Account
- DSMod
 - `dsmod computer ComputerDN -disabled yes`
`dsmod computer ComputerDN -disabled no`



Key Points

If a computer is taken offline or is not to be used for an extended period of time, you should consider disabling the account. This recommendation reflects the security principle that an identity store should allow authentication only of the minimum number of accounts required to achieve the goals of an organization. Disabling the account does not modify the computer's SID or group membership, so when the computer is brought back online, the account can be enabled.

To disable a computer in the Active Directory Users and Computers snap-in, right-click the computer, and then click Disable Account.

A disabled account appears with a down-arrow icon in the Active Directory Users And Computers snap-in, as shown here:



While an account is disabled, the computer cannot create a secure channel with the domain. The result is that users who have not previously logged on to the computer, and who therefore do not have cached credentials on the computer, will be unable to log on until the secure channel is reestablished by enabling the account.

To enable a computer account, right-click the computer, and then click Enable Account.

To disable or enable a computer from the command prompt, use the DSMod command. The syntax used to disable or enable computers is:

```
dsmod computer ComputerDN -disabled yes  
dsmod computer ComputerDN -disabled no
```

Delete and Recycle Computer Accounts

- Delete a computer with Active Directory Users and Computers
 - Right-click the computer, and then click Delete
- Delete a computer with DSRm
 - `dsrcm ObjectDN`
- Delete destroys SID and group memberships
 - If replacing or reinstalling a computer, if computer will play same role, *reset computer account* instead of deleting it
 - Preserves all attributes of computer, including SID and group memberships
 - You can rename object if computer is being renamed during reinstallation/upgrade
 - This "recycles" the computer account

Key Points

You have learned that each computer account, like each user account, maintains a unique SID, which enables an administrator to grant permissions to computers. Also like user accounts, computers can belong to groups. Therefore, like user accounts, it is important to understand the effect of deleting a computer account. When a computer account is deleted, its group memberships and SID are lost. If the deletion is accidental, and another computer account is created with the same name, it is nonetheless a new account, with a new SID. Group memberships must be reestablished, and any permissions assigned to the deleted computer must be reassigned to the new account. Delete computer objects only when you are certain that you no longer require those security-related attributes of the object.

To delete a computer account using Active Directory Users And Computers:

1. Right-click the computer object, and then click **Delete**.

You are prompted to confirm the deletion and, because deletion is not reversible, the default response to the prompt is No.

2. Click **Yes** to delete the object.

The DSRm command allows you to delete a computer object from the command prompt. To delete a computer with DSRm, type:

```
dsrm ObjectDN
```

Where ObjectDN is the distinguished name of the computer, such as “CN=Desktop154, OU=NYC,OU=Client Computers,DC=contoso,DC=com.” Again, you will be prompted to confirm the deletion.

Recycling Computers

If a computer account’s group memberships and SID, and the permissions assigned to that SID, are important to the operations of a domain, you do not want to delete that account. So what would you do if a computer was replaced with a new system, with upgraded hardware? That is another scenario in which you would reset a computer account.

Resetting a computer account resets its password but maintains all of the computer object’s properties. With a reset password, the account becomes, in effect, available for use. Any computer can then join the domain using that account, including the upgraded system. In effect, you’ve recycled the computer account, assigning it to a new piece of hardware. You can even rename the account. The SID and group memberships remain the same.

As you learned earlier in this lesson, the Reset Account command is available in the context menu when you right-click a computer object. The DSMod command can also be used to reset a computer account, when you type dsmod computer "ComputerDN" -reset.

Lab B: Administer Computer Objects and Accounts

- Exercise 1: Administer Computer Objects Through Their Life cycle
- Exercise 2: Administer and Troubleshooting Computer Accounts

Logon information

Virtual machine	6425B-HQDC01-A	6425B-SERVER01-B
Logon user name	Pat.Coleman	Pat.Coleman
Administrative user name	Pat.Coleman_Admin	Administrator
Password	Pa\$\$w0rd	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

You are an administrator for Contoso, Ltd. During a security audit, a number of computer accounts were discovered. Those computers no longer exist in the domain. You've been tasked with improving the management of computer accounts, and identifying best practices for administering the entire life cycle of a computer account.

Exercise 1: Administer Computer Objects Through Their Life Cycle

In this exercise, you will configure common attributes of computer objects, including description and ManagedBy. You will also manage the group membership of computers and move computers between OUs.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Configure computer object attributes.
3. Add computers to software management groups.
4. Move a computer between OUs.
5. Disable, enable and delete computers.

► Task 1: Prepare for the lab

The virtual Machines should already be started and available after completing Lab A. However, if they are not, you should complete steps 1 to 3 below and then step through exercises 1 to 3 in Lab A before continuing. You will be unable to successfully complete Lab B unless you have completed Lab A.

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Start 6425B-SERVER01-B.

► Task 2: Configure computer object attributes

1. On HQDC01, run **Active Directory Users and Computers** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Client Computers\SEA OU**, use the **Managed By** tab of computer objects to assign **LNO8538** to **Linda Mitchell** and **LOT9179** to **Scott Mitchell**.
3. Because Scott and Linda Mitchell will occasionally use each other's computer, use multiselect to change the description of both **LNO8538** and **LOT9179** to **Scott and Linda Mitchell**.

► Task 3: Add computers to software management groups

Microsoft Office Project is required on both Scott's and Linda's computers. Contoso uses security groups as collections for scoping the deployment of software. You will add each of their computers to the group APP_Project using two different methods.

1. In the **Client Computers\SEA OU**, right-click **LOT9179**, and then click **Add to a group**.

2. Type **APP_** and press ENTER.

The Multiple Items Found dialog box appears.

3. Click **APP_Project** and click **OK**.

A message appears: "The Add to Group operation was successfully completed."

4. Click **OK**.

5. In the console tree, expand the **Groups OU**, and then click **Application**.

6. Right-click **APP_Project**, and then click **Properties**.

7. Click the **Members** tab.

8. Click **Add**.

9. Type **LNO8538** and press ENTER.

The Name Not Found dialog box appears.

By default, the Select Users, Computers, or Groups interface does not search for computer objects.

10. Click **Object Types**.

11. Select the check box next to **Computers**, and then click **OK**.

12. Click **OK** to close the **Name Not Found** dialog box.

Both computers can now be seen on the Members tab.

13. Click **OK**.

► **Task 4: Move a computer between OUs**

Scott and Linda are relocating to the Vancouver office. You will move their computers to the new OU using two different methods.

1. In the **Client Computers\SEA** OU, click **LOT9179**.
2. Drag **LOT9179** into the **VAN** OU, visible in the console tree.
A message appears that reminds you to be careful about moving objects in Active Directory.
3. Click **Yes**.
4. Right-click **LNO8538**, and then click **Move**.
The Move dialog box appears.
5. In the console tree, expand **Client Computers**, and then click **VAN**.
6. Click **OK**.

► **Task 5: Disable, enable, and delete computers**

1. In the **Client Computers\SEA** OU, disable, then enable the account for **DEP6152**.
2. Delete the account for **DEP6152**.

Exercise 2: Administer and Troubleshooting Computer Accounts

In this exercise, you will administer and troubleshoot computer accounts and the secure channel.

The main tasks for this exercise are as follows:

1. Reset a computer account.
2. Experience a secure channel problem.
3. Reset the secure channel.

► Task 1: Reset a computer account

Recently, Scott Mitchell's computer required reinstallation. Contoso's naming convention is that the name of a computer object is its asset tag, assigned by the IT inventory team. Because Scott reinstalled his computer on the same piece of hardware, the computer name is the same: LOT9179. He now wants to join the machine to the domain, but there is already an account for LOT9179, and the account is a member of groups that ensure the correct software (including Microsoft Office Project) and configuration are applied to the system. Therefore, it is important that the account not be deleted, so that group memberships can be retained.

- In the **Client Computers\VAN OU**, reset the account for **LOT9179**.

You could now join Scott's reinstalled computer to the domain.

► Task 2: Experience a secure channel problem

1. Demonstrate that you can log on successfully to SERVER01 as **Pat.Coleman** with the password **Pa\$\$w0rd**. After the desktop appears, log off.
2. To "break" the secure channel, use Active Directory Users and Computers on HQDC01 to reset the account for SERVER01.
3. Attempt to log on to SERVER01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.

► Task 3: Reset the secure channel

To solve a broken trust relationship between a domain member and the domain, you can reset the computer's account, then move the computer into a workgroup, and then rejoin the domain.

- Reset the computer account for SERVER01.

After resetting the secure channel, you could move SERVER01 into a workgroup, and then rejoin the domain. It will join its reset account, thereby retaining its group memberships. Do not perform that step at this time.

Results: After this exercise, you will have a user account named Chris Mayo in the Employees OU.



Note: After completing this exercise, turn off all virtual machines and discard undo disks.

Lab Review Question

Question: What insights did you gain into the issues and procedures regarding computer accounts and administering computer accounts through their life cycle?

MCT USE ONLY. STUDENT USE PROHIBITED

Module 6

Implement a Group Policy Infrastructure

Contents:

Lesson 1: Understand Group Policy	6-4
Lesson 2: Implement GPOs	6-21
Lab A: Implement Group Policy	6-38
Lesson 3: A Deeper Look at Settings and GPOs	6-42
Lab B: Manage Settings and GPOs	6-64
Lesson 4: Manage Group Policy Scope	6-71
Lab C: Manage Group Policy Scope	6-102
Lesson 5: Group Policy Processing	6-110
Lesson 6: Troubleshoot Policy Application	6-120
Lab D: Troubleshoot Policy Application	6-132

Module Overview

- Understand Group Policy
- Implement GPOs
- A Deeper Look at Settings and GPOs
- Manage Group Policy Scope
- Group Policy Processing
- Troubleshoot Policy Application

In Module 1, you learned that Active Directory® Domain Services (AD DS) provides the foundational services of an identity and access solution for enterprise networks running Windows®, and that AD DS goes further to support the management and configuration of even the largest, most complex networks. In Modules 2 through 5, you learned how to administer Active Directory directory service security principals: users, groups, and computers. Now you will begin an examination of the management and configuration of users and computers using Group Policy. Group Policy provides an infrastructure within which settings can be defined centrally and deployed to users and computers in the enterprise.

In an environment managed by a well-implemented Group Policy infrastructure, little or no configuration needs to be made by directly touching a desktop. All configuration is defined, enforced, and updated using settings in GPOs that affect a portion of the enterprise as broad as an entire site or domain or as narrow as a single organizational unit (OU) or group. In this module, you will learn what Group Policy is, how it works, and how best to implement Group Policy in your organization. Several subsequent modules will apply Group Policy to specific management tasks such as security configuration, software deployment, password policy, and auditing.

Objectives

After completing this module, you will be able to:

- Identify the business drivers for configuration management.
- Understand the components and technologies that comprise the Group Policy framework.
- Manage GPOs.
- Configure and understand a variety of policy setting types.
- Scope GPOs using links, security groups, WMI filters, loopback processing, and Preference targeting.
- Explain GPO storage, replication, and versioning.
- Administer a Group Policy infrastructure.
- Evaluate GPO inheritance, precedence, and RSoP.
- Locate the event logs containing Group Policy-related events.

Lesson 1

Understand Group Policy

- What is Configuration Management?
- Policy Settings (Also Known as *Policies*)
- Group Policy Objects
- GPO Scope
- Group Policy Client and Client-Side Extensions
- Group Policy Refresh
- Resultant Set of Policy
- Review and Discuss the Components of Group Policy

A Group Policy infrastructure has a lot of moving parts. It is important that you understand not only what each part does but also how the parts work together and why you might want to assemble them in various configurations. In this lesson, you will get a comprehensive overview of Group Policy: its components, its functions, and its inner workings.

Objectives

After completing this lesson, you will be able to:

- Identify the business drivers for configuration management.
- Understand the core components and terminology of Group Policy.
- Explain the fundamentals of Group Policy processing.

What Is Configuration Management?

- A centralized approach to applying one or more changes to one or more users or computers
- *Setting*: Definition of a change or configuration
- *Scope*: Definition of the user(s) or computer(s) to which the change applies
- *Application*: A mechanism that applies the setting to users and computers within the scope
- Group Policy: The framework for configuration management in an AD DS domain
 - Setting
 - Scope
 - Application
 - Tools for management, configuration, and troubleshooting

Key Points

If you have only one computer in your environment—at home, for example—and you need to make a change—modify the desktop background, for example—there are several ways to do that. Most people would probably open Personalization from Control Panel and make the change using the Windows interface. That works well for one user, but becomes tedious if you want to make the change across multiple users. Say, for example, that you want the same background for yourself and your family. You have to make the change multiple times, and then if you ever change your mind and want to change the background yet again, you have to return to each user's profile and make the change. Implementing the change, and maintaining a consistent environment, becomes even more difficult across multiple computers.

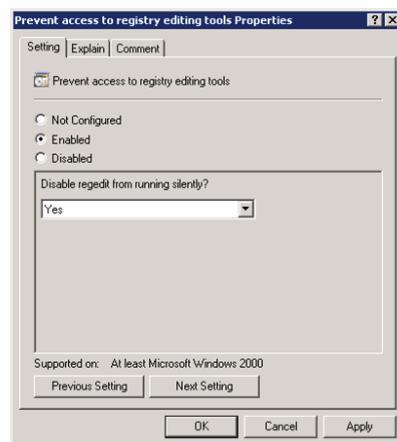
In the end, *configuration management* is a *centralized approach to applying one or more changes to one or more users or computers*. If you keep that in mind, everything else will be easier to understand. So let's repeat. The key elements of configuration management are:

- A centralized definition of a change, which we will also call a *setting*. The setting brings a user or a computer to a desired state of configuration.
- A definition of the user(s) or computer(s) to whom the change applies, which we will call the *scope* of the change.
- A mechanism that ensures that the setting is applied to users and computers within the scope. We will call this process the *application*.

Group Policy is a framework within Windows—with components that reside in Active Directory, on domain controllers, and on each Windows server and client—that enables you to manage configuration in an AD DS domain. As we turn our attention to Group Policy, which can become very complex, always remember that everything boils down, in the end, to just these few basic elements of configuration management.

Policy Settings (Also Known as Policies)

- The granular definition of a change or configuration
 - Prevent access to registry-editing tools
 - Rename the Administrator account
- Divided between
 - User Configuration ("user policies")
 - Computer Configuration ("computer policies")
- Define a setting
 - Not configured (default)
 - Enabled
 - Disabled
- Read explanatory text
- Test all settings



Key Points

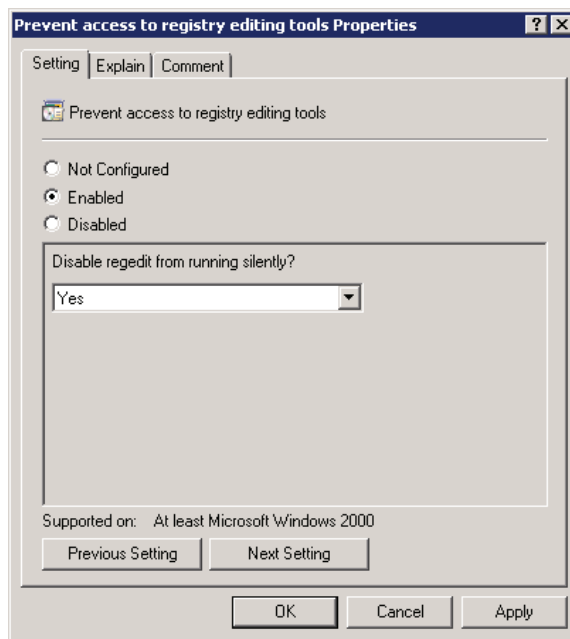
The most granular component of the Group Policy is an individual *policy setting*, also known simply as a *policy*, that defines a specific configuration change to apply. For example, a policy setting exists that prevents a user from accessing registry-editing tools. If you define that policy setting and apply it to the user, the user will be unable to run tools such as Regedit.exe. Another policy setting is available that enables you to rename the local Administrator account. You can use this policy setting to rename the Administrator account on all user desktops and laptops, for example.

These two examples illustrate an important point: that some policy settings affect a user, regardless of the computer to which the user logs on, and other policy settings affect a computer, regardless of which user logs on to that computer. Policy settings such as the setting that prevents access to registry-editing tools are often referred to as *user configuration settings* or *user settings*. Policy settings such as the one that disables the Administrator account and similar settings are often referred to as *computer configuration settings* or *computer settings*. You will also hear these referred to as "user policies" and "computer policies." The terminology used in the industry is not exact.

There are thousands of policy settings that can be managed by Group Policy, and the framework is extensible so, in the end, you could manage just about anything with Group Policy.

To define a policy setting, double-click the policy setting.

The policy setting's Properties dialog box appears. An example is shown here:



A policy setting can have three states: Not Configured, Enabled, and Disabled.

In a new GPO, every policy setting is Not Configured. This means that the GPO will not modify the existing configuration of that particular setting for a user or computer. If you enable or disable a policy setting, a change will be made to the configuration of users and computers to which the GPO is applied.

The effect of the change depends on the policy setting itself. For example, if you enable the Prevent Access To Registry Editing Tools policy setting, users will be unable to launch the Regedit.exe Registry Editor. If you disable the policy setting, you ensure that users can launch the Registry Editor. Notice the double negative in this policy setting: You disable a policy that prevents an action, so you allow the action.

Some policy settings bundle several configurations into one policy and might require additional parameters. In the screenshot above, you can see that by enabling the policy to restrict registry editing tools, you can also define whether registry files can be merged into the system silently, using regedit /s.



Note: Understand and test all policy settings. Many policy settings are complex, and the effect of enabling or disabling them might not be immediately clear. Also, some policy settings affect only certain versions of Windows. Be sure to review a policy setting's explanatory text in the Group Policy Management Editor (GPME) detail pane or on the Explain tab of the policy setting's Properties dialog box. Additionally, always test the effects of a policy setting, and its interactions with other policy settings, before deploying a change in the production environment.

You will explore policy settings and how to manage them in Lesson 3.

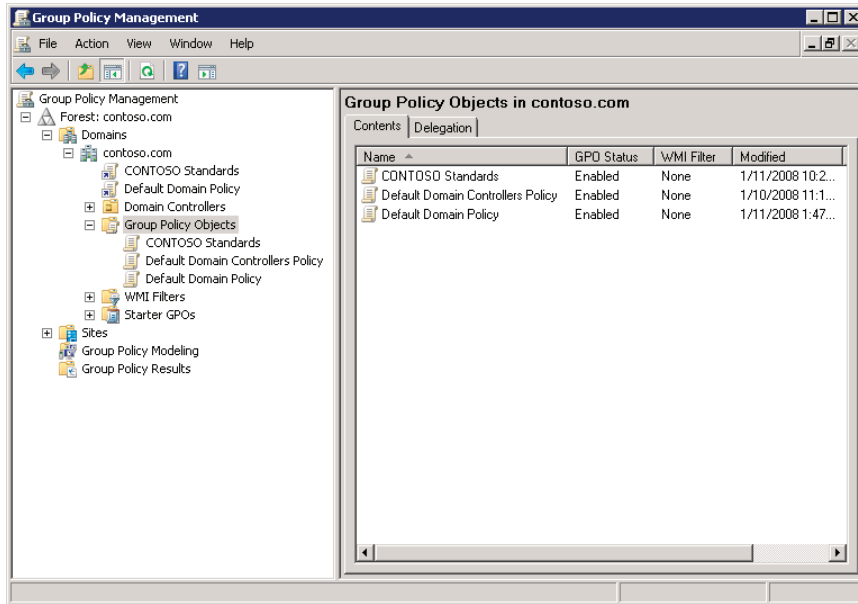
Group Policy Objects

- The container for one or more policy settings
- Managed with the Group Policy Management console (GPMC)
 - Group Policy Objects container
- Edited with the Group Policy Management Editor (GPME)

Key Points

Policy settings are defined and exist within a *Group Policy object (GPO)*. A GPO is an object that contains one or more policy settings and thereby applies one or more configuration settings for a user or computer.

GPOs can be managed in Active Directory by using the Group Policy Management console (GPMC), shown here:

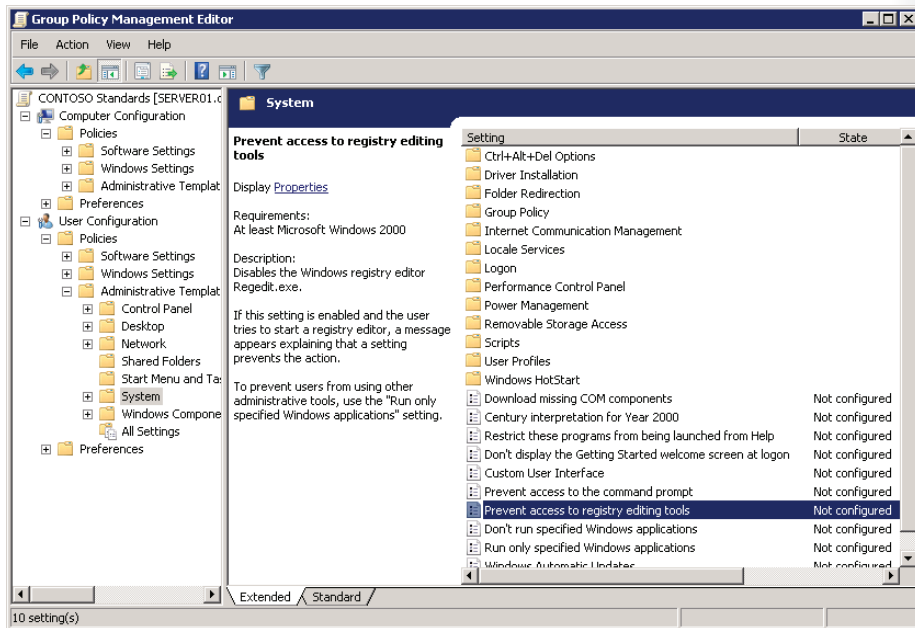


GPOs are displayed in a container named Group Policy Objects.

To create a new GPO in a domain, right-click the Group Policy Objects container, and then click New.

To modify the configuration settings in a GPO, right-click the GPO and choose Edit.

The GPO opens in the Group Policy Management Editor (GPME) snap-in, formerly known as the Group Policy Object Editor (GPO Editor), shown here:



The GPME displays the thousands of policy settings available in a GPO in an organized hierarchy that begins with the division between computer settings and user settings: the Computer Configuration node and the User Configuration node. The next levels of the hierarchy are two nodes called Policies and Preferences. You will learn about the difference between these two nodes as this lesson progresses. Drilling deeper into the hierarchy, you will see that the GPME displays folders, also called nodes or policy setting groups. Within the folders are the policy settings themselves. Prevent Access To Registry Editing Tools is selected in the screenshot shown here.

You will learn how to implement and manage GPOs in Lesson 2.

GPO Scope

- **Scope.** Definition of objects (users or computers) to which GPO applies
- **GPO link.** GPO can be linked to site, domain, or organizational unit (OU) (SDOU)
 - GPO can be linked to multiple site(s) or OU(s)
 - GPO link(s) define *maximum* scope of GPO
- **Security group filtering**
 - Apply or deny application of GPO to members of global security group
 - Filter application of scope of GPO within its link scope
- **WMI filtering**
 - Refine scope of GPO within link based on WMI query
- **Preference targeting**

Key Points

Configuration is defined by policy settings in GPOs. However, the configuration changes in a GPO do not affect computers or users in your enterprise until you have specified the computers or users to which the GPO applies. This is called *scoping* a GPO. The *scope* of a GPO is the collection of users and computers that will apply the settings in the GPO.

You can use several methods to manage the scope of GPOs. The first is the *GPO link*. GPOs can be linked to sites, domains, and OUs in Active Directory. The site, domain, or OU then becomes the maximum scope of the GPO. All computers and users within the site, domain, or OU, including those in child OUs, will be affected by the configurations specified by policy settings in the GPO. A single GPO can be linked to more than one site or OU.

You can further narrow the scope of the GPO with one of two types of filters: *security filters* that specify global security groups to which the GPO should or should not apply, and *Windows Management Instrumentation (WMI) filters* that specify a scope by using characteristics of a system such as operating system version or free disk space. Use security filters and WMI filters to narrow or specify the scope within the initial scope created by the GPO link.

Windows Server® 2008 introduced a new component of Group Policy: Group Policy Preferences. Settings that are configured by Group Policy Preferences within a GPO can be filtered, or *targeted*, based on a number of criteria. *Targeted preferences* allow you to further refine the scope of Preferences within a single GPO.

Scoping GPOs is detailed in Lesson 4.

Group Policy Client and Client-Side Extensions

- How GPOs and their settings are *applied*
- *Group Policy Client* retrieves *ordered list of GPOs*
- GPOs are downloaded (then cached)
- Components called *client-side extensions (CSEs)* process the settings to apply the changes
 - One for each major category of policy settings: security, registry, script, software installation, mapped drive preferences, etc.
 - Most CSEs apply settings only if GPO (as a whole) has changed
 - Improves performance
 - Security CSE applies changes every 16 hours
 - GPO application is client driven ("pull")

Key Points

How, exactly, are the policy settings applied? When Group Policy refresh begins, a service running on all Windows systems (called the *Group Policy Client* in Windows Vista® and Windows Server 2008) determines which GPOs apply to the computer or user. It downloads any GPOs that it does not already have cached. Then a series of processes called *client-side extensions (CSEs)* do the work of interpreting the settings in a GPO and making appropriate changes to the local computer or to the currently logged-on user. There are CSEs for each major category of policy setting. For example, there is a security CSE that applies security changes, a CSE that executes startup and logon scripts, a CSE that installs software, and a CSE that makes changes to registry keys and values. Each version of Windows has added CSEs to extend the functional reach of Group Policy. There are several dozen CSEs now in Windows.

One of the more important concepts to remember about Group Policy is that it is really client driven. The Group Policy client pulls the GPOs from the domain, triggering the CSEs to apply settings locally. Group Policy is not a “push” technology.

The behavior of CSEs can be configured using Group Policy, in fact. Most CSEs will apply settings in a GPO only if that GPO has changed. This behavior improves overall policy processing by eliminating redundant applications of the same settings. Most policies are applied in such a way that standard users cannot change the setting on their system—they will always be subject to the configuration enforced by Group Policy. However, some settings can be changed by standard users, and many can be changed if a user is an administrator on that system. If users in your environment are administrators on their computers, consider configuring CSEs to reapply policy settings even if the GPO has not changed. That way, if an administrative user changes a configuration so that it is no longer compliant with policy, the configuration will be reset to its compliant state at the next Group Policy refresh.



Note: Configure CSEs to reapply policy settings even if the GPO has not changed.

You can configure CSEs to reapply policy settings, even if the GPO has not changed, at background refresh. To do so, configure a GPO scoped to computers and define the settings in the Computer Configuration\Policies\Administrative Templates\System\Group Policy node. For each CSE you want to configure, open its policy processing policy setting—for example, Registry Policy Processing for the Registry CSE. Click Enabled, and select the check box labeled Process Even If The Group Policy Objects Have Not Changed.

An important exception to the default policy processing settings is settings managed by the security CSE. Security settings are reapplied every 16 hours even if a GPO has not changed.



Note: The Always Wait For Network At Startup And Logon policy setting. It is highly recommended that you enable the Always Wait For Network At Startup And Logon policy setting for all Windows clients. Without this setting, by default, Windows XP, Windows Vista, and Windows 7 clients perform only background refreshes, meaning that a client might start up and a user might log on without receiving the latest policies from the domain. The setting is located in Computer Configuration\Policies\Administrative Templates\System\Logon. Be sure to read the policy setting's explanatory text. The contoso.com domain used in this course has been pre-configured with this additional Group Policy setting.

Group Policy application is discussed in detail in Lesson 5.

Group Policy Refresh

- When GPOs and their settings are *applied*
- Computer Configuration
 - Startup
 - Every 90-120 minutes
 - Triggered: GPUpdate command
- User Configuration
 - Logon
 - Every 90-120 minutes
 - Triggered: GPUpdate command

Key Points

When are policies applied? Policy settings in the Computer Configuration node are applied at system startup and every 90 to 120 minutes thereafter. User Configuration policy settings are applied at logon and every 90 to 120 minutes thereafter. The application of policies is called Group Policy *refresh*.

You can also force a policy refresh using the GPUpdate command.

You will learn more about Group Policy refresh in Lesson 5.

Resultant Set of Policy

- The "cumulative" effect of Group Policy
 - A user or computer is usually within the scope of many GPOs
 - Potentially conflicting settings: precedence
- Tools to report the settings that were applied and which GPO "won" in the case of conflicting settings
- Tools to model the effects of changes to the Group Policy infrastructure or to the location of objects in Active Directory

Key Points

Computers and users within the scope of a GPO will apply the policy settings specified in the GPO. An individual user or computer is likely to be within the scope of multiple GPOs linked to the sites, domain, or OUs in which the user or computer exists. This leads to the possibility that policy settings might be configured differently in multiple GPOs. You must be able to understand and evaluate the *Resultant Set of Policy (RSOP)*, which determines the settings that are applied by a client when the settings are configured divergently in more than one GPO.

RSOP will be examined in Lesson 6.

Review and Discuss the Components of Group Policy

- Setting
- Scope
- Application
- Tools

Key Points

Review the key components of Group Policy.

Additional Reading

TechNet contains detailed technical and operational guides to Group Policy, including the following:

- Windows Server Group Policy
<http://go.microsoft.com/fwlink/?LinkId=99449>
- How Core Group Policy Works
<http://go.microsoft.com/fwlink/?LinkId=99468>

- Deploying Group Policy Using Windows Vista
<http://go.microsoft.com/fwlink/?LinkId=169357>
- Summary of New or Expanded Group Policy Settings
<http://go.microsoft.com/fwlink/?LinkId=99450>
- What's New in Group Policy in Windows Vista
<http://go.microsoft.com/fwlink/?LinkId=99451>

Lesson 2

Implement GPOs

- Local GPOs
- Domain-Based GPOs
- Demonstration: Create, Link, and Edit GPOs
- GPO Storage
- Demonstration: Policy Settings

Now that you have a broad understanding of Group Policy and its components, you can look more closely at each component. In this section, you will examine GPOs in detail.

Objectives

After completing this lesson, you will be able to:

- Create, edit, and link Group Policy objects.
- Identify change and configuration management capabilities of Group Policy.
- Configure policy settings.
- Explain GPO storage, replication, and versioning.

Local GPOs

- **Apply before domain-based GPOs**
 - Any setting specified by a domain-based GPO will override the setting specified by the local GPOs.
- **Local GPO**
 - *One* local GPO in Windows 2000, Windows XP, Windows Server® 2003
 - Multiple local GPOs in Windows Vista® and later
 - Local GPO: Computer settings and settings for all users
 - Administrators GPO: Settings for users in Administrators
 - Non-administrators GPO: Settings for users not in Admins
 - Per-user GPO: Settings for a specific user
- **If domain members can be centrally managed using domain-linked GPOs, in what scenarios might local GPOs be used?**

Key Points

To manage configuration for users and computers, you create GPOs that contain the policy settings you require. Each computer has several GPOs stored locally on the system—the local GPOs—and can be within the scope of any number of domain-based GPOs.

Computers running Windows 2000, Windows XP, and Windows Server 2003 each have one local GPO, which can manage configuration of that system. The local GPO exists whether or not the computer is part of a domain, a workgroup, or a non-networked environment. It is stored in %SystemRoot%\System32\GroupPolicy. The policies in the local GPO affect only the computer on which the GPO is stored. By default, only the Security Settings policies are configured on a system's local GPO. All other policies are set at Not Configured.

When a computer does not belong to an Active Directory domain, the local policy is useful to configure and enforce configuration on that computer. However, in an Active Directory domain, settings in GPOs that are linked to the site, domain, or OUs will override local GPO settings and are easier to manage than GPOs on individual computers.

Windows Vista and Windows Server 2008 and later systems have multiple local GPOs. The Local Computer GPO is the same as the GPO in previous versions of Windows. In the Computer Configuration node, configure all computer-related settings. In the User Configuration node, configure settings you want to apply to all users on the computer. The user settings in the Local Computer GPO can be modified by the user settings in two new local GPOs: Administrators and Non-Administrators. These two GPOs apply user settings to logged-on users according to whether they are members of the local Administrators group (and thus would use the Administrators GPO) or not members of the Administrators group (and thus would use the Non-Administrators GPO). . You can further refine user settings with a local GPO that applies to a specific user account. User-specific local GPOs are associated with local, not domain, user accounts.

RSOP is easy for computer settings: the Local Computer GPO is the only local GPO that can apply computer settings. User settings in a user-specific GPO will override conflicting settings in the Administrators and Non-Administrators GPOs, which themselves override settings in the Local Computer GPO. The concept is simple: the more specific the local GPO, the higher the precedence of its settings.

To create and edit local GPOs:

1. Click the **Start** button and then, in the **Start Search** box, type **mmc.exe** and press ENTER.

An empty Microsoft® Management console (MMC) opens.

2. Click **File**, and then click **Add/Remove Snap-in**.
3. Select the **Group Policy Object Editor** and click **Add**.

A dialog box will appear, prompting you to select the GPO to edit.

4. The **Local Computer GPO** is selected by default. If you want to edit another local GPO, click the **Browse** button. On the **Users** tab, you will find the **Non-Administrators** and **Administrators** GPOs and one GPO for each local user. Select the GPO and click **OK**.
5. Click **Finish** and then **OK** to close each of the dialog boxes.

The Group Policy Object editor snap-in is added, focused on the selected GPO.

Question: If domain members can be centrally managed using domain-linked GPOs, in what scenarios might local GPOs be used?

Additional Reading

- Multiple Local Group Policy objects
<http://go.microsoft.com/fwlink/?LinkId=112463>
- Step-by-Step Guide to Managing Multiple Local Group Policy Objects
<http://go.microsoft.com/fwlink/?LinkId=99457>

Domain-Based GPOs

- Created in Active Directory, stored on domain controllers
- Two default GPOs
 - Default Domain Policy
 - Define account policies for the domain: Password, account lockout, and Kerberos policies
 - Default Domain Controllers Policy
 - Define auditing policies for domain controllers and Active Directory

Key Points

Domain-based GPOs are created in Active Directory and stored on domain controllers. They are used to manage configuration centrally for users and computers in the domain. The remainder of this course refers to domain-based GPOs rather than local GPOs, unless otherwise specified.

When AD DS is installed, two default GPOs are created:

Default Domain Policy

This GPO is linked to the domain and has no security group or WMI filters. Therefore, it affects all users and computers in the domain (including computers that are domain controllers). This GPO contains policy settings that specify password, account lockout, and Kerberos policies. As discussed in Module 9, you will modify the default settings in this GPO to align with your enterprise password and account lockout policies. You should not add unrelated policy settings to this GPO. If you need to configure other settings to apply broadly in your domain, create additional GPOs linked to the domain.

Default Domain Controllers Policy

This GPO is linked to the Domain Controllers OU. Because computer accounts for domain controllers are kept exclusively in the Domain Controllers OU, and other computer accounts should be kept in other OUs, this GPO affects only domain controllers. The Default Domain Controllers GPO should be modified to implement your auditing policies, as discussed in Modules 7 through 9. It should also be modified to assign user rights required on domain controllers.

Demonstration: Create, Link, and Edit GPOs

In this demonstration, we will:

- Delete a GPO
- Create a GPO
- Link a GPO
- Open a GPO for editing
- Delegate the management of GPOs
 - Creation
 - Linking
 - Editing
- Discuss the default connection to the PDC Emulator

Key Points

To create a GPO, right-click the Group Policy Objects container and choose New.

You must have permission to the Group Policy Objects container to create a GPO. By default, the Domain Admins group and the Group Policy Creator Owners group are delegated the ability to create GPOs.

To delegate permission to create GPOs to other groups, select the Group Policy Objects container in the GPMC console tree and then click the Delegation tab in the console details pane.

After you have created a GPO, you can create the initial scope of the GPO by linking it to a site, domain, or OU.

To link a GPO, right-click the site, domain, or OU and then click Link An Existing GPO.

You can also create and link a GPO with a single step: right-click a site, domain, or OU, and then click Create A GPO In This Domain And Link It Here.

Note that you will not see your sites in the Sites node of the GPMC until you right-click Sites, choose Show Sites, and then select the Sites you want to manage.

You must have permission to link GPOs to a site, domain, or OU. In the GPMC, select the container in the console tree, and then click the Delegation tab in the console details pane. From the Permission drop-down list, select Link GPOs. The users and groups displayed hold the permission for the selected OU. Click the Add or Remove buttons to modify the delegation.

To edit a GPO, right-click the GPO in the Group Policy Objects container and choose Edit.

The GPO is opened in the GPME. You must have at least Read permission to open the GPO in this way.

To make changes to a GPO, you must have Write permission to the GPO. Permissions for the GPO can be set by selecting the GPO in the Group Policy Objects container and then clicking the Delegation tab in the details pane.

The GPME will display the name of the GPO as the root node. The GPME also displays the domain in which the GPO is defined and the server from which the GPO was opened and to which changes will be saved. The root node is in the GPOName [ServerName] format. In the screenshot of the GPME on an earlier page in this module, the root node is CONTOSO Standards [SERVER01.contoso.com] Policy. The GPO name is CONTOSO Standards, and it was opened from SERVER01.contoso.com, meaning that the GPO is defined in the contoso.com domain.

By default, both the GPMC and the GPME console connect to a specific domain controller in your environment: the domain controller acting as the PDC Emulator. In a later module, you will learn to identify and manage which domain controller has this role.

This is done in order to reduce the possibility that a single GPO might be changed on two different domain controllers, at which point during replication there would be no way to reconcile the changes, and only one version of the entire GPO would "win" and be replicated. Focusing the administrative tools on one domain controller helps ensure that changes are made in one place.

However, in a large, distributed environment, the PDC Emulator may be in a distant site, resulting in slow performance for the GPMCs. You can right-click the root node of each console and connect to a specific domain controller closer to you. Just be cognizant of the replication issue: If you are the only one who is editing a GPO, it is perfectly acceptable for you to do so on a local, higher performing domain controller.

Demonstration Steps

Create a GPO

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Run **Group Policy Management** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
4. In the console tree, expand **Forest: contoso.com**, **Domains**, and **contoso.com**, and then click the **Group Policy Objects** container.
5. In the console tree, right-click the **Group Policy Objects** container, and then click **New**.
6. In **Name**: type **CONTOSO Standards**, and then click **OK**.

Open a GPO for editing

1. In the details pane of the Group Policy Management console (GPMC), right-click the **CONTOSO Standards** GPO, and then click **Edit**.

The Group Policy Management Editor (GPME) appears.

2. Close the GPME.

Link a GPO

1. In the GPMC console tree, right-click the **contoso.com** domain, and then click **Link an Existing GPO**.
2. Select **CONTOSO Standards** and click **OK**.

Delegate the management of GPOs

1. In the GPMC console tree, click the **contoso.com** domain.
2. In the details pane, click the **Delegation** tab.
3. Review the default delegation.
4. In the GPMC console tree, expand the **Group Policy Objects** container, and then click the **CONTOSO Standards** GPO.
5. In the details pane, click the **Delegation** tab.
6. Review the default delegation.

7. Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
8. In the console tree, click the **Users** container.
9. In the details pane, double-click the **Group Policy Creator Owners** group, and then click the **Members** tab.
10. Review the default membership.

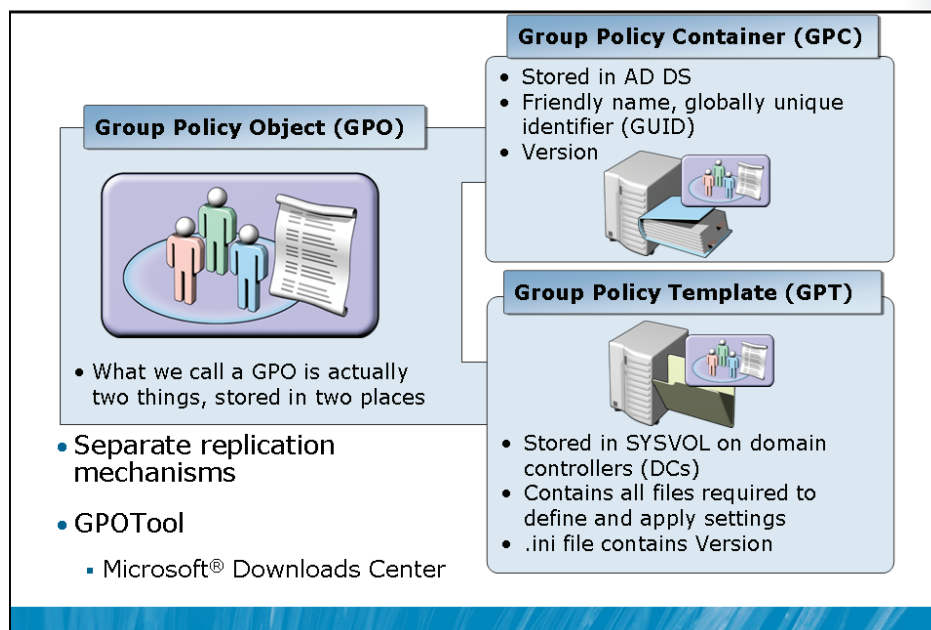
Delete a GPO

1. In the GPMC console tree, in the **Group Policy Objects** container, right-click the **CONTOSO Standards** GPO, and then click **Delete**.
2. Click **No**.

Discuss the default connection to the PDC Emulator

1. Switch to the GPMC.
2. In the GPMC console tree, right-click the **contoso.com** domain, and then click **Change Domain Controller**.
3. Review the default settings.

GPO Storage



Key Points

Group Policy settings are presented as GPOs in Active Directory user interface tools, but a GPO is actually two components: a *Group Policy Container (GPC)* and a *Group Policy Template (GPT)*.

The GPC is an Active Directory object stored in the Group Policy Objects container within the domain-naming context of the directory. Like all Active Directory objects, each GPC includes a *globally unique identifier (GUID)* attribute that uniquely identifies the object within Active Directory. The GPC defines basic attributes of the GPO, but it does not contain any of the settings. The settings are contained in the GPT, a collection of files stored in the SYSVOL of each domain controller in the %SystemRoot%\SYSVOL\Domain\Policies\GPOGUID path, where GPOGUID is the GUID of the GPC. When you make changes to the settings of a GPO, the changes are saved to the GPT of the server from which the GPO was opened.

By default, when Group Policy refresh occurs, the CSEs apply settings in a GPO only if the GPO has been updated.

The Group Policy Client can identify an updated GPO by its version number. Each GPO has a version number that is incremented each time a change is made. The version number is stored as an attribute of the GPC and in a text file, GPT.ini, in the GPT folder. The Group Policy Client knows the version number of each GPO it has previously applied. If, during Group Policy refresh, it discovers that the version number of the GPC has been changed, the CSEs will be informed that the GPO is updated.

GPO Replication

The two parts of a GPO are replicated between domain controllers by using distinct mechanisms.

The GPC in Active Directory is replicated by the Directory Replication Agent (DRA), using a topology generated by the Knowledge Consistency Checker (KCC) that can be defined or refined manually. You will learn more about Active Directory Replication in Module 12. The result is that the GPC is replicated within seconds to all domain controllers in a site and is replicated between sites based on your intersite replication configuration, which will also be discussed in Module 12.

The GPT in the SYSVOL is replicated using one of two technologies. The File Replication Service (FRS) is used to replicate SYSVOL in domains running Windows Server 2008, Windows Server 2003, and Windows 2000. If all domain controllers are running Windows Server 2008, you can configure SYSVOL replication by using Distributed File System Replication (DFSR), a much more efficient and robust mechanism.

Because the GPC and GPT are replicated separately, it is possible for them to become out of sync for a short time.

Typically, when this happens, the GPC will replicate to a domain controller first. Systems that obtained their ordered list of GPOs from that domain controller will identify the new GPC, will attempt to download the GPT, and will notice that the version numbers are not the same. A policy processing error will be recorded in the event logs. If the reverse happens, and the GPO replicates to a domain controller before the GPC, clients obtaining their ordered list of GPOs from that domain controller will not be notified of the new GPO until the GPC has replicated.

You can download from the Microsoft Download Center the Group Policy Verification Tool, GPTool.exe, which is part of Windows Resource Kits. This tool reports the status of GPOs in the domain and can identify instances in which, on a domain controller, the GPC and the GPT do not have the same version. For more information about GPTool.exe, type `gpoutil /?` at the command line.

Demonstration: Policy Settings

In this demonstration, we will explore some of the thousands of settings in a Group Policy object

Key Points

Group Policy settings, also known simply as policies, are contained in a GPO and are viewed and modified using the GPME. In this demonstration, you will look more closely at the categories of settings available in a GPO.

Computer Configuration and User Configuration

There are two major divisions of policy settings: computer settings, contained in the Computer Configuration node, and user settings, contained in the User Configuration node.

- The *Computer Configuration* node contains the settings that are applied to computers, regardless of who logs on to them. Computer settings are applied when the operating system starts up and during background refresh every 90–120 minutes thereafter.
- The *User Configuration* node contains settings that are applied when a user logs on to the computer and during background refresh every 90–120 minutes thereafter.

Within the Computer Configuration and User Configuration nodes are the *Policies* and *Preferences* nodes. Policies are settings that are configured and behave similarly to the policy settings in earlier versions of Windows. Preferences are introduced in Windows Server 2008. The following sections examine these nodes.

Within the Policies nodes within Computer Configuration and User Configuration are a hierarchy of folders containing policy settings. Because there are thousands of settings, it is beyond the scope of the exam and of this course to examine individual settings. It is worthwhile, however, to define the broad categories of settings in the folders.

Software Settings Node

The first of these nodes is the Software Settings node, which contains only the Software Installation extension. The Software Installation extension helps you specify how applications are installed and maintained within your organization. It also provides a place for independent software vendors to add settings. Software deployment with Group Policy is discussed in Module 7.

Windows Settings Node

In both the Computer Configuration and User Configuration nodes, the Policies node contains a Windows Settings node that includes the Scripts, Security Settings, and Policy-Based QoS nodes.

The Scripts extension enables you to specify two types of scripts: startup/shutdown (in the Computer Configuration node) and logon/logoff (in the User Configuration node). Startup/shutdown scripts run at computer startup or shutdown. Logon/logoff scripts run when a user logs on or off the computer. When you assign multiple logon/logoff or startup/shutdown scripts to a user or computer, the Scripts CSE executes the scripts from top to bottom. You can determine the order of execution for multiple scripts in the Properties dialog box. When a computer is shut down, the CSE first processes logoff scripts, followed by shutdown scripts. By default, the timeout value for processing scripts is 10 minutes. If the logoff and shutdown scripts require more than 10 minutes to process, you must adjust the timeout value with a policy setting. You can use any ActiveX® scripting language to write scripts. Some possibilities include Microsoft Visual Basic® Scripting Edition (VBScript), Microsoft JScript®, Perl, and Microsoft MS-DOS®-style batch files (.bat and .cmd). Logon scripts on a shared network directory in another forest are supported for network logon across forests.

The Security Settings node allows a security administrator to configure security by using GPOs. This can be done after, or instead of, using a security template to set system security. For a detailed discussion of system security and the Security Settings node, refer to Module 7.

The Policy-Based QoS node defines policies that manage network traffic. For example, you might want to ensure that users in the Finance department have priority for running a critical network application during the end-of-year financial reporting period. Policy-Based QoS enables you to do that.

In the User Configuration node only, the Windows Settings folder contains the additional Remote Installation Services, Folder Redirection, and Internet Explorer Maintenance nodes. Remote Installation Services (RIS) policies control the behavior of a remote operating system installation. Folder Redirection enables you to redirect user data and settings folders (AppData, Desktop, Documents, Pictures, Music, and Favorites, for example) from their default user profile location to an alternate location on the network, where they can be centrally managed. Internet Explorer Maintenance enables you to administer and customize Microsoft Internet Explorer®.

Administrative Templates Node

In both the Computer Configuration and User Configuration nodes, the Administrative Templates node contains registry-based Group Policy settings. There are thousands of such settings available for configuring the user and computer environment. As an administrator, you might spend a significant amount of time manipulating these settings. To assist you with the settings, a description of each policy setting is available in two locations:

- **On the Explain tab in the Properties dialog box for the setting.** In addition, the Settings tab in the Properties dialog box for the setting lists the required operating system or software for the setting.
- **On the Extended tab of the GPME.** The Extended tab appears on the bottom of the right details pane and provides a description of each selected setting in a column between the console tree and the settings pane. The required operating system or software for each setting is also listed.

The Administrative Templates node is discussed in detail later in this module.

Preferences Node

Underneath both Computer Configuration and User Configuration is a Preferences node. New to Windows Server 2008, preferences provide more than 20 CSEs to help you manage an incredible number of additional settings, including:

- Applications such as Microsoft Office 2003 and Office 2007
- Mapped drives
- Registry settings

- Power options
- Folder options
- Regional options
- Start menu options

Preferences also enables you to deploy the following:

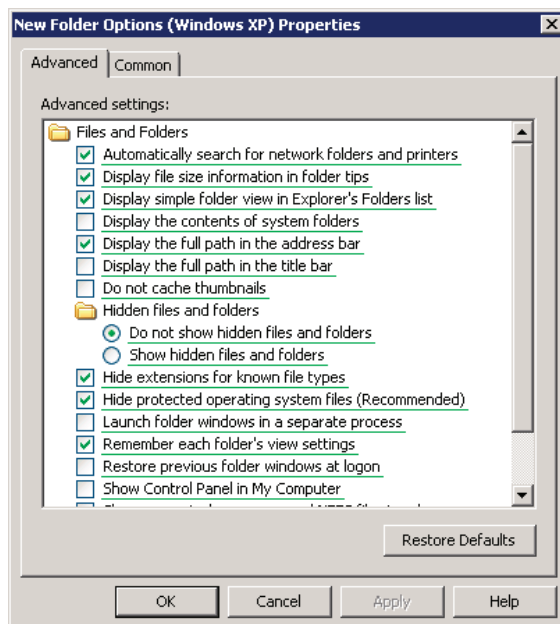
- Files and folders
- Shortcuts
- Printers
- Scheduled tasks
- Network connections

Many enterprises will also benefit from preferences because the options can be used to enable or disable hardware devices or classes of devices. For example, you can use preferences to prevent USB hard drives, including personal media players, from being connected to computers.

You must use the new version of the GPME to configure preferences. This new version is part of the Remote Server Administration Tools that can be installed on Windows Server 2008, Windows Vista, and later operating systems.

To apply preferences, systems require the preferences CSEs, which is included with Windows Server 2008 and Windows 7. CSEs for Windows XP, Windows Server 2003, and Windows Vista can be downloaded from the Microsoft Download Center.

The interface you use to configure many preferences looks identical to the Windows user interface in which you would make the change manually.



The figure above shows a Folder Options (Windows XP) preference item—a collection of settings that are processed by the preferences CSE. You will recognize the similarity to the Folder Options application in Control Panel.

Demonstration Steps

1. Switch to HQDC01.
2. Right-click the **CONTOSO Standards** GPO, and then click **Edit**.
3. Spend time exploring the settings that are available in a GPO. Do not make any changes.

Lab A: Implement Group Policy

- Exercise 1: Create, Edit, and Link Group Policy Objects

Logon information

Virtual machine	6425B-HQDC01-A	6425B-Desktop101-A
Logon user name	Pat.Coleman	Pat.Coleman
Administrative user name	Pat.Coleman_Admin	
Password	Pa\$\$w0rd	Pa\$\$w0rd

Estimated time: 30 minutes

Scenario

You are responsible for managing change and configuration at Contoso, Ltd. Contoso corporate IT security policies specify that computers cannot be left unattended and logged onto for more than 10 minutes. You will therefore configure the screen-saver timeout and password-protected screen-saver policy settings. Additionally, you will lock down access to registry editing tools.

Exercise 1: Create, Edit, and Link Group Policy Objects

In this exercise, you will create a GPO that implements a setting mandated by the corporate security policy of Contoso, Ltd., and scope the setting to all users and computers in the domain. You will then experience the effect of the GPO. Any remaining time can be used exploring settings that are made available within a Group Policy object.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Create a GPO.
3. Edit the settings of a GPO.
4. Scope a GPO with a GPO link.
5. View the effects of Group Policy application.
6. Explore GPO settings.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Start 6425B-DESKTOP101-A but do not log on to the system.

► Task 2: Create a GPO

1. Run **Group Policy Management** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Create a **Group Policy** object named **CONTOSO Standards** in the **Group Policy** objects container.

► Task 3: Edit the settings of a GPO

1. Edit the **CONTOSO Standards** GPO.
2. Navigate to the **User Configuration, Policies, Administrative Templates, System** folder.
3. Prevent users from running **Registry Editor** and **regedit /s**.

4. Navigate to the **User Configuration, Policies, Administrative Templates, Control Panel, Display** folder.
5. Examine the explanatory text for the **Screen Saver** timeout policy setting.
6. Configure the **Screen Saver** timeout policy to **600** seconds.
7. Enable the **Password protect the screen saver** policy setting.

► **Task 4: Scope a GPO with a GPO link**

- Link the **CONTOSO Standards** GPO to the **contoso.com** domain.

► **Task 5: View the effects of Group Policy application**

1. Log on to DESKTOP101 as **Pat.Coleman**.
2. Attempt to change the **Screen Saver** timeout and password protection. You will be prevented from doing so by Group Policy.
3. Attempt to run **Registry Editor**. You will be prevented from doing so by Group Policy.

► **Task 6: Explore GPO settings**

- On HQDC01, edit the **CONTOSO Standards** GPO and spend time exploring the settings that are available in a GPO. Do not make any changes.

Results: After this exercise, you will have a GPO named **CONTOSO Standards** that configures password-protected screen saver, screen-saver timeout, and registry editing tool restrictions.



Note: Do not shut down the virtual machines once you are finished with this lab as the settings you have configured here will be used in subsequent labs.

Lab Review Questions

Question: What policy settings are already being deployed using Group Policy in your organization?

Question: What policy settings did you discover that you might want to implement in your organization?

Lesson 3

A Deeper Look at Settings and GPOs

- Registry Policies in the Administrative Templates Node
- Managed Settings, Unmanaged Settings, and Preferences
- Administrative Templates
- The Central Store
- Demonstration: Work with Settings and the GPOs
- Managed GPOs and their Settings

In Lessons 1 and 2, you learned enough fundamentals to implement Group Policy in an AD DS domain. However, to really master Group Policy and to manage Group Policy in a real-world, complex enterprise, you must understand more detail about settings, GPOs, and management of Group Policy.

Objectives

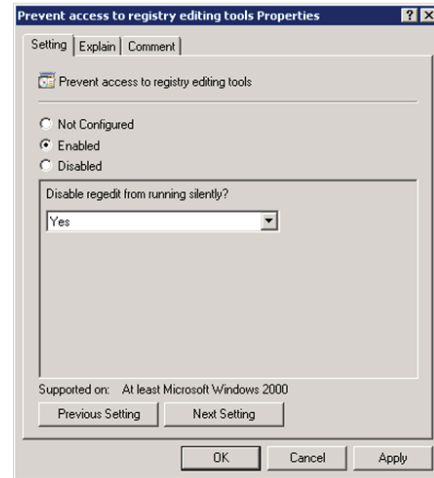
After completing this lesson, you will be able to:

- Understand the differences between policies, preferences, and managed and unmanaged settings.
- Create the central store for administrative templates.
- Document GPO and policy settings by using comments.

- Search for specific policy settings in a GPO.
- Create a GPO from a Starter GPO.
- Back up a GPO.
- Create a GPO with settings from a backed up GPO.

Registry Policies in the Administrative Templates Node

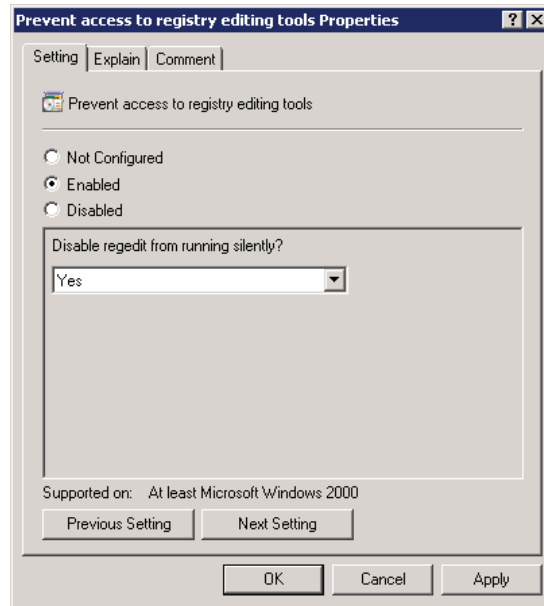
- Policy settings in the Administrative Templates node make changes to the registry
- HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System
 - DisableRegeditMode
 - 1 - Regedit UI tool only
 - 2 - Also disable regedit /s



Key Points

In the Administrative Templates node, you will find thousands of settings that allow you to control many aspects of Windows.

Below, you can see the Properties dialog box for the Prevent Access To Registry Editing Tools policy setting:



If this setting is enabled and the user tries to start a registry editor, a message appears explaining that a setting prevents the action.



Note: To prevent users from using other administrative tools, use the **Run Only Specified Windows Applications** setting, or use Software Restriction Policies, which are beyond the scope of this course.

Policies in the Administrative Templates node make changes to the registry. settings in the Computer Configuration node modify registry values in the HKEY_LOCAL_MACHINE (HKLM) key. Settings in the Administrative Templates node in the User Configuration node modify registry values in the HKEY_CURRENT_USER (HKCU) key.

In the case of this policy setting, the following registry value is modified:

```
HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\System\Disable  
RegeditMode
```

If you choose to restrict Regedit from running silently, that value is set to 2. If you choose to restrict only the Registry Editor UI tool, the value is set to 1.

Managed Settings, Unmanaged Settings, and Preferences

- Administrative templates
 - Managed policy setting
 - User interface (UI) is locked; user cannot make a change to the setting
 - Changes are made in one of four reserved registry keys
 - Change and UI lock are "released" when the user/computer falls out of scope
 - Unmanaged policy setting
 - UI not locked
 - Makes a change that is persistent; "tattoos" the registry
 - Only managed setting shown by default
 - Set Filter Options to view unmanaged settings
- Preferences
 - Effects vary

Key Points

There is a nuance to the registry policy settings configured by the Administrative Templates node that is important to understand: the difference between managed and unmanaged policy settings.

A *managed policy setting* has the following characteristics:

- **The user interface (UI) is locked so a user cannot change the setting.** Managed policy settings result in the appropriate UI being disabled. For example, if you configure the Screensaver Timeout policy setting, a user cannot change the timeout delay in the UI.
- **Changes are made in one of four keys in the registry reserved for managed policy settings:**
 - HKLM\Software\Policies (computer settings)
 - HKCU\Software\Policies (user settings)

- HKLM\Software\Microsoft\Windows\Current Version\Policies (computer settings)
- HKCU\Software\Microsoft\Windows\Current Version\Policies (user settings)

These keys are secured so that only administrators can make a change. Together with UI lockout, this means that non-administrative users will receive the change specified by the policy setting and cannot modify the setting on their computer.

- **Changes made by a Group Policy setting, and the UI lockout, are "released" if the user or computer falls out of scope of the GPO.** For example, if you delete a GPO, managed policy settings that had applied to a user will be released. This means that, generally, the setting reverts back to its previous state. Additionally, the UI interface for the setting is enabled.

The registry policy settings that have been discussed so far and that are encountered in the practices of this chapter are examples of managed policy settings. A managed policy setting effects a configuration change of some kind when the setting is applied by a GPO. When the user or computer is no longer within the scope of the GPO, the configuration is released automatically.

For example, if a GPO prevents access to registry editing tools, and then the GPO is deleted, disabled, or scoped so that it no longer applies to users, those users will regain access to registry editing tools at the next policy refresh, which is Windows' default behavior, unless you have implemented a restriction at some other level.

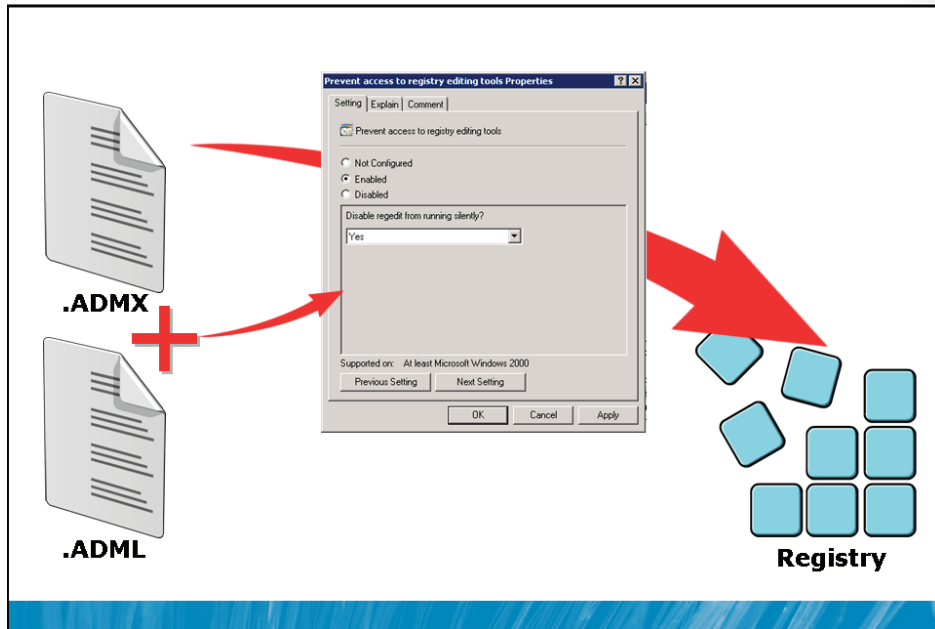
In contrast, an *unmanaged policy setting* makes a change that is persistent in the registry. If the GPO no longer applies, the setting remains. This is often called "tattooing" the registry—making a permanent change. To reverse the effect of the policy setting, you must deploy a change that reverts the configuration to the desired state. Additionally, an unmanaged policy setting does not lock the UI for that setting.

By default, the GPME hides unmanaged policy settings to discourage you from implementing a configuration that is difficult to revert. However, you can make many useful changes with unmanaged policy settings, particularly for custom administrative templates to manage configuration for applications.

To control which policy settings are visible, right-click Administrative Templates and click Filter Options, and then make a selection from the Managed drop-down list.

Later in this module, you will work with Group Policy preferences. When a change is made by a preference, the change tattoos the system. However, some preferences include an option to remove the preference when it no longer applies to the user or computer. This is not the same as a managed policy setting, which is released and often returned to its original value. Instead, when a preference is removed, the setting is actually deleted entirely.

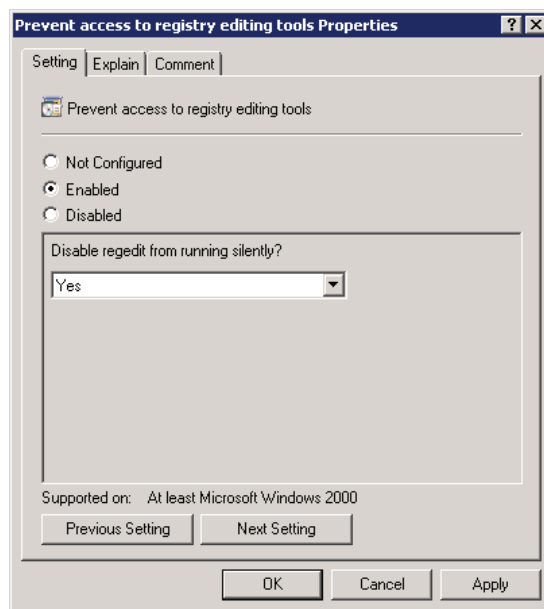
Administrative Templates



Key Points

Why are the Administrative Templates nodes called *Administrative Templates*? Because the settings that the nodes contain are derived from files called administrative templates.

An administrative template is a text file that specifies the registry change to be made and that generates the user interface to configure the Administrative Templates policy settings in the GPME. The screen shot here shows the properties dialog box for the Prevent Access To Registry Editing Tools policy setting:



The fact that the setting exists, and that it provides a drop-down list with which to disable Regedit.exe from running silently, is determined in an administrative template. The registry setting that is made based on how you configure the policy is also defined in the administrative template.

Some software vendors provide administrative templates as a mechanism to manage the configuration of their application centrally. For example, you can obtain administrative templates for all recent versions of Microsoft Office from the Microsoft Downloads Center. You can also create your own custom administrative templates. A tutorial on creating custom administrative templates is beyond the scope of this course.

.ADM Files

In versions of Windows prior to Windows Vista, an administrative template had an .ADM extension. .ADM files have several drawbacks. First, all localization must be performed within the .ADM file. That is, if you want to create an .ADM file to help deploy configuration in a multilingual organization, you would need separate .ADM files for each language to provide a user interface for administrators who speak that language. If you were to decide later to make a modification related to the registry settings managed by the templates, you would need to make the change to each .ADM file.

The second problem with .ADM files is the way they are stored. An .ADM file is stored as part of the GPT in the SYSVOL. If an .ADM file is used in multiple GPOs, it is stored multiple times, contributing to SYSVOL bloat. There were also challenges maintaining version control over .ADM files.

To add classic administrative templates to the GPME, right-click the Administrative Templates node and then click Add/Remove Templates.

.ADMX/.ADML Files

In Windows Vista and Windows Server 2008, an administrative template is a pair of XML files, one with an .ADMX extension that specifies changes to be made to the registry, and the other with an .ADML extension that provides a language-specific user interface in the GPME. When changes need to be made to settings managed by the administrative template, they can be made to the single .ADMX file. Any administrator who modifies a GPO that uses the template accesses the same .ADMX file and calls the appropriate .ADML file to populate the user interface.

To add .ADMX/.ADML administrative templates to the GPME, Copy the .ADMX file into the %SystemRoot%\PolicyDefinitions folder on your client, or in the central store. Copy the .ADML file into the language-and-region-specific subfolder, such as *en-us*, of %SystemRoot%\PolicyDefinitions on your client, or in the central store. The central store will be discussed in the next topic.

No Need to Take Sides

.ADM and .ADMX/.ADML administrative templates can coexist. Settings generated by .ADM files will appear under the Administrative Templates node in a node labeled Classic Administrative Templates (ADM).

Migrate Classic Administrative Templates (.ADM) to .ADMX

The ADMX Migrator enables you to convert ADM files to the ADMX format. For more information, see:

- ADMX Migrator
<http://go.microsoft.com/fwlink/?LinkId=99466>
- ADMX Migrator download (Blog)
<http://go.microsoft.com/fwlink/?LinkId=113124>

The Central Store

- **.ADM files**
 - Stored in the GPT
 - Leads to version control and GPO bloat problems
- **.ADMX/.ADML files**
 - Retrieved from the client
 - Problematic if the client doesn't have the appropriate files
- **Central Store**
 - Create a folder called PolicyDefinitions on a DC
 - Remotely: `\\contoso.com\SYSVOL\contoso.com\Policies\PolicyDefinitions`
 - Locally: `%SystemRoot%\SYSVOL\contoso.com\Policies\PolicyDefinitions`
 - Copy .ADMX files from your `%SystemRoot%\PolicyDefinitions`
 - Copy .ADML file from language-specific subfolders (such as en-us)

Key Points

As was previously stated, .ADM files are stored as part of the GPO itself, in the GPT. When you edit a GPO that uses administrative templates in the .ADM format, the GPME loads the .ADM from the GPT to produce the user interface. When .ADMX/.ADML files are used as administrative templates, the GPO contains only the data that the client needs for processing Group Policy, and when you edit the GPO, the GPME pulls the .ADMX and .ADML files from the local workstation.

This works well for smaller organizations, but for complex environments that include custom administrative templates or that require more centralized control, Windows Server 2008 introduces Central Store. Central Store is a single folder in SYSVOL that holds all the .ADMX and .ADML files that are required. Once you have set up Central Store, the GPME recognizes it and loads all administrative templates from Central Store instead of from the local machine.

To create a central store:

1. Create a folder called **PolicyDefinitions** in the `\\FQDN\SYSVOL\FQDN\Policies` path.

For example, the central store for the contoso.com domain would be:

```
\\contoso.com\SYSVOL\contoso.com\Policies\PolicyDefinitions
```

If you log on to a domain controller, locally or by using Remote Desktop, the local path to the PolicyDefinitions folder is:

```
%SystemRoot%\SYSVOL\domain\Policies\PolicyDefinitions
```

2. Copy all .ADMX files from the %SystemRoot%\PolicyDefinitions folder of a Windows Server 2008 system to the new SYSVOL PolicyDefinitions folder.
3. Copy the .ADML files from the appropriate language-specific subfolder of %SystemRoot%\PolicyDefinitions into the language-specific subfolder of the new SYSVOL PolicyDefinitions folder.

For example, English (United States) .ADML files are located in %SystemRoot%\PolicyDefinitions\en-us. Copy them into `\\FQDN\SYSVOL\FQDN\Policies\PolicyDefinitions\en-us`.

4. If additional languages are required, copy the folder that contains the .ADML files to Central Store.

When you have copied all .ADMX and .ADML files, the PolicyDefinitions folder on the domain controller should contain the .ADMX files and one or more folders containing language-specific .ADML files.



Note: You can use the Central Store in a mixed environment, with clients and servers running operating systems earlier than Windows Vista and Windows Server 2008. However, you must use a Windows Vista, Windows Server 2008, or later operating system to *manage* Group Policy. That is, your administrative workstation must be running a version of Windows that is able to work with the Central Store. The GPOs you create can be applied to previous versions of Windows.

Demonstration: Work with Settings and GPOs

In this demonstration, we will:

- Use Filter Options to locate policies in Administrative Templates
- Add comments to a policy setting
- Add comments to a GPO
- Create a new GPO from a starter GPO
- Create a new GPO by copying an existing GPO
- Create a new GPO by importing settings that were exported from another GPO

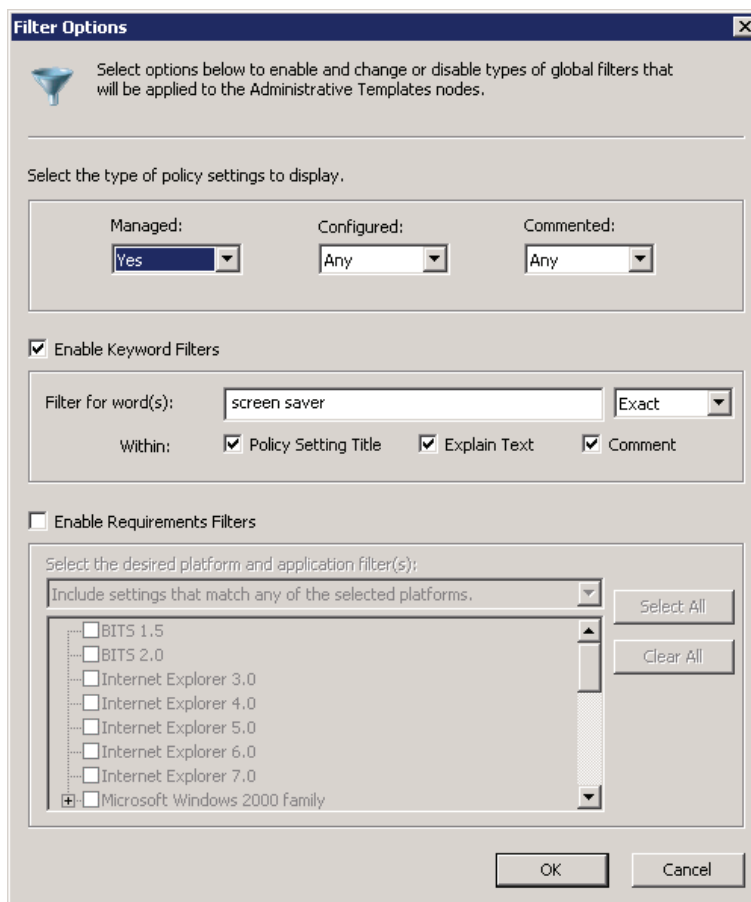
Key Points

Filter Administrative Template Policy Settings

A weakness of the Group Policy editing tools in previous versions of Windows is the inability to search for a specific policy setting. With thousands of policies to choose from, it can be difficult to locate exactly the setting you want to configure. The new GPME in Windows Server 2008 solves this problem for Administrative Template settings: you can now create filters to locate specific policy settings.

To create a filter:

1. Right-click **Administrative Templates** and choose **Filter Options**.
2. To locate a specific policy, select **Enable keyword filters**, enter the words with which to filter, and select the fields within which to search. The screen shot here shows an example of a search for policy settings related to the screen saver:



In the top section of the Filter Options dialog box shown above, you can filter the view to show only policy settings that are configured. This can help you locate and modify settings that are already specified in the GPO.

You can also filter for Group Policy settings that apply to specific versions of Windows, Internet Explorer, and other Windows components.

Unfortunately, the filter only applies to settings in the Administrative Templates nodes.

Comments

You can also search and filter based on policy-setting comments. Windows Server 2008 enables you to add comments to policy settings in the Administrative Templates node. To do so, double-click a policy setting and click the Comment tab.

It is a best practice to add comments to configured policy settings as a way to document the justification for a setting and its intended effect. You should also add comments to the GPO itself. Windows Server 2008 enables you to attach comments to a GPO: In the GPME, right-click the root node in the console tree and choose Properties, and then click the Comment tab.

Starter GPOs

Another new Group Policy feature in Windows Server 2008 is starter GPOs. A starter GPO contains Administrative Template settings. You can create a new GPO from a starter GPO, in which case the new GPO is prepopulated with a copy of the settings in the starter GPO. A starter GPO is, in effect, a template. Unfortunately, Microsoft had already been using the term *template* in the context of administrative templates, so another name had to be found. When you create a new GPO, you can still choose to begin with a blank GPO, or you can select one of the preexisting starter GPOs or a custom starter GPO.

After you have created a GPO from a starter GPO, there is no "link" to the starter GPO. Changes to the starter GPO do not affect the GPOs that were previously created from the starter GPO.

Other Ways to Copy GPO Settings

Starter GPOs can contain only Administrative Templates policy settings. There are two other ways to copy settings from one GPO into another, new GPO.

- You can copy and paste entire GPOs in the Group Policy Objects container of the GPMC so that you have a new GPO with all the settings of the source GPO.
- To transfer settings between GPOs in different domains or forests, right-click a GPO and choose Back Up. In the target domain, create a new GPO, right-click it, and choose Import Settings. You will be able to import the settings of the backed up GPO.

Demonstration Steps

User Filter Options to locate policies in Administrative Templates

1. Switch to HQDC01.
2. Run **Group Policy Management** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
3. In the console tree, expand **Forest: contoso.com, Domains**, and **contoso.com**, and then click the **Group Policy Objects** container.
4. In the details pane, right-click the **CONTOSO Standards** GPO, and then click **Edit**.

The Group Policy Management Editor appears.

5. In the console tree, expand **User Configuration** and **Policies**, and then click **Administrative Templates**.
6. Right-click **Administrative Templates**, and then click **Filter Options**.
7. Select the **Enable Keyword Filters** check box.
8. In the **Filter for word(s)** text box, type **screen saver**.
9. In the drop-down list next to the text box, select **Exact**, and click **OK**.

Administrative Templates policy settings are filtered to show only those that contain the words *screen saver*.

10. Spend a few moments examining the settings that you have found.
11. In the console tree, right-click **Administrative Templates** under **User Configuration**, and then click **Filter Options**.
12. Clear the **Enable Keyword Filters** check box.
13. In the **Configured** drop-down list, select **Yes**, and then click **OK**.

Administrative Template policy settings are filtered to show only those that have been configured (enabled or disabled).

14. Spend a few moments examining those settings.
15. In the console tree, right-click **Administrative Templates** under **User Configuration** and clear the **Filter On** option.

Add comments to a policy setting

1. In the console tree, expand **User Configuration, Policies, Administrative Templates**, and then click **Display**.
2. Double-click the **Screen Saver** policy setting.
3. Click the **Comment** tab.
4. Type **Corporate IT Security Policy implemented with this policy in combination with Password Protect the Screen Saver**, and click **OK**.
5. Double-click the **Password protect the screen saver** policy setting.
6. Click the **Comment** tab.
7. Type **Corporate IT Security Policy implemented with this policy in combination with Screen Saver Timeout**, and click **OK**.

Add comments to a GPO

1. In the console tree of the GPMC, right-click the root node, **CONTOSO Standards**, and then click **Properties**.
2. Click the **Comment** tab.
3. Type **Contoso corporate standard policies. Settings are scoped to all users and computers in the domain. Person responsible for this GPO: *your name*.**
This comment appears on the **Details** tab of the GPO in the GPMC.
4. Click **OK**.

Create a new GPO from a starter GPO

1. In the console tree of the GPMC, click the **Starter GPOs** container.
2. In the details pane, click the **Create Starter GPOs Folder** button.
3. In the console tree, right-click the **Starter GPOs** container, and then click **New**.
4. In **Name**: type **CONTOSO Starter GPO**, and then click **OK**.
5. In the details pane, right-click **CONTOSO Starter GPO**, and then click **Edit**.
The Group Policy Management Editor appears. Review and edit the settings as desired.
6. Close the Group Policy Management Editor.

7. In the details pane, right-click **CONTOSO Starter GPO**, and then click **New GPO From Starter GPO**.
8. In **Name**: type **CONTOSO Desktop**, and then click **OK**.

Create a new GPO by copying an existing GPO

1. In the GPMC console tree, expand the **Group Policy Objects** container, right-click the **CONTOSO Desktop** GPO, and then click **Copy**.
2. Right-click the **Group Policy Objects** container, click **Paste**, and then click **OK**.
3. Click **OK**.

Create a new GPO by importing settings that were exported from another GPO

1. In the GPMC console tree, expand the **Group Policy Objects** container, right-click the **CONTOSO Desktop** GPO, and then click **Back Up**.
2. In **Location**: type **D:\Labfiles\Lab06b**, and then click **Back Up**.
3. When the backup finishes, click **OK**.
4. In the GPMC console tree, right-click the **Group Policy Objects** container, and then click **New**.
5. In **Name**: type **CONTOSO Import**, and then click **OK**.
6. In the GPMC console tree, right-click the **CONTOSO Import** GPO, and then click **Import Settings**.

The Import Settings Wizard appears.

7. Click **Next** three times.
8. Select the **CONTOSO Desktop** GPO, and then click **Next** two times.
9. Click **Finish**, and then click **OK**.

Manage GPOs and Their Settings

- **Copy** (and *Paste* into a Group Policy Objects container)
 - Create a new "copy" GPO and modify it
 - Transfer a GPO to a trusted domain, such as test-to-production
- **Back Up** all settings, objects, links, permissions (access control lists [ACLs])
- **Restore** into same domain as backup
- **Import Settings** into a new GPO in same or any domain
 - Migration table for source-to-destination mapping of UNC paths and security group names
 - *Replaces all settings* in the GPO – not a "merge"
- **Save Report**
- **Delete**
- **Rename**

Key Points

When you right-click a GPO in the GPMC, you are presented with a menu of useful management commands:

- **Copy.** You can copy a GPO and then right-click the Group Policy Objects container and choose Paste to create a copy of the GPO. This is useful when you want to create a new GPO in the same domain and to start with the same settings as an existing GPO. It is also useful to copy a GPO into another domain, for example between a test domain and a production domain. To copy a GPO between domains, add the target trusted domain to the GPMC. You must have permission to create GPOs in the target domain. When you paste a GPO, you are given the option to copy the access control list (ACL) from the original GPO, which preserves the security filtering, or to use the default ACL for new GPOs in the target domain.

- **Back Up.** As with any critical data, it's important to back up GPOs. Because a GPO consists of several files, objects, permissions, and links, managing the backup and restore of GPOs could be quite difficult. Luckily, the Back Up command pulls all of those pieces into a single place and makes restore a breeze.
- **Restore from Backup.** Restore an entire GPO, including its files, objects, permissions, and links into the same domain in which the GPO originally existed.
- **Import Settings.** Import only the settings from a backed up GPO. This operation does not import permissions or links, it can be useful for transferring GPOs between non-trusted domains that cannot use copy and paste. If a GPO includes potentially domain specific settings, including the UNC paths or names of security groups, you will be prompted as to whether you want to import those settings exactly as they were backed up, or to use a migration table that maps source to destination names.
- **Save Report.** Use this to save an HTML report of the GPO settings.
- **Delete.**
- **Rename.**

Additional Reading

- GPO Operations:
<http://go.microsoft.com/fwlink/?LinkId=168655>
- Backing up, Restoring, Migrating, and Copying GPOs:
<http://go.microsoft.com/fwlink/?LinkId=168656>

Lab B: Manage Settings and GPOs

- Exercise 1: Use Filtering and Commenting
- Exercise 2: Manage Administrative Templates

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 30 minutes

Scenario

You were recently hired as the domain administrator for Contoso, Ltd., replacing the previous administrator, who retired. You are not certain what policy settings have been configured, so you decide to locate and document GPOs and policy settings. You also discover that the company has not leveraged either the functionality or the manageability of administrative templates.

Exercise 1: Use Filtering and Commenting

In this exercise, you will use the new commenting and filtering features of Group Policy to locate and document policy settings.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Search and filter policy settings.
3. Document GPOs and settings with comments.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.

► Task 2: Search and filter policy settings

1. In the **User Configuration\Policies\Administrative Templates** folder, filter the view to show only policy settings that contain the phrase **screen saver**. Spend a few moments examining those settings.
2. Filter the view to show only configured policy settings. Spend a few moments examining those settings.
3. Turn off the filter from **Administrative Templates**.

► **Task 3: Document GPOs and settings with comments**

1. Edit the comment to the **CONTOSO Standards** GPO and add the following comment to the GPO: **Contoso corporate standard policies. Settings are scoped to all users and computers in the domain. Person responsible for this GPO: *your name*.**

This comment appears on the Details tab of the GPO in the GPMC.

2. Add the following comment to the screen saver policy setting: **Corporate IT Security Policy implemented with this policy in combination with Password Protect the Screen Saver.**
3. Add the following comment to the **Password Protect the Screen Saver** policy setting: **Corporate IT Security Policy implemented with this policy in combination with Screen Saver Timeout.**

Results: After this exercise, you will have added comments to your Group Policy object and settings.

Exercise 2: Manage Administrative Templates

Administrative templates provide the instructions with which the GPME creates a user interface to configure Administrative Templates policy settings and specify the registry changes that must be made based on those policy settings. In this exercise, you will examine and manage administrative templates. You will also create a central store of administrative templates to centralize the management of templates.

The main tasks for this exercise are as follows:

1. Explore the syntax of an Administrative Template.
2. Manage classic administrative templates (.ADM files).
3. Manage .ADMX and .ADML files.
4. Create the central store.

► Task 1: Explore the syntax of an administrative template

1. On HQDC01, click **Start**, then click **Run**, then type **%SystemRoot%\PolicyDefinitions** and press ENTER. The **PolicyDefinitions** folder opens.
2. Open the **en-us** folder or the folder for your region and language.
3. Double-click **ControlPanelDisplay.adml**.
4. Choose the **Select a program from a list of installed programs** option and click **OK**.
5. Select **Notepad** and click **OK**.
6. Click the **Format** menu and select **Word wrap**.
7. Search for the text **ScreenSaverIsSecure**.
This is a definition of a string variable called ScreenSaverIsSecure.
8. Note the text between the **<string>** and **</string>** tags.
9. Note the name of the variable on the following line, **ScreenSaverIsSecure_Help**, and the text between the **<string>** and **</string>** tags.
10. Close the file.
11. Navigate up to the **PolicyDefinitions** folder.

12. Double-click **ControlPanelDisplay.admx**.
13. Choose the **Select a program from a list of installed programs** option and click **OK**.
14. Select **Notepad** and click **OK**.
15. Search for the text, **ScreenSaverIsSecure**.
16. Examine the code in the file, also shown below:

```
<policy name="ScreenSaverIsSecure" class="User"
displayName="$(string.ScreenSaverIsSecure)"
explainText="$(string.ScreenSaverIsSecure_Help)"
key="Software\Policies\Microsoft\Windows\Control Panel\Desktop"
valueName="ScreenSaverIsSecure">
  <parentCategory ref="Display" />
  <supportedOn ref="windows:SUPPORTED_Win2kSP1" />
  <enabledValue>
    <string>1</string>
  </enabledValue>
  <disabledValue>
    <string>0</string>
  </disabledValue>
</policy>
```

17. Identify the parts of the template that define the following:
 - The name of the policy setting that appears in the GPME
 - The explanatory text for the policy setting
 - The registry key and value affected by the policy setting
 - The data put into the registry if the policy is enabled
 - The data put into the registry if the policy is disabled
18. Close the file, and then close Windows Explorer.

► **Task 2: Manage classic administrative templates (.ADM files)**

1. Open the GPME and, in the **User Configuration\Policies\Administrative Templates** folder, add the **office12.adm** template from **D:\Labfiles\Lab06b\Office 2007 Administrative Templates**.

Classic administrative templates (.ADM files) are provided primarily for enterprises that do not manage Group Policy with Windows Vista or Windows Server 2008 or later operating systems.

You should use a computer running the most recent version of Windows to manage Group Policy. By doing so, you will be able to view and modify all available policy settings, including those that apply to previous versions of Windows. If you have at least one computer running Windows Vista, Windows Server 2008, or later, you should use that computer to manage Group Policy, and then you will not need classic administrative templates (.ADM files) when .ADMX/.ADML files are available.

Note that the template format affects only the *management* of Group Policy. Settings will apply to versions of Windows as described in the Supported on or Requirements section of the policy setting properties.

2. Examine the settings exposed by this administrative template.
3. Remove the template.

► **Task 3: Manage .ADMX and .ADML files**

- Copy all .ADMX files and the **en-us** subfolder (or the appropriate subfolder for your language and region) from **D:\Labfiles\Lab06b\Office 2007 Administrative Templates** to **%SystemRoot%\PolicyDefinitions**. When you paste the files, you will be prompted for administrative credentials. Use the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
- Close and then re-open the GPME for **CONTOSO Standards**. In the console tree, expand **User Configuration\Policies\Administrative Templates**. Note the addition of Microsoft® Office 2007 policy setting folders.

► Task 4: Create the central store

1. In the GPME, select the **Administrative Templates** node underneath **User Configuration\Policies**, and note the heading in the details pane reports: **Policy definitions (ADMX files) retrieved from the local machine**.
2. Close the GPME.
3. Copy all .ADMX files from %systemroot%\PolicyDefinitions to \\contoso.com\SYSVOL\contoso.com\Policies\PolicyDefinitions.
4. Copy all .ADML files from %systemroot%\PolicyDefinitions\en-us (or the appropriate folder for your language and region) to \\contoso.com\SYSVOL\contoso.com\Policies\PolicyDefinitions\en-us (or the appropriate folder for your language and region).
5. Edit the **CONTOSO Standards** GPO and, in the GPME, select the **Administrative Templates** node underneath **User Configuration\Policies**, and note the heading in the details pane reports: **Policy definitions (ADMX files) retrieved from the central store**.

Results: After this exercise, you will have created a central store of administrative templates and added the Microsoft Office 2007 templates.



Note: Do not shut down the virtual machines once you are finished with this lab as the settings you have configured here will be used in subsequent labs.

Lab Review Questions

Question: Describe the relationship between administrative template files (both .ADMX and .ADML files) and the GPME.

Question: When does an enterprise get a central store? What benefits does it provide?

Question: What are the advantages of managing Group Policy from a client running the latest version of Windows? Do settings you manage apply to previous versions of Windows?

Lesson 4

Manage Group Policy Scope

- GPO Links
- GPO Inheritance and Precedence
- Group Policy Processing Order
- Use Security Filtering to Modify GPO Scope
- WMI Filters
- Enable or Disable GPOs and GPO Nodes
- Target Preferences
- Loopback Policy Processing

A GPO is, by itself, just a collection of configuration instructions that will be processed by the CSEs of computers. Until the GPO is scoped, it does not apply to any users or computers. The GPO's scope determines which computers' CSEs will receive and process the GPO, and only the computers or users within the scope of a GPO will apply the settings in that GPO. In this lesson, you will learn to manage the scope of a GPO. Several mechanisms are used to scope a GPO:

- The GPO link to a site, domain, or OU and whether that link is enabled
- The Enforce option of a GPO
- The Block Inheritance option on an OU
- Security group filtering
- WMI filtering

- Policy node enabling or disabling
- Preferences targeting
- Loopback policy processing

You must be able to define the users or computers to which configuration is deployed, and therefore you must master the art of scoping GPOs. In this lesson, you will learn each of the mechanisms with which you can scope a GPO and, in the process, you will master the concepts of Group Policy application, inheritance, and precedence.

Objectives

After completing this lesson, you will be able to:

- Manage GPO links.
- Identify the relationship between OU structure and GPO application.
- Evaluate GPO inheritance and precedence.
- Understand the Block Inheritance and Enforced link options.
- Apply security filtering to narrow the scope of a GPO.
- Apply a WMI filter to a GPO.
- Target Group Policy preferences.
- Identify best practices for scoping Group Policy.

GPO Links

- GPO link
 - Causes policy settings in GPO to apply to *users* or *computers* within that container
 - Links GPO to site, domain, or OU (SDOU)
 - Must enable sites in the GPM console
 - GPO can be linked to multiple sites or OUs
 - Link can exist but be disabled
 - Link can be deleted, but GPO remains

Key Points

A GPO can be linked to one or more Active Directory sites, domains, or OUs. After a policy is linked to a site, domain, or OU, the users or computers and users in that container are within the scope of the GPO, including computers and users in child OUs.

As you learned in Lesson 1, you can link a GPO to the domain or to an OU.

To link a GPO, right-click the domain or OU in the GPMC console tree, and then click Link An Existing GPO. If you have not yet created a GPO, click Create A GPO In This {Domain | OU | Site} And Link It Here.

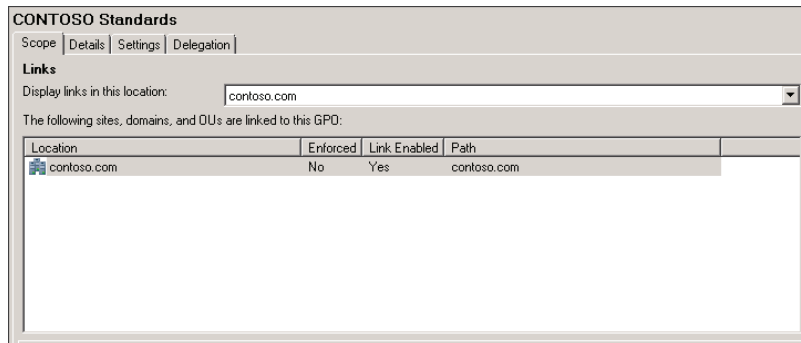
You can choose the same commands to link a GPO to a site, but by default, your Active Directory sites are not visible in the GPMC.

To show sites in the GPMC, right-click Sites in the GPMC console tree and choose Show Sites.



Note: Site-Linked GPOs and Domain Controller Placement. A GPO linked to a site affects all computers in the site without regard to the domain to which the computers belong (as long as all computers belong to the same Active Directory forest). Therefore, when you link a GPO to a site, that GPO can be applied to multiple domains within a forest. Site-linked GPOs are stored on domain controllers in the domain in which the GPO was created. Therefore, domain controllers for that domain must be accessible for site-linked GPOs to be applied correctly. If you implement site-linked policies, you must consider policy application when planning your network infrastructure. Either place a domain controller from the GPO's domain in the site to which the policy is linked, or ensure that Wide Area Network (WAN) connectivity provides accessibility to a domain controller in the GPO's domain.

When you link a GPO to a site, domain, or OU, you define the initial scope of the GPO. Select a GPO and click the Scope tab to identify the containers to which the GPO is linked. In the details pane of the GPMC, the GPO links are displayed in the first section of the Scope tab, as seen here:



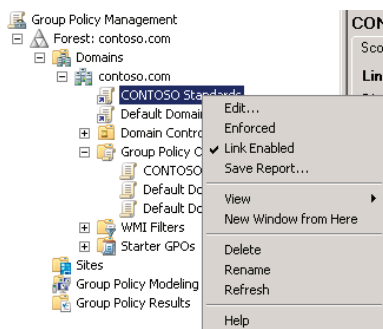
The impact of the GPO's links is that the Group Policy Client will download the GPO if either the computer or the user objects fall within the scope of the link. The GPO will be downloaded only if it is new or updated. The Group Policy Client caches the GPO to make policy refresh more efficient.

Link a GPO to Multiple OUs

You can link a GPO to more than one site or OU. It is common, for example, to apply configuration to computers in several OUs. You can define the configuration in a single GPO and link that GPO to each OU. If you later change settings in the GPO, your changes will apply to all OUs to which the GPO is linked.

Delete or Disable a GPO Link

After you have linked a GPO, the GPO link appears in the GPMC underneath the site, domain, or OU. The icon for the GPO link has a small shortcut arrow. When you right-click the GPO link, a context menu appears, as shown here:



To delete a GPO link, right-click the GPO link in the GPMC console tree and then click Delete.

Deleting a GPO link does not delete the GPO itself, which remains in that GPO container. Deleting the link does change the scope of the GPO so that it no longer applies to computers and users within a site, domain, or OU to which it was previously linked.

You can also modify a GPO link by disabling it.

To disable a GPO link, right-click the GPO link in the GPMC console tree and then deselect the Link Enabled option.

Disabling the link also changes the scope of the GPO so that it no longer applies to computers and users within that container. However, the link remains so that it can be easily re-enabled.

GPO Inheritance and Precedence

- The application of GPOs linked to each container results in a cumulative effect called *inheritance*
 - Default Precedence: Local → Site → Domain → OU → OU... (LSDOU)
 - Seen on the Group Policy Inheritance tab
- Link order (attribute of GPO Link)
 - Lower number → Higher on list → Precedent
- Block Inheritance (attribute of OU)
 - Blocks the processing of GPOs from above
- Enforced (attribute of GPO Link)
 - Enforced GPOs “blast through” Block Inheritance
 - Enforced GPO settings win over conflicting settings in lower GPOs

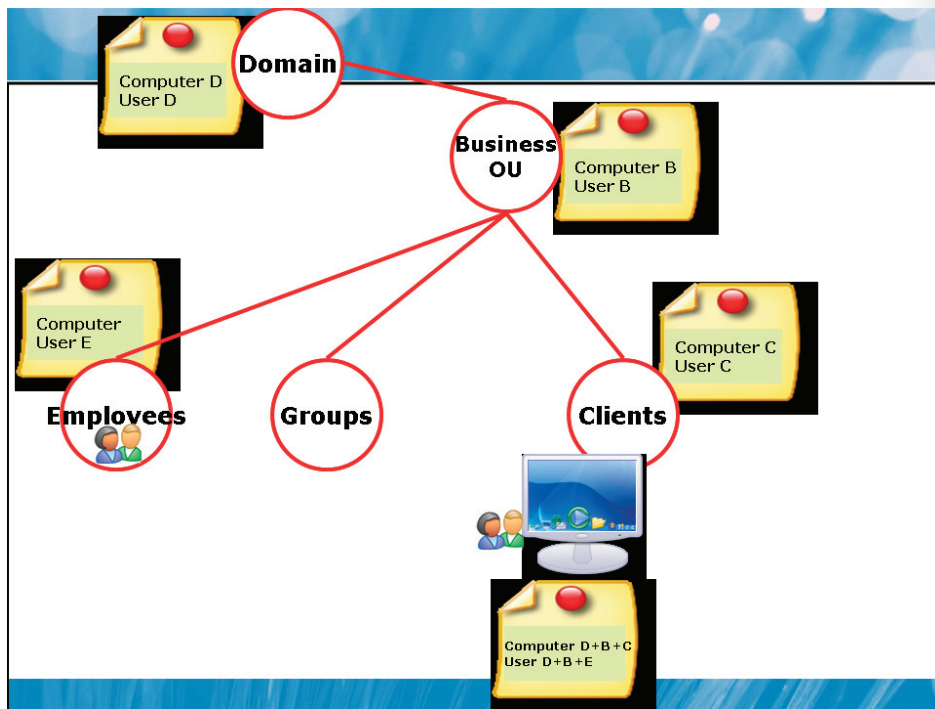
Key Points

A policy setting can be configured in more than one GPO, and GPOs can be in conflict with one another. For example, a policy setting can be enabled in one GPO, disabled in another GPO, and not configured in a third GPO. In this case, the precedence of the GPOs determines which policy setting the client applies. A GPO with higher precedence will prevail over a GPO with lower precedence. Precedence is shown as a number in the GPMC. The smaller the number—that is, the closer to 1—the higher the precedence, so a GPO with a precedence of 1 will prevail over other GPOs. Select the domain or OU and then click the Group Policy Inheritance tab to view the precedence of each GPO.

When a policy setting is enabled or disabled in a GPO with higher precedence, the configured setting takes effect. However, remember that policy settings are set to Not Configured by default. If a policy setting is not configured in a GPO with higher precedence, the policy setting (either enabled or disabled) in a GPO with lower precedence will take effect.

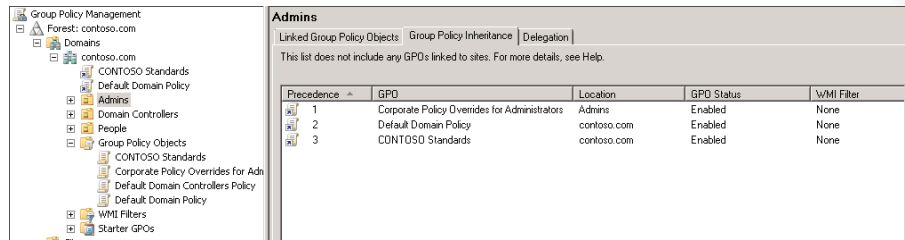
A site, domain, or OU can have more than one GPO linked to it. The link order of GPOs determines the precedence of GPOs in such a scenario. GPOs with higher-link order take precedence over GPOs with lower-link order. When you select an OU in the GPMC, the Linked Group Policy Objects tab shows the link order of GPOs linked to that OU.

The default behavior of Group Policy is that GPOs linked to a higher-level container are inherited by lower-level containers. When a computer starts up or a user logs on, the Group Policy Client examines the location of the computer or user object in Active Directory and evaluates the GPOs with scopes that include the computer or user. Then the client-side extensions apply policy settings from these GPOs. Policies are applied sequentially, beginning with the policies linked to the site, followed by those linked to the domain, followed by those linked to OUs—from the top-level OU down to the OU in which the user or computer object exists. It is a layered application of settings, so a GPO that is applied later in the process, because it has higher precedence, will override settings applied earlier in the process. This default order of applying GPOs is illustrated below:



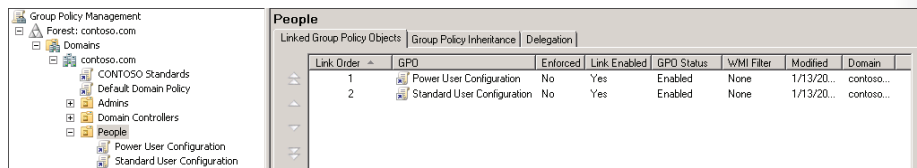
This sequential application of GPOs creates an effect called *policy inheritance*. Policies are inherited, so the resultant set of group policies for a user or computer will be the cumulative effect of site, domain, and OU policies.

By default, inherited GPOs have lower precedence than GPOs linked directly to the container. For example, you might configure a policy setting to disable the use of registry-editing tools for all users in the domain by configuring the policy setting in a GPO linked to the domain. That GPO, and its policy setting, will be inherited by all users within the domain. However, you probably want administrators to be able to use registry-editing tools, so you will link a GPO to the OU that contains administrators' accounts and configure the policy setting to allow the use of registry-editing tools. Because the GPO linked to the administrators' OU takes higher precedence than the inherited GPO, administrators will be able to use registry-editing tools. The figure below shows Group Policy Inheritance:



Precedence of Multiple Linked GPOs

An OU, domain, or site can have more than one GPO linked to it. In the event of multiple GPOs, the objects' link order determines their precedence. In the figure below, two GPOs are linked to the People OU:



The object higher on the list, with a link order of 1, has the highest precedence. Therefore, settings that are enabled or disabled in the Power User Configuration GPO will have precedence over these same settings in the Standard User Configuration GPO.

To change the precedence of a GPO link:

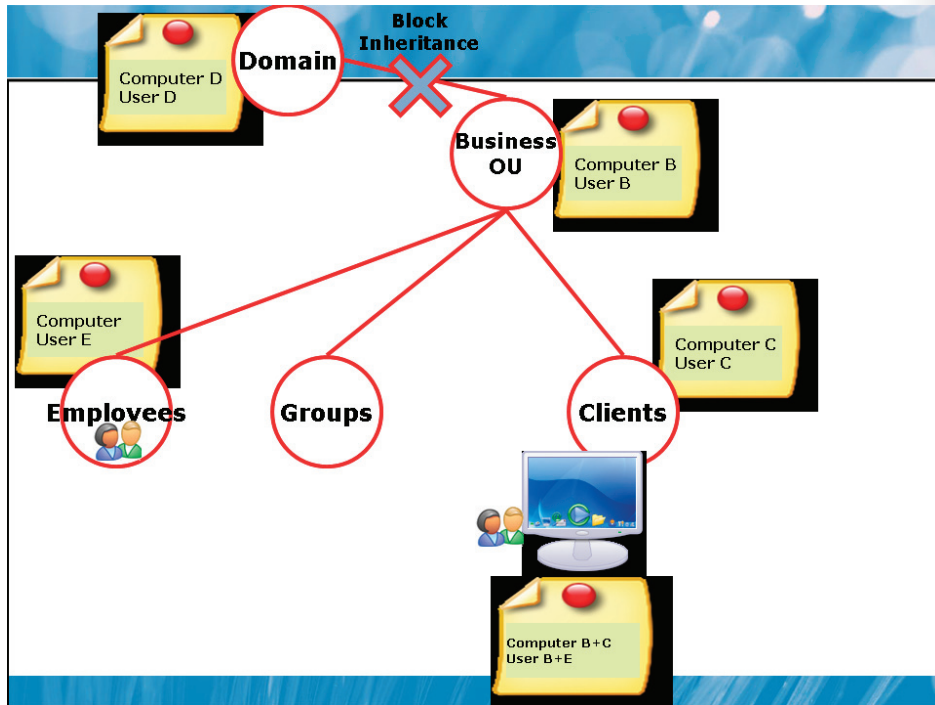
1. Select the OU, site, or domain in the GPMC console tree.
2. Click the **Linked Group Policy Objects** tab in the details pane.
3. Select the GPO.
4. Use the **Up**, **Down**, **Move To Top**, and **Move To Bottom** arrows to change the link order of the selected GPO.

Block Inheritance

A domain or OU can be configured to prevent the inheritance of policy settings.

To block inheritance, right-click the domain or OU in the GPMC console tree and choose Block Inheritance.

The Block Inheritance option is a property of a domain or OU, so it blocks all Group Policy settings from GPOs linked to parents in the Group Policy hierarchy. When you block inheritance on an OU, for example, GPO application begins with any GPOs linked directly to that OU—GPOs linked to higher-level OUs, the domain, or the site will not apply.



The Block Inheritance option should be used sparingly, if ever. Blocking inheritance makes it more difficult to evaluate Group Policy precedence and inheritance. In a later topic, you will learn how to scope a GPO so that it applies to only a subset of objects or so that it is prevented from applying to a subset of objects. With security group filtering, you can carefully scope a GPO so that it applies to only the correct users and computers in the first place, making it unnecessary to use the Block Inheritance option.

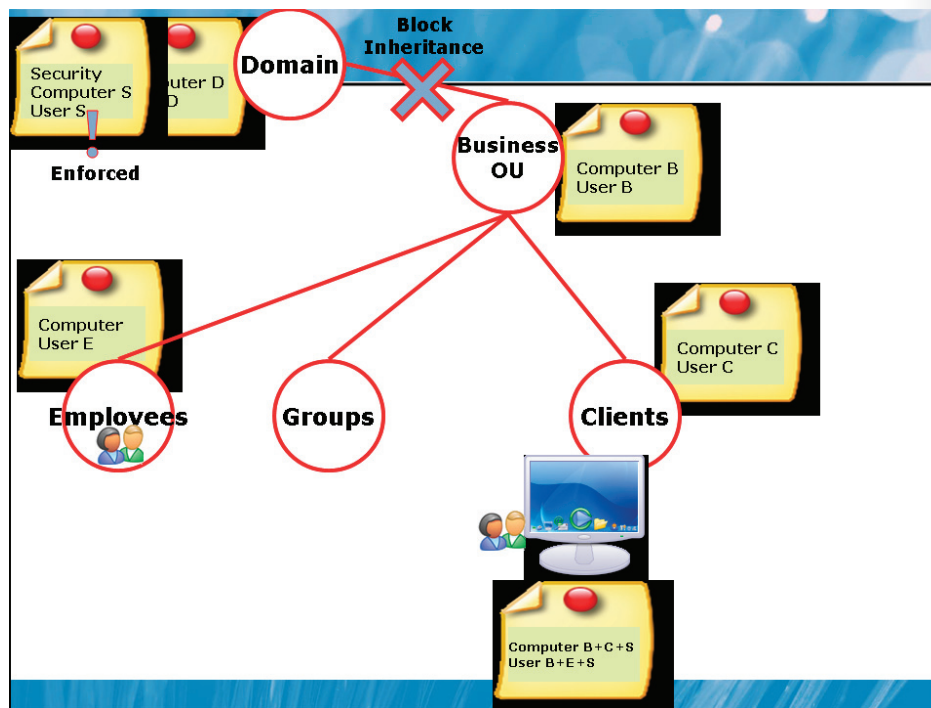
Enforce a GPO Link

In addition, a GPO link can be set to Enforced.

To enforce a GPO link, right-click the GPO link in the console tree and choose Enforced from the context menu.

When a GPO link is set to Enforced, the GPO takes the highest level of precedence; policy settings in that GPO will prevail over any conflicting policy settings in other GPOs. In addition, a link that is enforced will apply to child containers even when those containers are set to Block Inheritance. The Enforced option causes the policy to apply to all objects within its scope. Enforced will cause policies to override any conflicting policies and will apply regardless of whether a Block Inheritance option is set.

In the figure on the following page, Block Inheritance has been applied to the Business OU. As a result, GPO D, which is applied to the domain, is blocked and does not apply when a user from the Employees OU logs on to a computer in the Clients OU. However, the Security GPO, GPO S, linked to the domain with the Enforced option, does apply. In fact, it is applied last in the processing order, meaning that its settings will override those of GPOs B, C, and E.



When you configure a GPO that defines configuration mandated by your corporate IT security and usage policies, you want to ensure that those settings are not overridden by other GPOs. You can do this by enforcing the link of the GPO. The figure here shows just this scenario:

Group Policy Management

Forest: contoso.com

Domains

contoso.com

CONTOSO Corporate IT Security & Usage

CONTOSO Standards

Default Domain Policy

Admins

Domain Controllers

People

Power User Configuration

Standard User Configuration

People

Linked Group Policy Objects | Group Policy Inheritance | Delegation

This list does not include any GPOs linked to sites. For more details, see Help.

Precedence	GPO	Location	GPO Status	WMI Filter
1 (Enforced)	CONTOSO Corporate IT Security & Usage ...	contoso.com	Enabled	None
2	Power User Configuration	People	Enabled	None
3	Standard User Configuration	People	Enabled	None
4	Default Domain Policy	contoso.com	Enabled	None
5	CONTOSO Standards	contoso.com	Enabled	None

Configuration mandated by corporate policies is deployed in the CONTOSO Corporate IT Security & Usage GPO, which is linked with an enforced link to the Contoso.com domain. The icon for the GPO link has a padlock on it—the visual indicator of an enforced link. On the People OU, the Group Policy Inheritance tab shows that the GPO takes precedence even over the GPOs linked to the People OU itself.

Evaluating Precedence

To facilitate evaluation of GPO precedence, you can simply select an OU (or domain) and click the Group Policy Inheritance tab. This tab will display the resulting precedence of GPOs, accounting for GPO link, link order, inheritance blocking, and link enforcement. This tab does not account for policies that are linked to a site, nor does it account for GPO security or WMI filtering.

Exam Tips

- Be certain to memorize the default domain policy processing order: site, domain, OU. And remember that domain policy settings are applied after—and therefore take precedence over—settings in local GPOs.
- Although it is recommended to use the Block Inheritance and Enforced options sparingly in your Group Policy infrastructure, the 70-640 exam will expect you to understand the effect of both options.

Use Security Filtering to Modify GPO Scope

- **Apply Group Policy permission**
 - GPO has an ACL (Delegation tab → Advanced)
 - Default: Authenticated Users have Allow Apply Group Policy
- **Scope *only* to users in selected global group(s)**
 - Remove Authenticated Users
 - Add appropriate *global* groups
 - Must be *global* groups (GPOs don't scope to domain local)
- **Scope to users *except for* those in selected group(s)**
 - On Delegation tab, click Advanced
 - Add appropriate *global* groups
 - *Deny* Apply Group Policy permission
 - Does not appear on Delegation tab or in filtering section ☹

Key Points

By now, you've learned that you can link a GPO to a site, domain, or OU. However, you might need to apply GPOs only to certain groups of users or computers rather than to all users or computers within the scope of the GPO. Although you cannot directly link a GPO to a security group, there is a way to apply GPOs to specific security groups. The policies in a GPO apply only to users who have Allow Read and Allow Apply Group Policy permissions to the GPO.

Each GPO has an access control list (ACL) that defines permissions to the GPO. Two permissions, Allow Read and Allow Apply Group Policy, are required for a GPO to apply to a user or computer. If a GPO is scoped to a computer, for example, by its link to the computer's OU, but the computer does not have Read and Apply Group Policy permissions, it will not download and apply the GPO. Therefore, by setting the appropriate permissions for security groups, you can filter a GPO so that its settings apply only to the computers and users you specify.

By default, Authenticated Users are given the Allow Apply Group Policy permission on each new GPO. This means that by default, all users and computers are affected by the GPOs set for their domain, site, or OU regardless of the other groups in which they might be members. Therefore, there are two ways of filtering GPO scope:

- Remove the Apply Group Policy permission (currently set to Allow) for the Authenticated Users group but do not set this permission to Deny. Then determine the groups to which the GPO should be applied and set the Read and Apply Group Policy permissions for these groups to Allow.
- Determine the groups to which the GPO should not be applied and set the Apply Group Policy permission for these groups to Deny. If you deny the Apply Group Policy permission to a GPO, the user or computer will not apply settings in the GPO, even if the user or computer is a member of another group that is allowed the Apply Group Policy Permission.

Filtering a GPO to Apply to Specific Groups

To apply a GPO to a specific security group:

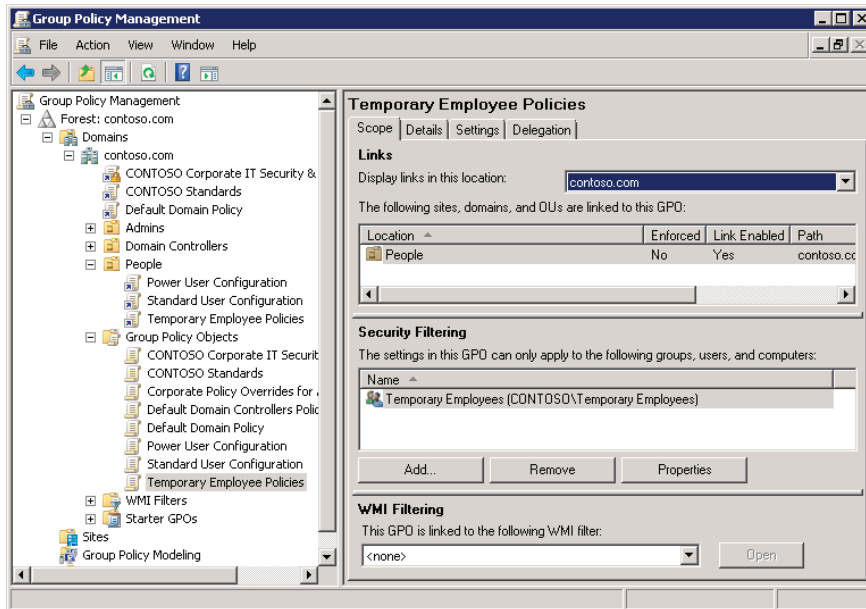
1. Select the GPO in the **Group Policy Objects** container in the console tree.
2. In the **Security Filtering** section, select the **Authenticated Users** group and click **Remove**.



Note: Use global security groups to filter GPOs. GPOs can be filtered only with global security groups—not with domain local security groups.

3. Click **OK** to confirm the change.
4. Click **Add**.
5. Select the group to which you want the policy to apply and click **OK**.

The result will look similar to the figure shown here—the Authenticated Users group is not listed, and the specific group to which the policy should apply is listed.



Filtering a GPO to Exclude Specific Groups

Unfortunately, the Scope tab of a GPO does not allow you to exclude specific groups. To exclude a group—that is, to deny the Apply Group Policy permission—you must use the Delegation tab.

To deny a group the Apply Group Policy permission:

1. Select the GPO in the **Group Policy Objects** container in the console tree.
2. Click the **Delegation** tab.
3. Click the **Advanced** button.

The Security Settings dialog box appears.

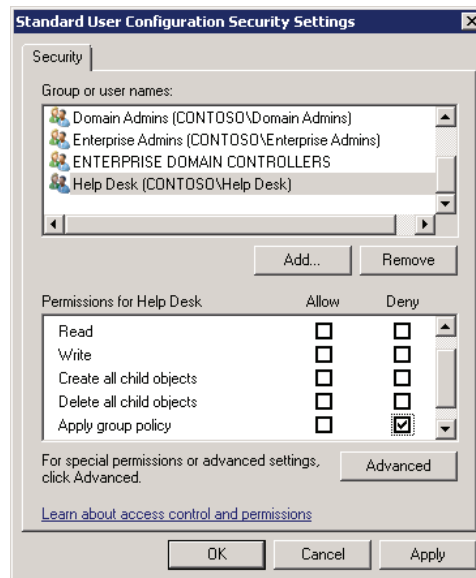
4. Click the **Add** button.
5. Select the group you want to exclude from the GPO. Remember, it must be a global group. GPO scope cannot be filtered by domain local groups.

6. Click **OK**.

The group you selected is given the Allow Read permission by default.

7. Deselect the **Allow Read permission** check box.
8. Select the **Deny Apply Group Policy** check box.

The figure here shows an example that denies the Help Desk group the Apply group policy permission and, therefore, excludes the group from the scope of the GPO.



9. Click **OK**.

You are warned that Deny permissions override other permissions.

Because Deny permissions override Allow permissions, it is recommended that you use Deny permissions sparingly. Microsoft Windows reminds you of this best practice with the warning message, and by the far more laborious process required to exclude groups with the Deny Apply Group Policy permission than to include groups in the Security Filtering section of the Scope tab.

10. Confirm that you want to continue.



Note: Important! Deny permissions are not exposed on the Scope tab.

Unfortunately, when you exclude a group, the exclusion is not shown in the Security Filtering section of the Scope tab. This is yet one more reason to use Deny permissions sparingly.

WMI Filters

- Windows Management Instrumentation (WMI)
- WMI Query Language (WQL)
 - Similar to T-SQL
 - `Select * FROM Win32_OperatingSystem WHERE Caption="Microsoft Windows XP Professional" AND CSDVersion="Service Pack 3"`
- Create a WMI filter
- Use the filter for one or more GPOs

Key Points

Windows Management Instrumentation (WMI) is a management infrastructure technology that enables administrators to monitor and control managed objects in the network. A WMI query is capable of filtering systems based on characteristics, including RAM, processor speed, disk capacity, IP address, operating system version and service pack level, installed applications, and printer properties. Because WMI exposes almost every property of every object within a computer, the list of attributes that can be used in a WMI query is virtually unlimited. WMI queries are written using WMI Query Language (WQL).

You can use a WMI query to create a WMI filter, with which a GPO can be filtered. A good way to understand the purpose of a WMI filter, both for the certification exams and for real-world implementation, is through examples. Group Policy can be used to deploy software applications and service packs—a capability that is discussed in Module 7. You might create a GPO to deploy an application and then use a WMI filter to specify that the policy should apply only to computers with a certain operating system and service pack—Windows XP SP3, for example. The WMI query to identify such systems is:

```
Select * FROM Win32_OperatingSystem WHERE Caption="Microsoft  
Windows XP Professional" AND CSDVersion="Service Pack 3"
```

When the Group Policy Client evaluates GPOs it has downloaded to determine which should be handed off to the CSEs for processing, it performs the query against the local system. If the system meets the criteria of the query, the query result is a logical True, and the CSEs will process the GPO.

WMI exposes namespaces, within which are classes that can be queried. Many useful classes, including Win32_Operating System, are found in a class called root\CIMv2.

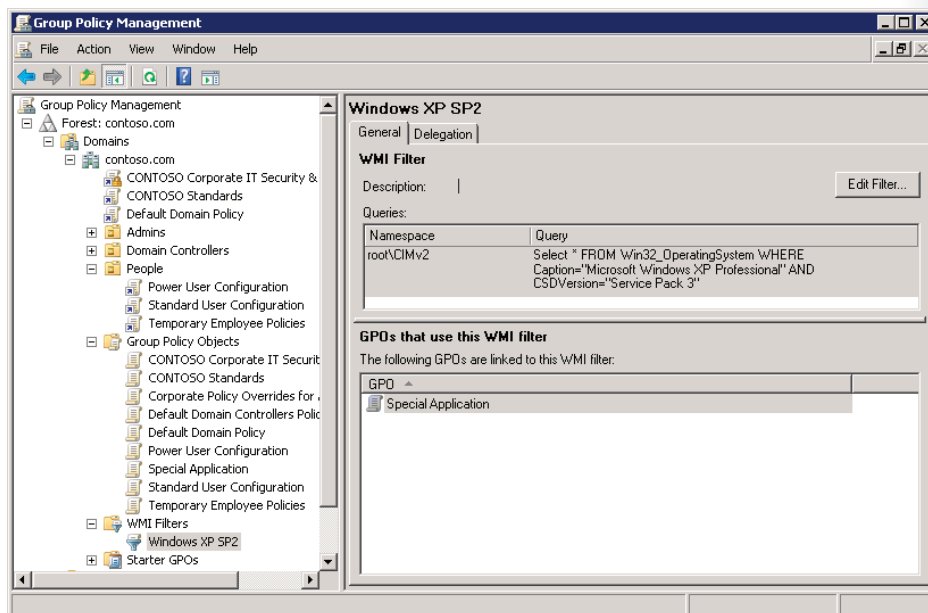
To create a WMI filter:

1. Right-click the **WMI Filters** node in the GPMC console tree and choose **New**.
Type a name and description for the filter, and then click the **Add** button.
2. In the **Namespace** box, type the namespace for your query.
3. In the **Query** box, enter the query.
4. Click **OK**.

To filter a GPO with a WMI filter:

1. Select the GPO or GPO link in the console tree.
2. Click the **Scope** tab.
3. Click the WMI drop-down list, and select the WMI filter.

A GPO can be filtered by only one WMI filter, but that WMI filter can be a complex query, using multiple criteria. A single WMI filter can be linked to, and thereby used to filter, one or more GPOs. The General tab of a WMI filter, shown in the figure here, displays the GPOs that use the WMI filter:



There are three significant caveats regarding WMI filters. First, the WQL syntax of WMI queries can be challenging to master. You can often find examples on the Internet when you search using the keywords *WMI filter* and *WMI query* along with a description of the query you want to create.

You can find examples of WMI filters at <http://technet2.microsoft.com/windowsserver/en/library/a16cfa4-83b3-430b-b826-9bf81c0d39a71033.mspx?mfr=true>.

You can also refer to the Windows Management Instrumentation (WMI) software development kit (SDK), located at <http://msdn2.microsoft.com/en-us/library/aa394582.aspx>.

Second, WMI filters are expensive in terms of Group Policy processing performance. Because the Group Policy Client must perform the WMI query at each policy processing interval, there is a slight impact on system performance every 90–120 minutes. With the performance of today's computers, the impact might not be noticeable, but you should certainly test the effects of a WMI filter prior to deploying it widely in your production environment.

Note that the WMI query is only processed *one* time, even if it is used to filter the scope of multiple GPOs.

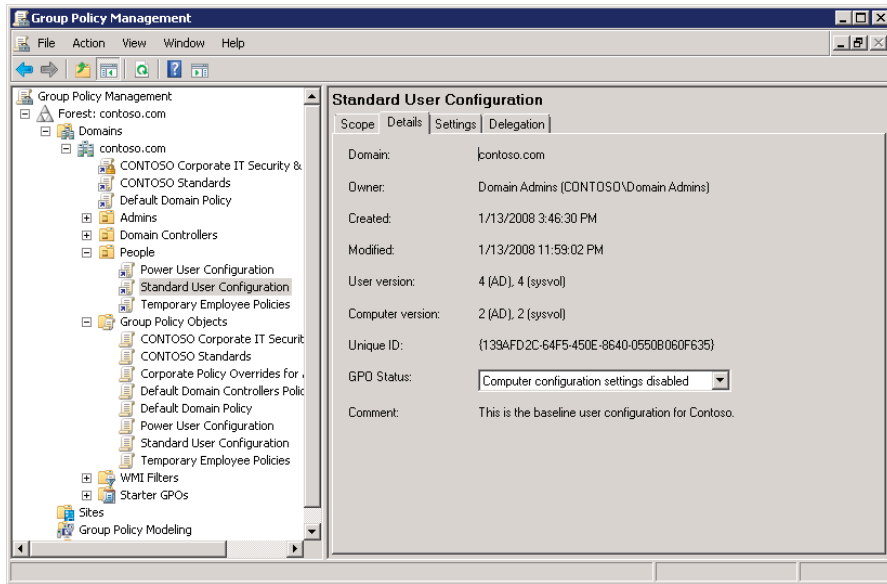
Third, WMI filters are not processed by computers running Windows 2000. If a GPO is filtered with a WMI filter, a Windows 2000 system ignores the filter and processes the GPO as if the results of the filter were True.

Enable or Disable GPOs and GPO Nodes

- GPO Details tab → GPO Status drop-down list
- Enabled: Both Computer Configuration and User Configuration settings will be applied by CSEs
- All settings disabled: CSEs will not process the GPO
- Computer Configuration settings disabled: CSEs will not process settings in Computer Configuration
- User Configuration settings disabled: CSEs will not process settings in User Configuration

Key Points

You can prevent the settings in the Computer Configuration or User Configuration nodes from being processed during policy refresh by changing GPO Status.



To enable or disable a GPO's nodes, select the GPO or GPO link in the console tree, then click the Details tab, shown in the figure above, and then choose one of the following from the GPO Status drop-down list :

- **Enabled.** Both computer configuration settings and user configuration settings will be processed by CSEs during policy refresh.
- **All Settings Disabled.** CSEs will not process the GPO during policy refresh.
- **Computer Configuration Settings Disabled.** During computer policy refresh, computer configuration settings in the GPO will not be applied.
- **User Configuration Settings Disabled.** During user policy refresh, user configuration settings in the GPO will not be applied.

You can configure GPO status to optimize policy processing. If a GPO contains only user settings, for example, setting the GPO Status option to disable computer settings will prevent the Group Policy client from attempting to process the GPO during computer policy refresh. Because the GPO contains no computer settings, there is no need to process the GPO, and you can save a few cycles of the processor.



Note: Use disabled GPOs for disaster preparedness. You can define a configuration that should take effect in case of an emergency, security incident, or other disasters in a GPO and link the GPO so that it is scoped to appropriate users and computers. Then disable the GPO. In the event that you require the configuration to be deployed, simply enable the GPO.

Target Preferences

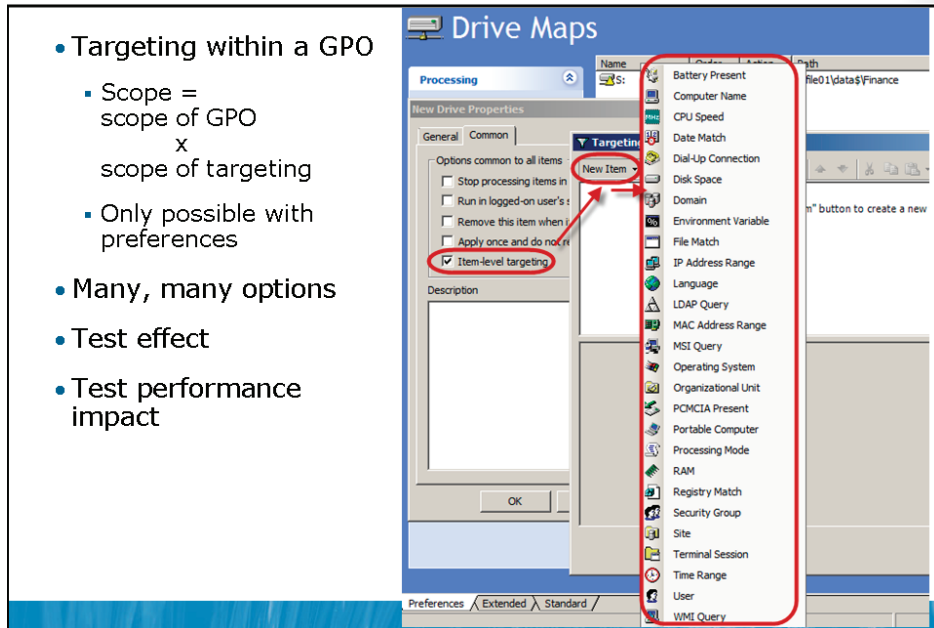
- Targeting within a GPO

- Scope =
scope of GPO
x
scope of targeting
- Only possible with preferences

- Many, many options

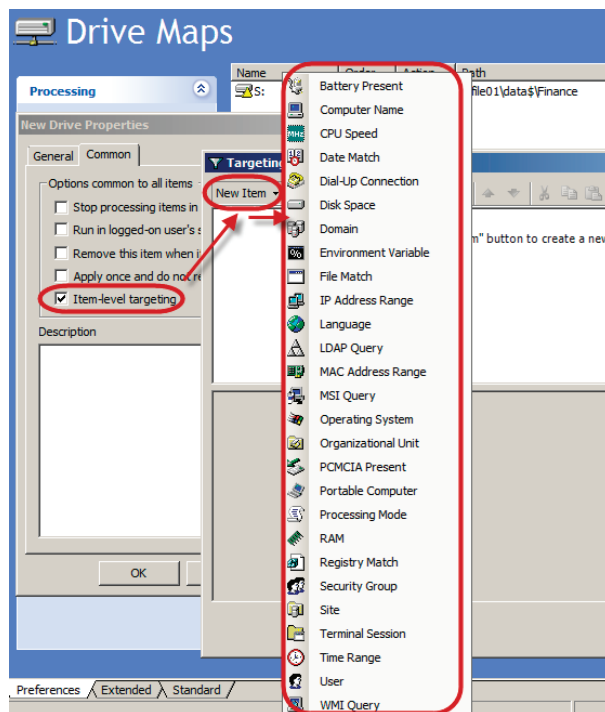
- Test effect

- Test performance impact



Key Points

Preferences, which are new to Windows Server 2008, have a built-in scoping mechanism called *item-level targeting*. You can have multiple preference items in a single GPO, and each preference item can be targeted or filtered. So, for example, you could have a single GPO with a preference that specifies folder options for engineers and another item that specifies folder options for sales people. You can target the items by using a security group or OU. There are over a dozen other criteria that can be used, including hardware and network characteristics, date and time, Lightweight Directory Access Protocol (LDAP) queries, and more.



Note: Preferences can target within a GPO. What's new about preferences is that you can target multiple preferences items within a single GPO instead of requiring multiple GPOs. With traditional policies, you often need multiple GPOs filtered to individual groups to apply variations of settings.

Like WMI filters, item-level targeting of preferences requires the CSE to perform a query to determine whether to apply the settings in a preferences item. You must be aware of the potential performance impact of item-level targeting, particularly if you use options such as LDAP queries, which require processing time and a response from a domain controller to process. As you design your Group Policy infrastructure, balance the configuration management benefits of item-level targeting against the performance impact you discover during testing in a lab.

Loopback Policy Processing

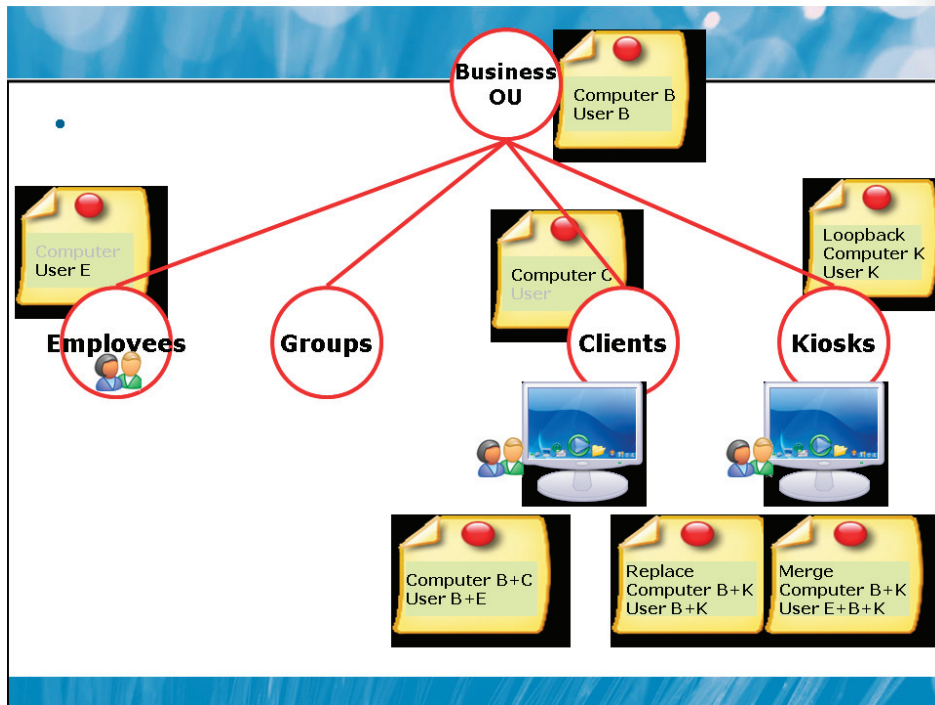
- At user login, user settings from GPOs scoped to *computer* object are applied
 - Create a consistent user experience on a computer
 - Conference rooms, kiosks, computer labs, VDI, RDS/TS, etc.
- Computer Configuration\Policies\Administrative Templates\System\Group Policy
 - User Group Policy loopback processing mode
- Replace mode
 - The user gets *none* of the User settings that are scoped to the user... *only* the User settings that are scoped to computer.
- Merge mode
 - The user gets the User settings scoped to the user, but those settings are overlaid with User settings scoped to the computer. The computer wins.

Key Points

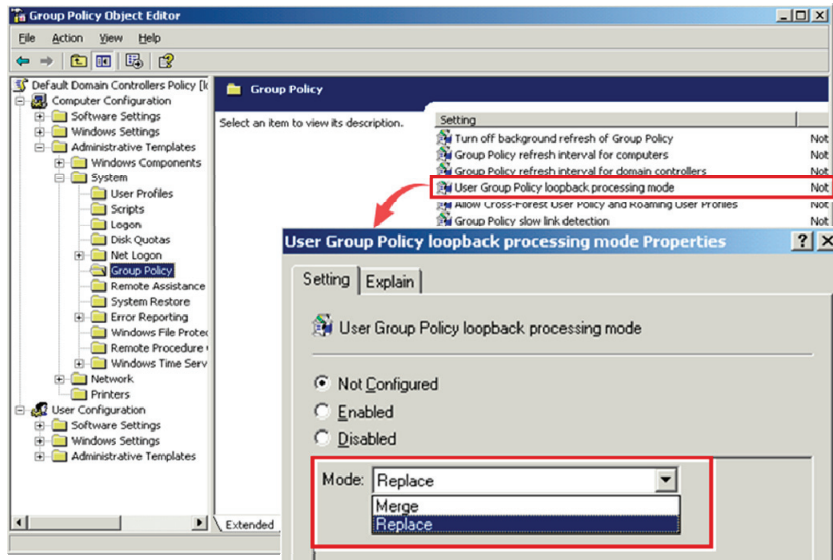
By default, a user's settings come from GPOs scoped to the user object in Active Directory. Regardless of which computer the user logs on to, the resultant set of policies that determine the user's environment will be the same. There are situations, however, in which you might want to configure a user differently, depending on the computer in use. For example, you might want to lock down and standardize user desktops when users log on to computers in closely managed environments such as conference rooms, reception areas, laboratories, classrooms, and kiosks. It is also important for virtual desktop infrastructure (VDI) scenarios, including remote virtual machines and Remote Desktop Services (Terminal Services).

Imagine a scenario in which you want to enforce a standard corporate appearance for the Windows desktop on all computers in conference rooms and other public areas of your office. How could you centrally manage this configuration, using Group Policy? Policy settings that configure desktop appearance are located in the User Configuration node of a GPO. Therefore, by default, the settings apply to users regardless of which computer they log on to. The default policy processing does not give you a way to scope user settings to apply to computers, regardless of which user logs on. That's where loopback policy processing comes in.

Loopback policy processing alters the default algorithm used by the Group Policy client to obtain the ordered list of GPOs that should be applied to a user's configuration. Instead of user configuration being determined by the User Configuration node of GPOs that are scoped to the user object, user configuration can be determined by the User Configuration node policies of GPOs that are scoped to the computer object.



The User Group Policy loopback processing mode policy, located in the Computer Configuration\Policies\Administrative Templates\System\Group Policy folder in GPME, can be, like all policy settings, set to Not Configured, Enabled, or Disabled.



When enabled, the policy can specify Replace or Merge mode.

- Replace.** In this case, the GPO list for the user (obtained in step 5 in the “Group Policy Processing”, the next section) is replaced in its entirety by the GPO list already obtained for the computer at computer startup (during step 2). The settings in the User Configuration policies of the computer’s GPOs are applied to the user. Replace mode is useful in a situation such as a classroom, where users should receive a standard configuration rather than the configuration applied to those users in a less managed environment.
- Merge.** In this case, the GPO list obtained for the computer at computer startup (step 2 in the “Group Policy Processing” section) is appended to the GPO list obtained for the user when logging on (step 5). Because the GPO list obtained for the computer is applied later, settings in GPOs on the computer’s list have precedence if they conflict with settings in the user’s list. This mode would be useful to apply additional settings to users’ typical configurations. For example, you might allow a user to receive his or her typical configuration when logging on to a computer in a conference room or reception area but replace the wallpaper with a standard bitmap and disable the use of certain applications or devices.



Note: It is an underdocumented fact that when you combine the loopback processing with security group filtering, the application of user settings during policy refresh uses the credentials of the computer to determine which GPOs to apply as part of the loopback processing, but the logged-on user must also have the Apply Group Policy permission for the GPO to be successfully applied.

Lab C: Manage Group Policy Scope

- Exercise 1: Configure GPO Scope with Links
- Exercise 2: Configure GPO Scope with Filtering
- Exercise 3: Configure Loopback Processing

Logon information

Virtual machine	6425B-HQDC01-A	6425B-DESKTOP101-A
Logon user name	Pat.Coleman	Do not Logon
Administrative user name	Pat.Coleman_Admin	
Password	Pa\$\$w0rd	

Estimated time: 30 minutes

Scenario

You are an administrator of the contoso.com domain. The CONTOSO Standards GPO, linked to the domain, configures a policy setting that requires a ten-minute screen saver timeout. An engineer reports that a critical application that performs lengthy calculations crashes when the screens saver starts, and the engineer has asked you to prevent the setting from applying to the team of engineers that uses the application every day. You have also been asked to configure conference room computers to use a 45-minute timeout, so that the screen saver does not launch during a meeting.

Exercise 1: Configure GPO Scope with Links

In this exercise, you will modify the scope of GPOs using GPO links, and you will explore inheritance, precedence, and the effects of Enforced links and Block Inheritance.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Create a GPO with a policy setting that takes precedence over a conflicting setting.
3. View the effect of an Enforced GPO link.
4. Apply Block Inheritance.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Start 6425B-DESKTOP101-A but do not log on to the system.

► Task 2: Create a GPO with a policy setting that takes precedence over a conflicting setting

1. Run **Active Directory Users and Computers** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **User Accounts\Employees** OU, create a sub-OU called **Engineers**, and then close Active Directory Users and Computers.
3. Run the Group Policy Management Console as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
4. Create a new GPO linked to the **Engineers** OU called **Engineering Application Override**.
5. Configure the **Screen saver** timeout policy setting to be disabled, and then close the GPME.

6. Select the **Engineers** OU, and then click the **Group Policy Inheritance** tab. Notice that the **Engineering Application Override** GPO has precedence over the **CONTOSO Standards** GPO.
7. The screen saver timeout policy setting you just configured in the **Engineering Application Override** GPO will be applied after the setting in the **CONTOSO Standards** GPO. Therefore, the new setting will overwrite the standards setting, and will "win." Screen saver timeout will be disabled for users within the scope of the **Engineering Application Override** GPO.

► **Task 3: View the effect of an Enforced GPO link**

1. In the GPMC console tree, select the **Domain Controllers** OU, and then click the **Group Policy Inheritance** tab.
2. Notice that the GPO named **6425B** has the highest precedence. Settings in this GPO will override any conflicting settings in any of the other GPOs.

The Default Domain Controllers GPO specifies, among other things, which groups are given the right to log on locally to domain controllers. To enhance the security of domain controllers, standard users are not given the right to log on locally. In order to allow a nonprivileged user account such as Pat.Coleman to log on to domain controllers in this course, the 6425B GPO gives Domain Users the right to log on locally to a computer. The 6425B GPO is linked to the domain, so its settings would normally be overridden by settings in the Default Domain Controllers GPO. Therefore, the 6425B GPO link to the domain is configured as Enforced. In this way, the conflict in user rights assignment between the two GPOs is "won" by the 6425B GPO.

► **Task 4: Apply Block Inheritance**

1. In the GPMC console, select the **Engineers** OU, and examine the precedence and inheritance of GPOs on the **Group Policy Inheritance** tab.
2. Block the inheritance of GPOs to the **Engineers** OU.

Question: What GPOs continue to apply to users in the Engineers OU? Where are those GPOs linked? Why did they continue to apply?

3. Turn off **Block Inheritance** from the **Engineers** OU.

Results: After this exercise, you will have created a GPO called Engineering Application Override, and linked it to the Engineers OU. You will also have an understanding of inheritance, precedence, and the effects of an Enforced link and Block Inheritance.

Exercise 2: Configure GPO Scope with Filtering

As time passes, you discover that only a small number of engineers require the screen saver timeout override that is currently applied to all users in the Engineers OU. In addition, you learn that a small number of users must be exempted from the screen saver timeout policy and other settings configured by the CONTOSO Standards GPO. You decide to use security filtering to manage the scope of the GPOs.

In this exercise, you will modify the scope of GPOs using filtering.

The main tasks for this exercise are as follows:

1. Configure policy application with security filtering.
2. Configure an exemption with security filtering.

► Task 1: Configure policy application with security filtering

1. Run **Active Directory Users and Computers** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Groups\Configuration** OU, create a global security group named **GPO_Engineering Application Override_Apply**.
3. In the GPMC console, select the **Engineering Application Override** GPO. Notice that in the **Security Filtering** section, the GPO applies by default to all authenticated users.
4. Configure the GPO to apply only to the **GPO_Engineering Application Override_Apply** group.

► **Task 2: Configure an exemption with security filtering**

1. Run **Active Directory Users and Computers** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Groups\Configuration** OU, create a global security group named **GPO_CONTOSO Standards_Exempt**.
3. In the GPMC console, select the **CONTOSO Standards** GPO. Notice that in the **Security Filtering** section, the GPO applies by default to all authenticated users.
4. Configure the GPO to deny **Apply Group Policy** permission to the **GPO_CONTOSO Standards_Exempt** group.

Results: After this exercise, you will have configured the Engineering Application Override GPO to apply only to the members of GPO_Engineering Application Override_Apply. You will have also configured a group with the Deny Apply Group Policy permission, which overrides the Allow permission. If any user requires exemption from the policies in the CONTOSO Standards GPO, you can simply add the computer to the group GPO_CONTOSO Standards_Exempt.

Exercise 3: Configure Loopback Processing

You have been asked to configure the screen saver timeout in conference rooms to 45 minutes, so that a screen saver does not appear in the middle of a meeting.

In this exercise, you will configure loopback GPO processing.

The main task for this exercise is as follows:

- Configure loopback processing.

► Task 1: Configure loopback processing

1. Create a new GPO named **Conference Room Policies** and link it to the **Kiosks\Conference Rooms OU**.
2. Confirm that the **Conference Room Policies** GPO is scoped to **Authenticated Users**.
3. Modify the **Screen Saver timeout** policy to launch the screen saver after 45 minutes. Modify the **User Group Policy loopback processing mode** policy setting to use **Merge** mode.

Results: After this exercise, you will have created a Conference Room Policies GPO that applies a 45-minute screen saver timeout to users when they log on to conference room computers.



Note: Do not shut down the virtual machines once you are finished with this lab as the settings you have configured here will be used in subsequent labs.

Lab Review Questions

Question: Many organizations rely heavily on security group filtering to scope GPOs, rather than linking GPOs to specific OUs. In these organizations, GPOs are typically linked very high in the Active Directory logical structure: to the domain itself or to a first-level OU. What advantages are gained by using security group filtering rather than GPO links to manage the scope of the GPO?

Question: Why might it be useful to create an exemption group—a group that is denied the Apply Group Policy permission—for every GPO that you create?

Question: Do you use loopback policy processing in your organization? In what scenarios and for what policy settings can loopback policy processing add value?

Lesson 5

Group Policy Processing

- A Detailed Review of Group Policy Processing
- Slow Links and Disconnected Systems
- Understand When Settings Take Effect

Now that you have learned more about the concepts, components, and scoping of Group Policy, you are ready to examine Group Policy processing closely.

Objectives

After completing this lesson, you will be able to:

- Understand, improve, and manually trigger policy refresh.
- Implement loopback policy processing.

A Detailed Review of Group Policy Processing

- Computer starts; Remote Procedure Call System Service (RPCSS) and Multiple Universal Naming Convention Provider (MUP) are started
- Group Policy Client starts and obtains an ordered list of GPOs that are scoped to the computer
 - Local → Site → Domain → OU → Enforced GPOs
- GPC processes each GPO in order
 - Should it be applied? (enabled/disabled/permission/WMI filter)
 - CSEs are triggered to process settings in GPO
 - Settings configured as Enabled or Disabled are processed
- User logs on
- Process repeats for user settings
- Every 90-120 minutes after startup, computer refresh
- Every 90-120 minutes after logon, user refresh

Key Points

This topic details Group Policy processing. As you read it, keep in mind that Group Policy is all about applying configurations defined by GPOs, that GPOs are applied in an order (site, domain, and OU), and that GPOs applied later in the order have higher precedence; their settings, when applied, will override settings applied earlier. The following sequence details the process through which settings in a domain-based GPO are applied to affect a computer or user:

1. The computer starts, and the network starts. Remote Procedure Call System Service (RPCSS) and Multiple Universal Naming Convention Provider (MUP) are started. The Group Policy Client is started.

2. The Group Policy Client obtains an ordered list of GPOs scoped to the computer.

The order of the list determines the order of GPO processing, which is, by default, local, site, domain, and OU:

- **Local GPOs.** Each computer running Windows Server 2003, Windows XP, and Windows 2000 has exactly one GPO stored locally. Windows Vista and Windows Server 2008 have multiple local GPOs. The precedence of local GPOs is discussed in the “Local GPOs” section in Lesson 2.
- **Site GPOs.** Any GPOs that have been linked to the site are added to the ordered list next. When multiple GPOs are linked to a site (or domain or OU), the link order, configured on the Scope tab, determines the order in which they are added to the list. The GPO that is highest on the list, with the number closest to 1, has the highest precedence, and is added to the list last. It will, therefore, be applied last, and its settings will override those of GPOs applied earlier.
- **Domain GPOs.** Multiple domain-linked GPOs are added as specified by the link order.



Note: Domain-linked policies are not inherited by child domains. Policies from a parent domain are not inherited by a child domain. Each domain maintains distinct policy links. However, computers in several domains might be within the scope of a GPO linked to a site.

- **OU GPOs.** GPOs linked to the OU highest in the Active Directory hierarchy are added to the ordered list, followed by GPOs linked to its child OU, and so on. Finally, the GPOs linked to the OU that contains the computer are added. If several group policies are linked to an OU, they are added in the order specified by the link order.
- **Enforced GPOs** are added at the end of the ordered list, so their settings will be applied at the end of the process and will, therefore, override settings of GPOs earlier in the list and in the process. As a point of trivia, enforced GPOs are added to the list in reverse order: OU, domain, and then site. This is relevant when you apply corporate security policies in a domain-linked enforced GPO. That GPO will be at the end of the ordered list and will be applied last, so its settings will take precedence.

3. The GPOs are processed synchronously in the order specified by the ordered list. This means that settings in the local GPOs are processed first, followed by GPOs linked to the site, the domain, and the OUs containing the user or computer. GPOs linked to the OU of which the computer or user is a direct member are processed last, followed by enforced GPOs.

As each GPO is processed, the system determines whether its settings should be applied based on the GPO status for the computer node (enabled or disabled) and whether the computer has the Allow Group Policy permission. If a WMI filter is applied to the GPO, and if the computer is running Windows XP or later, it performs the WQL query specified in the filter.

4. If the GPO should be applied to the system, CSEs trigger to process the GPO settings. Policy settings in GPOs will overwrite policies of previously applied GPOs in the following ways:
 - If a policy setting is configured (set to Enabled or Disabled) in a GPO linked to a parent container (OU, domain, or site), and the same policy setting is Not Configured in GPOs linked to its child container, the resultant set of policies for users and computers in the child container will include the parent's policy setting. If the child container is configured with the Block Inheritance option, the parent setting is not inherited unless the GPO link is configured with the Enforced option.
 - If a policy setting is configured (set to Enabled or Disabled) for a parent container, and the same policy setting is configured for a child, the child container's setting overrides the setting inherited from the parent. If the parent GPO link is configured with the Enforced option, the parent setting has precedence.
 - If a policy setting of GPOs linked to parent containers is Not Configured, and the child OU setting is also Not Configured, the resultant policy setting is the setting that results from the processing of local GPOs. If the resultant setting of local GPOs is also Not Configured, the resultant configuration is the Windows default setting.
5. When the user logs on, the process is repeated for user settings. The client obtains an ordered list of GPOs scoped to the user, examines each GPO synchronously, and hands over GPOs that should be applied to the appropriate CSEs for processing. This step is modified if User Loopback Group Policy Processing is enabled. Loopback policy processing is discussed in the next topic.



Note: Some Policy settings are in both the Computer Configuration and User Configuration nodes. Most policy settings are specific to either the User Configuration or Computer Configuration node. A small handful of settings appear in both nodes. Although in most situations, the setting in the Computer Configuration node will override the setting in the User Configuration node, it is important to read the explanatory text accompanying the policy setting to understand the setting's effect and its application.

6. Every 90–120 minutes after computer startup, computer policy refresh occurs, and the process is repeated for computer settings.
7. Every 90–120 minutes after user logon, user policy refresh occurs, and the process is repeated for user settings.

Slow Links and Disconnected Systems

- Group Policy Client determines whether link to domain should be considered slow link
 - By default, less than 500 kilobits per second (kbps)
 - Each CSE can use determination of slow link to decide whether it should process or not
 - Software CSE, for example, does not process
- Disconnected
 - Settings previously applied will continue to take effect
 - Exceptions include startup, logon, logoff, and shutdown scripts
- Connected
 - Windows Vista and later operating systems detect new connection and perform Group Policy refresh if refresh window was missed while disconnected

Key Points

One of the tasks that can be automated and managed with Group Policy is software installation. In Module 7, you'll learn about Group Policy Software Installation (GPSI), which is provided by the software installation CSE. You can configure a GPO to install one or more software packages.

Imagine, however, if a user were to connect to your network over a slow connection. You would not want large software packages to be transferred over the slow link because performance would be problematic.

The Group Policy Client addresses this concern by detecting the speed of the connection to the domain and determining whether the connection should be considered a *slow link*. That determination is then used by each CSE to decide whether to apply settings. The software extension, for example, is configured to forgo policy processing so that software is not installed if a slow link is detected. By default, a link is considered to be slow if it is less than 500 kilobits per second (kbps).

If a user is working while disconnected from the network, the settings previously applied by Group Policy will continue to take effect, so a user's experience is identical whether he or she is on the network or working away from the network. There are exceptions to this rule, most notably that startup, logon, logoff, and shutdown scripts will not run if the user is disconnected.

If a remote user connects to the network, the Group Policy Client wakes up and determines whether a Group Policy refresh window has been missed. If so, it performs a Group Policy refresh to obtain the latest GPOs from the domain. Again, the CSEs determine, based on their policy processing settings, whether settings in those GPOs are applied. This does not apply to Windows XP or Windows Server 2003 systems—only to Windows Vista, Windows Server 2008, and later operating systems.

Additional Reading

- How Core Group Policy Works::
<http://go.microsoft.com/fwlink/?LinkId=168658>

Understand When Settings Take Effect

- GPO replication must happen
 - GPC and GPT must replicate
- Group changes must be incorporated
 - Logoff/logon for user; restart for computer
- Group Policy refresh must occur
 - Windows XP, Windows Vista, and Windows 7 clients
 - Always wait for network at startup and logon
- Settings may require logoff/logon (user) or restart (computer) to take effect
- Manually refresh: `GPOUpdate [/force] [/logoff] [/boot]`
- Most CSEs do not re-apply settings if GPO has not changed
 - Configure in Computer\Admin Templates\System\Group Policy

Key Points

Understand the following GPO Settings:

GPO Replication Must Happen

Before a GPO can take effect, the Group Policy container (GPC) in Active Directory must be replicated to the domain controller from which the Group Policy Client obtains its ordered list of GPOs. Additionally, the Group Policy template (GPT) in SYSVOL must replicate to the same domain controller.

Group Changes Must Be Incorporated

Finally, if you have added a new group, or changed the membership of a group that is used to filter the GPO, that change must also have replicated, and the change must be in the security token of the computer and the user, which requires a restart (for the computer to update its group membership) or a logoff and logon (for the user to update its group membership).

The User or Computer Group Policy Refresh Must Occur

As you know, refresh happens at startup (for computer settings) and logon (for user settings) and every 90-120 minutes thereafter, by default.



Note: An average of 45-60 minutes. Keep in mind that the practical impact of the Group Policy refresh interval is that when you make a change in your environment, it will be *on average* one-half that time, or 45 to 60 minutes, before the change *starts* to take effect.

By default, Windows XP, Windows Vista, and Windows 7 clients perform only background refreshes at startup and logon, meaning that a client might start up and a user might log on *without receiving the latest policies from the domain*. It is highly recommended that you change this default behavior so that policy changes are implemented in a managed, predictable way. Enable the policy setting Always Wait For Network At Startup And Logon for all Windows clients. The setting is located in Computer Configuration\Policies\Administrative Templates\System\Logon. Be sure to read the policy setting's explanatory text. Note that this does *not* affect the startup or logon time for computers that are not connected to a network. If the computer detects that it is disconnected, it simply moves on and does not literally "wait" for a network. The contoso.com domain used in this course has been pre-configured with this additional Group Policy setting.

Settings Might Not Take Effect Immediately

Although most settings are applied during a background policy refresh, some CSEs do not apply the setting until the next startup or logon event. Newly added startup and logon script policies, for example, will not run until the next computer startup or logon. Software installation, discussed in Module 7, will occur at the next startup if the software is assigned in computer settings. Changes to folder redirection policies will not take effect until the next logon.

Manually Refresh Group Policy with GPOUpdate

When you are experimenting with Group Policy or trying to troubleshoot Group Policy processing, you might need to initiate a Group Policy refresh manually so that you do not have to wait for the next background refresh. The GPOUpdate command can be used to initiate a Group Policy refresh. Used on its own, GPOUpdate triggers processing identical to a background Group Policy refresh. Both computer policy and user policy are refreshed. Use the /target:computer or /target:user parameter to limit the refresh to computer or user settings, respectively. During background refresh, by default, settings are applied only if the GPO has been updated. The /force switch causes the system to reapply all settings in all GPOs scoped to the user or computer. Some policy settings require a logoff or reboot before they actually take effect. The /logoff and /boot switches of GPOUpdate cause a logoff or reboot, respectively, if settings are applied that require one.

So the command that will cause a total refresh, application and (if necessary) reboot and logon to apply updated policy settings is:

```
gpupdate /force /logoff /boot
```

In Windows 2000, the Secedit.exe command was used to refresh policy, so you might encounter a mention of the Secedit.exe command on the exam.

Most CSEs Do Not Re-apply Settings if the GPO Has Not Changed

Remember that most CSEs apply settings in a GPO only if the GPO version has changed. That means that if a user can change a setting that was originally specified by Group Policy, the setting will not be brought back into compliance with the settings specified by the GPO until the GPO changes. Luckily, most policy settings cannot be changed by a nonprivileged user. However, if a user is an administrator of their computer, or if the policy setting affects a part of the registry or of the system that the user has permissions to change, this could be a real problem.

You have the option of instructing each CSE to reapply the settings of GPOs even if the GPOs have not been changed. Processing behavior of each CSE can be configured in policy settings found in Computer Configuration\Administrative Templates\System\Group Policy.

Lesson 6

Troubleshoot Policy Application

- Resultant Set of Policy
- Generate RSoP Reports
- Perform What-If Analyses with the Group Policy Modeling Wizard
- Examine Policy Event Logs

Group Policy application can be complex to analyze and understand, with the interaction of multiple settings in multiple GPOs scoped using a variety of methods. You must be equipped to effectively evaluate and troubleshoot your Group Policy implementation, to identify potential problems before they arise, and to solve unforeseen challenges. Microsoft Windows provides two tools that are indispensable for supporting Group Policy: Resultant Set of Policy (RSOP) and the Group Policy Operational Logs. In this lesson, you will explore the use of these tools in both proactive and reactive troubleshooting and support scenarios.

Objectives

After completing this lesson, you will be able to:

- Analyze the set of GPOs and policy settings that have been applied to a user or computer.
- Proactively model the impact of Group Policy or Active Directory changes on the Resultant Set of Policy.
- Locate the event logs containing Group Policy–related events.

Resultant Set of Policy

- Inheritance, filters, loopback, and other policy scope and precedence factors are complex!
- RSoP
 - The "end result" of policy application
 - Tools to help evaluate, model, and troubleshoot the application of Group Policy settings
- RSoP analysis
 - The Group Policy Results Wizard
 - The Group Policy Modeling Wizard
 - GPRresult.exe

Key Points

In Lesson 4, you learned that a user or computer can be within the scope of multiple GPOs. Group Policy inheritance, filters, and exceptions are complex, and it's often difficult to determine just which policy settings will apply.

Resultant Set of Policy (RSoP) is the net effect of GPOs applied to a user or computer, taking into account GPO links, exceptions such as Enforced and Block Inheritance, and the application of security and WMI filters.

RSoP is also a collection of tools that help you evaluate, model, and troubleshoot the application of Group Policy settings. RSoP can query a local or remote computer and report back the exact settings that were applied to the computer and to any user who has logged on to the computer. RSoP can also model the policy settings that are anticipated to be applied to a user or computer under a variety of scenarios, including moving the object between OUs or sites or changing the object's group membership. With these capabilities, RSoP can help you manage and troubleshoot conflicting policies.

Windows Server 2008 provides the following tools for performing RSoP analysis:

- The Group Policy Results Wizard
- The Group Policy Modeling Wizard
- GPResult.exe

Generate RSoP Reports

- Group Policy Results Wizard
 - Queries WMI to report *actual* Group Policy application
- Requirements
 - Administrative credentials on the target computer
 - Access to WMI (firewall)
 - User must have logged on at least once
- RSoP report
 - Can be saved
 - View in Advanced mode
 - Shows some settings that do not show in the HTML report
 - View Group Policy processing events
 - `GPRresult.exe /s ComputerName /h filename`

Key Points

To help you analyze the cumulative effect of GPOs and policy settings on a user or computer in your organization, the GPMC includes the Group Policy Results Wizard. If you want to understand exactly which policy settings have applied to a user or computer, and why, the Group Policy Results Wizard is the tool to use.

The Group Policy Results Wizard is able to reach into the WMI provider on a local or remote computer running Windows Vista, Windows XP, Windows Server 2003, or Windows Server 2008. The WMI provider can report everything there is to know about the way Group Policy was applied to the system. It knows when processing occurred, which GPOs were applied, which GPOs were not applied and why, errors that were encountered, and the exact policy settings that took precedence and their source GPO.

There are several requirements for running the Group Policy Results Wizard:

- You must have administrative credentials on the target computer.
- The target computer must be running Windows XP or later. The Group Policy Results Wizard cannot access Windows 2000 systems.
- You must be able to access WMI on the target computer. That means that it must be powered on, connected to the network, and accessible through ports 135 and 445.



Note: Enable remote administration of client computers. Performing RSoP analysis by using Group Policy Results Wizard is just one example of remote administration. To perform remote administration, you may need to configure inbound rules for the firewall used by your clients and servers.

- The WMI service must be started on the target computer.
- If you want to analyze RSoP for a user, that user must have logged on at least once to the computer. It is not necessary for the user to be currently logged on.

After you have ensured that the requirements are met, you are ready to run an RSoP analysis.

To run an RSoP report, right-click Group Policy Results in the GPMC console tree and then click Group Policy Results Wizard.

The wizard prompts you to select a computer. It then connects to the WMI provider on that computer and provides a list of users that have logged on to it. You can then select one of the users or opt to skip RSoP analysis for user configuration policies.

The wizard produces a detailed RSoP report in a dynamic HTML format. If Internet Explorer Enhanced Security Configuration is set, you will be prompted to allow the console to display the dynamic content. You can expand or collapse each section of the report by clicking the Show or Hide link or by double-clicking the heading of the section.

The report is displayed on three tabs:

- **Summary.** The Summary tab displays the status of Group Policy processing at the last refresh. You can identify information that was collected about the system, the GPOs that were applied and denied, security group membership that might have affected GPOs filtered with security groups, WMI filters that were analyzed, and the status of CSEs.
- **Settings.** The Settings tab displays the resultant set of policy settings applied to the computer or user. This tab shows you exactly what has happened to the user through the effects of your Group Policy implementation. A tremendous amount of information can be gleaned from the Settings tab, but *some data isn't reported*, such as IPsec, wireless, and disk quota policy settings.
- **Policy Events.** The Policy Events tab displays Group Policy events from the event logs of the target computer.

After you have generated an RSoP report with the Group Policy Results Wizard, you can right-click the report to rerun the query, print the report, or save the report as either an XML file or an HTML file that maintains the dynamic expanding and collapsing sections. Either file type can be opened with Internet Explorer, so the RSoP report is portable outside the GPMC.

If you right-click the node of the report itself, underneath the Group Policy Results folder in the console tree, you can switch to Advanced View. In Advanced View, RSoP is displayed using the RSoP snap-in, which exposes all applied settings, including IPsec, wireless, and disk quota policies.

Generate RSoP Reports with GPRresult.exe

The GPRresult.exe command is the command-line version of the Group Policy Results Wizard. GPRresult taps into the same WMI provider as the wizard, produces the same information and, in fact, enables you to create the same graphical reports. GPRresult runs on Windows Vista, Windows XP, Windows Server 2003, and Windows Server 2008. Windows 2000 includes a GPRresult.exe command, which produces a limited report of Group Policy processing but is not as sophisticated as the command included in later versions of Windows.

When you run the GPRresult command, you are likely to use the following options:

```
/s computername
```

This option specifies the name or IP address of a remote system. If you use a dot (.) as the computer name, or do not include the /s option, the RSoP analysis is performed on the local computer.

```
/scope [user | computer]
```

This displays RSoP analysis for user or computer settings. If you omit the /scope option, RSoP analysis includes both user and computer settings.

```
/user username
```

This specifies the name of the user for which RSoP data is to be displayed.

```
/r
```

This option displays a summary of RSoP data.

```
/v
```

This options displays verbose RSoP data, which presents the most meaningful information.

```
/z
```

This displays super verbose data, including the details of all policy settings applied to the system. Often, this is more information than you will require for typical Group Policy troubleshooting.

```
/u domain\user /p password
```

This provides credentials that are in the Administrators group of a remote system. Without these credentials, GPRresult runs using the credentials with which you are logged on.

```
[/x | /h] filename
```

This option saves the reports in XML or HTML format, respectively. These options are available in Windows Vista SP1 and later and Windows Server 2008 and later.

Troubleshoot Group Policy with the Group Policy Results Wizard and GPRresult.exe

As an administrator, you will likely encounter scenarios that require Group Policy troubleshooting. You might need to diagnose and solve problems, including the following:

- GPOs are not being applied at all.
- The resultant set of policies for a computer or user is not what was expected.

The Group Policy Results Wizard and GPRresult.exe will often provide the most valuable insight into Group Policy processing and application problems. Remember that these tools examine the WMI RSoP provider to report exactly what happened on a system. Examining the RSoP report will often point you to GPOs that are scoped incorrectly or policy processing errors that prevented the application of GPO settings.

Perform What-If Analyses with the Group Policy Modeling Wizard

- Group Policy Modeling Wizard
 - Emulates Group Policy application to report *anticipated* RSoP

Key Points

If you move a computer or user between sites, domains, or OUs, or change its security group membership, the GPOs scoped to that user or computer will change and, therefore, the RSoP for the computer or user will be different. The RSoP will also change if slow link or loopback processing occurs or if there is a change to a system characteristic that is targeted by a WMI filter.

Before you make any of these changes, you should evaluate the potential impact to the RSoP of the user or computer. The Group Policy Results Wizard can perform RSoP analysis only on what has actually happened. To predict the future and to perform what-if analyses, you can use the Group Policy Modeling Wizard.

To perform Group Policy Modeling, right-click the Group Policy Modeling node in the GPMC console tree, then click Group Policy Modeling Wizard and then perform the steps in the wizard.

Modeling is performed by conducting a simulation on a domain controller, so you are first asked to select a domain controller that is running Windows Server 2003 or later. You do not need to be logged on locally to the domain controller, but the modeling request will be performed on the domain controller. You are then asked to specify the settings for the simulation:

- Select a user or computer object to evaluate, or specify the OU, site, or domain to evaluate.
- Choose whether slow link processing should be simulated.
- Specify to simulate loopback processing and, if so, choose Replace or Merge mode.
- Select a site to simulate.
- Select security groups for the user and for the computer.
- Choose which WMI filters to apply in the simulation of user and computer policy processing.

When you have specified the settings for the simulation, a report is produced that is very similar to the Group Policy Results report discussed earlier. The Summary tab shows an overview of which GPOs will be processed, and the Settings tab details the policy settings that will be applied to the user or computer. This report, too, can be saved by right-clicking it and choosing Save Report.

Examine Policy Event Logs

- **System log**
 - High-level information about Group Policy
 - Errors elsewhere in the system that could impact Group Policy
- **Application log**
 - Events recorded by CSEs
- **Group Policy Operational log**
 - Detailed trace of Group Policy application

Key Points

Windows Vista and Windows Server 2008 improve your ability to troubleshoot Group Policy not only with RSoP tools but also with improved logging of Group Policy events.

- In the System log, you will find high-level information about Group Policy, including errors created by the Group Policy client when it cannot connect to a domain controller or locate GPOs.
- The Application log captures events recorded by CSEs.
- A new log, called the Group Policy Operational Log, provides detailed information about Group Policy processing.

To find the Group Policy logs, open the Event Viewer snap-in or console. The System and Application logs are in the Windows Logs node. The Group Policy Operational Log is found in Applications And Services Logs\Microsoft\Windows\GroupPolicy\Operational.

Lab D: Troubleshoot Policy Application

- Exercise 1: Perform RSoP Analysis
- Exercise 2: Use the Group Policy Modeling Wizard
- Exercise 3: View Policy Events

Logon information

Virtual machine	6425B-HQDC01-A	6425B-DESKTOP101-A
Logon user name	Pat.Coleman	Pat.Coleman
Administrative user name	Pat.Coleman_Admin	
Password	Pa\$\$w0rd	Pa\$\$w0rd

Estimated time: 30 minutes

Scenario

You are responsible for administering and troubleshooting the Group Policy infrastructure at Contoso, Ltd. You want to evaluate the resultant set of policies for users in your environment in order to ensure that the Group Policy infrastructure is healthy, and that all policies are applied as they were intended.

Exercise 1: Perform RSoP Analysis

In this exercise, you will evaluate resultant set of policy using both the Group Policy Results Wizard and the GPRResults command.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Refresh Group Policy.
3. Create a Group Policy results RSoP report.
4. Analyze RSoP with GPRResults.

► Task 1: Prepare for the lab

The virtual machines should already be started and available after completing Labs A, B and C. However, if they are not, you should complete the below steps then step through the exercises in Labs A, B and C before continuing.

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Start 6425B-DESKTOP101-A.
4. Log on to DESKTOP101 as **Pat.Coleman** with the password **Pa\$\$w0rd**.

► Task 2: Refresh Group Policy

1. Run the command prompt as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Run the command **gpupdate /force**. After the command has completed, make a note of the current system time, which you will need to know for a task later in this lab.
3. Restart DESKTOP101 and wait for it to restart before proceeding with the next task.

► **Task 3: Create a Group Policy results RSoP report**

1. On HQDC01, run **Group Policy Management** console as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Use the **Group Policy Results Wizard** to run an RSoP report for **Pat.Coleman** on DESKTOP101.
3. Review the **Group Policy Summary** results. For both user and computer configuration, identify the time of the last policy refresh and the list of allowed and denied GPOs. Identify the components that were used to process policy settings.
4. Click the **Settings** tab. Review the settings that were applied during user and computer policy application, and identify the GPO from which the settings were obtained.
5. Click the **Policy Events** tab, and locate the event that logs the policy refresh you triggered with the **GPUpdate** command in Task 1.
6. Click the **Summary** tab, right-click the page, and choose **Save Report**. Save the report as an HTML file to drive D with a name of your choice. Then open the RSoP report from drive D.

► **Task 4: Analyze RSoP with GPResults**

1. Log on to DESKTOP101 as **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. Run the command prompt with administrative credentials.
3. Type **gpresult /r** and press ENTER.
RSoP summary results are displayed. The information is very similar to the Summary tab of the RSoP report produced by the Group Policy Results Wizard.
4. Type **gpresult /v** and press ENTER.
A more detailed RSoP report is produced. Notice that many of the Group Policy settings applied by the client are listed in this report.
5. Type **gpresult /z** and press ENTER.
The most detailed RSoP report is produced.

6. Type **gpresult /h:"%userprofile%\Desktop\RSOP.html"** and press ENTER.
An RSoP report is saved as an HTML file to your desktop.
7. Open the saved RSoP report from your desktop.
8. Compare the report, its information, and its formatting to the RSoP report you saved in the previous task.

Results: After this exercise, you will have learned how to do a resultant set of policy two ways, using a wizard and from the command line.

Exercise 2: Use the Group Policy Modeling Wizard

Before you roll out the Conference Room Policies GPO for production use, you want to evaluate the effect it will have on users who log on to conference room computers. In this exercise, you will use the Group Policy Modeling Wizard to model the resultant set of policies applied to a user, Mike Danseglio, if he were to log on to a conference room computer, DESKTOP101.

The main task for this exercise is as follows:

- Perform Group Policy results modeling.

► Task 1: Perform Group Policy results modeling

1. Switch to HQDC01.
2. In the Group Policy Management console tree, expand **Forest:Contoso.com**, and then click **Group Policy Modeling**.
3. Right-click **Group Policy Modeling**, and then click **Group Policy Modeling Wizard**.

The Group Policy Modeling Wizard appears.

4. Click **Next**.
5. On the **Domain Controller Selection** page, click **Next**.
6. On the **User And Computer Selection** page, in the **User Information** section, click the **User** option button, and then click **Browse**.

The Select User dialog box appears.

7. Type **Mike.Danseglio** and then press ENTER.
8. In the **Computer Information** section, click the **Computer** option button, and then click **Browse**.

The Select Computer dialog box appears.

9. Type **DESKTOP101** and then press ENTER.
10. Click **Next**.

11. On the **Advanced Simulation Options** page, select the **Loopback Processing** check box and then click **Merge**.

Even though the Conference Room Policies GPO specifies the loopback processing, you must instruct the Group Policy Modeling Wizard to consider loopback processing in its simulation.

12. Click **Next**.
13. On the **Alternate Active Directory Paths** page, click the **Browse** button next to **Computer location**.

The Choose Computer Container dialog box appears.

14. Expand **contoso.com** and **Kiosks**, and then click **Conference Rooms**.

You are simulating the effect of DESKTOP101 as a conference room computer.

15. Click **OK**.
16. Click **Next**.
17. On the **User Security Groups** page, click **Next**.
18. On the **Computer Security Groups** page, click **Next**.
19. On the **WMI Filters for Users** page, click **Next**.
20. On the **WMI Filters for Computers** page, click **Next**.
21. Review your settings on the **Summary of Selections** page, and then click **Next**.
22. Click **Finish**.
23. On the **Summary** tab, scroll to and expand, if necessary, **User Configuration**, **Group Policy Objects**, and **Applied GPOs**.
24. Will the **Conference Room Policies** GPO apply to Mike Danseglio as a User policy when he logs on to DESKTOP101 if DESKTOP101 is in the Conference Rooms OU?

If not, check the scope of the Conference Room Policies GPO. It should be linked to the Conference Rooms OU with security group filtering that applies the GPO to the Authenticated Users special identity. You can right-click the modeling query to rerun the query. If the GPO is still not applying, try deleting and re-building the Group Policy Modeling report, and be very careful to follow each step precisely.
25. Click the **Settings** tab.

26. Scroll to, and expand if necessary, **User Configuration, Policies, Administrative Templates** and **Control Panel/Display**.
27. Confirm that the screen saver timeout is 2700 seconds (45 minutes), the setting configured by the **Conference Room Policies** GPO that overrides the 10-minute standard configured by the **CONTOSO Standards** GPO.

Results: After this exercise, you will have used the Group Policy Modeling Wizard to confirm that the Conference Room Policies GPO will in fact apply its settings to users logging on to conference room computers.

Exercise 3: View Policy Events

As a client performs a policy refresh, Group Policy components log entries to the Windows event logs. In this exercise, you will locate and examine Group Policy-related events.

The main task for this exercise is as follows:

- View policy events.

► Task 1: View policy events

1. On DESKTOP101, where you are logged on as **Pat.Coleman_Admin**, run Event Viewer as an administrator.
2. Locate and review **Group Policy** events in the **System** log.
3. Locate and review **Group Policy** events in the **Application** log.
4. In the **Group Policy Operational** log, locate the first event related in the **Group Policy** refresh you initiated in Exercise 1, with the **GPUpdate** command. Review that event and the events that followed it.

Results: After this exercise, you will have identified Group Policy events in the event logs.



Note: After completing this exercise, turn off all virtual machines and discard undo disks.

Lab Review Questions

Question: In what situations have you used RSoP reports to troubleshoot Group Policy application in your organization?

Question: In what situations have you used, or could you anticipate using, Group Policy modeling?

Question: Have you ever diagnosed a Group Policy application problem based on events in one of the event logs?

MCT USE ONLY. STUDENT USE PROHIBITED

Module 7

Manage Enterprise Security and Configuration with Group Policy Settings

Contents:

Lesson 1: Delegate the Support of Computers	7-4
Lab A: Delegate the Support of Computers	7-16
Lesson 2: Manage Security Settings	7-20
Lab B: Manage Security Settings	7-48
Lesson 3: Manage Software with GPSI	7-61
Lab C: Manage Software with GPSI	7-80
Lesson 4: Auditing	7-86
Lab D: Audit File System Access	7-99

Module Overview

- Delegate the Support of Computers
- Manage Security Settings
- Manage Software with GPSI
- Auditing

Group Policy can be used to manage the configuration of an enormous variety of components and features of Windows®. In the previous module, you learned how to configure a Group Policy infrastructure. In this module, you will learn to apply that infrastructure to manage several types of configuration related to security and software installation. You will also discover tools, such as the Security Configuration Wizard, that make it easier to determine which settings should be configured based on a server's roles. Finally, you will learn how to configure auditing of files and folders.

Objectives

After completing this module, you will be able to:

- Delegate the support of computers.
- Use Restricted Groups policies to modify or enforce the membership of groups.
- Use Group Policy Preferences to modify the membership of groups.

- Configure security settings by using the Local Security policy.
- Create and apply security templates to manage security configuration.
- Analyze security configuration based on security templates.
- Create, edit, and apply security policies by using the Security Configuration Wizard.
- Deploy security configuration with Group Policy.
- Deploy software by using GPSI.
- Remove software originally installed with GPSI.

Lesson 1

Delegate the Support of Computers

- Understand the Support of Computers
- Demonstration: Delegate Administration by Using Restricted Groups Policies
- Define Group Membership with Group Policy Preferences

Many enterprises have one or more members of personnel dedicated to supporting end users, a role often referred to as the help desk, desktop support, or just support. Help desk personnel are often asked to perform troubleshooting, configuration, or other support tasks on client computers, and these tasks often require administrative privileges. Therefore, the credentials used by support personnel must be at the level of a member of the local Administrators group on client computers, but desktop support personnel do not need the high level of privilege given to the Domain Admins group, so it is not recommended to place them in that group. Instead, configure client systems so that a group representing support personnel is added to the local Administrators group. Restricted groups policies enable you to do just that, and in this lesson, you will learn how to use restricted groups policies to add the help desk personnel to the local Administrators group of clients and, thereby, to delegate support of those computers to the help desk. The same approach can be used to delegate the administration of any scope of computers to the team responsible for those systems.

Objectives

After completing this lesson, you will be able to:

- Delegate the administration of computers.
- Use Restricted Groups policies to modify or enforce the membership of groups.
- Use Group Policy Preferences to modify the membership of groups.

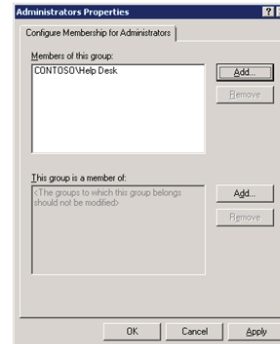
Understand Restricted Groups Policies

- Restricted Groups policies enable you to manage the membership of groups.



Member Of

- Policy is for a domain group
- Specify its membership in a local group
- Cumulative

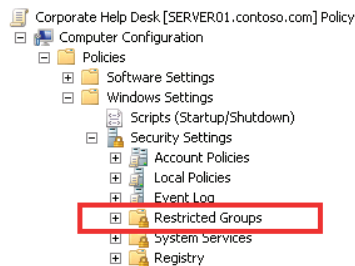


Members

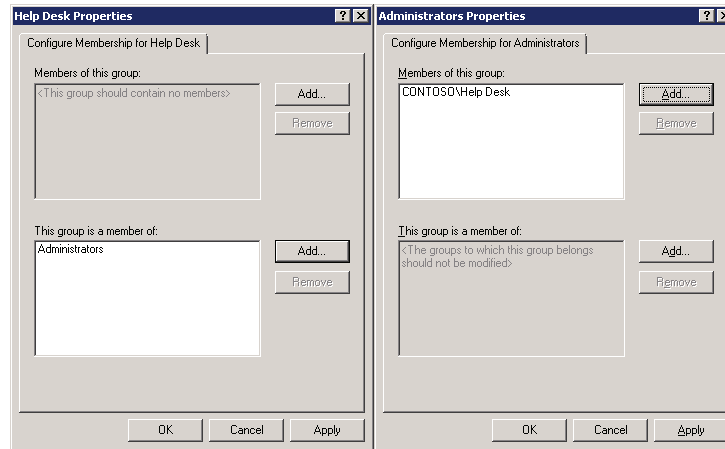
- Policy is for a local group
- Specify its members (groups and users)
- Authoritative

Key Points

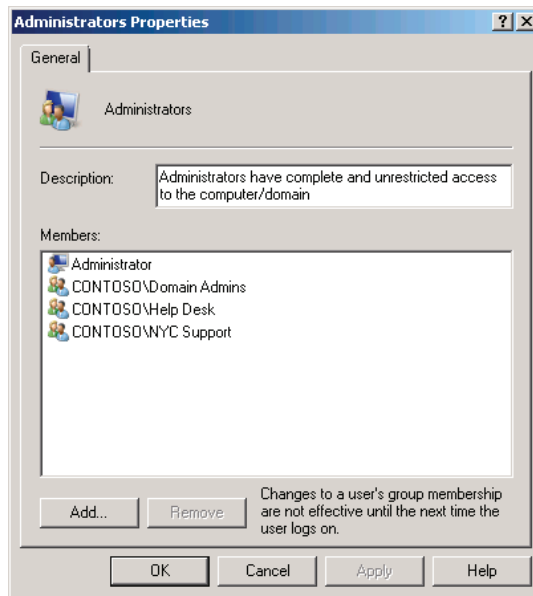
When you edit a Group Policy object (GPO) and expand the Computer Configuration node, the Policies node, the Windows Settings node, and the Security Settings node, you will find the Restricted Groups policy node, shown in the following screen shot.



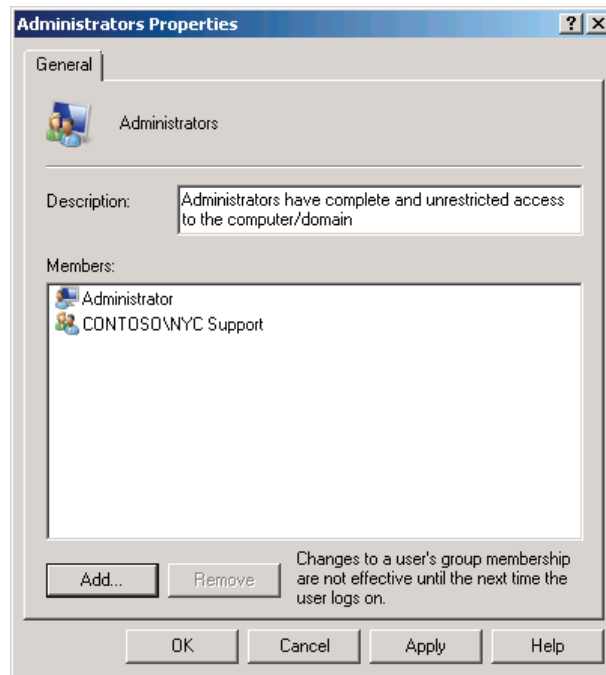
Restricted Groups policy settings enable you to manage the membership of groups. There are two types of settings: **This group is a member of** (the *Member Of* setting) and **Members of this group** (the *Members* setting).



It's very important to understand the difference between these two settings. A Member Of setting specifies that the group specified by the policy is a member of another group. On the left side of the previous screen shot, you can see a typical example: The CONTOSO\Help Desk group is a member of the Administrators group. When a computer applies this policy setting, it ensures that the Help Desk group from the domain becomes a member of its local Administrators group. If there is more than one GPO with restricted groups policies, each Member Of policy is applied. For example, if a GPO linked to the Client Computers organizational unit (OU) specifies CONTOSO\Help Desk as a member of Administrators, and a second GPO linked to the SEA OU (a sub-OU of the Client Computers OU) specifies CONTOSO\SEA Support as a member of Administrators, a computer in the SEA OU will add both the Help Desk and SEA Support groups to its Administrators group in addition to any existing members of the group, such as Domain Admins. This example is illustrated in the following screen shot.



As you can see, restricted groups policies that use the Member Of setting are cumulative. The second type of restricted groups policy setting is the Members setting, which specifies the entire membership of the group specified by the policy. The dialog box on the right of the side-by-side dialog boxes shown earlier is a typical example: The Administrators group's Members list is specified as CONTOSO\Help Desk. When a computer applies this policy setting, it ensures that the local Administrators group's membership consists *only* of CONTOSO\Help Desk. Any members not specified in the policy are removed, including Domain Admins. The Members setting is the authoritative policy—it defines the final list of members. If there is more than one GPO with restricted group policies, the GPO with the highest priority will prevail. For example, if a GPO linked to the Client Computers OU specifies the Administrators group membership as CONTOSO\Help Desk, and another GPO linked to the SEA OU specifies the Administrators group membership as CONTOSO\SEA Support, computers in the SEA OU will have only the SEA Support group in their Administrators group. This example is illustrated in the following screen shot.



If you use both Members and Member Of restricted groups policies, the precedent Members policy setting sets the authoritative baseline membership for the group, and then the cumulative memberships of Member Of policies augment that baseline.

In your enterprise, be careful to design and test your restricted groups policies to ensure that they achieve the desired result.

Demonstration: Delegate Administration by Using Restricted Groups Policies

In this demonstration, we will

- Add a domain support group to the local Administrators group of client computers
- Define the authoritative membership of the local Administrators group of client computers.

Key Points

You can use restricted groups policies with the Member Of setting to manage the delegation of administrative privileges for computers by following these steps:

Demonstration Steps

1. Start 6425B-HQDC01-A and log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
2. On HQDC01 click **Start >Administrative Tools** and run **Group Policy Management** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
3. In the console tree, expand the **Forest:contoso.com, Domains and contoso.com**, and then click the **Group Policy Objects** container.
4. Right-click the **Group Policy Objects** container, and then click **New**.
5. In the **Name** box, type **Corporate Help Desk**, and then click **OK**.

6. In the details pane, right-click the **Corporate Help Desk**, and then click **Edit**.
The Group Policy Management Editor appears.
7. In the **Group Policy Management Editor**, navigate to **Computer Configuration\Policies\Windows Settings\Security Settings\Restricted Groups**.
8. Right-click **Restricted Groups**, and choose **Add Group**.
9. Click the **Browse** button and, in the **Select Groups** dialog box, type the name of the group you want to add to the Administrators group—for example, **CONTOSO\Help Desk**—and click **OK**.
10. Click **OK** to close the **Add Group** dialog box.
A Properties dialog box appears.
11. Click the **Add** button next to the **This Group Is A Member Of** section.
12. Type **Administrators**, and click **OK**.
The Properties group policy setting should look similar to the dialog box on the left of the side-by-side dialog boxers shown earlier.
13. Click **OK** again to close the **Properties** dialog box.

Delegating the membership of the local Administrators group in this manner adds the group specified in step 9 to that group. It does not remove any existing members of the Administrators group. The Group Policy setting simply tells the client, “Make sure this group is a member of the local Administrators group.” This allows for the possibility that individual systems could have other users or groups in their local Administrators group. This group policy setting is also cumulative. If multiple GPOs configure different security principals as members of the local Administrators group, all will be added to the group.

To take complete control of the local Administrators group, follow these steps:

Demonstration Steps

1. In the **Group Policy Management Editor**, navigate to **Computer Configuration\Windows Settings\Security Settings\Restricted Groups**.
2. Right-click **Restricted Groups**, and choose **Add Group**.
3. Type **Administrators**, and click **OK**.
A Properties dialog box appears.
4. Click the **Add** button next to the **Members Of This Group** section.

5. Click the **Browse** button and enter the name of the group you want to make the sole member of the Administrators group—for example, CONTOSO\Help Desk—and click **OK**.
6. Click **OK** again to close the **Add Member** dialog box.

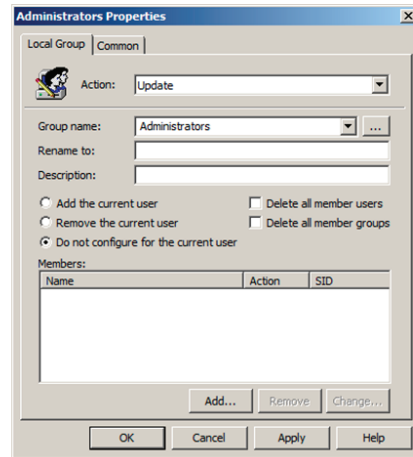
The group policy setting Properties should look similar to the dialog box on the left of the side-by-side dialog boxes shown earlier.
7. Click **OK** again to close the **Properties** dialog box.

When you use the Members setting of a restricted groups policy, the Members list defines the final membership of the specified group. The steps just listed result in a GPO that authoritatively manages the Administrators group. When a computer applies this GPO, it will add all members specified by the GPO and will remove all members not specified by the GPO, including Domain Admins. Only the local Administrator account will not be removed from the Administrators group because Administrator is a permanent and unremovable member of Administrators.

Define Group Membership with Group Policy Preferences

- Create, delete, or replace a local group
- Rename a local group
- Change the Description
- Modify group membership

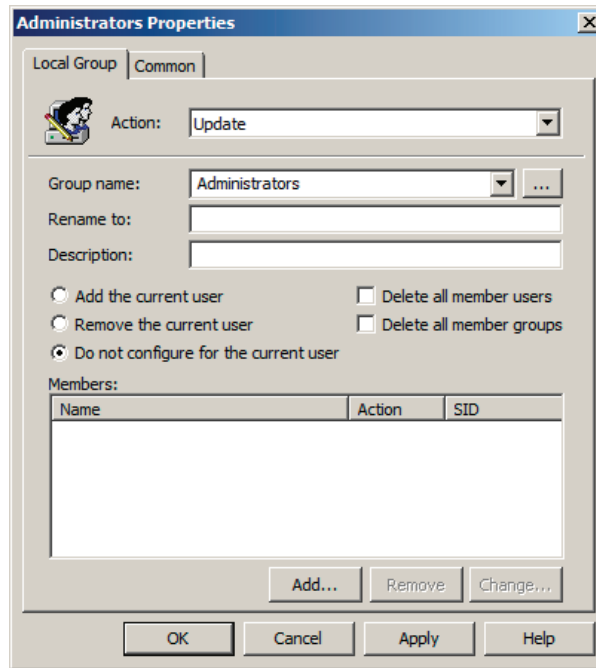
- Local Group preferences are available in both Computer Configuration and User Configuration



Key Points

Group Policy Preferences can also be used to define the membership of groups.

Local Group preferences are available in both Computer Configuration and User Configuration. The settings for a Local Group preference are shown below.



The three options related to "current user" are available only in the Local Group preference in User Configuration.

You have the ability to create, delete, replace, or modify (update) a local group. As you can see in the previous screen shot, you can rename the group, change its description, or make modifications to the group's membership.

Local Group preferences cannot remove members from a group if those members were added to a group using a restricted groups policy setting. Additionally, if a restricted groups policy setting uses the Members method to define the authoritative membership of a group, preferences can neither add nor remove members.

The interactions between Members restricted groups policy settings, Member Of restricted groups policy settings, Local Group preferences scoped as computer settings, and Local Group preferences scoped as user settings can be complex to understand. Be sure to thoroughly test the results if you choose to implement multiple methods of managing group membership with Group Policy.

Discussion Questions

1. In what scenarios, or for what reasons might you want to delete all members users or groups?
2. Why might you want to add the currently logged on user?
3. In what scenario might you want to modify the membership of the local Administrators group of a computer using a Local Group preference in the User Configuration node of a GPO that scopes the preference not to specific computers but to specific users?

Additional Reading

- Group Policy Management Console Help, "Local Users and Groups Extension"

Lab A: Delegate the Support of Computers

- Exercise 1: Configure the Membership of Administrators by Using Restricted Groups Policies

Logon information

Virtual machine	6425B-HQDC01-A	6425B-DESKTOP101-A
Logon user name	Pat.Coleman	Do not Logon
Administrative user name	Pat.Coleman_Admin	
Password	Pa\$\$w0rd	

Estimated time: 15 minutes

Scenario

You have been asked by the corporate security team to lock down the membership of the Administrators group on client computers. However, you need to provide the centralized help desk with the ability to perform support tasks for users throughout the organization. Additionally, you must empower the local site desktop support team to perform administrative tasks for client computers in that site.

Exercise 1: Configure the Membership of Administrators by Using Restricted Groups Policies

In this exercise, you will use Group Policy to delegate the membership of the Administrators group. You will first create a GPO with a restricted groups policy setting that ensures that the Help Desk group is a member of the Administrators group on all client systems. You will then create a GPO that adds the SEA Support group to Administrators on clients in the SEA OU. Finally, you will confirm that in the SEA OU, both the Help Desk and SEA Support groups are administrators.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Delegate the administration of all clients in the domain.
3. Create a Seattle Support group.
4. Delegate the administration of a subset of clients in the domain.
5. Confirm the cumulative application of Member Of policies.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Start 6425B-DESKTOP101-A but do not log on to the system.

► Task 2: Delegate the administration of all clients in the domain

1. Run **Group Policy Management** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Create a **GPO** named **Corporate Help Desk**, scoped to all computers in the Client Computers OU.
3. Configure a **Restricted Groups** policy setting that ensures that the Help Desk group is a member of the Administrators group on all client systems.

► **Task 3: Create a Seattle Support group**

1. Run **Active Directory Users and Computers** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Groups\Role** OU, create a global security group called **SEA Support**.
3. Close Active Directory Users and Computers.

► **Task 4: Delegate the administration of a subset of clients in the domain**

1. In **Group Policy Management**, create a GPO named **Seattle Support**, scoped to all computers in the Client Computers\SEA OU.
2. Configure a **Restricted Groups** policy setting that ensures that the SEA Support group is a member of the Administrators group on all client systems in the SEA OU.

► **Task 5: Confirm the cumulative application of Member Of policies**

- Use **Group Policy Modeling** to confirm that a computer in the SEA OU will include both the Help Desk and SEA Support groups in its Administrators group.

Results: After this exercise, you will have created a Corporate Help Desk GPO that ensures that the Help Desk group is a member of the local Administrators group on all computers in the Client Computers OU. Additionally, you will have created a Seattle Support GPO that adds the Seattle Support group to the local Administrators group on all client computers in the SEA OU.



Important: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in subsequent labs.

Lab Review Question

Question: If you wanted to ensure that the *only* members of the local Administrators group on a client computer were the Help Desk in the site-specific Support group, and to remove any other members from the local Administrators group, how would you achieve that using only restricted groups policies?

Lesson 2

Manage Security Settings

- What Is Security Policy Management?
- Configure the Local Security Policy
- Manage Security Configuration with Security templates
- Demonstration: Create and Deploy Security Templates
- Use Security Configuration and Analysis
- The Security Configuration Wizard
- Settings, Templates, Policies, and GPOs

Security is a primary concern for all Windows administrators. Windows Server® 2008 includes numerous settings that affect the services that are running, the ports that are open, the network packets that are allowed into or out of the system, the rights and permissions of users, and the activities that are audited. There is an enormous number of settings that can be managed, and unfortunately, there is no magic formula that applies the perfect security configuration to a server. The appropriate security configuration for a server depends on the roles that server plays, the mix of operating systems in the environment, and the security policies of the organization, which themselves depend on compliance regulations enforced from outside the organization.

Therefore, you must work to determine and configure the security settings that are required for servers in your organization, and you must be prepared to manage those settings in a way that centralizes and optimizes security configuration. Windows Server 2008 provides several mechanisms with which to configure security settings on one or more systems. In this lesson, you will discover these mechanisms and their interactions.

Objectives

After completing this lesson, you will be able to:

- Configure security settings on a computer using the Local Security policy.
- Create and apply security templates to manage security configuration.
- Analyze security configuration based on security templates.
- Create, edit, and apply security policies using the Security Configuration Wizard.
- Deploy security configuration with Group Policy.

What Is Security Policy Management?

- Enterprise IT Security Policy
 - security configuration
 - settings
- Manage security configuration
 - Create the security policy
 - Apply the security policy to one or more systems
 - Analyze security settings against the policy
 - Update the policy, or correct the discrepancies on the system
- Tools
 - Local Group Policy and Domain Group Policy
 - Security Templates snap-in
 - Security Configuration and Analysis snap-in
 - Security Configuration Wizard

Key Points

Security policy management involves designing, deploying, managing, analyzing, and revising security settings for one or more configurations of Windows systems. There are likely to be several configurations in a typical enterprise: desktops and laptops, servers, and domain controllers. Most enterprises end up defining even more configurations, for example by delineating various types or roles of servers.

The first words are key: Security Policy. Before you even touch the technology, you need to understand what your enterprise security policy requires; and if you do not yet have a written security policy, begin by creating one. Once you know where you are heading, you are ready to start the journey.

Your security policy, and the requirements it contains, will likely require multiple customizations to the default, out-of-box security configuration of Windows client and server operating systems.

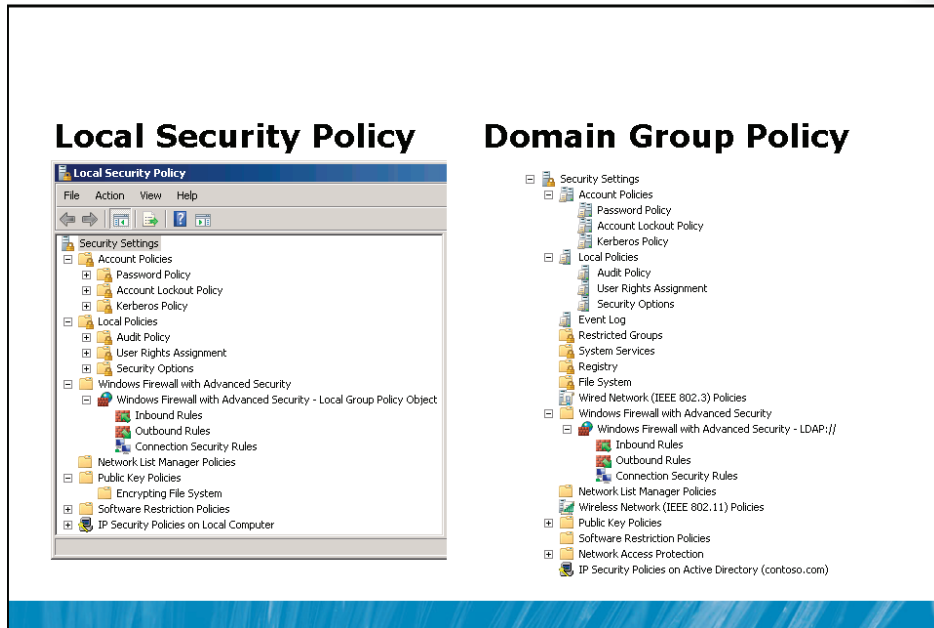
To manage security configuration you will need to:

- Create a security policy for a new application or server role not included in Server Manager.
- Use security policy management tools to apply security policy settings that are unique to your environment.
- Analyze server security settings to ensure that the security policy applied to a server is appropriate for the server role.
- Update a server security policy when the server configuration is modified.

This lesson covers the tools, concepts, and processes required to perform these tasks. The tools you will encounter in this lesson include:

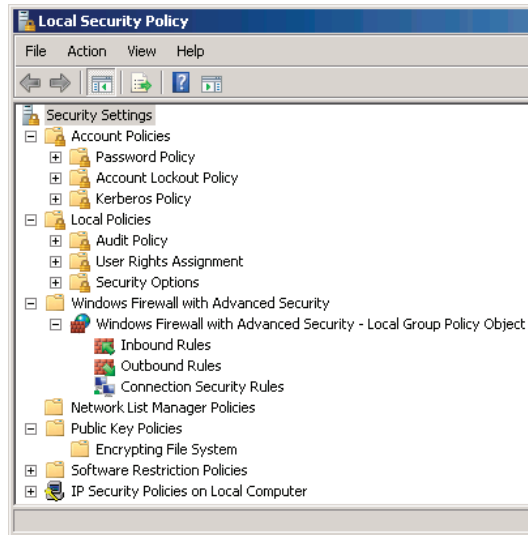
- Local Group Policy
- Security Configuration Wizard
- Security Templates snap-in
- Security Configuration And Analysis snap-in
- Domain Group Policy

Configure the Local Security Policy



Key Points

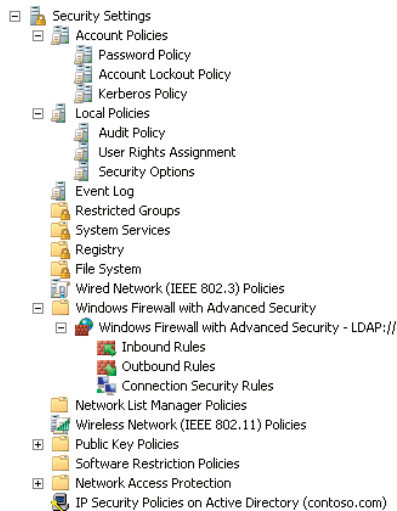
Each server running Windows Server 2008 maintains a collection of security settings that can be managed using the local GPO. You can configure the local GPO by using the Group Policy Object Editor snap-in or the Local Security Policy console. The available policy setting categories are shown on the next page.



This lesson focuses on the mechanisms with which to configure and manage security settings rather than on the details of the settings themselves. Many of the settings—including account policies, audit policy, and user rights assignment—are discussed elsewhere in this course.

Because domain controllers (DCs) do not have local user accounts—only domain accounts—the policies in the Account Policies container of the local GPO on DCs cannot be configured. Instead, account policies for the domain should be configured as part of a domain-linked GPO such as the Default Domain Policy GPO. Account policies are discussed in Module 8.

The settings found in the local Security Settings policies are a subset of the policies that can be configured using domain-based Group Policy, shown below:



As you learned in Module 6, it is a best practice to manage configuration by using domain-based Group Policy rather than on a machine-by-machine basis using local Group Policy. This is particularly true for domain controllers. The Default Domain Controllers Policy GPO is created when the first domain controller is promoted for a new domain. It is linked to the Domain Controllers OU and should be used to manage baseline security settings for all DCs in the domain so that DCs are consistently configured.

Additional Reading

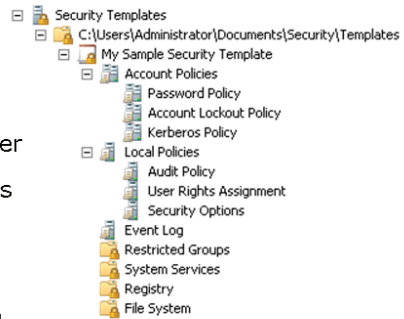
- Server Security Policy Settings:
<http://go.microsoft.com/fwlink/?LinkId=168675>

Manage Security Configuration with Security Templates

- Settings are a subset of domain GPO settings but different than local GPO

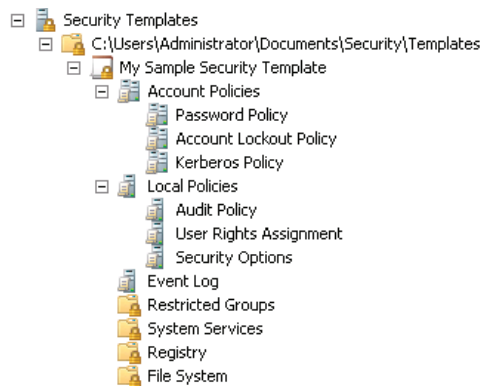
- Security Templates

- Plain text files
- Can be applied directly to a computer
 - Security Configuration & Analysis
 - Secedit.exe
- Can be deployed with Group Policy
- Can be used to analyze a computer's current security settings against the security template's



Key Points

The second mechanism for managing security configuration is the security template. A security template is a collection of configuration settings stored as a text file with the .inf extension. As you can see in the screen shot on the next page, a security template contains settings that are a subset of the settings available in a domain-based GPO but a somewhat different subset than those managed by the local GPO.



The tools used to manage security templates present settings in an interface that enables you to save your security configurations as files and deploy them when and where they are needed. You can also use a security template to analyze the compliance of a computer's current configuration against the desired configuration.

There are several advantages to storing security configuration in security templates. Because the templates are plain text files, you can work with them manually as with any text file, cutting and pasting sections as needed. Second, templates make it easy to store security configurations of various types so that you can easily apply different levels of security to computers performing different roles.

Security templates enable you to configure any of the following types of policies and settings:

- **Account Policies:** Enables you to specify password restrictions, account lockout policies, and Kerberos policies.
- **Local Policies:** Enables you to configure audit policies, user rights assignments, and security options policies.
- **Event Log Policies:** Enables you to configure maximum event log sizes and rollover policies.
- **Restricted Groups:** Enables you to specify the users who are permitted to be members of specific groups.

- **System Services:** Enables you to specify the startup types and permissions for system services.
- **Registry Permissions:** Enables you to set access control permissions for specific registry keys.
- **File System Permissions:** Enables you to specify access control permissions for NTFS files and folders.

You can deploy security templates in a variety of ways, using Active Directory Group Policy Objects, the Security Configuration and Analysis snap-in, or Secedit.exe. When you associate a security template with an Active Directory Group Policy object, the settings in the template become part of the GPO. You can also apply a security template directly to a computer, in which case, the settings in the template become part of the computer's local policies. You will learn about each of these options in this lesson.

Demonstration: Create and Deploy Security Templates

In this demonstration, we will:

- Build a custom MMC with the Security Templates snap-in
- Create a security template
- Import the template into the Security Settings node of a Group Policy object

Key Points

Using the Security Templates Snap-in

To work with security templates, you use the Security Templates snap-in. Windows Server 2008 does not include a console with the Security Templates snap-in, so you have to create one yourself using the MMC Add/Remove Snap-in command. The snap-in creates a folder called Security and a subfolder called Templates in your Documents folder, and the Documents\Security\Templates folder becomes the template search path, where you can store one or more security templates.

To create a new security template:

- Right-click the node that represents your template search path—C:\Users\Documents\Administrator\Security\Templates, for example—and choose **New Template**.

You can also create a template that reflects the current configuration of a server; you'll learn how to do that later in this lesson.

Settings are configured in the template in the same way that settings are configured in a GPO. The Security Templates snap-in is used to configure settings in a security template. It is just an editor—it does not play any role in actually applying those settings to a system. Configure security settings in a template by using the Security Templates snap-in. Although the template itself is a text file, the syntax can be confusing. Using the snap-in ensures that settings are changed using the proper syntax.

The exception to this rule is adding Registry settings that are not already listed in the Local Policies\Security Option portion of the template. As new security settings become known, if they can be configured using a Registry key, you can add them to a security template. To do so, you add them to the Registry Values section of the template.



Note: Save your settings. Be sure to save your changes to a security template by right-clicking the template and choosing Save.

When you install a server or promote it to a domain controller, a default security template is applied by Windows. You can find that template in the %SystemRoot%\Security\Templates folder. On a domain controller, the template is called DC security.inf. You should not modify this template directly, but you can copy it to your template search path and modify the copy.



Note: Security templates in Windows Server 2008 and in earlier versions of Windows. In previous versions of Windows, a number of security templates were available to modify and apply to a computer. The new role-based configuration of Windows Server 2008 and the improved Security Configuration Manager have made these templates unnecessary.

Deploying Security Templates by Using Group Policy Objects

Creating and modifying security templates does not improve security unless you apply those templates. To configure a number of computers in a single operation, you can import a security template into the Group Policy Object for a domain, site, or organizational unit object in Active Directory.

To import a security template into a GPO:

- Right-click the **Security Settings** node and choose **Import Policy**.

In the Import Policy From dialog box, if you select the Clear This Database Before Importing check box, all security settings in the GPO will be erased prior to importing the template settings, so the GPO's security settings will match the template's settings. If you leave the Clear This Database Before Importing check box cleared, the GPO's security policy settings will remain and the template's settings will be imported. Any settings defined in the GPO that are also defined in the template will be replaced with the template's setting.

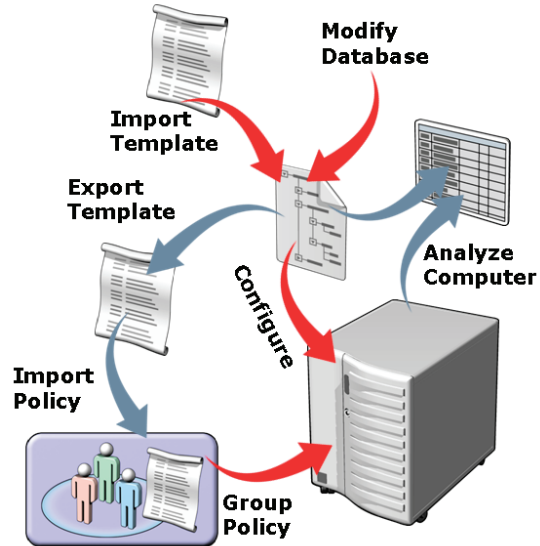
Demonstration Steps

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Click **Start** and in the search box type **mmc.exe** and press ENTER, when prompted supply administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
4. Click **File**, and then click **Add/Remove Snap-in**.
5. In the **Available snap-ins** list, select **Security Templates**, then click **Add**.
6. Click **OK**.
7. Click **File**, and then click **Save**.
The Save As dialog box appears.
8. Type **D:\Security Management**, and then press ENTER.
9. In the console tree, expand **Security Templates**.
10. Right-click **C:\Users\Pat.Coleman_Admin\Documents\Security Templates**, and then click **New Template**.
11. Type **DC Remote Desktop**, and then click **OK**.
12. Click **Start>Administrative Tools** and run **Group Policy Management** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
13. In the console tree, expand the **Forest:contoso.com, Domains and contoso.com**, and then click the **Group Policy Objects** container.
14. In the details pane, right-click the **Corporate Help Desk**, and then click **Edit**.
The Group Policy Management Editor appears.

15. In the console tree, expand **Computer Configuration, Policies, Windows Settings**, and then click **Security Settings**.
16. Right-click **Security Settings**, and then click **Import Policy**.
17. Select the **DC Remote Desktop** template, and then click **Open**.

Use Security Configuration and Analysis

- Build-your-own MMC
 - Create a database
 - Import template(s)
- Use the database
 - Analyze computer
 - Correct discrepancies
 - Configure computer
 - Export as template
- Secedit.exe



Key Points

Security Configuration and Analysis

You can use the Security Configuration and Analysis snap-in to apply a security template to a computer interactively. The snap-in also provides the ability to analyze the current system security configuration and compare it to a baseline saved as a security template. This enables you to determine quickly whether someone has changed a computer's security settings and whether the system conforms to your organization's security policies.

As with the Security Templates snap-in, Windows Server 2008 does not include a console with the Security Configuration and Analysis snap-in, so you must add the snap-in to a console yourself.

Creating a Database

To use the Security Configuration and Analysis snap-in, you must first create a database that will contain a collection of security settings. The database is the interface between the actual security settings on the computer and the settings stored in your security templates.

To create a database (or open an existing one)

- Right-click the **Security Configuration and Analysis** node in the console tree.

You can then import one or more security templates. If you import more than one template, you must decide whether to clear the database. If the database is cleared, only the settings in the new template will be part of the database. If the database is not cleared, additional template settings that are defined will override settings from previously imported templates. If settings in newly imported templates are not defined, the settings in the database from previously imported templates will remain.

To summarize, the Security Configuration and Analysis snap-in creates a database of security settings composed of imported security template settings. The settings in the database can be applied to the computer or used to analyze the computer's compliance and discrepancies with the desired state. Remember that settings in a database do not modify the computer's settings or the settings in a template until that database is either used to configure the computer or exported to a template.

Applying Database Settings to a Computer

After you have imported one or more templates to create the database, you can apply the database settings to the computer.

To apply a database

- Right-click **Security Configuration and Analysis** and choose **Configure Computer Now**.

You will be prompted for a path to an error log that will be generated during the application of settings. After applying the settings, examine the error log for any problems.

Analyzing the Security Configuration of a Computer

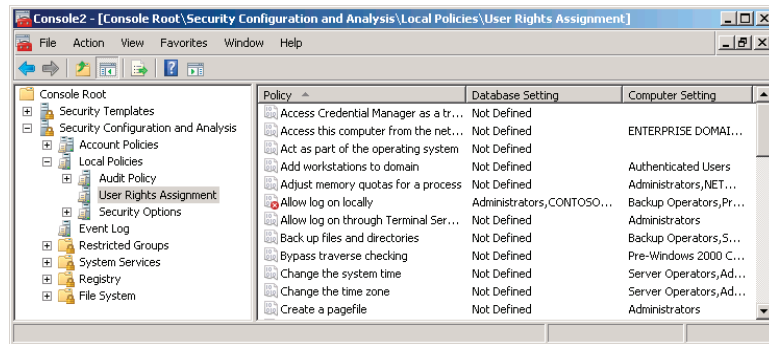
Before applying the database settings to a computer, you might want to analyze the computer's current configuration to identify discrepancies.

To analyze the security configuration of a computer

- Right-click **Security Configuration and Analysis** and choose **Analyze Computer Now**.

The system prompts you for the location of its error log file and then proceeds to compare the computer's current settings to the settings in the database.

After the analysis is complete, the console produces a report such as the one shown in the following screen shot:



Unlike the display of policy settings in the Group Policy Management Editor, Group Policy Object Editor, Local Security Policy, or Security Templates snap-ins, the report shows for each policy the setting defined in the database (which was derived from the templates you imported) and the computer's current setting. The two settings are compared, and the comparison result is displayed as a flag on the policy name. For example, the Allow Log On Locally policy setting is showing a discrepancy between the database setting and the computer setting. The meanings of the flags are as follows:

- **X in a red circle.** Indicates that the policy is defined both in the database and on the computer but that the configured values do not match.
- **Green check mark in a white circle.** Indicates that the policy is defined both in the database and on the computer and that the configured values do match.
- **Question mark in a white circle.** Indicates that the policy is not defined in the database and, therefore, was not analyzed or that the user running the analysis did not have the permissions needed to access the policy on the computer.
- **Exclamation point in a white circle.** Indicates that the policy is defined in the database but does not exist on the computer.
- **No flag.** Indicates that the policy is not defined in the database or on the computer.

Correcting Security Setting Discrepancies

As you examine the elements of the database and compare its settings with those of the computer, you might find discrepancies and want to make changes to the computer's configuration or to the database to bring the two settings into alignment. You can double-click any policy setting to display its Properties dialog box and modify its value in the database. After you've made changes to the database, you can apply the database settings to the computer by performing the steps described earlier, in the section, "Manage Security Configuration with Security Templates."

Applying or exporting database changes

Modifying a policy value in the Security Configuration and Analysis snap-in changes the database value only, not the actual computer setting. For the changes you make to take effect on the computer, you must either apply the database settings to the computer using the Configure Computer Now command or export the database to a new template and apply it to the computer, using a GPO or the Secedit.exe command (discussed in the "Secedit.exe" section later in this lesson.)

Alternatively, you can modify the computer's security settings directly by using the Local Security Policy console, by modifying the appropriate Group Policy Object, or by manually manipulating file system or registry permissions. After making such changes, return to the Security Configuration and Analysis snap-in and choose the Analyze Computer Now command to refresh the analysis of the computer's settings compared to the database.

Creating a Security Template

You can create a new security template from the database.

To create a security template from the database:

- Right-click **Security Configuration and Analysis** and select **Export Template**.

The template will contain the settings in the database, which have been imported from one or more security templates and which you have modified to reflect the current settings of the analyzed computer. The Export Template feature creates a new template from the current database settings at the time you execute the command, not from the computer's current settings.

Secedit.exe

Secedit.exe is a command-line utility that can perform the same functions as the Security Configuration and Analysis snap-in. The advantage of Secedit.exe is that you can call it from scripts and batch files, enabling you to automate your security template deployments. Another big advantage of Secedit.exe is that you can use it to apply only part of a security template to a computer, something you cannot do with the Security Configuration and Analysis snap-in or with Group Policy Objects. For example, if you want to apply the file system's permissions from a template but leave all the other settings alone, Secedit.exe is the only way to do it.

To use Secedit.exe, you run the program from the command prompt with one of the following six main parameters, plus additional parameters for each function:

- **Configure.** Applies all or part of a security database to the local computer. You can also configure the program to import a security template into the specified database before applying the database settings to the computer.
- **Analyze.** Compares the computer's current security settings with those in a security database. You can configure the program to import a security template into the database before performing the analysis. The program stores the results of the analysis in the database itself, which you can view later using the Security Configuration and Analysis snap-in.
- **Import.** Imports all or part of a security template into a specific security database.
- **Export.** Exports all or part of the settings from a security database to a new security template.
- **Validate.** Verifies that a security template is using the correct internal syntax.
- **Generaterollback.** Creates a security template you can use to restore a system to its original configuration after applying another template.

For example, to configure the machine by using a template called BaselineSecurity, use the following command:

```
secdit /configure /db BaselineSecurity.sdb  
/cfg BaselineSecurity.inf /log BaselineSecurity.log
```

To create a rollback template for the BaselineSecurity template, use the following command:

```
secdit /generaterollback /cfg BaselineSecurity.inf  
/rbk BaselineSecurityRollback.inf  
/log BaselineSecurityRollback.log
```

Additional Reading

- For full details regarding Secedit.exe and its switches, see <http://go.microsoft.com/fwlink/?LinkId=168677>

Discussion Question

Question: What procedure is used to apply a security template to a computer.

The Security Configuration Wizard

- Security policy: .xml file that configures
 - Role-based service configuration
 - Network security, including firewall rules
 - Registry values
 - Audit policy
 - Can incorporate a security template (.inf)
- Create the policy
- Edit the policy
- Apply the policy
- Roll back the policy
- Transform the policy into a Group Policy object
 - `scwcmd transform /p:"MySecurity.xml" /g:"My New GPO"`

Key Points

The Security Configuration Wizard can be used to enhance the security of a server by closing ports and disabling services not required for the server's roles.

The Security Configuration Wizard can be launched from the home page of Server Manager, in the Security Information section, or from the Administrative Tools folder.

There is also a command-line version of the tool, `scwcmd.exe`. Type `scwcmd.exe /?` at the command prompt for help on the command or see <http://go.microsoft.com/fwlink/?LinkId=168678>.

The Security Configuration Wizard is a next-generation security management tool, more advanced than the Security Configuration And Analysis snap-in. The Security Configuration Wizard is role-based in accordance with the new role-based configuration of Windows Server 2008. The Security Configuration Wizard creates a security policy—an .xml file—that configures:

- Services
- Network security including firewall rules
- Registry values
- Audit policy
- Other settings based on the roles of a server

That security policy can then be modified, applied to another server, or transformed into a GPO for deployment to multiple systems.

Creating a Security Policy

To create a security policy

1. Launch the Security Configuration Wizard from the **Administrative Tools** folder or the **Security Information** section on the home page of Server Manager.

You can open the Security Configuration Wizard Help file by clicking the Security Configuration Wizard link on the first page of the wizard.

2. Click **Next**.
3. On the **Configuration Action** page, choose **Create a New Security Policy**, and then click **Next**.
4. Enter the name of the server to scan and analyze, and then click **Next**.

The security policy will be based on the roles being performed by the specified server. You must be an administrator on the server for the analysis of its roles to proceed. Ensure also that all applications using inbound IP ports are running prior to running the Security Configuration Wizard.

The Security Configuration Wizard begins the analysis of the selected server's roles. It uses a security configuration database that defines services and ports required for each server role supported by the Security Configuration Wizard. The security configuration database is a set of .xml files installed in %SystemRoot%\Security\Msscw\Kbs.



Note: Centralizing the security configuration database. In an enterprise environment, centralize the security configuration database so that administrators use the same database when running the Security Configuration Wizard. Copy the files in the %SystemRoot%\Security\Msscw\Kbs folder to a network folder. Then launch the Security Configuration Wizard with the Scw.exe command, using the syntax scw.exe /kb DatabaseLocation. For example, the command scw.exe /kb \\server01\scwkb launches the Security Configuration Wizard, using the security configuration database in the shared folder scwkb on SERVER01.

The Security Configuration Wizard uses the security configuration database to scan the selected server and identifies the following:

- Roles that are installed on the server
- Roles likely being performed by the server
- Services installed on the server but not defined in the security configuration database
- IP addresses and subnets configured for the server

The information discovered about the server is saved in a file named Main.xml. This server-specific file is called the configuration database, not to be confused with the security configuration database used by the Security Configuration Wizard to perform the analysis.

To display the configuration database:

- Click the **View Configuration Database** button on the **Processing Security Configuration** page.

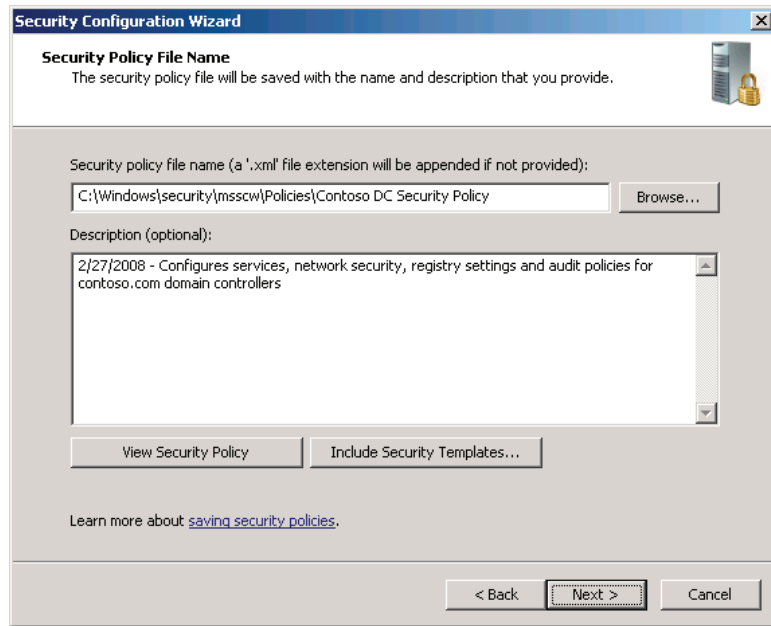
The initial settings in the configuration database are called the baseline settings. After the server has been scanned and the configuration database has been created, you have the opportunity to modify the database, which will then be used to generate the security policy to configure services, firewall rules, registry settings, and audit policies. The security policy can then be applied to the server or to other servers playing similar roles. The Security Configuration Wizard presents each of these four categories of the security policy in a section—a series of wizard pages:

- Role-based service configuration
- Network security

- Registry settings
- Audit policy

The Security Policy

You can skip any of the last three sections you do not want to include in your security policy.



When all the configuration sections have been completed or skipped, the Security Configuration Wizard presents the Security Policy section. The Security Policy File Name page, shown in the preceding screen shot, enables you to specify a path, a name, and a description for the security policy.

To examine the settings of the security policy

- Click the **View Security Policy** button.

The settings are very well documented by the Security Configuration Wizard.

You can also import a security template into the security policy.

- Click the **Include Security Templates** button.

Security templates, discussed earlier in this lesson, contain settings that are not provided by Managing Security Configuration with Security Templates, including restricted groups, event log policies, and file system and registry security policies. By including a security template, you can incorporate a richer collection of configuration settings in the security policy. If any settings in the security template conflict with the Security Configuration Wizard, the settings in the Security Configuration Wizard will take precedence. When you click the Next button, you are given the option to apply the security template to the server immediately or to apply the policy later.

Editing a Security Policy

To edit a saved security policy:

1. Open the Security Configuration Wizard.
2. On the **Configuration Action** page, choose **Edit an Existing Security Policy**.
3. Click the **Browse** button to locate the policy .xml file. When prompted to select a server, select the server that was used to create the security policy.

Applying a Security Policy

To apply a security policy to a server:

1. Open the Security Configuration Wizard.
2. On the **Configuration Action** page, choose **Apply an Existing Security Policy**.
3. Click the **Browse** button to locate the policy .xml file.
4. On the **Select Server** page, select a server to which to apply the policy.

Many of the changes specified in a security policy, including the addition of firewall rules for applications already running and the disabling of services require that you restart the server. Therefore, as a best practice, it is recommended to restart a server any time you apply a security policy.

Rolling Back an Applied Security Policy

If a security policy is applied and causes undesirable results, you can roll back the changes. To roll back an applied security policy:

1. Open the Security Configuration Wizard.
2. On the **Configuration Action** page, choose **Rollback the Last Applied Security Policy**.

When a security policy is applied by the Security Configuration Wizard, a rollback file is generated that stores the original settings of the system. The rollback process applies the rollback file.

Modifying Settings of an Applied Security Policy

Alternatively, if an applied security template does not produce an ideal configuration, you can manually change settings by using the Local Security Policy console discussed at the beginning of this lesson in the “Configuring the Local Security Policy” section. Thus, you can see the whole picture of security configuration from manual settings to the generation of security templates to the creation of security policies with the Security Configuration Wizard, which can incorporate security templates, to the application of security policies, and back to the manual configuration of settings.

Deploying a Security Policy Using Group Policy

You can apply a security policy created by the Security Configuration Wizard to a server by using the Security Configuration Wizard itself, by using the `Scwcmd.exe` command, or by transforming the security policy into a GPO.

To transform a security policy into a GPO:

- Log on as a domain administrator and run **Scwcmd.exe** with the transform command.

For example:

```
scwcmd transform /p:"Contoso DC Security.xml" /g:"Contoso DC Security GPO"
```

This command will create a GPO called Contoso DC Security GPO with settings imported from the Contoso DC Security.xml security policy file. The resulting GPO can then be linked to an appropriate scope—site, domain, or OU—by using the Group Policy Management console. Be sure to type `scwcmd.exe transform /?` for help and guidance about this process.

Additional Reading

- Security Configuration Wizard:
<http://go.microsoft.com/fwlink/?LinkId=168678>

Settings, Templates, Policies, and GPOs

- Direct configuration of security-related settings
- Local Security Policy
- Security templates
 - .inf files that define a wide variety of security settings
 - Security Templates, Security Configuration and Analysis
 - Import into a Group Policy object
- Security policies
 - .xml files that define role-based service startup, firewall rules, audit policies, and registry settings
 - Can include security templates
 - Security Configuration Wizard or scwcmd.exe
 - Transform into a GPO using scwcmd
- Modify Group Policy object

Key Points

As suggested in the introduction to the lesson, there are a number of mechanisms with which to manage security settings. You can use tools such as the Local Security Policy console to modify settings on an individual system. You can use security templates, which have existed since Windows 2000, to manage settings on one or more systems and to compare the current state of a system's configuration against the desired configuration defined by the template. Security policies generated by the Security Configuration Wizard are the most recent addition to the security configuration management toolset. They are role-based .xml files that define service startup modes, firewall rules, audit policies, and some registry settings. Security policies can incorporate security templates. Both security templates and security policies can be deployed using Group Policy.

The plethora of tools available can make it difficult to identify the best practice for managing security on one or more systems. Plan to use Group Policy whenever possible to deploy security configuration. You can generate a GPO from a role-based security policy produced by the Security Configuration Wizard, which itself incorporates additional settings from a security template. After the GPO has been generated, you can make additional changes to the GPO by using the Group Policy Management Editor snap-in. Settings not managed by Group Policy can be configured on a server-by-server basis, using the local GPO security settings.

Lab B: Manage Security Settings

- Exercise 1: Manage Local Security Settings
- Exercise 2: Create a Security Template
- Exercise 3: Use Security Configuration and Analysis
- Exercise 4: Use the Security Configuration Wizard

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 30 minutes

Scenario

You are an administrator of the contoso.com domain. As part of your effort to secure the directory service, you want to establish a security configuration to apply to domain controllers that, among other things, specifies who can log on to domain controllers using Remote Desktop to perform administrative tasks.

Exercise 1: Manage Local Security Settings

In this exercise, you will create a group that allows you to manage who is allowed to log on to HQDC01, a domain controller, using Remote Desktop. You will do so by configuring security settings directly on HQDC01.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Enable Remote Desktop on HQDC01.
3. Create a global security group named SYS_DC Remote Desktop.
4. Add SYS_DC Remote Desktop to the Remote Desktop Users group.
5. Configure the Local Security Policy to allow remote desktop connections by SYS_DC Remote Desktop.
6. Revert the local security policy to its default setting.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.

► Task 2: Enable Remote Desktop on HQDC01

1. Run **Server Manager** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Server Summary** section, click **Configure Remote Desktop**, and then click **Allow connections only from computers running Remote Desktop with Network Level Authentication** (more secure).
3. Close Server Manager.

► Task 3: Create a global security group named SYS_DC Remote Desktop

1. Run **Active Directory Users and Computers** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Admins\Admin Groupsp\Server Delegation** OU, create a global security group named **SYS_DC Remote Desktop**.

► **Task 4: Add SYS_DC Remote Desktop to the Remote Desktop Users group**

To connect using Remote Desktop, a user must have the user logon right to log on through Terminal Services, which you will grant to the SYS_DC Remote Desktop group in the next task.

Additionally, the user must have permission to connect to the RDP-Tcp connection. By default, the Remote Desktop Users group and the Administrators group have permission to connect to the RDP-Tcp connection. Therefore, you should add the user (or the SYS_DC Remote Desktop group in this case) to the Remote Desktop Users group.

1. Add the **SYS_DC Remote Desktop** group to the **Remote Desktop Users** group, found in the **Builtin** container.
2. Close Active Directory Users and Computers.



Note: Instead of adding the group to Remote Desktop Users, you could add the SYS_DC Remote Desktop group to the access control list (ACL) of the RDP-Tcp connection, using the Terminal Services Configuration console. Right-click RDP-Tcp and choose Properties; then click the Security tab, click the Add button, and type SYS_DC Remote Desktop. Click OK twice to close the dialog boxes.

► **Task 5: Configure the Local Security Policy to allow Remote Desktop connections by SYS_DC Remote Desktop**

On a domain member (workstation or server), the Remote Desktop Users group has permission to connect to the RDP-Tcp connection and has the user right to log on through Terminal Services. Therefore, on a domain member server or workstation, the easiest way to manage both the user right and the permission on RDP-Tcp connection is to add a user or group directly to the Remote Desktop Users group.

Because HQDC01 is a domain controller, only Administrators has the right to log on with Terminal Services. Therefore, you must explicitly grant the SYS_DC Remote Desktop group the user logon right to log on through Terminal Services.

- Run **Local Security Policy** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
- Modify the configuration of the user rights policy setting, **Allow Log On Through Terminal Services**, and add **SYS_DC Remote Desktop**.

► **Task 6: Revert the local security policy to its default setting**

You will now revert the policy to its default in preparation for following Exercises.

1. Modify the configuration of the user rights policy setting, **Allow Log On Through Terminal Services**, and then remove **SYS_DC Remote Desktop**.
2. Close Local Security Policy.

Results: After this exercise, you should have configured each of the local settings necessary to allow SYS_DC Remote Desktop to log on to HQDC01 using remote desktop.

Exercise 2: Create a Security Template

In this exercise, you will create a security template that gives the SYS_DC Remote Desktop group the right to log on using Remote Desktop.

The main tasks for this exercise are as follows:

1. Create a custom MMC console with the Security Templates snap-in.
2. Create a security template.

► Task 1: Create a custom MMC console with the Security Templates snap-in

1. Run **mmc.exe** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Add the **Security Templates** snap-in.
3. Save the console as **D:\Security Management.msc**.

► Task 2: Create a security template

1. In the **Security Templates** snap-in, create a new security template named **DC Remote Desktop**.
2. Modify the configuration of the user rights policy setting, **Allow log on through Terminal Services**, and then add **SYS_DC Remote Desktop**.
3. Using a **Restricted Groups** setting, configure the template to give **SYS_DC Remote Desktop** to the **Remote Desktop Users** group.
4. Save the changes you made to the template.

Results: After this exercise, you will have configured a security template named DC Remote Desktop that adds the SYS_DC Remote Desktop group to the Remote Desktop Users group, and gives the SYS_DC Remote Desktop group the user logon right to log on through Terminal Services

Exercise 3: Use Security Configuration and Analysis

In this exercise, you will analyze the configuration of HQDC01, using the DC Remote Desktop security template to identify discrepancies between the server's current configuration and the desired configuration defined in the template. You will then create a new security template.

The main tasks for this exercise are as follows:

1. Add the Security Configuration and Analysis snap-in to a custom console.
2. Create a security database and import a security template.
3. Analyze the configuration of a computer using the security database.
4. Configure security settings using a security database.

► **Task 1: Add the Security Configuration And Analysis snap-in to a custom console**

- Add the **Security Configuration and Analysis** snap-in to a custom console and save the change to the console.

► **Task 2: Create a security database and import a security template**

- Create a new security database called **HQDC01Test**.
- Import the **DC Remote Desktop** security template.

► **Task 3: Analyze the configuration of a computer by using the security database**

1. In the console tree, right-click **Security Configuration and Analysis**, and then click **Analyze Computer Now**.
2. Click **OK** to confirm the default path for the error log.
The snap-in performs the analysis.
3. In the console tree, expand **Security Configuration and Analysis** and **Local Policies**, and then click **User Rights Assignment**.

Notice that the Allow log on through Terminal Services policy is flagged with a red circle and an X. This indicates a discrepancy between the database setting and the computer setting.

4. Double-click **Allow log on through Terminal Services**.

Notice the discrepancies. The computer is not configured to allow the SYS_DC Remote Desktop Users group to log on through Terminal Services.

Notice also that the Computer setting currently allows Administrators to log on through Terminal Services. This is an important setting that should be incorporated into the database.

5. Confirm that the **Define this policy in the database** check box is selected.
6. Select the **Administrators** check box, under **Database Setting**.

This will add the right for Administrators to log on through Terminal Services to the database. It does not change the template, and it does not affect the current configuration of the computer.

7. Click **OK**.
8. In the console tree, select **Restricted Groups**.
9. In the details pane, double-click **CONTOSO\SYS_DC Remote Desktop**.
10. Click the **Member Of** tab.

Notice that the database specifies that the SYS_DC Remote Desktop group should be a member of Remote Desktop Users, but the computer is not currently in compliance with that setting.

11. Confirm that the **Define this group in the database** check box is selected.
12. Click **OK**.
13. Right-click **Security Configuration and Analysis**, and then click **Save**.

This saves the security database, which includes the settings imported from the template plus the change you made to allow Administrators to log on through Terminal Services.

The hint displayed in the status bar when you hover over the Save command suggests that you are saving the template. That is incorrect. You are saving the database.

14. Right-click **Security Configuration and Analysis**, and then click **Export Template**.

The Export Template To dialog box appears.

15. Select **DC Remote Desktop**, and then click **Save**.

You have now replaced the template created in Exercise 2 with the settings defined in the database of the Security Configuration and Analysis snap-in.

► **Task 4: Configure security settings by using a security database**

1. Close your Security Management console. If you are prompted to save your settings, click **Yes**.

Closing and reopening the console is necessary to refresh fully the settings shown in the Security Templates snap-in.

2. Run **D:\Security Management.msc** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
3. In the console tree, expand **Security Templates**, **C:\Users\Pat.Coleman_Admin\Documents\Security\Templates**, **DC Remote Desktop**, **Local Policies**, and then click **User Rights Assignment**.
4. In the details pane, double-click **Allow log on through Terminal Services**.

Notice that both the Administrators and SYS_DC Remote Desktop groups are allowed to log on through Terminal Services in the security template.

5. Click **OK**.
6. Right-click **Security Configuration and Analysis**, and then click **Configure Computer Now**.
7. Click **OK** to confirm the error log path. The settings in the database are applied to the server. You will now confirm that the change to the user right was applied.
8. Run **Local Security Policy** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
9. In the console tree expand **Local Policies**, and then click **User Rights Assignment**.
10. Double-click **Allow Log On Through Terminal Services**.

The Allow Log On Through Terminal Services Properties dialog box opens.

11. Confirm that both **Administrators** and **SYS_DC Remote Desktop** are listed.
The Local Security Policy console displays the actual, current settings of the server.
12. Close the Local Security Policy console.
13. Close your custom Security Management console.

Results: After this exercise, you will have created and applied a security template that gives the SYS_DC Remote Desktop the right to log on through Terminal Services, and adds the group as a member of the Remote Desktop Users group.

Exercise 4: Use the Security Configuration Wizard

In this exercise, you will use the Security Configuration Wizard to create a security policy for domain controllers in the contoso.com domain based on the configuration of HQDC01. You will then convert the security policy into a GPO, which could then be deployed to all domain controllers by using Group Policy.

The main tasks for this exercise are as follows:

1. Create a security policy.
2. Transform a security policy into a Group Policy object.

► Task 1: Create a security policy

1. Run the Security Configuration Wizard, in the Administrative Tools folder, with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. On the **Welcome to the Security Configuration Wizard** page, click **Next**.
3. On the **Configuration Action** page, select **Create a new security policy**, and then click **Next**.
4. On the **Select Server** page, accept the default server name, **HQDC01**, and click **Next**.
5. On the **Processing Security Configuration Database** page, you can optionally click **View Configuration Database** and explore the configuration that was discovered on HQDC01.
6. Click **Next**.
7. On the **Role Based Service Configuration** section introduction page, click **Next**.
8. On the **Select Server Roles** page, you can optionally explore the settings that were discovered on HQDC01, but do not change any settings. Click **Next**.
9. On the **Select Client Features** page, you can optionally explore the settings that were discovered on HQDC01, but do not change any settings. Click **Next**.
10. On the **Select Administration And Other Options** page, you can optionally explore the settings that were discovered on HQDC01, but do not change any settings. Click **Next**.

11. On the **Select Additional Services** page, you can optionally explore the settings that were discovered on HQDC01, but do not change any settings. Click **Next**.
12. On the **Handling Unspecified Services** page, do not change the default setting, **Do not change the startup mode of the service**. Click **Next**.
13. On the **Confirm Service Changes** page, in the **View** list, choose **All Services**.
14. Examine the settings in the **Current Startup Mode** column, which reflect service startup modes on HQDC01, and compare them to the settings in the **Policy Startup Mode** column.
15. In the **View** list, select **Changed Services**.
16. Click **Next**.
17. On the **Network Security** section introduction page, click **Next**.
18. On the **Network Security Rules** page, you can optionally examine the firewall rules derived from the configuration of HQDC01. Do not change any settings. Click **Next**.
19. On the **Registry Settings** section introduction page, click **Next**.
20. On each page of the **Registry Settings** section, examine the settings, but do not change any of them, then click **Next**. When the **Registry Settings Summary** page appears, examine the settings and click **Next**.
21. On the **Audit Policy** section introduction page, click **Next**.
22. On the **System Audit Policy** page, examine but do not change the settings. Click **Next**.
23. On the **Audit Policy Summary** page, examine the settings in the **Current Setting** and **Policy Setting** columns. Click **Next**.
24. On the **Save Security Policy** section introduction page, click **Next**.
25. In the **Security Policy File Name** text box, click at the end of the file path and type **DC Security Policy**.
26. Click the **Include Security Templates** button.
27. Click **Add**.

28. Browse to locate the **DC Remote Desktop** template created in Exercise 3, located in your Documents\Security\Templates folder. When you have located and selected the template, click **Open**.

Be careful that you add the Documents\Security\Templates\DC Remote Desktop.inf file and *not* the DC Security.inf default security template.

29. Click **OK** to close the **Include Security Templates** dialog box.
30. Click the **View Security Policy** button.

You are prompted to confirm the use of the ActiveX control.
31. Click **Yes**.
32. Examine the security policy. Notice that the DC Remote Desktop template is listed in the **Templates** section.
33. Close the window after you have examined the policy.
34. In the Security Configuration Wizard, click **Next**.
35. On the **Apply Security Policy** page, accept the **Apply Later** default setting, and then click **Next**.
36. Click **Finish**.

► **Task 2: Transform a security policy into a Group Policy object**

1. Run the Command Prompt as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Type **cd c:\windows\security\msscw\policies**, and then press ENTER.
3. Type **scwcmd transform /?**, and then press ENTER.
4. Use the **scwcmd.exe** command to transform the security policy named "DC Security Policy.xml" to a GPO named "DC Security Policy".

5. Run **Group Policy Management** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
6. Examine the settings of the **DC Security Policy GPO**. Confirm that the BUILTIN\Administrators and CONTOSO\SYS_DC Remote Desktop groups are given the **Allow log on through Terminal Services** user right. Also confirm that the CONTOSO\SYS_DC Remote Desktop group is a member of BUILTIN\Remote Desktop Users.

Results: After this exercise, you will have used the Security Configuration Wizard to create a security policy named DC Security Policy, and transformed the security policy to a Group Policy object named DC Security Policy.



Important: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in subsequent labs

Lab Review Question

Question: Describe the relationship between security settings on a server, Local Group Policy, security templates, the database used in Security Configuration And Analysis, the security policy created by the Security Configuration Wizard, and domain-based Group Policy.

Lesson 3

Manage Software with GPSI

- Understand Group Policy Software Installation (GPSI)
- Demonstration: Create a Software Distribution Point
- Maintain Software Deployed with GPO
- GPSI and Slow Links

You might be aware of several tools that can be used to deploy software within an organization, including Microsoft® System Center Configuration Manager (SCCM or, simply, Configuration Manager) and its predecessor Microsoft Systems Management Server (SMS). Although these tools provide great benefits, including features to meter software use and inventory systems, you can effectively deploy most software without these tools, using only Group Policy software installation (GPSI).

Objectives

After completing this lesson, you will be able to:

- Deploy software using GPSI.
- Remove software originally installed with GPSI.

Understand Group Policy Software Installation (GPSI)

- Client-side extension (CSE)
- Installs supported packages
 - Windows Installer packages (.msi)
 - Optionally modified by Transform (.mst) or patches (.msp)
 - GPSI automatically installs with elevated privileges
 - Downlevel application package (.zap)
 - Supported by “publish” option only
 - Requires user has admin privileges
 - SCCM and other deployment tools can support a wider variety of installation and configuration packages
- No “feedback”
 - No centralized indication of success or failure
 - No built-in metering, auditing, license management

Key Points

Group Policy software installation (GPSI) is used to create a managed software environment that has the following characteristics:

- Users have access to the applications they need to do their jobs, no matter which computer they log on to.
- Computers have the required applications, without intervention from a technical support representative.
- Applications can be updated, maintained, or removed to meet the needs of the organization.

The software installation extension is one of the many client-side extensions (CSEs) that support change and configuration management using Group Policy. CSEs were discussed in Module 6. The extension enables you to manage centrally the initial deployment, the upgrades, and the removal of software. All configuration of the software deployment is managed within a GPO, using procedures detailed later in this lesson.

Windows Installer Packages

GPSI uses the Windows Installer service to install, maintain, and remove software. The Windows Installer service manages software, using information contained in the application's Windows Installer package. The Windows Installer package is in a file with an .msi extension that describes the installed state of the application. The package contains explicit instructions regarding the installation and removal of an application. You can customize Windows Installer packages by using one of the following types of files:

- **Transform (.mst) files.** These files provide a means for customizing the installation of an application. Some applications provide wizards or templates that permit a user to create transforms. For example, Adobe provides an enterprise deployment tool for Adobe Acrobat Reader that generates a transform. Many enterprises use the transform to configure agreement with the end-user license agreement and to disable certain features of the application, such as automatic updates that involve access to the Internet.
- **Patch (.msp) files.** These files are used to update an existing .msi file for security updates, bug fixes, and service packs. An .msp file provides instructions about applying the updated files and registry keys in the software patch, service pack, or software update. For example, updates to Microsoft Office 2003 and later are provided as .msp files.



Note: Installation of .msp and .mst files. You cannot deploy .mst or .msp files alone. They must be applied to an existing Windows Installer package.

GPSI can make limited use of non-MSI application files (.zap file), also known as down-level application packages, that specify the location of the software distribution point (SDP) and the setup command. See knowledge base article 231747 at <http://support.microsoft.com/?kbid=231747> for details. Most organizations do not use .zap files, however, because the installation of the application requires the user to have administrative privileges on the system. When GPSI installs an application by using a Windows Installer package, the user does not require administrative privileges, allowing for a more secure enterprise.



Note: GPSI and Windows Installer packages. GPSI can fully manage applications only if the applications are deployed using Windows Installer packages. Other tools, including Configuration Manager and SMS, can manage applications that use other deployment mechanisms.

The .msi file, transforms, and other files required to install an application are stored in a shared software distribution point (SDP).

Software Deployment Options

- Software deployment options
 - Assign application to users
 - Start menu shortcuts appear
 - Install-on-demand
 - File associations made (optional "Auto Install")
 - Install-on-document invocation
 - Optionally, configure to install at logon
 - Publish application to users
 - Advertised in Programs And Features (Control Panel)
 - Install-on-request
 - Assign to computers
 - Install at startup

You can deploy software by assigning applications to users or computers or by publishing applications for users. You assign required or mandatory software to users or to computers. You publish software that users might find useful in performing their jobs.

Assigning Applications

When you assign an application to a user, the application's local registry settings, including file name extensions, are updated and its shortcuts are created on the Start menu or desktop, thus advertising the availability of the application. The application advertisement follows the user regardless of which physical computer he or she logs on to. This application is installed the first time the user activates the application on the computer, either by selecting the application on the Start menu or by opening a document associated with the application. When you assign an application to the computer, the application is installed during the computer's startup process.

Publishing Applications

When you publish an application to users, the application does not appear as if it is installed on the users' computers. No shortcuts are visible on the desktop or Start menu. Instead, the application appears as an available application for the user to install using Add Or Remove Programs in Control Panel on a Windows XP system or in Programs And Features on a Windows Server 2008, Windows Vista®, or Windows 7 system. Additionally, the application can be installed when a user opens a file type associated with the application. For example, if Acrobat Reader is advertised to users, it will be installed if a user opens a file with a .pdf extension.

Given that applications can be either assigned or published and targeted to users or computers, you can establish a workable combination to meet your software management goals. The following table details the different software deployment options.

Software Deployment Options

	Publish (User Only)	Assign (User)	Assign (Computer)
After deployment of the GPO, the software is available for installation:	The next time a user logs on.	The next time a user logs on.	The next time the computer starts.
Typically, the user installs the software from:	Add Or Remove Programs in Control Panel (Windows XP) or Programs And Features (Windows Server 2008, Windows Vista, Windows 7).	Start menu or desktop shortcut. An application can also be configured to install automatically at logon.	The software is installed automatically when the computer starts up.
If the software is not installed and the user opens a file associated with the software, does the software install?	Yes (if auto-install is enabled).	Yes.	Does not apply; the software is already installed.

(continued)

	Publish (User Only)	Assign (User)	Assign (Computer)
Can the user remove the software by using Control Panel?	Yes, and the user can choose to install it again from Control Panel.	Yes, and the software is available for installation again from the Start menu shortcuts or file associations.	No. Only a local administrator can remove the software; a user can run a repair on the software.
Supported installation files:	Windows Installer packages (.msi files), .zap files.	Windows Installer packages (.msi files).	Windows Installer packages (.msi files).

Additional Reading

- Group Policy Software Installation overview:
<http://go.microsoft.com/fwlink/?LinkId=168691>

Demonstration: Create a Software Distribution Point

In this demonstration, we will:

- Create a software distribution point

Key Points

Now that you understand GPSI at a high level, you are ready to prepare the SDP. The SDP is simply a shared folder from which users and computers can install applications. Create a shared folder and create a separate folder for each application. Then copy the software package, modifications, and all other necessary files to the application folders. Set appropriate permissions on the folders that allow users or computers Read & Execute permission—the minimum permission required to successfully install an application from the SDP. The administrators of the SDP must be able to change and delete files in order to maintain the SDP over time.

Demonstration Steps

1. Start 6425B-HQDC01-A and log on as **Pat.Coleman** with the password, **Pa\$\$w0rd**.
2. Start 6425B-SERVER01-A but do not log on.
3. Switch to HQDC01.

4. Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
5. In the console tree, expand the **contoso.com** domain and the **Groups** OU, and then click the **Application** OU.
6. Right-click the **Application** OU, point to **New**, and then click **Group**.
7. Type **APP_XML Notepad**, and then press ENTER.
8. In the console tree, expand the **contoso.com** domain and the **Servers** OU, and then click the **File** OU.
9. In the details pane, right-click **SERVER01**, and then click **Manage**.
The Computer Management console opens, focused on SERVER01.
10. In the console tree, expand **System Tools** and **Shared Folders**, and then click **Shares**.
11. Right-click **Shares**, and then click **New Share**. The Create A Shared Folder Wizard appears.
12. Click **Next**.
13. In the **Folder Path** box, type **C:\Software**, and then click **Next**.
A message appears asking if you want to create the folder.
14. Click **Yes**.
15. Accept the default Share name, **Software**, and then click **Next**.
16. Click **Customize permissions**, and then click the **Custom** button.
17. Click the **Security** tab.
18. Click **Advanced**.
The Advanced Security Settings dialog box appears.
19. Click **Edit**.
20. Clear the option, **Include inheritable permissions from this object's parent**.
A dialog box appears asking if you want to Copy or Remove inherited permissions.
21. Click **Copy**.
22. Select the first permission assigned to the **Users** group, and then click **Remove**.

23. Select the remaining permission assigned to the **Users** group, and then click **Remove**.
24. Select the permission assigned to **Creator Owner**, and then click **Remove**.
25. Click **OK** two times to close the **Advanced Security Settings** dialog boxes.
26. In the **Customize Permissions** dialog box, click the **Share Permissions** tab.
27. Select the check box next to **Full Control** and below **Allow**.

Security management best practice is to configure least privilege permissions in the ACL of the resource, which will apply to users regardless of how users connect to the resource, at which point you can use the Full Control permission on the SMB shared folder. The resultant access level will be the more restrictive permissions defined in the ACL of the folder.

28. Click **OK**.
29. Click **Finish**.
30. Click **Finish** to close the wizard.
31. Click **Start**, click **Run**, type `\\SERVER01\c$`, and then press ENTER.

The Connect to SERVER01 dialog box appears.

32. In the **User name** box, type `CONTOSO\Pat.Coleman_Admin`.
33. In the **Password** box, type `Pa$$w0rd`, and then press ENTER.

A Windows Explorer window opens, focused on the root of the C drive on SERVER01.

34. Open the **Software** folder.
35. Click the **File** menu, point to **New**, and then click **Folder**.

A new folder is created and is in "rename mode."

36. Type **XML Notepad**, and then press ENTER.
37. Right-click the **XML Notepad** folder, and then click **Properties**.
38. Click the **Security** tab.
39. Click **Edit**.

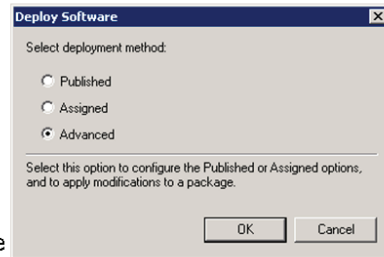
40. Click **Add**. The Select Users, Computers, or Groups dialog box appears.
41. Type **APP_XML Notepad**, and then press ENTER.

The group is given the default, Read & Execute permission.

42. Click **OK** twice to close all open dialog boxes.
43. Open the **XML Notepad** folder.
44. Open the **D:\Labfiles\Lab07b** folder in a new window.
45. Right-click **XMLNotepad.msi**, and then click **Copy**.
46. Switch to the Windows Explorer window displaying **\\server01\c\$\Software\XML Notepad**.
47. Right-click in the empty details pane, and then click **Paste**.
XML Notepad is copied into the folder on SERVER01.
48. Close all open Windows Explorer windows.
49. Close the Computer Management console.

Create and Scope a Software Deployment GPO

- Computer [or User] Configuration \ Policies \ Software Settings \ Software Installation
 - Right-click → New → Package
 - Browse to .msi file through *network* path (\\server\share)
 - Choose deployment option recommend: Advanced
- Managing the scope of a software deployment GPO
 - Typically easiest to manage with security group filtering
 - Create an app group, for example APP_XML Notepad
 - Put users into the group: allows users to access software share in the event that repairs or reinstalls are necessary
 - Put computers into the group if assigning to computers



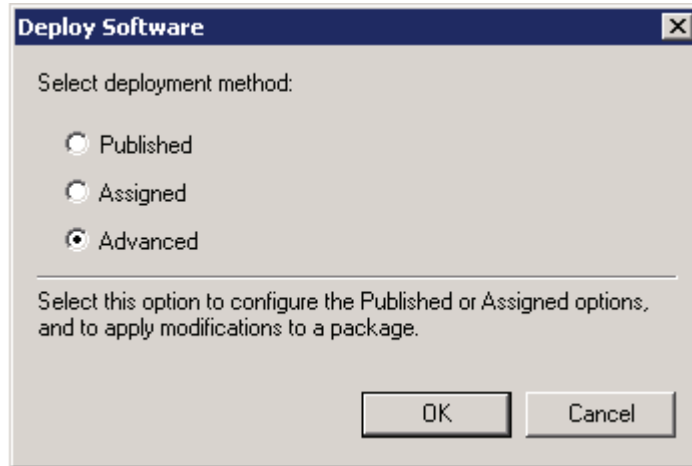
Key Points

To create a software deployment GPO:

1. Use the Group Policy Management console to create a new GPO or select an existing GPO.
2. Edit the GPO using the **Group Policy Management Editor**.
3. Expand the console nodes **Computer Configuration\Policies\Software Settings\Software Installation**. Alternatively, select the **Software Installation** node in the **User Configuration** branch.
4. Right-click **Software Installation**, choose **New**, and then select **Package**.

5. Browse to locate the .msi file for the application. Click **Open**.

The Deploy Software dialog box appears, shown in the following screen shot:



6. Select **Published**, **Assigned**, or **Advanced**.

You cannot publish an application to computers, so the option will not be available if you are creating the package in the Software Installation node in Computer Configuration.

The Advanced option enables you to specify whether the application is published or assigned and gives you the opportunity to configure advanced properties of the software package. Therefore, *it is recommended that you choose Advanced*. The package properties dialog box then appears. Among the more important properties that you can configure are the following choices:

- **Deployment Type:** On the Deployment tab, configure Published or Assigned.
- **Deployment Options:** Based on the selected deployment type, different choices will appear in the Deployment Options section. These options, along with other settings on the Deployment tab, manage the behavior of the application installation.
- **Uninstall This Application When It Falls Out Of the Scope Of Management:** If this option is selected, the application will be automatically removed when the GPO no longer applies to the user or computer.

- **Upgrades:** On the Upgrades tab, you can specify the software that this package will upgrade. Upgrades are discussed in the “Maintain Software Deployed with GPSI” section later in this lesson.
- **Categories:** The Categories tab enables you to associate the package with one or more categories. Categories are used when an application is published to a user. When the user goes to Control Panel to install a program, applications published using GPSI are presented in groups based on these categories.

To create categories that are available to associate with packages, right-click Software Installation and choose Properties; then click the Categories tab.

- **Modifications:** If you have a transform (.mst file) that customizes the package, click the Add button to associate the transform with the package. Most tabs in the package Properties dialog box are available for you to change settings at any time. However, the Modifications tab is available only when you create the new package and choose the Advanced option.

Managing the Scope of a Software Deployment GPO

After you have created a software deployment GPO, you can scope the GPO to distribute the software to appropriate computers or users. In many software management scenarios, applications should be assigned to computers rather than to users. This is because most software licenses allow an application to be installed on one computer, and if the application is assigned to a user, the application will be installed on each computer to which the user logs on.

As you learned in Module 6, you can scope a GPO by linking the GPO to an OU or by filtering the GPO so that it applies only to a selected global security group. Many organizations have found that it is easiest to manage software by linking an application's GPO to the domain and filtering the GPO with a global security group that contains the users and computers to which the application should be deployed. For example, a GPO that deploys the XML Notepad tool (available from the Microsoft downloads site at <http://www.microsoft.com/downloads>) would be linked to the domain and filtered with a group containing developers that require the tool. The group would have a descriptive name that indicates its purpose to manage the deployment of XML Notepad—APP_XML Notepad, for example.

Maintain Software Deployed with GPSI

- **Redeploy application**
 - After successful install, client will not attempt to reinstall app
 - You might make a change to the package
 - Package → All Tasks → Redeploy Application
- **Upgrade application**
 - Create new package in same or different GPO.
 - Advanced → Upgrades → Select package to upgrade
 - Uninstall old version first; or install over old version
- **Remove application**
 - Package → All Tasks → Remove
 - Uninstall immediately (forced removal) or Prevent new installations (optional removal)
 - Don't delete or unlink GPO until all clients have applied setting

Key Points

After a computer has installed an application by using the Windows Installer package specified by a GPO, the computer will not attempt to reinstall the application at each Group Policy refresh. There might be scenarios in which you want to force systems to reinstall the application. For example, small changes might have been made to the original Windows Installer package.

To redeploy an application deployed with Group Policy:

- Right-click the package in the GPO, click **All Tasks**, and then select **Redeploy Application**.

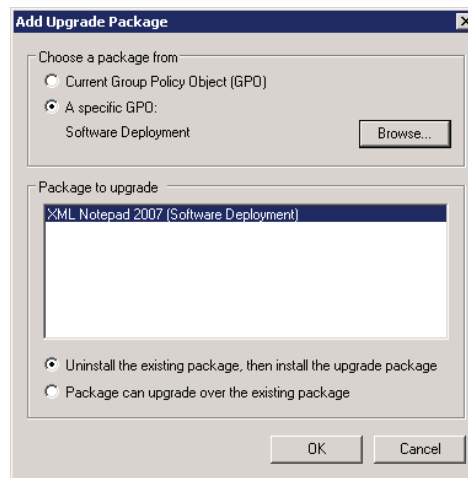
You can also upgrade an application that has been deployed with GPSI.

1. Create a package for the new version of the application in the **Software Installation** node of the GPO.

The package can be in the same GPO as the package for the previous version or in any different GPO.

2. Right-click the package and choose **Properties**.
3. Click the **Upgrades** tab, and then click the **Add** button.

The Add Upgrade Package dialog box appears.



4. Select whether the package for the previous version of the application is in the current GPO or in another GPO. If the previous package is in another GPO, click the **Browse** button to select that GPO.
5. Then select the package from the **Package to upgrade** list.
6. Based on your knowledge of the application's upgrade behavior, choose one of the upgrade options shown at the bottom of the dialog box.
 - Uninstall the existing package, then install the upgrade package
 - Package can upgrade over the existing package
7. Click **OK**.

You can also remove an application that was deployed with GPSI.

1. Right-click the package, click **All Tasks**, and then select **Remove**.
2. In the **Remove Software** dialog box, choose one of the following two options:
 - **Immediately uninstall the software from users and computers.** This option, known as forced removal, causes computers to remove the application. The software installation extension will remove an application when the computer restarts if the application was deployed with a package in the Computer Configuration portion of the GPO. If the package is in the User Configuration portion, the application will be uninstalled the next time the user logs on.
 - **Allows Users To Continue To Use The Software, But Prevents New Installations.** This setting, known as optional removal, causes the software installation extension to avoid adding the package to systems that do not yet have the package installed. Computers that had previously installed the application do not forcibly uninstall the application, so users can continue using it.

If you use one of these two options to remove software using GPSI, it is important that you allow the settings in the GPO to propagate to all computers within the scope of the GPO before you delete, disable, or unlink the GPO. Clients need to receive this setting that specifies forced or optional removal. If the GPO is deleted or no longer applied before all clients have received this setting, the software is not removed according to your instructions. This is particularly important in environments with mobile users on laptop computers that might not connect to the network on a regular basis.

If, when creating the software package, you chose the Uninstall This Application When It Falls Out Of The Scope Of Management option, you can simply delete, disable, or unlink the GPO and the application will be forcibly removed by all clients that have installed the package with that setting.

GPSI and Slow Links

- The Group Policy Client determines whether the domain controller providing GPOs is on the other side of a slow link
 - < 500 kbps by default
- Each CSE uses the “slow link” determination to decide whether to process
 - By default, GPSI does not process over a slow link
- You can change slow link processing behavior of each CSE
 - Computer Configuration\Policies\Administrative Templates\System\Group Policy
- You can change the slow link threshold
 - Computer [or User] Configuration\Policies\Administrative Templates\System\Group Policy

Key Points

When a client performs a Group Policy refresh, it tests the performance of the network to determine whether it is connected using a slow link defined by default as 500 kilobits per second (kbps). Each client-side extension is configured to process Group Policy or to skip the application of settings on a slow link. By default, GPSI does not process Group Policy settings over a slow link because the installation of software over a slow link could cause significant delays.

You can change the slow link policy processing behavior of each client-side extension, using policy settings located in Computer Configuration\Policies\Administrative Templates\System\Group Policy. For example, you could modify the behavior of the software installation extension so that it does process policies over a slow link.

You can also change the connection speed threshold that constitutes a slow link. By configuring a low threshold for the connection speed, you can convince the client-side extensions that a connection is not a slow link, even if it actually is. There are separate Group Policy Slow Link Detection policy settings for computer policy processing and user policy processing. The policies are in the Administrative Templates\System\Group Policy folders in Computer Configuration and User Configuration.

Lab C: Manage Software with GPSI

- Exercise 1: Deploy Software with GPSI
- Exercise 2: Upgrade Applications with GPSI

Logon information

Virtual machine	6425B-HQDC01-A	6425B-DESKTOP101-A	6425B-SERVER01-A
Logon user name	Pat.Coleman	Pat.Coleman	Do not Logon
Administrative user name	Pat.Coleman_Admin		
Password	Pa\$\$w0rd	Pa\$\$w0rd	

Estimated time: 15 minutes

Scenario

You are an administrator at Contoso, Ltd. Your developers require XML Notepad to edit XML files, and you want to automate the deployment and life cycle management of the application. You decide to use Group Policy Software Installation. Most applications are licensed per computer, so you will deploy XML Notepad to the developers' computers, rather than associating the application with their user accounts.

Exercise 1: Deploy Software with GPSI

In this exercise, you will use GPSI to deploy XML Notepad to computers including DESKTOP101.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Create a software distribution folder.
3. Create a software deployment GPO.
4. Deploy software to computers.
5. Confirm the successful deployment of software.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A and log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
2. Start 6425B-SERVER01-A but do not log on.
3. Wait for both SERVER01 to finish startup before continuing with the next task.

► Task 2: Create a software distribution folder

1. On HQDC01, run **Active Directory Users and Computers** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Groups\Application** OU, create a new global security group named **APP_XML Notepad**.
3. In the **Servers\File** OU, right-click **SERVER01**, and then click **Manage**.
4. Use the **Shared Folders** snap-in to create a new shared folder, **C:\Software**, with a share name of **Software**. Configure the NTFS permissions as described below:
 - System::Allow::Full Control
 - Administrators::Allow::Full Control

And configure the Share permission such that the Everyone group is allowed Full Control.

Security management best practice is to configure least privilege permissions in the ACL of the resource, which will apply to users regardless of how users connect to the resource, at which point you can use the Full Control permission on the SMB shared folder. The resultant access level will be the more restrictive permissions defined in the ACL of the folder.

5. Open the administrative share for the C drive on SERVER01 (\\SERVER01\\c\$) as **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
6. Inside the **Software** folder on SERVER01, create a folder called **XML Notepad**.
7. Add permission to the **XML Notepad** folder so that the **APP_XML Notepad** group is allowed **Read & Execute permission**.
8. Copy **XML Notepad.msi** from **D:\\Labfiles\\Lab07b** to **\\SERVER01\\c\$\\Software\\XML Notepad**.
9. Close any opened Windows Explorer windows.
10. Close the Computer Management console.

► **Task 3: Create a software deployment GPO**

1. Run **Group Policy Management** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. In the **Group Policy Objects** container, create a new GPO called **XML Notepad**. Edit that GPO.
3. Expand **Computer Configuration, Policies, Software Settings**, and then click **Software Installation**.
4. Right-click **Software Installation**, point to **New**, and then click **Package**.
5. In the **File name** text box, type the network path to the software distribution folder, **\\server01\\software\\XML Notepad**, and then press ENTER.
6. Select the Windows Installer package, **XmlNotepad.msi**; and then click **Open**.
After a few moments, the Deploy Software dialog box appears.
7. Click **Advanced**, and then click **OK**.
8. On the **General** tab, note that the name of the package includes the version, **XML Notepad 2007**.

9. Click the **Deployment** tab.

Note that when deploying software to computers, Assigned is the only option. Examine the options that would be available if you were assigning or publishing the application to users.

10. Select **Uninstall This Application When It Falls Out Of The Scope Of Management**.
11. Click **OK**.
12. Close the Group Policy Management Editor.
13. Scope the GPO to apply only to members of APP_XML Notepad, and not to Authenticated Users.
14. Link the GPO to the **Client Computers** OU.

► **Task 4: Deploy software to computers**

1. Add DESKTOP101 to the APP_XML Notepad group.
2. Start 6425B-DESKTOP101-A, but do not log on.

► **Task 5: Confirm the successful deployment of software**

1. Log on to DESKTOP101 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
2. Confirm that **XML Notepad** installed successfully.



Note: When verifying the deployment of the xml notepad and it may take two startups to be successful. I.e. if you do not see Notepad installed restart the virtual machine. You may need to do this a couple of times.

Results: After this exercise, you will have deployed XML Notepad to DESKTOP101.

Exercise 2: Upgrade Applications with GPSI

In this exercise, you will simulate deploying an upgraded version of XML Notepad.

The main task for this exercise is as follows:

- Create an upgrade package using GPSI.

► Task 1: Create an upgrade package by using GPSI

1. Switch to HQDC01.
2. In the Group Policy Management console tree, right-click the **XML Notepad** GPO in the **Group Policy Objects** container, and then click **Edit**.

The Group Policy Management Editor opens.

3. In the console tree, expand **Computer Configuration, Policies, Software Settings**, and then click **Software Installation**.
4. Right-click **Software Installation**, point to **New**, and then click **Package**.
5. In the **File name** text box, type the network path to the software distribution folder, **\\server01\software\XML Notepad**, and then press ENTER.

This exercise will use the existing XmlNotepad.msi file as if it is an updated version of XML Notepad.

6. Select the Windows Installer package, **XmlNotepad.msi**, and then click **Open**.

The Deploy Software dialog box appears.

7. Click **Advanced**, and then click **OK**.
8. On the **General** tab, change the name of the package to suggest that it is the next version of the application. Type **XML Notepad 2010**.
9. Click the **Deployment** tab. Because you are deploying the application to computers, **Assigned** is the only deployment type option.
10. Click the **Upgrades** tab.
11. Click the **Add** button.
12. Click the **Current Group Policy Object (GPO)** option.
13. In the **Package to upgrade** list, select the package for the simulated earlier version, **XML Notepad 2007**.

14. Click the **Uninstall the existing package and then select then install the upgrade package** option.
15. Click **OK**.
16. Click **OK**.

If this were an actual upgrade, the new package would upgrade the previous version of the application as clients applied the XML Notepad GPO. Because this is only a simulation of an upgrade, you can remove the simulated upgrade package.

17. Right-click **XML Notepad 2010**, which you just created to simulate an upgrade, point to **All Tasks**, and then select **Remove**.
18. In the **Remove Software** dialog box, click **Immediately uninstall the software from users and computers**, and then click **OK**.

Results: After this exercise, you will have simulated an upgrade of XML Notepad by using GPSI.



Important: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in subsequent labs.

Lab Review Questions

Question: Consider the NTFS permissions you applied to the Software and XML Notepad folders on SERVER01. Explain why these least privilege permissions are preferred to the default permissions.

Question: Consider the methods used to scope the deployment of XML Notepad: Assigning the application to computers, filtering the GPO to apply to the APP_XML Notepad group that contains only computers, and linking the GPO to the Client Computers OU. Why is this approach advantageous for deploying most software? What would be the disadvantage of scoping software deployment to users rather than to computers?

Lesson 4

Auditing

- An Overview of Audit Policies
- Specify Auditing Settings on a File or Folder
- Enable Audit Policy
- Evaluate Events in the Security Log

Auditing is an important component of security. Auditing logs specified activities in your enterprise to the Windows Security log, which you can then monitor to understand those activities and to identify issues that warrant further investigation. Auditing can log successful activities to provide documentation of changes. It can also log failed and potentially malicious attempts to access enterprise resources. Auditing involves up to three management tools: audit policy, auditing settings on objects, and the Security log. In this lesson, you will learn how to configure auditing to address several common scenarios.

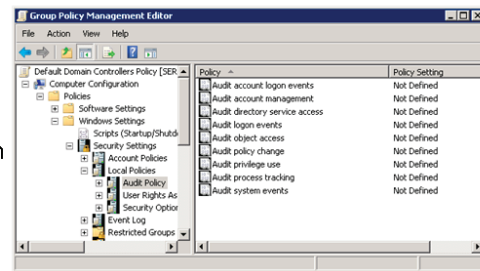
Objectives

After completing this lesson, you will be able to:

- Configure audit policy.
- Configure auditing settings on file system objects.
- View the Security log using the Event Viewer snap-in.

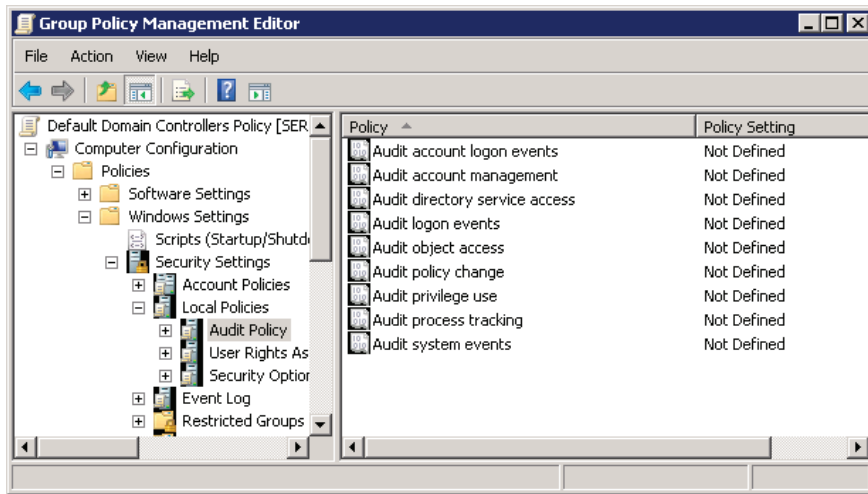
An Overview of Audit Policies

- Audit events in a category of activities
 - Access to NTFS files/folders
 - Account or object changes in Active Directory
 - Logon
 - Assignment or use of user rights
- By default, DCs audit success events for most categories
- Goal: Align audit policies with corporate security policies and reality
 - Over-auditing: logs are too big to find the events that matter
 - Under-auditing: important events are not logged
 - Tools that help you consolidate and crunch logs can be helpful



Key Points

Audit Policy configures a system to audit categories of activities. If Audit Policy is not enabled, a server will not audit those activities. The screen shot on the next page shows the Audit Policy node of a GPO expanded:



To configure auditing, you must define the policy setting. Double-click any policy setting and select the Define These Policy Settings check box. Then select whether to enable auditing of Success events, Failure events, or both.

The following table defines each audit policy and its default settings on a Windows Server 2008 domain controller.

Audit Policies

Audit Policy Setting	Explanation	Default Setting for Windows Server 2008 Domain Controllers
Audit Account Logon Events	Creates an event when a user or computer attempts to authenticate using an Active Directory account. For example, when a user logs on to any computer in the domain, an account logon event is generated.	Successful account logons are audited.

(continued)

Audit Policy Setting	Explanation	Default Setting for Windows Server 2008 Domain Controllers
Audit Logon Events	Creates an event when a user logs on interactively (locally) to a computer or over the network (remotely). For example, if a workstation and a server are configured to audit logon events, the workstation audits a user logging on directly to that workstation. When the user connects to a shared folder on the server, the server logs that remote logon. When a user logs on, the domain controller records a logon event because logon scripts and policies are retrieved from the DC.	Successful logons are audited.
Audit Account Management	Audits events, including the creation, deletion, or modification of user, group, or computer accounts and the resetting of user passwords.	Successful account management activities are audited.
Audit Directory Service Access	Audits events that are specified in the system ACL (SACL), which is seen in an Active Directory object's Properties Advanced Security Settings dialog box. In addition to defining the audit policy with this setting, you must also configure auditing for the specific object or objects using the SACL of the object or objects. This policy is similar to the Audit Object Access policy used to audit files and folders, but this policy applies to Active Directory objects.	Successful directory service access events are audited, but few objects' SACLs specify audit settings.

(continued)

Audit Policy Setting	Explanation	Default Setting for Windows Server 2008 Domain Controllers
Audit Policy Change	Audits changes to user rights assignment policies, audit policies, or trust policies.	Successful policy changes are audited.
Audit Privilege Use	Audits the use of a privilege or user right. See the explanatory text for this policy in the Group Policy Management Editor (GPME).	No auditing is performed by default.
Audit System Events	Audits system restart, shutdown, or changes that affect the system or security log.	Successful system events are audited.
Audit Process Tracking	Audits events such as program activation and process exit. See the explanatory text for this policy in the GPME.	No events are audited.
Audit Object Access	Audits access to objects such as files, folders, registry keys, and printers that have their own SACLs. In addition to enabling this audit policy, you must configure the auditing entries in objects' SACLs.	No events are audited.

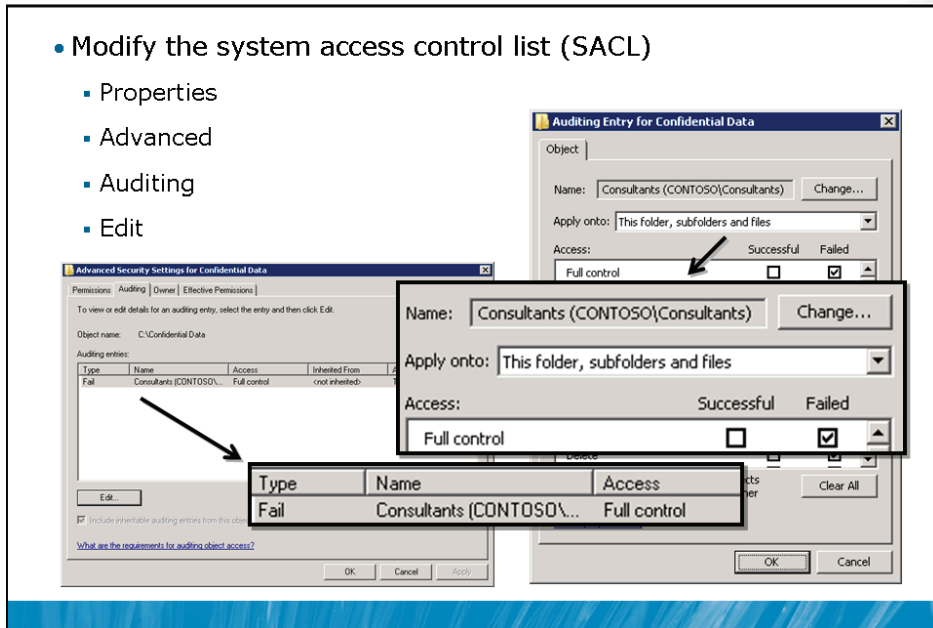
As you can see, most major Active Directory events are already audited by domain controllers, assuming that the events are successful. Therefore, the creation of a user, the resetting of a user's password, the logon to the domain, and the retrieval of a user's logon scripts are all logged.

However, not all failure events are audited by default. You might need to implement additional failure auditing based on your organization's IT security policies and requirements. Auditing failed account logon events, for example, will expose malicious attempts to access the domain by repeatedly trying to log on as a domain user account without yet knowing the account's password. Auditing failed account management events can reveal someone attempting to manipulate the membership of a security-sensitive group.

One of the most important tasks you must fulfill is to balance and align the audit policy with your corporate policies and reality. Your corporate policy might state that all failed logons and successful changes to Active Directory users and groups must be audited. That's easy to achieve in Active Directory. But how, exactly, are you going to use that information? Verbose auditing logs are useless if you don't know how or don't have the tools to manage those logs effectively. To implement auditing, you must have the business requirement to audit, a well-configured audit policy, and the tools with which to manage audited events.

Specify Auditing Settings on a File or Folder

- Modify the system access control list (SACL)
 - Properties
 - Advanced
 - Auditing
 - Edit



Key Points

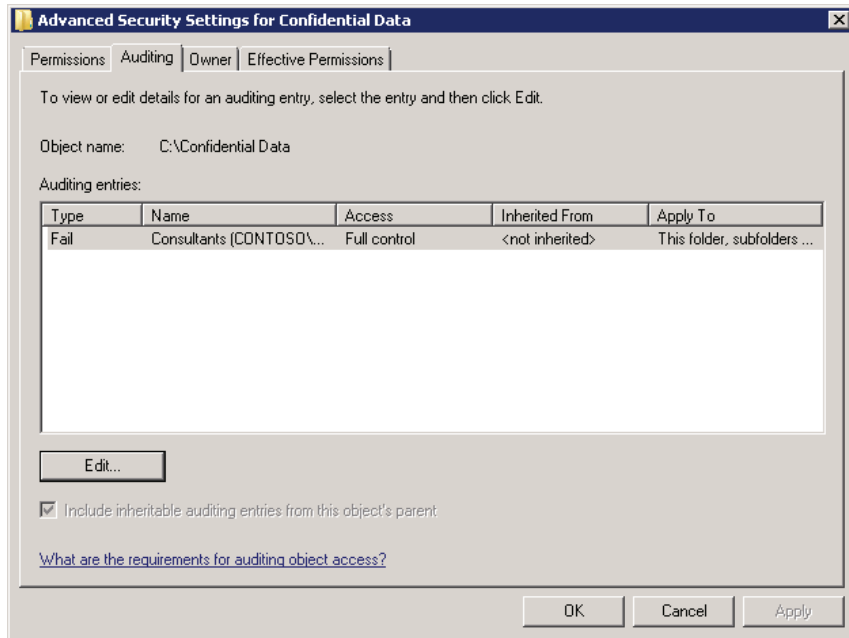
Many organizations elect to audit file system access to provide insight into resource usage and potential security issues. Windows Server 2008 supports granular auditing based on user or group accounts and the specific actions performed by those accounts. To configure auditing, you must complete three steps: specify auditing settings, enable audit policy, and evaluate events in the security log.

You can audit access to a file or folder by adding auditing entries to its system access control list (SACL).

1. Open the properties dialog box of the file or folder, and then click the **Security** tab.
2. Click the **Advanced** button.

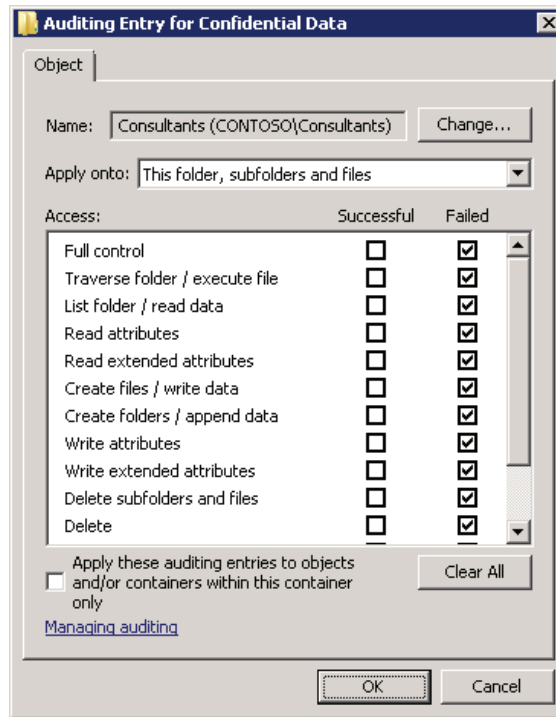
- Click the **Auditing** tab.

The Advanced Security Settings dialog box of a folder named Confidential Data is shown in the following screen shot:



- To add an entry, click the **Edit** button to open the **Auditing** tab in **Edit** mode.
- Click the **Add** button to select the user, group, or computer to audit.

6. In the **Auditing Entry** dialog box shown in the following screen shot, indicate the type of access to audit:



You are able to audit for successes, failures, or both as the specified user, group or computer attempts to access the resource by using one or more of the granular access levels.

You can audit successes for the following purposes:

- To log resource access for reporting and billing
- To monitor access that would suggest users are performing actions greater than what you had planned, indicating that permissions are too generous
- To identify access that is out of character for a particular account, which might be a sign that a user account has been breached by a hacker

Auditing failed events enables you:

- To monitor for malicious attempts to access a resource to which access has been denied.
- To identify failed attempts to access a file or folder to which a user does require access. This would indicate that the permissions are not sufficient to achieve a business requirement.

Auditing entries directs Windows to audit the successful or failed activities of a security principal (user, group, or computer) to use a specific permission. The example in the screenshot of the Auditing Entry dialog box shown previously, audits for unsuccessful attempts by users in the Consultants group to access data in the Confidential Data folder at any level. It does that by configuring an auditing entry for Full Control access. Full Control includes all the individual access levels, so this entry covers any type of access. If a Consultant group member attempts access of any kind and fails, the activity will be logged.

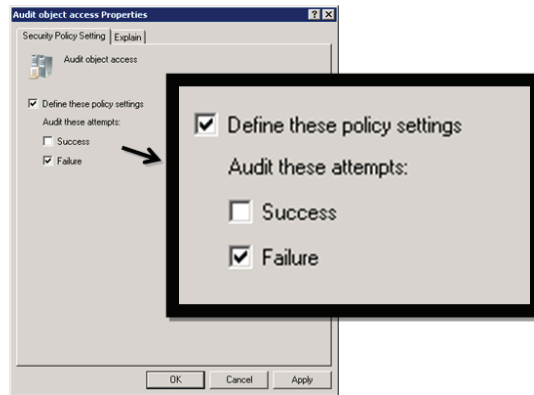
Typically, auditing entries reflect the permission entries for the object. In other words, you would configure the Confidential Data folder with permissions that prevent Consultants from accessing its contents. You would then use auditing to monitor Consultants who nonetheless attempt to access the folder. Keep in mind, of course, that a member of the Consultants group can also belong to another group that does have permission to access the folder. Because that access will be successful, the activity is not logged. Therefore, if you really are concerned about keeping users out of a folder and making sure they do not access it in any way, monitor failed access attempts; however, also audit successful access to identify situations in which a user is accessing the folder through another group membership that is potentially incorrect.



Important: Don't over-audit. Audit logs have the tendency to get quite large quite rapidly, so a golden rule for auditing is to configure the bare minimum required to achieve the business task. Specifying to audit the successes and failures on an active data folder for the Everyone group using Full Control (all permissions) would generate enormous audit logs that could affect the performance of the server and make locating a specific audited event all but impossible.

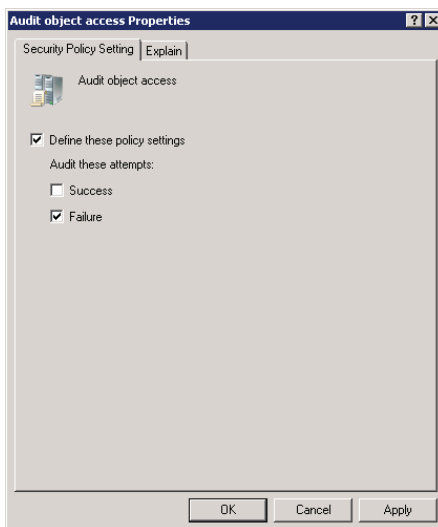
Enable Audit Policy

- Enable auditing for Object Access: Success and/or Failure
- GPO must be scoped to the server
- Success/Failure policy setting must match auditing settings (success/failure)



Key Points

Configuring auditing entries in the security descriptor of a file or folder does not, in itself, enable auditing. Auditing must be enabled by defining the Audit Object Access setting shown on the following page:



After auditing is enabled, the security subsystem begins to pay attention to the audit settings and to log access as directed by those settings.

The policy setting must be applied to the server that contains the object being audited. You can configure the policy setting in the server's local GPO or use a GPO scoped to the server.

You can define the policy then to audit Success events, Failure events, or both. The policy setting (shown above) must specify auditing of Success or Failure attempts that match the type of auditing entry in the object's SACL (shown in the previous topic). For example, to log a failed attempt by Consultants to access the Confidential Data folder, you must configure the Audit Object Access policy to audit failures, and you must configure the SACL of the Confidential Data folder to audit failures. If the audit policy audits successes only, the failure entries in the folder's SACL will not trigger logging.



Note: Making sure audit policy matches auditing entries. Remember that access that is audited and logged is the combination of the audit entries on specific files and folders and the settings in Audit Policy. If you've configured audit entries to log failures, but the policy enables only logging for successes, your audit logs will remain empty.

Evaluate Events in the Security Log

- Security Log



- Summary

- Audit Object Access policy must be enabled to audit Success or Failure
 - GPO must be scoped to the server
- SACL must be configured to audit successful or failed access
- Security Log must be examined

Key Points

After you have enabled the Audit Object Access policy setting and specified the access you want to audit, using object SACLs, the system will begin to log access according to the audit entries. You can view the resulting events in the Security log of the server. Open the Event Viewer console from Administrative Tools. Expand Windows Logs\Security.

Lab D: Audit File System Access

- Exercise 1: Configure Permissions and Audit Settings
- Exercise 2: Configure Audit Policy
- Exercise 3: Examine Audit Events

Logon information

Virtual machine	6425B-HQDC01-A	6425B-DESKTOP101-A	6425B-SERVER01-A
Logon user name	Pat.Coleman	Pat.Coleman and Mike.Danseglio	Pat.Coleman
Administrative user name	Pat.Coleman_Admin		Pat.Coleman_Admin
Password	Pa\$\$w0rd	Pa\$\$w0rd	Pa\$\$w0rd

Estimated time: 30 minutes

Scenario

In this Lab, you will configure auditing settings, enable audit policies for object access, and filter for specific events in the Security log. The business objective is to monitor a folder containing confidential data that should not be accessed by users in the Consultants group.

Exercise 1: Configure Permissions and Audit Settings

In this exercise, you will configure permissions on the Confidential Data folder to deny access to consultants. You will then enable auditing of attempts by consultants to access the folder.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Create and secure a shared folder.
3. Configure auditing settings on a folder.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A and log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
2. Start 6425B-SERVER01-A but do not log on.
3. Start 6425B-DESKTOP101-A but do not log on.
4. Wait for all virtual machines to complete startup before continuing to the next task.

► Task 2: Create and secure a shared folder

1. Switch to HQDC01.
2. Run **Active Directory Users and Computers** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
3. In the **Groups\Role** OU, create a new global security group named **Consultants**.
4. Add **Mike.Danseglio** to the **Consultants** group.
5. Create a new folder in **\\server01\c\$\data** called **Confidential Data**.
6. Configure NTFS permissions that deny the **Consultants** group all access to the folder.

► **Task 3: Configure auditing settings on a folder**

- Configure auditing settings on the **Confidential Data** folder to audit for any failed access by the Consultants group.

Exercise 2: Configure Audit Policy

In this exercise, you will enable auditing of file system access on file servers using Group Policy.

The main tasks for this exercise are as follows:

- Enable auditing of file system access using Group Policy.

► Task 1: Enable auditing of file system access by using Group Policy

1. Run **Group Policy Management** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Create a new GPO named **File Server Auditing**.
3. Configure the GPO to audit for failed object access.
4. Link the GPO to the Servers\File OU.

Results: After this exercise, you will have configured for auditing of failed access to file system objects on servers in the Servers\File OU.

Exercise 3: Examine Audit Events

In this exercise, you will generate audit failure events and then examine the resulting security event log messages.

The main tasks for this exercise are as follows:

1. Generate audit events.
2. Examine audit event log messages.

► Task 1: Generate audit events

1. Log on to SERVER01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
2. Run the Command Prompt as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
3. Refresh **Group Policy** to apply the new auditing settings by executing the command **gpupdate.exe /force**.
4. Log off of SERVER01.
5. Log on to DESKTOP101 as **Mike.Danseglio** with the password **Pa\$\$w0rd**.
6. Attempt to open **\\server01\data\Confidential Data**. You will receive an Access Denied message.

► Task 2: Examine audit event log messages

1. Switch to SERVER01.
2. Run **Event Viewer** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
3. Locate the audit failure events related to Mike Danseglio's access to the **Confidential Data** folder.

Question: What is the Task Category for the event? What is the Event ID? What type of access was attempted?

Results: After this exercise, you will have validated the auditing of failed access to the Confidential Data folder by members of the Consultants group.



Note: After completing this exercise, turn off all virtual machines and discard undo disks.

Lab Review Questions

Question: What are the three major steps required to configure auditing of file system and other object access?

Question: What systems should have auditing configured? Is there a reason not to audit all systems in your enterprise? What types of access should be audited, and by whom should they be audited? Is there a reason not to audit all access by all users?

Module 8

Secure Administration

Contents:

Lesson 1: Delegate Administrative Permissions	8-4
Lab A: Delegate Administration	8-25
Lesson 2: Audit Active Directory® Changes	8-33
Lab B: Audit Active Directory Changes	8-39

Module Overview

- Delegate Administrative Permissions
- Audit Active Directory Administration

"Security is Job #1" at most enterprises today, and just as organizations are working to remove unnecessary administrative privileges that have, in the past, been assigned to users on their workstations, organizations are also striving to lock down and manage the privileges given to administrators themselves. If you want to manage the security of Active Directory administration, you must understand how to delegate specific administrative tasks and how to audit changes that are made to the directory.

Objectives

After completing this module, you will be able to:

- Describe the business purpose of delegation.
- Assign permissions to Active Directory objects using the security editor user interfaces and the Delegation Of Control Wizard.
- View and report permissions on Active Directory objects by using user-interface and command-line tools.

- Reset the permissions on an object to its default.
- Describe the relationship between delegation and organizational unit (OU) design.
- Configure Directory Service Changes auditing.
- Specify auditing settings on Active Directory objects.
- Identify event log entries created by Directory Access auditing and Directory Service Changes auditing.

Lesson 1

Delegate Administrative Permissions

- Understand Delegation
- View of the ACL of an Active Directory Object
- Property Permissions, Property Sets, Control Access Rights, and Object Permissions
- Demonstration: Assign a Permission by Using the Advanced Security Settings Dialog Box
- Understand and Manage Permissions with Inheritance
- Demonstration: Delegate Administrative Tasks with the Delegation of Control Wizard
- Report and View Permissions
- Remove or Reset Permissions on an Object
- Understand Effective Permissions
- Design an OU Structure to Support Delegation

In previous modules, you've learned how to create users, groups, computers, and OUs, and how to access the properties of those objects. Your ability to perform those actions was dependent on your membership in the Administrators group of the domain. You would not want every user on your help desk team to be a member of the domain's Administrators group just to reset user passwords and unlock user accounts. Instead, you should enable the help desk, and each role in your organization, to perform the tasks that are required of the role, and no more. In this lesson, you'll learn how to delegate specific administrative tasks within Active Directory. This is achieved by changing the access control lists (ACLs) on Active Directory objects.

Objectives

After completing this lesson, you will be able to:

- Describe the business purpose of delegation.
- Assign permissions to Active Directory objects using the security editor user interfaces and the Delegation of Control Wizard.

- View and report permissions on Active Directory objects by using user-interface and command-line tools.
- Reset the permissions on an object to its default.
- Describe the relationship between delegation and OU design.

Understand Delegation

- Simple example
 - The help desk needs to be able to reset passwords for users and force users to change the temporary password at next logon
 - The help desk cannot create or delete users: delegation is *specific* or *granular*
 - The help desk can reset passwords of normal user accounts, not administrative or service accounts: delegation has a *scope*
- Every Active Directory object has permissions.
 - Permissions are called access control entries (ACEs)
 - ACEs are on the discretionary access control list (DACL)
 - The DACL is part of the object's access control list (ACL)
 - The ACL also contains the system access control list (SACL)
 - The SACL specifies (among other things) auditing settings

Key Points

In most organizations, there is more than one administrator, and as organizations grow, administrative tasks are often distributed to various administrators or support organizations. For example, at many organizations, the help desk is able to reset user passwords and unlock the accounts of users who are locked out. This capability of the help desk is a delegated administrative task.

The help desk cannot usually create new user accounts, but it can make specific changes to existing user accounts. The capability that is delegated is *specific*, or *granular*.

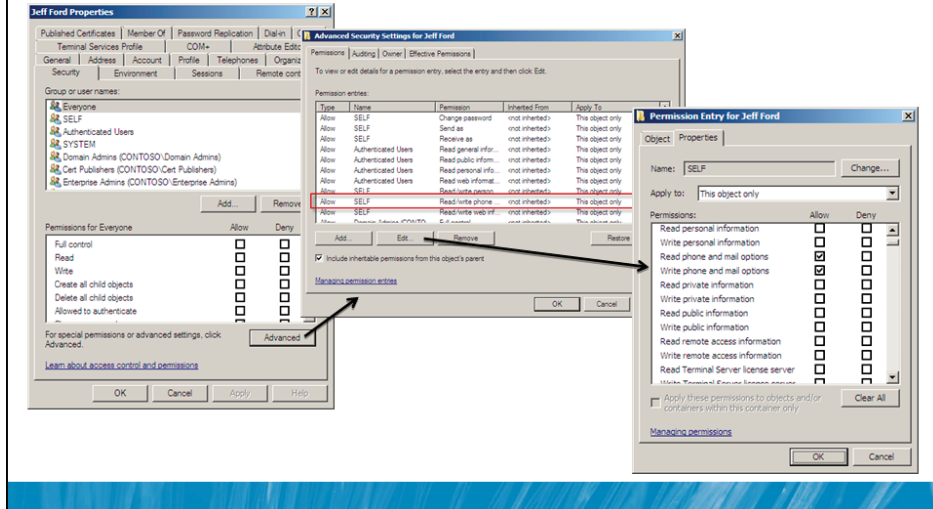
Continuing the example, in most organizations, the help desk's ability to reset passwords would apply to normal user accounts, but not to accounts used for administration or to service accounts. The delegation is thus said to be *scoped* to standard user accounts.

All Active Directory objects, such as the users, computers, and groups you created in the previous module, can be secured by using a list of permissions. So you could give your help desk permission to reset passwords on user objects. The permissions on an object are called access control entries (ACEs), and they are assigned to users, groups, or computers (called *security principals*). ACEs are saved in the object's discretionary access control list (DACL). The DACL is a part of the object's access control list (ACL), which also contains the system access control list (SACL) that includes auditing settings. This may sound familiar to you if you have studied the permissions on files and folders—the terms and concepts are identical.

The delegation of administrative control, also called the *delegation of control* or just *delegation*, simply means assigning permissions that manage access to objects and properties in Active Directory. Just as you can give a group the ability to change files in a folder, you can give a group the ability to reset passwords on user objects.

View the ACL of an Active Directory Object

- Ensure Advanced Features are enabled in the View menu
- Properties → Security → Advanced → Edit



Key Points

To view the ACL on an object:

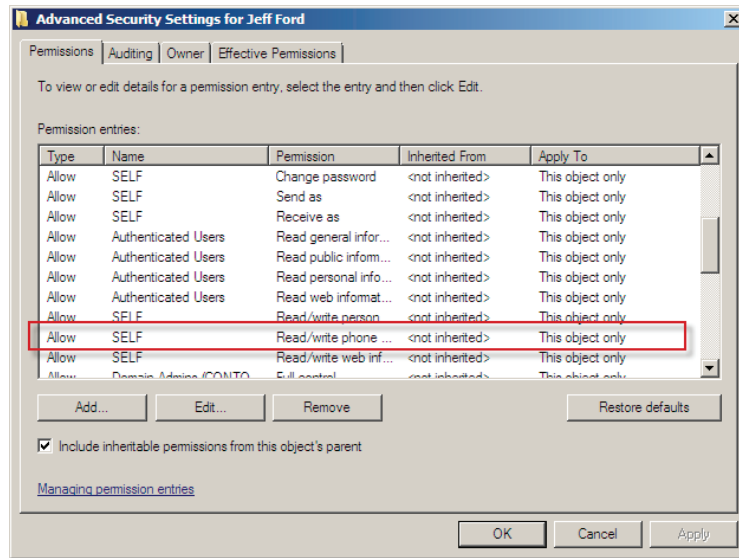
1. Open the **Active Directory Users and Computers** snap-in.
2. Click the **View** menu and select **Advanced Features**.
3. Right-click an object and choose **Properties**.
4. Click the **Security** tab.

If Advanced Features is not enabled, you will not see the Security tab in an object's Properties dialog box.

5. Click the **Advanced** button.

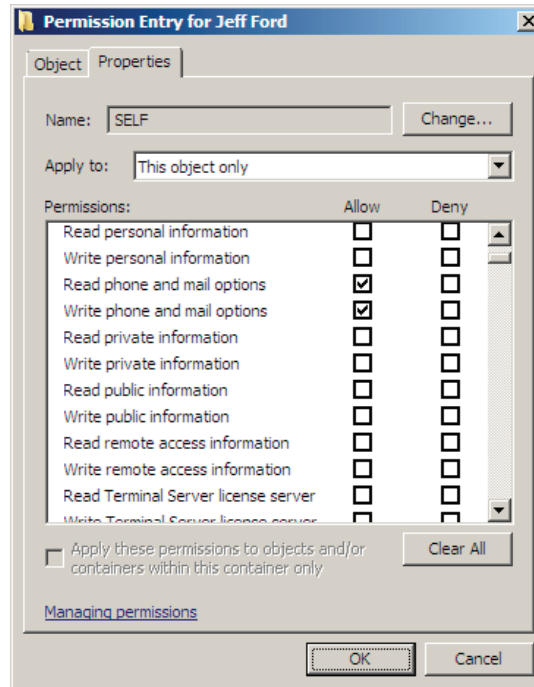
The Security tab shows a very high-level overview of the security principals that have been given permissions to the object, but in the case of Active Directory ACLs, the Security tab is rarely detailed enough to provide the information you need to interpret or manage the ACL. You should always click Advanced to open the Advanced Security Settings dialog box.

The Advanced Security Settings dialog box appears, shown below.



The Permissions page of the Advanced Security Settings dialog box shows the DACL of the object. You can see in the screen shot that ACEs are summarized on a line of the Permission entries list. In this dialog box, you are not seeing the granular ACEs of the DACL. For example, the permission entry that is highlighted above is actually comprised of two ACEs.

6. To see the granular ACEs of a permission entry, select the entry and click **Edit**. The Permission Entry dialog box appears, detailing the specific ACEs that make up the entry.



Property Permissions, Property Sets, Control Access Rights, and Object Permissions

- Permissions can allow (or deny) changes to a specific *property*
 - Example: Allow Write Mobile Number
- Permissions can allow (or deny) changes to a *property set*
 - Example: Allow Write Phone and Mail Options
 - Bundle of properties: Phone and mail properties
 - One-click management of permissions for related properties
- Permissions can allow (or deny) *control access rights*
 - Allow Change Password: Must enter old password, then new
 - Allow Reset Password: Enter new password (do not need old)
- Permissions can allow (or deny) changes to the *object*
 - Allow Modify Permissions
 - Allow Create Computer Objects

Key Points

The DACL of an object allows you to assign permissions to specific properties of an object. For example, you can allow (or deny) permission to change phone and email options. This is, in fact, not just one property, it is a property set that includes multiple specific properties. Property sets make it easier to manage permissions to commonly used collections of properties. But you could get even more granular and allow or deny permission to change just the mobile telephone number, or just the home street address.

Permissions can also be assigned to manage control access rights, which are actions such as changing or resetting a password. The difference between those two control access rights is important to understand. If you have the right to change a password, you must know and enter the current password before making the change. If you have the right to reset a password, you are not required to know the previous password.

Finally, permissions can be assigned to objects. For example, the ability to change permissions on an object is controlled by the Allow Modify Permissions ACE. Object permissions also control whether you are able to create child objects. For example, you might give your desktop support team permissions to create computer objects in the Client Computers OU. The Allow Create Computer Objects ACE would be assigned to the desktop support team at the OU.

The type and scope of permissions are managed using the Object tab and the Properties tab, and the Apply To drop-down lists on each tab.

Demonstration: Assign a Permission Using the Advanced Security Settings Dialog Box

- In this demonstration, we will delegate the help desk permission to change the password for Jeff Ford

Key Points

Imagine a scenario in which you want to allow the help desk to change the password on Jeff Ford's user account, and *only* Jeff Ford's account. In this section, you will learn to do it the most complicated way first: by assigning the ACE on the DACL of the user object. Later, you'll learn how to perform the delegation using the Delegation Of Control Wizard for the entire OU of users, and you'll see why this latter practice is recommended.

Demonstration Steps

1. Start 6425B-HQDC01-A log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
2. Click **Start>Administrative Tools** and run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
3. Click the **View** menu and select **Advanced Features**.
4. Right-click an object and choose **Properties**.

5. Click the **Security** tab.
6. Click the **Advanced** button.
7. Click the **Add** button.

If you have User Account Control enabled, you may need to click Edit, and perhaps enter administrative credentials, before the Add button will appear.

8. In the **Select** dialog box, select the security principal to which permissions will be assigned.

It is an important best practice to assign permissions to groups, not to individual users.

In this example, you would select your Help Desk group and press ENTER. The Permission Entry dialog box appears.

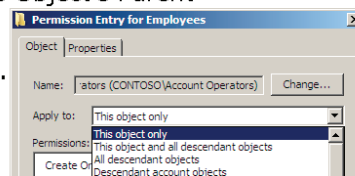
9. Configure the permissions you want to assign.

For our example, on the Object tab, scroll down the list of Permissions and select Allow::Reset Password.

10. Click **OK** to close each dialog box.

Understand and Manage Permissions with Inheritance

- Child objects inherit the permissions of the parent organizational unit or container
 - Top-level OUs inherit permissions from the domain
- By default, each new object is created with the option Include Inheritable Permissions From This Object's Parent
- Not every permission is inheritable. Inheritance of a permission is scoped.
- There are three ways to modify the effects of inheritance:
 - Turn off inheritance on child object
 - Deselect Include Inheritable Permissions...
 - Assign an explicit permission
 - Explicit permissions override inherited permissions
 - Change the scope of inheritance on the parent (Apply To)



Key Points

You can imagine that assigning the help desk permission to reset passwords for each individual user object would be a nightmare. Luckily, you don't have to, and in fact it's a terrible practice to assign permissions to individual objects in Active Directory. Instead, you will assign permissions to organizational units. The permissions you assign to an OU will be inherited by all objects in the OU. So, if you give the help desk permission to reset passwords for user objects, and you attach that permission to the OU that contains your users, all user objects within that OU will inherit that permission. With one step, you'll have delegated that administrative task.

Inheritance is an easy concept to understand. Child objects inherit the permissions of the parent container or OU. That container or OU in turn inherits its permissions from its parent container, OU or, if it is a first-level container or OU, from the domain itself. The reason child objects inherit permissions from their parents is that, by default, each new object is created with the Include Inheritable Permissions From This Object's Parent option enabled.

However, note that as the option indicates, only inheritable permissions will be inherited by the child object. Not every permission is inheritable. For example, the permission to reset passwords, when assigned to an OU, would not be inherited by group objects because group objects do not have a password attribute. So inheritance can be scoped to specific object classes: passwords are applicable to user objects, not groups. Additionally, you can use the Apply To box of the Permission Entry dialog box to scope the inheritance of a permission. The conversation can start to get very complicated. What you should know is that, by default, new objects inherit inheritable permissions from their parent object—usually an OU or container.

What if the permission that is being inherited is not appropriate? Three things can be done to modify the permissions that a child object is inheriting:

- First, you can disable inheritance by deselecting the Include Inheritable Permissions From This Object's Parent option in the Advanced Security Settings dialog box. When you do, the object will no longer inherit any permissions from its parent—all permissions will be explicitly defined for the child object. This is generally not a good practice, as it creates an exception to the rule that is being created by permissions of parent containers.
- The second option is to allow inheritance, but to override the inherited permission with a permission assigned specifically to the child object—an explicit permission. Explicit permissions always override permissions that are inherited from parent objects. This has an important implication: an explicit permission that allows access will actually override an inherited permission that denies the same access. If that sounds counterintuitive to you, it is not: the rule (Deny) is being defined by a parent, but the child object has been configured to be an “exception” (Allow).
- Third, you can change the scope of inheritance on the parent permission itself by changing the option in the Apply To drop-down list in the Permission Entry dialog box. In most cases, this is the best practice. What you are doing, in effect, is defining the security policy in the form of the ACL more accurately at its source, rather than trying to override it further down the tree.

Demonstration: Delegate Administrative Tasks with the Delegation of Control Wizard

- In this demonstration, we will use the Delegation Of Control Wizard to assign permissions to the AD_User Account_Support group to
 - Reset passwords
 - Force users to change passwords at the next logonon the User Accounts OU.

Key Points

You've seen the complexity of the DACL, and you've probably gleaned that managing permissions using the Permission Entry dialog box is not a simple task. Luckily, the best practice is not to manage permissions using the security interfaces, but rather to use the Delegation Of Control Wizard. The following procedure details the use of the wizard.

Demonstration Steps

1. On HQDC01 click **Start >Administrative Tools** and run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$w0rd**.

2. Right-click the node (domain or OU) for which you want to delegate administrative tasks or control, and choose **Delegate Control**.

In our example, you would select the OU that contains your users.

The Delegation of Control Wizard appears, to guide you through the required steps.

3. Click **Next**.

You will first select the administrative group to which you are granting privileges.

4. In the **Users or Groups** page, click the **Add** button.

5. Use the **Select** dialog box to select the group, and click **OK**.

6. Click **Next**.

You will next specify the specific task you wish to assign to that group.

7. On the **Tasks to Delegate** page, select the task.

In our example, you would select Reset User Passwords and Force Password Change at Next Logon.

8. Click **Next**.

9. Review the summary of the actions that have been performed, and click **Finish**.

The Delegation of Control Wizard applies the ACEs that are required to enable the selected group to perform the specified task.

Report and View Permissions

- Use the Advanced Security Settings dialog box
- Use DSACLs (dsac ls.exe)
 - `dsac ls ObjectDN`
 - Example:
`dsac ls "ou=User Accounts,dc=contoso,dc=com"`

Key Points

There are several other ways to view and report permissions when you need to know who can do what. You've already seen that you can view permissions on the DACL using the Advanced Security Settings and Permission Entry dialog boxes.

DSACLs (dsac ls.exe) is also available as a command-line tool that reports on directory service objects. If you type the command followed by the distinguished name of an object you will see a report of the object's permissions. For example, this command will produce a report of the permissions associated with the User Accounts OU:

```
dsac ls.exe "ou=User Accounts,dc=contoso,dc=com"
```

DSACLs can also be used to set permissions—to delegate. Type `dsac ls.exe /?` for help regarding the syntax and utilization of DSACLs.

Remove or Reset Permissions on an Object

- No "undelegate" command
- Remove permissions manually in the Advanced Security Settings and Permission Entry dialog boxes
- Reset permissions to default with Active Directory Users and Computers
 - Advanced Security Settings dialog box → Restore Defaults
 - Applies default ACL defined in the schema for the object class
- Reset permissions to default with DSACLs
 - `dsacls ObjectDN /s /t`
 - Example:
`dsacls "ou=User Accounts,dc=contoso,dc=com" /s /t`

Key Points

How do you remove or reset permissions that have been delegated? Unfortunately, there is no "undelegate" command. You must do one of the following:

- Open the Advanced Security Settings and Permission Entry dialog boxes to remove permissions.
- If you want to reset the permissions on the object back to the defaults, open the Advanced Security Settings dialog box and click Restore Defaults. The default permissions are defined by the Active Directory schema for the class of object. After you've restored the defaults, you can reconfigure the explicit permissions you want to add to the DACL.
- DSACLs also provides the /s switch to reset permissions to the schema-defined defaults, and the /t switch to make the change for the entire "tree"—the object and all of its child objects. For example, to reset permissions on the People OU and all of its child OUs and objects, you would enter:

```
dsacls "ou=User Accounts,dc=contoso,dc=com" /s /t
```


Understand Effective Permissions

- Permissions assigned to you and your groups cumulate
 - Best practice is to assign permissions to groups, not to individual users
- In the event of conflicts
 - Deny permissions override Allow permissions
 - Explicit permissions override Inherited permissions
 - Explicit Allow overrides Inherited Deny
- Evaluating effective permissions
 - The Effective Permissions tab: helpful but not very granular
 - Manual analysis
 - Third-party tools
 - Role-based management

Key Points

Effective permissions are the resulting permissions for a security principal, such as a user or group, based on the cumulative effect of each inherited and explicit ACE. Your ability to reset a user's password, for example, may be due to your membership in a group that was allowed Reset Password permission on an OU several levels above the user object. The inherited permission assigned to a group to which you belong resulted in an effective permission of Allow::Reset Password. Your effective permissions can be complicated, when you consider Allow and Deny permissions, explicit and inherited ACEs, and the fact that you may belong to multiple groups, each of which may be assigned different permissions.

Permissions, whether assigned to your user account or to a group to which you belong, are equivalent. In the end, an ACE applies to you, the user. The best practice is to manage permissions by assigning them to groups, but it is also possible to assign ACEs to individual users or computers. A permission that has been assigned directly to you, the user, is neither more important nor less important than a permission assigned to a group to which you belong.

Permissions that allow access (*Allow permissions*) are cumulative. When you belong to several groups, and those groups have been granted permissions that allow a variety of tasks, you will be able to perform all of the tasks assigned to all of those groups, as well as tasks assigned directly to your user account.

Permissions that deny access (*Deny permissions*) override equivalent Allow permissions. If you are in one group that has been allowed the permission to reset passwords, and another group that has been denied permission to reset passwords, the Deny permission will prevent you from resetting passwords.



Note: It is generally unnecessary to assign Deny permissions: If you simply do not assign an Allow permission, users cannot perform the task. Before assigning a Deny permission, check to see if you could achieve your goal by removing an Allow permission instead. Use Deny permissions rarely, and thoughtfully.

Each permission is granular. Just because you've been denied the ability to reset passwords, you may still have the ability, through other Allow permissions, to change the user's logon name or email address.

Finally, you learned earlier in this lesson that child objects inherit the inheritable permissions of parent objects by default, and that explicit permissions can override inheritable permissions. This means that an explicit Allow permission will actually override an inherited Deny permission.

Unfortunately, the complex interaction of user, group, explicit, inherited, Allow, and Deny permissions can make evaluating effective permissions a bit of a chore. There is an Effective Permissions tab in the Advanced Security Settings of an Active Directory object, but the tab is practically useless: it does not expose enough permissions to provide the kind of detailed information you will require. You can use the permissions reported by the DSACLs command, or on the Permissions tab of the Advanced Security Settings dialog box, to begin evaluating effective permissions, but it will be a manual task.

Additional Reading

- The best way to manage delegation in Active Directory is through role-based access control. Although this approach will not be covered on the certification exam, it is well worth understanding for real-world implementation of delegation. See the *Windows® Administration Resource Kit: Productivity Solutions for IT Professionals* by Dan Holme (Microsoft® Press, 2008) for more information.

Design an OU Structure to Support Delegation

- OUs perform three functions:
 - **Delegation.** Scope permissions for administrative tasks in Active Directory
 - **Configuration.** Scope the application of Group Policy objects (GPOs)
 - **Presentation.** Organize and present objects in a logical manner
- Best practice Active Directory OU design:
 1. Create OUs to scope delegation.
 - Top/higher-level OUs reflect the *administrative model*
 2. Then divide those OUs to provide scopes for GPOs.
 - If there isn't a way to scope a GPO by linking it to an OU in your design, link the GPO higher and use security group filtering to manage its scope.
 3. Then, if necessary, create sub-OUs to organize and present.
 - Better yet, use Saved Queries to organize and present.

Key Points

OUs are, as you now know, administrative containers. They contain objects that share similar requirements for administration, configuration, and visibility. You now understand the first of those requirements: administration. Objects that will be administered the same way, by the same administrators, should be contained within a single OU. By placing your users in a single OU perhaps called *User Accounts*, you could delegate the help desk permission to change all users' passwords by assigning one permission to one OU. Any other permissions that affect what an administrator can do to a user object would be assigned at the User Accounts OU. For example, you might allow your Human Resources managers to disable user accounts in the event of an employee's termination. You would delegate that permission, again, to the User Accounts OU.

Remember that administrators should be logging on to their systems with user credentials and launching administrative tools with the credentials of a secondary account that has appropriate permissions to perform administrative tasks. Those secondary accounts are the administrative accounts of the enterprise. It is not appropriate for the front-line help desk to be able to reset passwords on such privileged accounts, and you probably would not want Human Resources managers to disable administrative accounts. Therefore, administrative accounts should be administered differently than “normal” user accounts. That’s why you would have a separate OU, such as *Admins*, for administrative user objects. That OU would be delegated quite differently than the User Accounts OU.

Similarly, you might delegate to the desktop support team the ability to add computer objects to an OU called *Client Computers*, which contains your desktops and laptops, but not to the Servers OU, where only your Server Administration group has permissions to create and manage computer objects.

The primary role of OUs is to efficiently scope delegation—to apply permissions to objects and sub-OUs. When you design an Active Directory environment, you always begin by designing an OU structure that will make delegation efficient—a structure that reflects the administrative model of your organization. Rarely does object administration in Active Directory look like your organizational chart. Typically, all normal user accounts are supported the same way, by the same team—so user objects are often found in a single OU or a single OU branch. Quite often, an organization that has a centralized help desk function to support users will also have a centralized desktop support function, in which case all client computer objects would be within a single OU or single OU branch. But if desktop support is decentralized, it would be likely the Client Computers OU divided into sub-OUs representing geographic locations, where each location was delegated to allow the local support team to add computer objects to the domain in that location.

Design OUs first to enable the efficient permissioning (delegation) of objects in the directory. Once you have achieved that design, you can refine the design to facilitate the configuration of computers and users through Group Policy.

Additional Reading

- See the *Windows Administration Resource Kit: Productivity Solutions for IT Professionals* by Dan Holme (Microsoft Press, 2008) for much more detail regarding OU design.

Lab A: Delegate Administration

- Exercise 1: Delegate permission to create and support user accounts
- Exercise 2: View delegated permissions
- Exercise 3: Remove and reset permissions

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

The enterprise security team at Contoso has asked you to lock down the administrative permissions delegated to support personnel.

Exercise 1: Delegate Permission to Create and Support User Accounts

In this exercise, you will delegate to the help desk permission to unlock user accounts, reset passwords, and require users to change passwords at the next logon. This permission will scope only to standard user accounts and will not allow the help desk to change passwords of administrative accounts. You will also delegate permission to the User Account Admins group to create and delete user accounts, as well as full control over user accounts.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Create security groups for role-based management.
3. Delegate control of user support with the Delegation of Control Wizard.
4. Delegate permission to create and delete users with the Access Control List Editor interface.
5. Validate the implementation of delegation.

► Task 1: Prepare for the lab 1. Start 6425B-HQDC01-A

1. Log on as **Pat.Coleman** with the password **Pa\$\$w0rd**.
2. Run **D:\Labfiles\Lab08a\Lab08a_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

► Task 2: Create security groups for role-based management

1. Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. In the **Groups\Role** OU, create the following role groups:
 - **Help Desk** (global security group)
 - **User Account Admins** (global security group)

3. Add the following users' *administrative accounts* to the **Help Desk** group. Be careful not to add the users' standard, non-privileged account.
 - **Aaron Painter**
 - **Elly Nkya**
 - **Julian Price**
 - **Holly Dickson**
4. Add the following users' *administrative accounts* to the **User Account Admins** group. Be careful not to add the users' standard, non-privileged account.
 - **Pat Coleman**
 - **April Meyer**
 - **Max Stevens**
5. In the **Admins\Admin Groups\AD Delegations** OU, create the following administrative access management groups:
 - **AD_User Accounts_Support** (domain local security group).
 - **AD_User Accounts_Full Control** (domain local security group).
6. Add the **Help Desk** as a member of **AD_User Accounts_Support**.
7. Add **User Account Admins** as a member of **AD_User Accounts_Full Control**.

► **Task 3: Delegate control of user support with the Delegation Of Control Wizard**

- Right-click the **User Accounts** OU and then click **Delegate Control**. Delegate to the **AD_User Accounts_Support** group the right to reset user passwords and force users to change passwords at next logon.

► **Task 4: Delegate permission to create and delete users with the Access Control List Editor interface**

1. Turn on the **Advanced Features** view of the **Active Directory Users and Computers** snap-in.
2. Right-click the **User Accounts** OU, and then click **Properties**. Click the **Security** tab, and then click **Advanced**.
3. Add permissions that give **AD_User Accounts_Full Control** the ability to create and delete users, and also gives the group full control over user objects. Be careful to limit the **Full Control** permission to user objects only.

► **Task 5: Validate the implementation of delegation**

1. Close Active Directory Users and Computers.
2. Run **Active Directory Users and Computers** as an administrator, with the username **Aaron.Painter_Admin** and the password **Pa\$\$w0rd**.
3. Confirm that you can reset the password for **Jeff Ford**, in the **Employees** OU, and that you can force him to change his password at the next logon.
4. Confirm that you cannot disable Jeff Ford's account.
5. Confirm that you cannot reset the password for **Pat Coleman (Admin)** in the **Admin Identities** OU.
6. Close Active Directory Users and Computers.
7. Run **Active Directory Users and Computers** as an administrator, with the username **April.Meyer_Admin** and the password **Pa\$\$w0rd**.
8. Confirm that you can create a user account in the **Employees** OU by creating an account with your own first and last name, the user name First.Last, and the password **Pa\$\$w0rd**.
9. Close Active Directory Users and Computers.

Results: After this exercise, you will have delegated to the help desk permission to unlock user accounts, reset passwords, and force users to change passwords at next logon, through the help desk's membership in the AD_User Accounts_Support group. You have also delegated full control of user objects to User Account Admins, through its membership in the AD_User Accounts_Full Control group. And you have tested both delegations to validate their functionality.

Exercise 2: View Delegated Permissions

In this exercise you will view, report, and evaluate the permissions that have been assigned to Active Directory objects.

The main tasks for this exercise are as follows:

1. View permissions in the Access Control List Editor interfaces.
2. Report permissions using DSACLs.
3. Evaluate effective permissions.

► Task 1: View permissions in the Access Control List Editor interfaces

1. Run **Active Directory Users and Computers** as an administrator, with the username **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Right-click the **User Accounts** OU, and then click **Properties**. Click the **Security** tab, and then click **Advanced**.
3. Sort so that permissions are displayed according to the group to which they are assigned.

Question: How many permission entries were created for the AD_User Accounts_Support group by the Delegation Of Control Wizard? Is it easy to tell what permissions were assigned in the Permission Entries list? List the permissions assigned to AD_User Accounts_Support.

► **Task 2: Report permissions using DSACLs**

- From the command prompt, use DSACLs to report the permissions assigned to the **User Accounts** OU. Type the command:

```
dsacl "ou=User Accounts,dc=contoso,dc=com"
```

and then press ENTER.

Question: What permissions are reported for AD_User Accounts_Support by the DSACLs command?

► **Task 3: Evaluate effective permissions**

1. Right-click the **User Accounts** OU, and then click **Properties**. Click the **Security** tab, and then click **Advanced**.
2. Using the **Advanced Security Settings** dialog box, evaluate the **Effective Permissions** for **April.Meyer_Admin**. Locate the permissions that allow her to create and delete users.

Question: Do you see the Reset Password in this list?

3. In the **Employees** OU, right-click the user account for **Aaron Lee**, and then click **Properties**. Click the **Security** tab, and then click **Advanced**.
4. Using the **Advanced Security Settings** dialog box, evaluate the **Effective Permissions** for **Aaron.Painter_Admin**. Locate the permissions that allow him to reset the password for **Aaron Lee**.

Results: After this exercise, you will have confirmed that the permissions you assigned in the previous exercise were applied successfully.

Exercise 3: Remove and Reset Permissions

In this exercise, you will remove delegated permissions and will reset an OU to its schema-defined default ACL.

The main tasks for this exercise are as follows:

1. Remove permissions assigned to AD_User Accounts_Support.
2. Reset the User Accounts OU to its default permissions.

► **Task 1: Remove permissions assigned to AD_User Accounts_Support**

1. Right-click the **User Accounts** OU, and then click **Properties**. Click the **Security** tab, and then click **Advanced**.
2. Sort so that permissions are displayed according to the group to which they are assigned.
3. Remove the permissions assigned to **AD_User Accounts_Support**.

► **Task 2: Reset the User Accounts OU to its default permissions**

1. Right-click the **User Accounts** OU, and then click **Properties**. Click the **Security** tab, and then click **Advanced**.
2. Click **Restore defaults**, and then click **Apply**.

Question: What do you achieve by clicking Reset To Default? What permissions remain?

Results: After this exercise, you will have reset the permissions on the User Accounts OU to its schema-defined defaults.



Note: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in the subsequent lab.

Lab Review Questions

Question: How does Active Directory Users and Computers indicate to you that you do not have permissions to perform a particular administrative task?

Question: When you evaluated effective permissions for April Meyer on the User Accounts OU, why didn't you see permissions such as Reset Password in this list? Why did the permission appear when you evaluated effective permissions for Aaron Painter on Aaron Lee's user account?

Question: Does Windows make it easy to answer the questions, "Who can reset user passwords?" and "What can XXX do as an administrator?"

Question: What is the benefit of a two-tiered, role-based management group structure when assigning permissions in Active Directory?



Note: Role-based management is a big topic, and there are other aspects of role-based management, including discipline and auditing, that are required to ensure that the members of a group such as AD_User Accounts_Support have the permissions they are supposed to have, and no other permissions, and that no other users or groups have been delegated the same permissions.

Question: What is the danger of resetting the ACL of an OU back to its schema-defined default?

Lesson 2

Audit Active Directory Administration

- Enable Audit Policy
- Specify Auditing Settings for Directory Service Changes
- View Audited Events in the Security Log

Just as auditing file and folder access allows you to log attempts to access those types of objects, the Audit Directory Service Access policy allows you to log attempts to access objects in Active Directory. Windows Server 2008 introduces another class of auditing for Active Directory: Directory Service Changes.

Objectives

After completing this lesson, you will be able to:

- Configure audit policy to enable Directory Service Changes auditing.
- Specify auditing settings on Active Directory objects.
- Identify event log entries created by Directory Access auditing and Directory Service Changes auditing.

Enable Audit Policy

- **Directory Service Access**
 - Same policy as in Windows Server® 2003
 - By default, configured to audit Success events
 - Event log entry says "a change was made to this object"
 - Difficult to identify what attribute was changed
 - Impossible to know old/new value of attribute
- **Directory Service Changes**
 - Identifies the object, the attribute, and the old/new values
 - Not enabled by default
 - Enable from the Command Prompt:
`auditpol /set /subcategory:"directory service changes" /success:enable`

Key Points

Just as the Audit Object Access policy allows you to log attempts to access objects such as files and folders, the Audit Directory Service Access policy allows you to log attempts to access objects in Active Directory. The same basic principles apply. You configure the policy to audit Success or Failure. You then configure the SACL of the Active Directory object to specify the types of access you want to audit.

As an example, if you want to monitor changes to the membership of a security-sensitive group, such as Domain Admins, you can enable the Audit Directory Service Access policy to audit Success events. You can then open the SACL of the Domain Admins group and configure an auditing entry for successful modifications of the group's Members attribute. In fact, in Windows Server 2008, the default configuration is to audit Success events for Directory Service Access, and to audit all changes to the Domain Admins group!

In Windows Server 2003 and Windows 2000 Server, you could audit directory service access and you would be notified that an object, or the property of an object, had been changed, but you could not identify the previous and new values of the attribute that had changed. For example, an event could be logged indicating that a particular user changed an attribute of Domain Admins, but you could not easily identify which attribute was changed, and there was no way to determine from the audit log exactly what change was made to that attribute.

Windows Server 2008 adds an auditing category called Directory Service Changes. The important distinction between Directory Service Changes and Directory Service Access is that with Directory Service Changes auditing, you can identify the previous and current values of a changed attribute.

Directory Service Changes is not enabled in Windows Server 2008 by default. Instead, Directory Service Access is enabled to mimic the auditing functionality of previous versions of Windows. To enable auditing of successful Directory Service Changes, open a command prompt on a domain controller and enter this command:

```
auditpol /set /subcategory:"directory service changes" /success:enable
```

Although you can use the preceding command to enable Directory Service Changes auditing in a lab and explore the events that are generated, don't implement this in a domain until you've read the documentation on TechNet, starting with the step-by-step guide found at: <http://go.microsoft.com/fwlink/?LinkId=168805>.

Specify Auditing Settings for Directory Service Changes

1. Right-click the object → **Properties** → **Security** → **Advanced**
2. Click the **Auditing** tab
3. Click **Add** to add an audit entry
4. Specify the group you want to audit (often, **Everyone**)
5. Select to audit **Success** or **Failure** events for one or more specific permissions
6. By default, Domain Admins is configured to audit successful changes to any property by any user (Everyone)

Key Points

You must still modify the SACL of objects to specify which attributes should be audited.

To access the SACL and its audit entries:

1. Open the **Properties** dialog box of the object you wish to audit.
2. Click the **Security** tab.
3. Click the **Advanced** button.
4. Click the **Auditing** tab.

To add an audit entry:

1. Click the **Add** button.
2. Select the user, group, or computer to audit. Often this will be the **Everyone** group.
3. In the **Auditing Entry** dialog box, indicate the type of access to audit.

You are able to audit for successes, failures, or both as the specified user, group, or computer attempts to access the resource using one or more of the granular access levels.

You can audit **Successes** for the following purposes:

- To log resource access for reporting and billing
- To monitor access that would suggest users are performing actions greater than what you had planned, indicating that permissions are too generous
- To identify access that is out of character for a particular account, which might be a sign that a user account has been breached by a hacker

Auditing failed events allows you:

- To monitor for malicious attempts to access resources to which access has been denied.
- To identify failed attempts to access a file or folder to which a user does require access. This would indicate that the permissions are not sufficient to achieve a business requirement.



Note: Don't over-audit. Audit logs have the tendency to get quite large quite rapidly, so a golden rule for auditing is to configure the bare minimum required to achieve the business task. Specifying to audit the successes and failures on an active data folder for the Everyone group using Full Control (all permissions) would generate enormous audit logs that could affect the performance of the server and make locating a specific audited event all but impossible.

View Audited Events in the Security Log

- Event Viewer → Windows Logs → Security
- Each event shows
 - Success/Failure
 - Time
 - Object accessed
 - Identity of user who generated the event
 - Task category

Key Points

After you have enabled the desired audit policy setting and specified the access you want to audit using object SACLs, the system will begin to log access according to the audit entries. You can view the resulting events in the Security Log of the server. Open the Event Viewer console from Administrative Tools. Expand Windows Logs, and select Security Log.

When Directory Service Changes auditing is enabled and auditing entries are configured in the SACL of directory service objects, events are logged to the Security Log that clearly indicate the attribute that was changed and the change made. In most cases, event log entries will show the previous and current value of the changed attribute.

Lab B: Audit Active Directory Changes

- Exercise 1: Audit changes to Active Directory by using default audit policy
- Exercise 2: Audit changes to Active Directory by using Directory Service Changes auditing

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 15 minutes

Scenario

The enterprise security team at Contoso has asked you to provide detailed reports regarding changes to the membership of security-sensitive groups, including Domain Admins. The reports must show the change that was made, who made the change, and when.

Exercise 1: Audit Changes to Active Directory by Using Default Audit Policy

In this exercise, you will see the Directory Service Access auditing that is enabled by default in Windows Server 2008 and Windows Server 2003.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Confirm that the Domain Admins group is configured to audit changes to its membership.
3. Make a change to the membership of Domain Admins.
4. Examine the events that were generated.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.

► Task 2: Confirm that the Domain Admins group is configured to audit changes to its membership

- Run **Active Directory Users and Computers** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
- Open the **Audit Settings** properties of the **Domain Admins** group.

Locate the entry that specifies the auditing of successful attempts to modify properties of the group such as membership.

Question: What is the Auditing Entry that achieves this goal?

► **Task 3: Make a change to the membership of Domain Admins**

- Add **Stuart Munson** (user logon name **Stuart.Munson**) to the **Domain Admins** group. Be sure to apply your change.
- Remove **Stuart Munson** from the **Domain Admins** group.
- Make a note of the time when you made the changes. That will make it easier to locate the audit entries in the event logs.

► **Task 4: Examine the events that were generated**

- Run **Event Viewer** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
- Open the **Security Log** and locate the events that were generated when you added and removed Stuart Munson.

Question: What is the Event ID of the event logged when you made your changes? What is the Task Category?

Question: Examine the information provided on the General tab. Can you identify the following in the event log entry?

- Who made the change?
- When the change was made?
- What object was changed?
- What type of access was performed?
- What attribute was changed? How is the changed attribute identified?
- What change was made to that attribute?

Results: After this exercise, you will have generated and examined Directory Service Access audit entries.

Exercise 2: Audit Changes to Active Directory by Using Directory Service Changes Auditing

In this exercise, you will implement the new Directory Services Changes auditing of Windows Server 2008 to reveal the details about changes to the Domain Admins group.

The main tasks for this exercise are as follows:

1. Enable Directory Services Changes auditing.
2. Make a change to the membership of Domain Admins.
3. Examine the events that were generated.

► Task 1: Enable Directory Services Changes auditing

- Run the command prompt as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
- Type the following command, and then press ENTER:

```
auditpol /set /subcategory:"directory service changes"  
/success:enable
```

► Task 2: Make a change to the membership of Domain Admins

- Add **Stuart Munson** (user logon name **Stuart.Munson**) to the **Domain Admins** group. Be sure to apply your change.
- Remove **Stuart Munson** from the **Domain Admins** group.
- Make a note of the time when you made the changes. That will make it easier to locate the audit entries in the event logs.

► Task 3: Examine the events that were generated

- Run **Event Viewer** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
- Open the **Security Log** and locate the new types of events that were generated when you added and removed Stuart Munson.

Question: What are the Event IDs of the event logged when you made your changes? What is the Task Category?

Question: Examine the information provided on the General tab. Can you identify the following in the event log entry?

- What type of change was made?
- Who made the change?
- What member was added or removed?
- What group was affected?
- When the change was made?

Results: After this exercise, you will have generated Directory Services Changes auditing entries.



Note: After completing this exercise, turn off all virtual machines and discard undo disks.

Lab Review Questions

Question: What details are captured by Directory Services Changes auditing that are not captured by Directory Service Access auditing?

Question: What types of administrative activities would you want to audit using Directory Services Changes auditing?

MCT USE ONLY. STUDENT USE PROHIBITED

Module 9

Improve the Security of Authentication in an Active Directory Domain Services (AD DS) Domain

Contents:

Lesson 1: Configure Password and Lockout Policies	9-4
Lab A: Configure Password and Account Lockout Policies	9-24
Lesson 2: Audit Authentication	9-30
Lab B: Audit Authentication	9-39
Lesson 3: Configure Read-Only Domain Controllers	9-43
Lab C: Configure Read-Only Domain Controllers	9-63

Module Overview

- Configure Password and Lockout Policies
- Audit Authentication
- Configure Read-Only Domain Controllers

When a user logs on to an Active Directory® domain, she enters her user name and password and the client uses those credentials to authenticate the user—to validate the user's identity against the user's Active Directory account. In Module 3, you learned how to create and manage user accounts and their properties, including their passwords. In this Module, you will explore the domain-side components of authentication, including the policies that specify password requirements and the auditing of authentication-related activities. You will also discover two features introduced by Windows Server® 2008 that can significantly improve the security of authentication in an Active Directory Domain Services (AD DS) domain: password settings objects (better known as fine-grained password policy) and read-only domain controllers.

Objectives

After completing this module, you will be able to:

- Implement your domain password and account lockout policy.
- Configure and assign fine-grained password policies.
- Configure auditing of authentication-related activity.
- Distinguish between account logon and logon events.
- Identify authentication-related events in the Security log.
- Identify the business requirements for RODCs.
- Install an RODC.
- Configure password replication policy.
- Monitor the caching of credentials on an RODC.

Lesson 1

Configure Password and Lockout Policies

- Understand Password Policies
- Understand Account Lockout Policies
- Configure the Domain Password and Lockout Policy
- Demonstration: Configure Domain Account Policies
- Fine-Grained Password and Lockout Policy
- Understand Password Settings Objects (PSOs)
- Demonstrations: Configure Fine-Grained Password Policy
- PSO Precedence and Resultant PSO

In a Windows Server 2008 domain, users are required to change their password every 42 days, and a password must be at least seven characters long and meet complexity requirements including the use of three of four character types: upper case, lower case, numeric, and non-alphanumeric. These three password policies—maximum password age, password length, and password complexity—are among the first policies encountered by administrators and users alike in an Active Directory domain. Rarely do these default settings align precisely with the password security requirements of an organization. Your organization might require passwords to be changed more or less frequently, or to be longer. In this lesson, you'll learn how to implement your enterprise's password and lockout policies by modifying the Default Domain Policy Group Policy object (GPO).

As you know, there are exceptions to every rule, and you likely have exceptions to your password policies. To enhance the security of your domain, you can place more restrictive password requirements for accounts assigned to administrators, for accounts used by services such as Microsoft® SQL Server®, or for a backup utility. In earlier versions of Windows®, this was not possible—a single password policy applied to all accounts in the domain. In this lesson, you will learn to configure fine-grained password policies, a new feature in Windows Server 2008 that allows you to assign different password policies to users and groups in your domain.

Objectives

After completing this lesson, you will be able to:

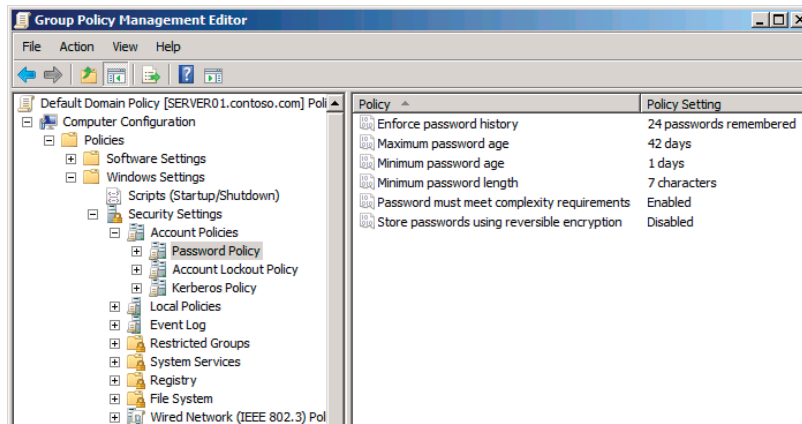
- Implement your domain password and account lockout policy.
- Configure and assign fine-grained password policies.

Understand Password Policies

- Password policies consist of
 - Enforce password history: 24 passwords
 - Max password age: 42 days
 - Min password age: 1 day
 - Min password length: 7 characters
 - Complex Password: enabled
 - Store password using reversible encryption: disabled

Key Points

Your domain's password policy is configured by a GPO scoped to the domain. Within the GPO, in the Computer Configuration > Policies > Windows Settings > Security Settings > Account Policies > Password Policy node, you can configure the policy settings that determine password requirements. The Password Policy node is shown in the following screen shot.



You can understand the effect of the policies by considering the life cycle of a user password. A user will be required to change his or her password within the number of days specified by the Maximum Password Age policy setting. When the user enters a new password, the length of the new password will be compared to the number of characters in the Minimum Password Length policy. If the Password and Must Meet Complexity Requirements policy is enabled, the password must contain at least three of four character types:

- Uppercase—for example, A to Z
- Lowercase—for example, a to z
- Numeric—0 to 9
- Non-alphanumeric—symbols such as !, #, %, or &

If the new password meets requirements, Active Directory puts the password through a mathematical algorithm that produces a representation of the password called the *hash code*. The hash code is unique—no two passwords can create the same hash code. The algorithm used to create the hash code is called a *one-way function*. You cannot put the hash code through a reverse function to derive the password. The fact that it is a hash code, and not the password itself, that is stored in Active Directory helps to increase the security of the user account.

Occasionally, there are applications that require the ability to read a user's password. This is not possible because, by default, only the hash code is stored in Active Directory. In order to support such applications, you can enable the Store Passwords Using Reversible Encryption policy. This policy is not enabled by default, but if you enable the policy, user passwords are stored in an encrypted form that can be decrypted by the application. Reversible encryption significantly reduces the security of your domain, so it is disabled by default, and you should strive to eliminate applications that require direct access to passwords.

Additionally, Active Directory can check a cache of the user's previous hash codes to make sure that the new password is not the same as the user's previous passwords. The number of previous passwords against which a new password is evaluated is determined by the Enforce Password History policy. By default, Windows maintains the previous 24 hash codes.

If a user is determined to reuse her password when the password expiration period occurs, she could simply change her password 25 times in order to work around the password history. To prevent that from happening, the Minimum Password Age policy specifies an amount of time that must pass between password changes. By default, it is one day. Therefore, the determined user would have to change her password once per day for 25 days in order to reuse a password. This type of deterrent is generally successful at discouraging such behavior.

These policy settings—history, minimum age, and maximum age—affect a user that changes his or her password. The settings do not affect an administrator using the Reset Password command to change another user's password.

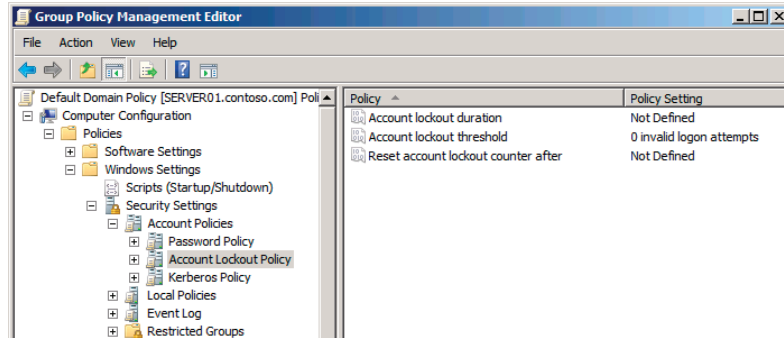
Understand Account Lockout Policies

- Account lockout policies consist of
 - Lockout duration: not defined
 - Lockout threshold: 0 invalid logon attempts
 - Reset account lockout after: not defined
- Help mitigate the threat of brute force attacks on user accounts
- Unlock
 - A user who is locked out can be unlocked by an administrator
 - The Reset account lockout policy can specify a "timeout" after which the account is automatically unlocked

Key Points

An intruder can gain access to the resources in your domain by determining a valid username and password. Usernames are relatively easy to identify, because most organizations create usernames from an employee's e-mail address, initials, combinations of first and last names, or employee IDs. After a username is known, the intruder must determine the correct password. This can be done by guessing, or by repeatedly logging on with combinations of characters or words until the logon is successful.

This type of attack, called *brute force*, can be thwarted by limiting the number of incorrect logons that are allowed. That is exactly what account lockout policies achieve. Account lockout policies are located in the node of the GPO directly below Password Policy. The Account Lockout Policy node is shown in the following screen shot.



There are three settings related to account lockout. The first, Account Lockout Threshold, determines the number of invalid logon attempts permitted within a time specified by the Account Lockout Duration policy. If an attack results in more unsuccessful logons within that timeframe, the user account is locked out. When an account is locked out, Active Directory will deny logon to that account, even if the correct password is specified.

An administrator can unlock a locked user account. You can also configure Active Directory to automatically unlock the account after a delay specified by the Reset Account Lockout After policy setting.

Configure the Domain Password and Lockout Policy

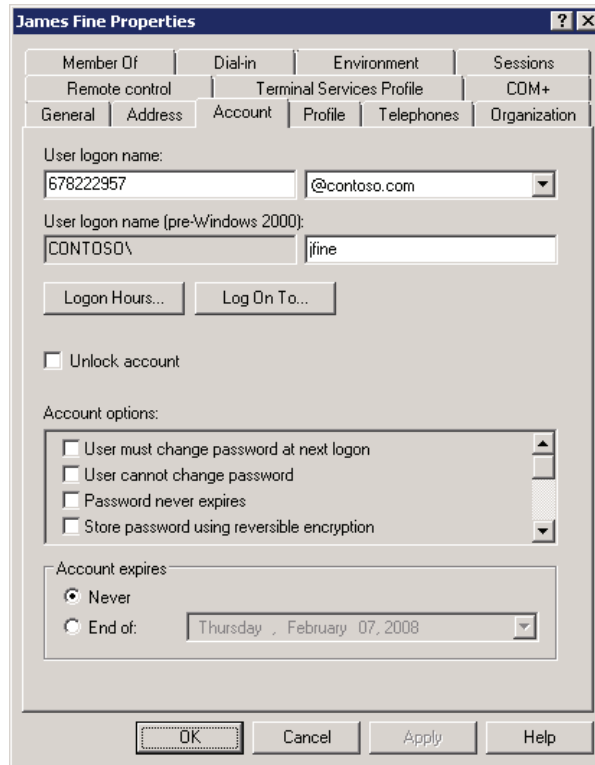
- Domain password policies are defined by the precedent Group Policy Object scoped to domain controllers
 - The Default Domain Policy GPO, by default
- Best practices:
 - Modify the settings in the Default Domain GPO for password, lockout, and Kerberos policies
 - Do not use the Default Domain GPO to deploy any other policy settings
 - Do not define password, lockout, or Kerberos settings for the domain in any other GPO
- Policy settings are overridden by options in user account
 - Password never expires
 - Store passwords using reversible encryption

Key Points

Active Directory supports one set of password and lockout policies for a domain. These policies are configured in a GPO that is scoped to the domain. A new domain contains a GPO called the Default Domain Policy that is linked to the domain and that includes the default policy settings for password, account lockout, and Kerberos policies. You can change the settings by editing the Default Domain Policy.

The best practice is to edit the Default Domain Policy GPO to specify the password policy settings for your organization. You should also use the Default Domain Policy GPO to specify account lockout policies and Kerberos policies. Do not use the Default Domain Policy GPO to deploy any other custom policy settings. In other words, the Default Domain Policy GPO defines the password, account lockout, and Kerberos policies for the domain, and nothing else. Additionally, do not define password, account lockout, or Kerberos policies for the domain in any other GPO.

The password settings configured in the Default Domain Policy affect all user accounts in the domain. The settings can be overridden, however, by the password-related properties of the individual user accounts. On the Account tab of a user's Properties dialog box, you can specify settings such as Password Never Expires or Store Passwords Using Reversible Encryption. For example, if five users have an application that requires direct access to their passwords, you can configure the accounts for those users to store their passwords using reversible encryption.



Additional Reading

- Windows Server 2003 Security Guide Chapter 3: The Domain Policy:
<http://go.microsoft.com/fwlink/?LinkId=99492>

Demonstration: Configure Domain Account Policies

In this demonstration, we will configure the domain account policies for Contoso, Ltd., according to their requirements.

- Password Requirements
 - A minimum of 8 characters long
 - Comply with Windows default complexity requirements
 - Users must change their password every 90 days
 - Users cannot change their own password more than once a week.
 - Users cannot reuse a password within a one-year time

Scenario

Configure the domain account policies to meet the following requirements for passwords:

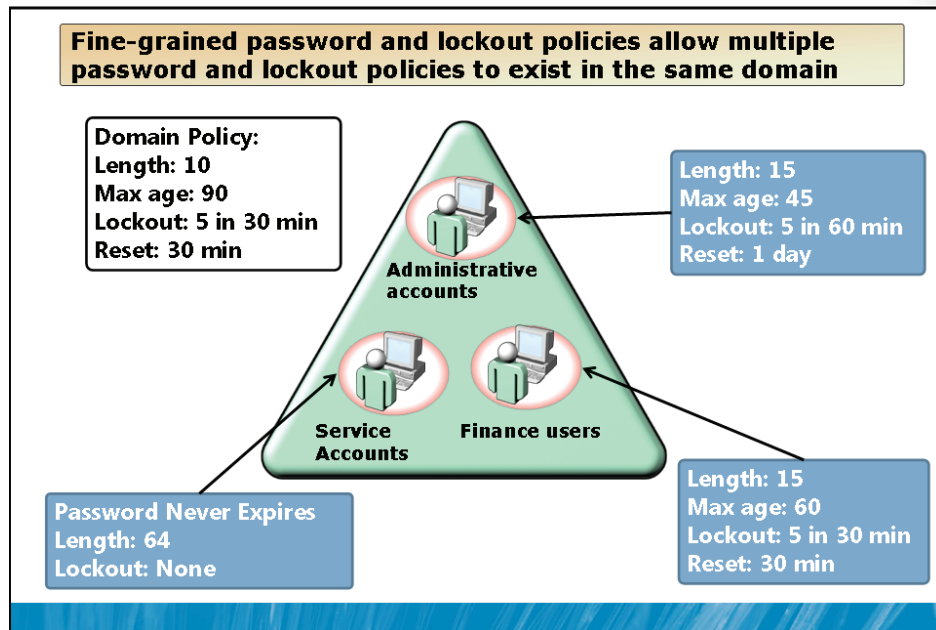
- A minimum of 8 characters long.
- Comply with Windows default complexity requirements.
- Users must change their password every 90 days.
- Users cannot change their own password more than once a week.
- A user cannot reuse a password within a one-year time.

Demonstration Steps

1. In the Group Policy Management console, run with administrative credentials, in the console tree, expand **Forest:contoso.com**, **Domains**, and **contoso.com**.
2. Right-click **Default Domain Policy** underneath the domain, contoso.com and click **Edit**.

3. In the Group Policy Management Editor console tree, expand **Computer Configuration, Policies, Windows Settings, Security Settings, and Account Policies**, and then click **Password Policy**.
4. Double-click the following policy settings in the console details pane and configure the settings as indicated:
 - **Enforce password history**: 53 passwords remembered
 - **Maximum password age**: 90 Days
 - **Minimum password age**: 7 days
 - **Minimum password length**: 8 characters
 - **Password must meet complexity requirements**: Enabled
5. Close the Group Policy Management Editor window.
6. Close the Group Policy Management window.

Fine-Grained Password and Lockout Policy



Key Points

You can override the domain password and lockout policy using a new feature of Windows Server 2008 called *fine-grained password and lockout policy*, often shortened to simply *fine-grained password policy*. Fine-grained password policy allows you to configure a policy that applies to one or more groups or users in your domain.

Fine-grained password policy is a highly anticipated addition to Active Directory. There are several scenarios for which fine-grained password policy can be used to increase the security of your domain. Accounts used by administrators are delegated privileges to modify objects in Active Directory; therefore if an intruder compromises an administrator's account, more damage can be done to the domain than could be done with the account of a standard user. For that reason, you should consider implementing stricter password requirements for administrative accounts. For example, you might require greater password length, and more frequent password changes.

Another type of account that requires special treatment in a domain are accounts used by services such as SQL Server. A service performs its tasks with credentials that must be authenticated with a username and password just like those of a human user. However, most services are not capable of changing their own password, so administrators configure service accounts with the Password Never Expires option enabled. When an accounts password will not be changed, you should make sure the password is difficult to compromise. You can use fine-grained password policies to specify an extremely long minimum password length.

Additional Reading

- AD DS: Fine-Grained Password Policies:
<http://go.microsoft.com/fwlink/?LinkId=99500>

Understand Password Settings Objects (PSOs)

A PSO has the following settings available:

- Password policies
- Account lockout policies
- PSO Link
- Precedence

Considerations when implementing PSOs:

- ☒ The Password Settings Container (PSC) and Password Setting Objects (PSOs) are new object classes defined by the Schema
- ☒ Windows Server 2008 domain functional level required
- ☒ PSOs can be created through ADSI Edit or LDIFDE
- ☒ PSOs can only be applied to users or global groups

Key Points

The settings managed by fine-grained password policy are identical to those in the Password Policy and Accounts Policy nodes of a GPO. However, fine-grained password policies are not implemented as part of Group Policy, nor are they applied as part of a GPO. Instead, there is a separate class of object in Active Directory that maintains the settings for fine-grained password policy—the Password Settings object (PSO).

Most Active Directory objects can be managed with user-friendly graphical user interface (GUI) tools, such as the Active Directory Users and Computers snap-in. You manage PSOs, however, with low-level tools including Active Directory Service Interface Editor (ADSIEdit).

You can create one or more PSOs in your domain. Each PSO contains a complete set of password and lockout policy settings. A PSO is applied by linking the PSO to one or more global security groups or users. For example, to configure a strict password policy for administrative accounts, create a global security group, add the service user accounts as members, and link a PSO to the group. Applying fine-grained password policies to a group in this manner is more manageable than applying the policies to each individual user account. If you create a new service account, you simply add it to the group and the account becomes managed by the PSO.

To use fine-grained password policy, your domain must be at the Windows Server 2008 domain functional level, which means that all of your domain controllers in the domain are running Windows Server 2008, and the domain functional level has been raised to Windows Server 2008.

To confirm and modify the domain functional level:

1. Open **Active Directory Domains and Trusts**.
2. In the console tree, expand **Active Directory Domains and Trusts**, and then expand the tree until you can see the domain.
3. Right-click the domain, and then choose **Raise domain functional level**.

Demonstration: Configure Fine-Grained Password Policy

In this demonstration, we will configure fine-grained password policy to enhance the security of accounts in the Domain Admins group.

- Confirm the domain functional level
- Create a PSO
- Link the PSO to the Domain Admins group
- Evaluate Resultant PSO

Demonstration Steps

1. Run **Active Directory Users and Computers** with administrative credentials and verify that the **Current domain functional level** is **Windows Server 2008**.
2. Run **ADSI Edit**, with administrative credentials, user name **Pat.Coleman_Admin** and password **Pa\$\$w0rd**.
3. Right-click **ADSI Edit**, and then click **Connect To**.
4. Accept all defaults. Click **OK**.
5. In the console tree, click **Default Naming Context**.
6. In the console tree, expand **Default Naming Context**, and then click **DC=contoso,DC=com**, and then click **CN=System**.
7. In the console tree, expand **CN=System**, and then click **CN=Password Settings Container**.

All PSOs are created and stored in the Password Settings Container (PSC).

8. Right-click the **PSC**, point to **New**, and then click **Object**.

The Create Objects dialog box appears. It prompts you to select the type of object to create. There is only one choice: *msDS-PasswordSettings*—the technical name for the object class referred to as a PSO.

9. Click **Next**.

You are then prompted for the value for each attribute of a PSO. The attributes are similar to those found in the domain account policies.

10. Configure each attribute as indicated below. Click **Next** after each attribute.

- *cn*: **My Domain Admins PSO**. This is the common name of the PSO.
- *msDS-PasswordSettingsPrecedence*: **1**. This PSO has the highest possible precedence.
- *msDS-PasswordReversibleEncryptionEnabled*: **False**. The password is not stored using reversible encryption.
- *msDS-PasswordHistoryLength*: **30**. The user cannot reuse any of the last 30 passwords.
- *msDS-PasswordComplexityEnabled*: **True**. Password complexity rules are enforced.
- *msDS-MinimumPasswordLength*: **15**. Passwords must be at least 15 characters long.
- *msDS-MinimumPasswordAge*: **1:00:00:00**. A user cannot change his or her password within one day of a previous change. The format is d:hh:mm:ss (days, hours, minutes, seconds).
- *msDS-MaximumPasswordAge*: **45:00:00:00**. The password must be changed every 45 days.
- *msDS-LockoutThreshold*: **5**. Five invalid logons within the time frame specified by XXX (the next attribute) will result in account lockout.
- *msDS-LockoutObservationWindow*: **0:01:00:00**. Five invalid logons (specified by the previous attribute) within one hour will result in account lockout.
- *msDS-LockoutDuration*: **1:00:00:00**. An account, if locked out, will remain locked for one day, or until it is unlocked manually. A value of zero will result in the account remaining locked out until an administrator unlocks it.

11. Click **Finish** and close **ADSI Edit**.

12. Run **Active Directory Users and Computers** as before and in the console tree, expand the **System** container.
If you do not see the System container, then click the View menu of the MMC console, and ensure that Advanced Features is selected.
13. In the console tree, click the **Password Settings Container**.
14. Right-click **My Domain Admins PSO**, click **Properties** and then click the **Attribute Editor** tab.
15. In the **Attributes** list, select **msDS-PSOAppliesTo**, and then click **Edit**.
The Multi-valued Distinguished Name With Security Principal Editor dialog box appears.
16. Click **Add Windows Account**.
The Select Users, Computers, or Groups dialog box appears.
17. Type **Domain Admins**, and then press ENTER.
18. Click **OK** twice to close the open dialog boxes.
19. In the console tree, expand the **contoso.com** domain and the **Admins** OU, and then click the **Admin Identities** OU.
20. Right-click **Pat Coleman (Administrator)** and click **Properties**.
21. Click the **Attribute Editor** tab.
22. Click the **Filter** button, and click the **Constructed** option, so that it is selected.
23. Open the value of the **msDS-ResultantPSO** attribute.

Additional Reading

- AD DS Fine-Grained Password and Account Lockout Policy Step-by-Step Guide: <http://go.microsoft.com/fwlink/?LinkId=113764>

PSO Precedence and Resultant PSO

- A PSO can be linked to more than one group or user
- A group or user can have more than one PSO linked to it
- Only one PSO "wins"—the **Resultant PSO**
 - *Precedence*: lower (closer to 1) has *higher* precedence
 - Global group PSO with highest precedence (closest to 1) wins
 - Any PSOs linked to user override all global group PSOs.
User-linked PSO with highest precedence (closest to 1) wins
- **msDS-ResultantPSO** attribute of user in Attribute Editor
 - Click the Filter button and ensure Constructed is selected
- If there are no PSOs, domain account policies apply
- Best practices
 - Use only group-linked PSOs. Do not link to user objects.
 - Avoid having two PSOs with the same *precedence* value
- PSOs cannot be "linked" to an OU
 - Create a *shadow group* that contains all users in the OU

Key Points

A PSO can be linked to more than one group or user, an individual group or user can have more than one PSO linked to it, and a user can belong to multiple groups. So which fine-grained password and lockout policy settings apply to a user? One and only one PSO determines the password and lockout settings for a user—this PSO is called the *resultant PSO*. Each PSO has an attribute that determines the PSOs precedence. The *precedence* value is any number greater than 0, where the number 1 indicates the highest precedence. If multiple PSOs apply to a user, the PSO with the highest precedence takes effect. The rules that determine precedence are as follows:

- If multiple PSOs apply to groups to which the user belongs, the PSO with the highest precedence wins.
- If one or more PSOs are linked directly to the user, PSOs linked to groups are ignored, regardless of their precedence. The user-linked PSO with highest precedence wins.

- If one or more PSOs have the same precedence value, Active Directory must make a choice. It picks the PSO with the lowest globally unique identifier (GUID). GUIDs are like serial numbers for Active Directory objects—no two objects have the same GUID. GUIDs have no particular meaning—they are just identifiers—so picking the PSO with the lowest GUID is, in effect, an arbitrary decision. You should configure PSOs with unique, specific precedence values so that you avoid this scenario.

These rules determine the resultant PSO. Active Directory exposes the resultant PSO in a user object attribute, *msDS-ResultantPSO*, so you can readily identify the PSO that will affect a user. PSOs contain all password and lockout settings, so there is no inheritance or merging of settings. The resultant PSO is the authoritative PSO.

To view the *msDS-ResultantPSO* attribute of a user:

1. Ensure that **Advanced Features** is enabled on the **View** menu.
2. Open the properties of the user account.
3. Click the **Attribute Editor** tab.
4. Click the **Filter** button and ensure that **Constructed** is selected.
5. Locate the *msDS-ResultantPSO* attribute.

PSOs, OUs, and Shadow Groups

PSOs can be linked to global security groups or users. PSOs cannot be linked to OUs. If you want to apply password and lockout policies to users in an OU, you must create a global security group that includes all of the users in the OU. This type of group is called a shadow group—its membership shadows, or mimics, the membership of an OU.

Lab A: Configure Password and Account Lockout Policies

- Exercise 1: Configure the Domain's Password and Lockout Policies
- Exercise 2: Configure Fine-Grained Password Policy

Logon information

Virtual machine	6425B-HQDC01-A
Logon user name	Pat.Coleman
Administrative user name	Pat.Coleman_Admin
Password	Pa\$\$w0rd

Estimated time: 30 minutes

Scenario

Contoso's security team has tasked you with increasing the security and monitoring of authentication against the enterprise's AD DS domain. Specifically, you are to enforce a specified password policy for all user accounts, and a more stringent password policy for security sensitive, administrative accounts.

Exercise 1: Configure the Domain's Password and Lockout Policies

In this exercise, you will modify the Default Domain Policy GPO to implement a password and lockout policy for users in the contoso.com domain.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Configure the domain account policies.

► Task 1: Start the virtual machines and log on

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.

► Task 2: Configure the domain account policies

1. Run **Group Policy Management** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Edit the **Default Domain Policy GPO**.
3. Configure the following password policy settings. Leave other settings at their default values.
 - **Maximum password age:** 90 Days
 - **Minimum password length:** 10 characters
5. Configure the following account lockout policy setting. Leave other settings at their default values.
 - **Account lockout threshold:** 5 Invalid Logon Attempts.
6. Close Group Policy Management Editor and Group Policy Management.

Results: After this exercise, you will have configured new settings for the domain account policies.

Exercise 2: Configure Fine-Grained Password Policy

In this exercise, you will create a PSO that applies a restrictive, fine-grained password policy to user accounts in the Domain Admins group. You will identify the PSO that controls the password and lockout policies for an individual user. Finally, you will delete the PSO that you created.

The main tasks for this exercise are as follows:

1. Create a PSO.
2. Link a PSO to a Group.
3. Identify the Resultant PSO for a User.
4. Delete a PSO.

► Task 1: Create a PSO

1. Click **Start**, point to **Administrative Tools**, right-click **ADSI Edit**, and choose **Run as administrator**.
2. Click **Use another account**.
3. In the **User name** box, type **Pat.Coleman_Admin**.
4. In the **Password** box, type **Pa\$\$w0rd**, and then press ENTER. ADSI Edit opens.
5. Right-click **ADSI Edit** and choose **Connect To**.
6. Accept all defaults. Click **OK**.
7. Click **Default Naming Context** in the console tree.
8. Expand **Default Naming Context**, and select **DC=contoso,DC=com**.
9. Expand **DC=contoso,DC=com**, and select **CN=System**.
10. Expand **CN=System**, and select **CN= Password Settings Container**.

All PSOs are created and stored in the Password Settings Container (PSC).

11. Right-click the **PSC** and choose **New, Object**. The **Create Objects** dialog box appears.

It prompts you to select the type of object to create. There is only one choice: *msDS-PasswordSettings*—the technical name for the object class referred to as a PSO.

12. Click **Next**. You are then prompted for the value for each attribute of a PSO. The attributes are similar to those found in the domain account policies.
13. Configure each attribute as indicated below. Click **Next** after each attribute.
 - *Common-Name*: **My Domain Admins PSO**. This is the friendly name of the PSO.
 - *msDS-PasswordSettingsPrecedence*: **1**. This PSO has the highest possible precedence.
 - *msDS-PasswordReversibleEncryptionEnabled*: **False**. The password is not stored using reversible encryption.
 - *msDS-PasswordHistoryLength*: **30**. The user cannot reuse any of the last 30 passwords.
 - *msDS-PasswordComplexityEnabled*: **True**. Password complexity rules are enforced.
 - *msDS-MinimumPasswordLength*: **15**. Passwords must be at least 15 characters long.
 - *msDS-MinimumPasswordAge*: **1:00:00:00**. A user cannot change his or her password within one day of a previous change. The format is d:hh:mm:ss (days, hours, minutes, seconds).
 - *msDS-MaximumPasswordAge*: **45:00:00:00**. The password must be changed every 45 days.
 - *msDS-LockoutThreshold*: **5**. Five invalid logons within the time frame specified by XXX (the next attribute) will result in account lockout.
 - *msDS-LockoutObservationWindow*: **0:01:00:00**. Five invalid logons (specified by the previous attribute) within one hour will result in account lockout.
 - *msDS-LockoutDuration*: **1:00:00:00**. An account, if locked out, will remain locked for one day, or until it is unlocked manually. A value of zero will result in the account remaining locked out until an administrator unlocks it.
14. Click **Finish**.
15. Close ADSI Edit.

► **Task 2: Link a PSO to a Group**

1. Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. In the console tree, expand the **System** container.
If you do not see the System container, then click the View menu of the MMC console, and ensure that Advanced Features is selected.
3. In the console tree, click the **Password Settings Container**.
4. Right-click **My Domain Admins PSO**, and then click the **Attribute Editor** tab.
5. In the **Attributes** list, select **msDS-PSOAppliesTo**, and then click **Edit**.
The Multi-valued Distinguished Name With Security Principal Editor dialog box appears.
6. Click **Add Windows Account**.
The Select Users, Computers, or Groups dialog box appears.
7. Type **Domain Admins**, and then press ENTER.
8. Click **OK** twice to close the open dialog boxes.

► **Task 3: Identify the Resultant PSO for a user**

1. Run **Active Directory Users and Computers** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Open the **Attribute Editor** in the **Properties** dialog box for the account **Pat.Coleman_Admin**.
3. Click the **Filter** button and ensure that **Constructed** is selected.
The attribute you will locate in the next step is a constructed attribute, meaning that the resultant PSO is not a hard-coded attribute of a user; rather it is calculated by examining the PSOs linked to a user in real-time.

Question: What is the resultant PSO for Pat Coleman (Administrator)?

► **Task 4: Delete a PSO**

1. With **Advanced Features** enabled in the **View** menu of **Active Directory Users and Computers**, open the **System** container and the **Password Settings Container**.
2. Delete the PSO you created, My Domain Admins PSO.

Results: After this exercise, you should have created a PSO, applied it to Domain Admins and confirmed its application, and then deleted the PSO.



Note: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in subsequent labs in this module

Lab Review Questions

Question: Where should you define the default password and account lockout policies for user accounts in the domain?

Question: What are the best practices for managing PSOs in a domain?

Question: How can you define a unique password policy for all of the service accounts in the Service Accounts OU?

Lesson 2

Audit Authentication

- Account Logon and Logon Events
- Configure Authentication-Related Audit Policies
- Scoping Audit Policies
- View Logon Events

Windows Server 2008 also allows you to audit the logon activity of users in a domain. By auditing successful logons, you can look for instances in which an account is being used at unusual times or in unexpected locations, which may indicate that an intruder is logging on to the account. Auditing failed logons can reveal attempts by intruders to compromise an account. In this lesson, you will learn to configure auditing of logon authentication.

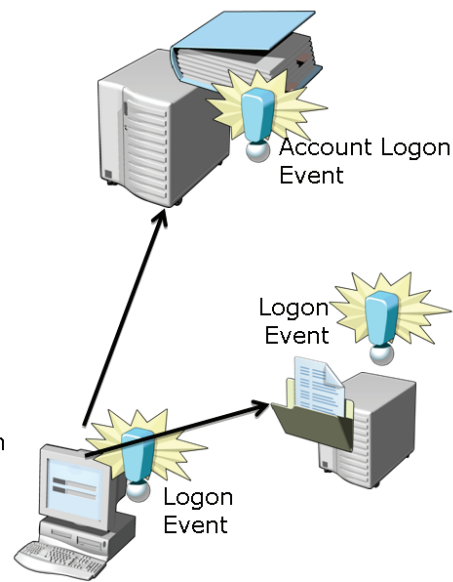
Objectives

After completing this lesson, you will be able to:

- Configure auditing of authentication-related activity.
- Distinguish between account logon and logon events.
- Identify authentication-related events in the Security log.

Account Logon and Logon Events

- **Account logon events**
 - Registered by the system that authenticates the account
 - For domain accounts: domain controllers
 - For local accounts: local computer
- **Logon events**
 - Registered by the machine at which (or to which) a user logged on
 - Interactive logon: user's system
 - Network logon: server



Key Points

This lesson examines two specific policy settings: Audit Account Logon Events and Audit Logon Events. It is important that you understand the difference between these two similarly named policy settings.

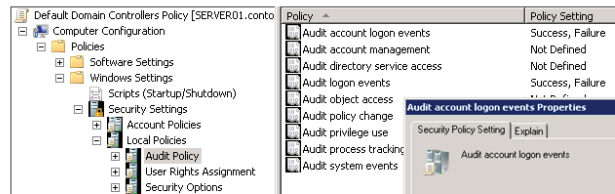
When a user logs on to any computer in the domain using a domain user account, a domain controller authenticates the attempt to log on to the domain account. This generates an account logon event on the domain controller.

The computer to which the user logs on—for example, the user's laptop—generates a logon event. The computer did not authenticate the user against his account—it passed the account to a domain controller for validation. The computer did, however, allow the user to log on interactively to the computer. Therefore, the event is a logon event.

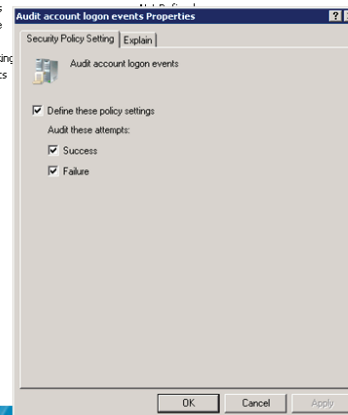
When the user connects to a folder on a server in the domain, that server authorizes the user for a type of logon called a network logon. Again, the server does not authenticate the user—it relies on the ticket given to the user by the domain controller. But the connection by the user generates a logon event on the server.

Configure Authentication-Related Audit Policies

- Computer Configuration > Windows Settings > Security Settings > Local Policies > Audit Policy

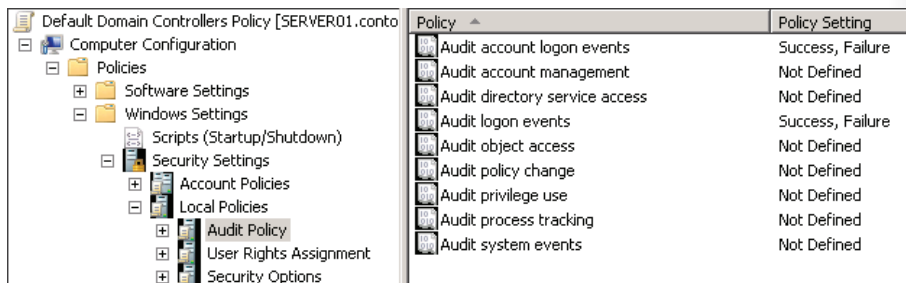


- Windows Server 2008 default is to audit SUCCESS events for both account logon and logon events

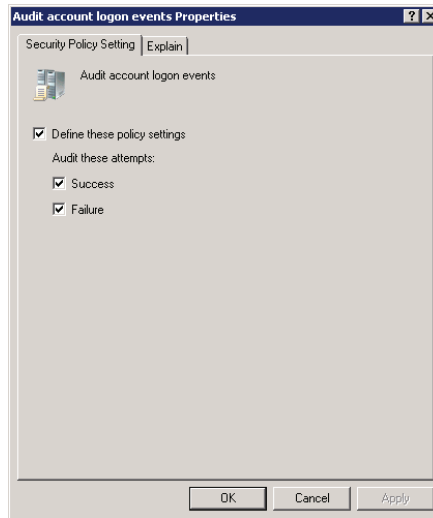


Key Points

Account logon and logon events can be audited by Windows Server 2008. These settings that manage auditing are located in a GPO in the Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > Audit Policy node. The Audit Policy node and the two settings are shown in the following screen shot.



To configure an audit policy, double-click the policy, and its properties dialog box appears. The Audit Account Logon Events Properties dialog box is shown in the following screen shot.



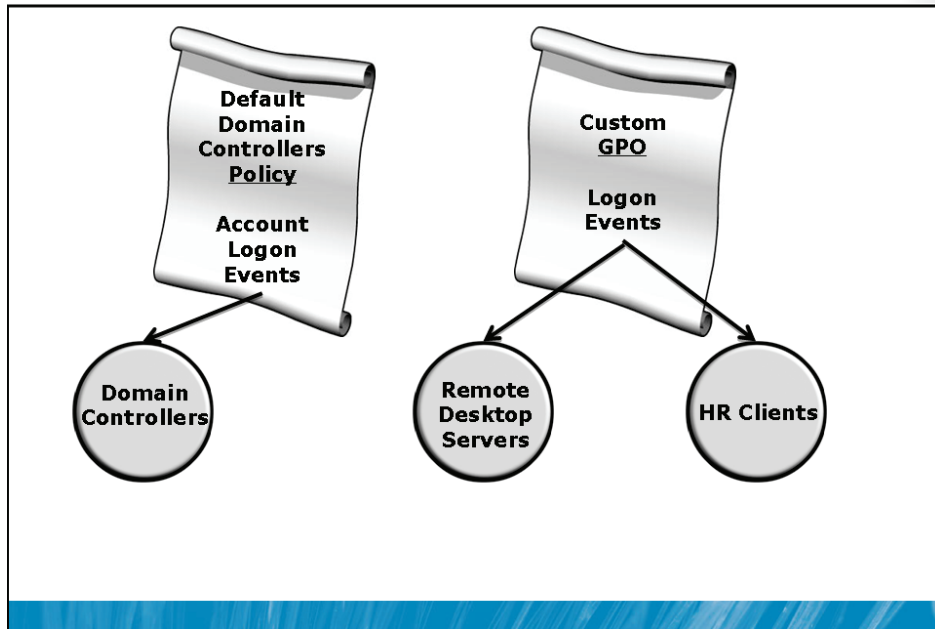
The policy setting can be configured to one of the following four states:

- **Not Defined:** If the Define These Policy Settings check box is cleared, the policy setting is not defined. In this case, the server will audit the event based on its default settings, or on the settings specified in another GPO.
- **Defined for no auditing:** If the Define These Policy Settings check box is selected, but the Success and Failure check boxes are cleared, then the server will not audit the event.
- **Audit successful events:** If the Define These Policy Settings check box is selected, and the Success check box is selected, the server will log successful events in its Security log.
- **Audit failed events:** If the Define These Policy Settings check box is selected, and the Failure check boxes selected, the server will log unsuccessful events in its Security log.

A server's audit behavior is determined by the one of these four settings that is applied as the resultant set of policy.

In Windows Server 2008, the default setting is to audit successful account logon events and successful logon events. So both types of events are, if successful, entered in the server's Security log. If you want to audit failures or to turn off auditing, you will need to define the appropriate setting in the audit policy.

Scoping Audit Policies



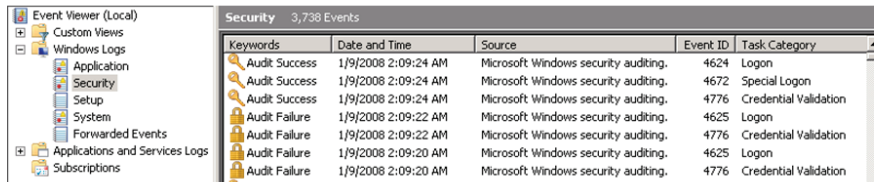
Key Points

As with all policy settings, you should be thoughtful to scope settings so that they affect the correct systems. For example, if you want to audit attempts by users to connect to remote desktop servers in your enterprise, you can configure logon event auditing in a GPO linked to the OU that contains your remote desktop servers. If, on the other hand, you want to audit logons by users to desktops in your human resources department, you can configure logon event auditing in a GPO linked to the OU containing human resources computer objects. Remember that domain users logging on to a client computer or connecting to a server will generate a logon event—not an account logon event—on that system.

Only domain controllers generate account logon events for domain users. Remember that an account logon event occurs on the domain controller that authenticates a domain user, regardless of where that user logs on. If you want to audit logons to domain accounts, you should scope account logon event auditing to affect only domain controllers. In fact, the Default Domain Controllers GPO that is created when you install your first domain controller is an ideal GPO in which to configure account logon audit policies.

View Logon Events

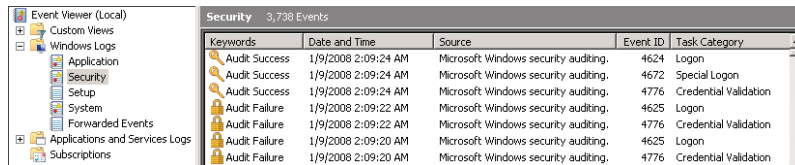
- Security log of the system that generated the event
 - The DC that authenticated the user: account logon
 - Note: Not replicated to other DCs
 - The system to which the user logged on or connected: logon



Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	1/9/2008 2:09:24 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	1/9/2008 2:09:24 AM	Microsoft Windows security auditing.	4672	Special Logon
Audit Success	1/9/2008 2:09:24 AM	Microsoft Windows security auditing.	4776	Credential Validation
Audit Failure	1/9/2008 2:09:22 AM	Microsoft Windows security auditing.	4625	Logon
Audit Failure	1/9/2008 2:09:22 AM	Microsoft Windows security auditing.	4776	Credential Validation
Audit Failure	1/9/2008 2:09:20 AM	Microsoft Windows security auditing.	4625	Logon
Audit Failure	1/9/2008 2:09:20 AM	Microsoft Windows security auditing.	4776	Credential Validation

Key Points

Account logon and logon events, if audited, appear in the Security log of the system that generated the event. An example is shown in the following screen shot.



Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	1/9/2008 2:09:24 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	1/9/2008 2:09:24 AM	Microsoft Windows security auditing.	4672	Special Logon
Audit Success	1/9/2008 2:09:24 AM	Microsoft Windows security auditing.	4776	Credential Validation
Audit Failure	1/9/2008 2:09:22 AM	Microsoft Windows security auditing.	4625	Logon
Audit Failure	1/9/2008 2:09:22 AM	Microsoft Windows security auditing.	4776	Credential Validation
Audit Failure	1/9/2008 2:09:20 AM	Microsoft Windows security auditing.	4625	Logon
Audit Failure	1/9/2008 2:09:20 AM	Microsoft Windows security auditing.	4776	Credential Validation

So if you are auditing logons to computers in the human resources department, the events are entered in each computer's Security log. Similarly, if you are auditing unsuccessful account logons in order to identify potential intrusion attempts, the events are entered in each domain controller's Security log. This means, by default, you will need to examine the Security logs of all domain controllers in order to get a complete picture of account logon events in your domain.

As you can imagine, in a complex environment with multiple domain controllers and many users, auditing account logons or logons can generate a tremendous number of events. If there are too many events, it can be difficult to identify problematic events worthy of closer investigation. You should balance the amount of logging you perform with the security requirements of your business and the resources you have available to analyze logged events.

Lab B: Audit Authentication

- Exercise 1: Audit Authentication

Logon information

Virtual machine	6425B-HQDC01-A	6425B-SERVER01-A
Logon user name	Pat.Coleman	Pat.Coleman
Administrative user name	Pat.Coleman_Admin	Pat.Coleman_Admin
Password	Pa\$\$w0rd	Pa\$\$w0rd

Estimated time: 30 minutes

Scenario

Contoso's security team has tasked you with increasing the security and monitoring of authentication against the enterprise's AD DS domain. Specifically, you are to create an audit trail of logons.

Exercise 1: Audit Authentication

In this exercise, you will use Group Policy to enable auditing of both successful and unsuccessful logon activity by users in the contoso.com domain. You will then generate logon events and view the resulting entries in the event logs.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Configure auditing of account logon events.
3. Configure auditing of logon events.
4. Force a refresh Group Policy.
5. Generate account logon events.
6. Examine account logon events
7. Examine logon events

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Open **D:\Labfiles\Lab09b**.
4. Run **Lab09b_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.

► Task 2: Configure auditing of account logon events

1. Run **Group Policy Management** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Modify the **Default Domain Controllers Policy GPO** to enable auditing events for both successful and failed account logon events.
3. Close Group Policy Management Editor.

► **Task 3: Configure auditing of logon events**

1. Create a **Group Policy Object** (GPO) linked to the **Servers\Important Project OU**. Name the GPO **Server Lockdown Policy**.
2. Modify the **Server Lockdown Policy** to enable auditing events for both successful and failed account logon events.
3. Close Group Policy Management Editor and Group Policy Management.

► **Task 4: Force a refresh Group Policy**

1. Start SERVER01-A. As the computer starts, it will apply the changes you made to Group Policy.
2. On HQDC01, run the **Command Prompt** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**, and then run the command **gpupdate.exe /force**. Close the Command Prompt.

► **Task 5: Generate account logon events**

1. Log on to SERVER01 as **Pat.Coleman**, but enter an incorrect password.
2. After you have been denied logon, log on again with the correct password, **Pa\$\$w0rd**.

► **Task 6: Examine account logon events**

1. Run **Event Viewer** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Identify the failed and successful events in the **Security** log.

Question: What Event ID is associated with the account logon failure events? (Tip: Look for the earliest of a series of failure events at the time you logged on incorrectly to SERVER01.)

Question: What Event ID is associated with the successful account logon? (Tip: Look for the earliest of a series of events at the time you logged on incorrectly to SERVER01.)

► Task 7: Examine logon events

1. On SERVER01, run **Event Viewer** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Identify the failed and successful events in the Security Log.

Question: What Event ID is associated with the logon failure events? (Tip: Look for the earliest of a series of failure events at the time you logged on incorrectly to SERVER01.)

Question: What Event ID is associated with the successful logon? (Tip: Look for the earliest of a series of events at the time you logged on incorrectly to SERVER01.)

Results: After this exercise, you will have established and reviewed auditing for successful and failed logons to the domain and to servers in the Important Project OU.



Note: Do not shut down the virtual machine once you are finished with this lab as the settings you have configured here will be used in subsequent labs in this module

Lab Review Questions

Question: What would be the disadvantage of auditing all successful and failed logons on all machines in your domain?

Question: You have been asked to audit attempts to log on to desktops and laptops in the Finance division using local accounts such as Administrator. What type of audit policy do you set, and in what GPO(s)?

Lesson 3

Configure Read-Only Domain Controllers

- Authentication and Domain Controller Placement in a Branch Office
- Read-Only Domain Controllers
- Deploy an RODC
- Demonstration: Password Replication Policy
- Demonstration: Password RODC Credentials Caching
- Administrative Role Separation

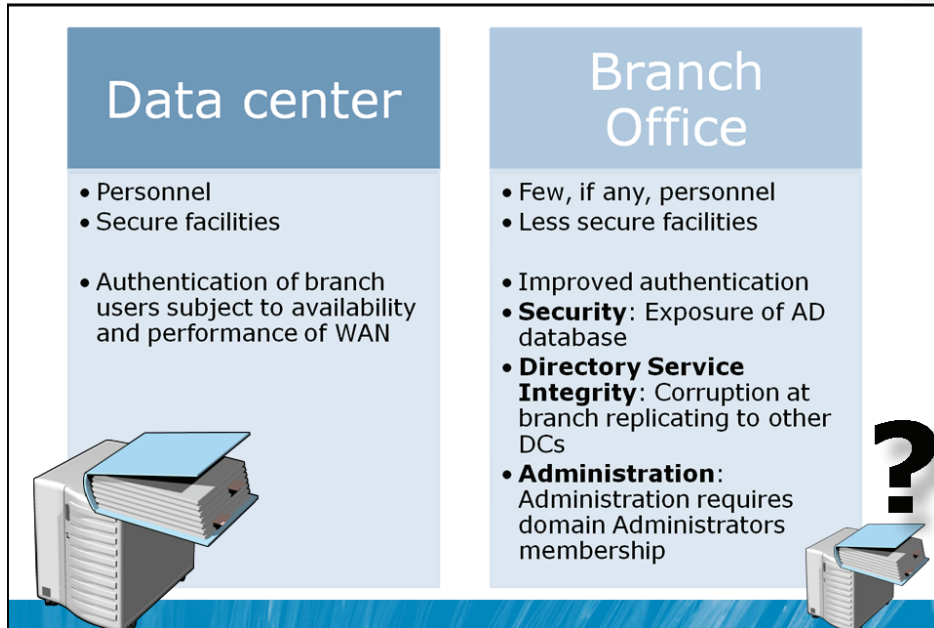
Branch offices present a unique challenge to an enterprise's information technology (IT) staff: If a branch office is separated from the hub site by a wide area network (WAN) link, should you place a domain controller (DC) in the branch office? In previous versions of Windows, the answer to this question was not a simple one. Windows Server 2008, however, introduces a new type of DC—the read-only domain controller (RODC)—that makes the question easier to answer. In this lesson, you will explore the issues related to branch office authentication and domain controller placement, and you will learn how to implement and support a branch-office RODC.

Objectives

After completing this lesson, you will be able to:

- Identify the business requirements for RODCs.
- Install an RODC.
- Configure password replication policy.
- Monitor the caching of credentials on an RODC.

Authentication and Domain Controller Placement in a Branch Office



Key Points

Consider a scenario in which an enterprise is characterized by a hub site and several branch offices. The branch offices connect to the hub site over wide area network (WAN) links that may be congested, expensive, slow, or unreliable. Users in the branch office must be authenticated by Active Directory in order to access resources in the domain. Should a DC be placed in the branch office?

In branch office scenarios, many of the IT services are centralized in the hub site that is carefully maintained by the IT staff. In larger organizations, the hub site may include a robust datacenter. Branch offices, however, are often smaller sites at which no datacenter exists. In fact, many branch offices have no significant IT presence, other than a small handful of servers. There may be no physically secure facility to house branch office servers. There may be few, if any, local IT staff to support the servers.

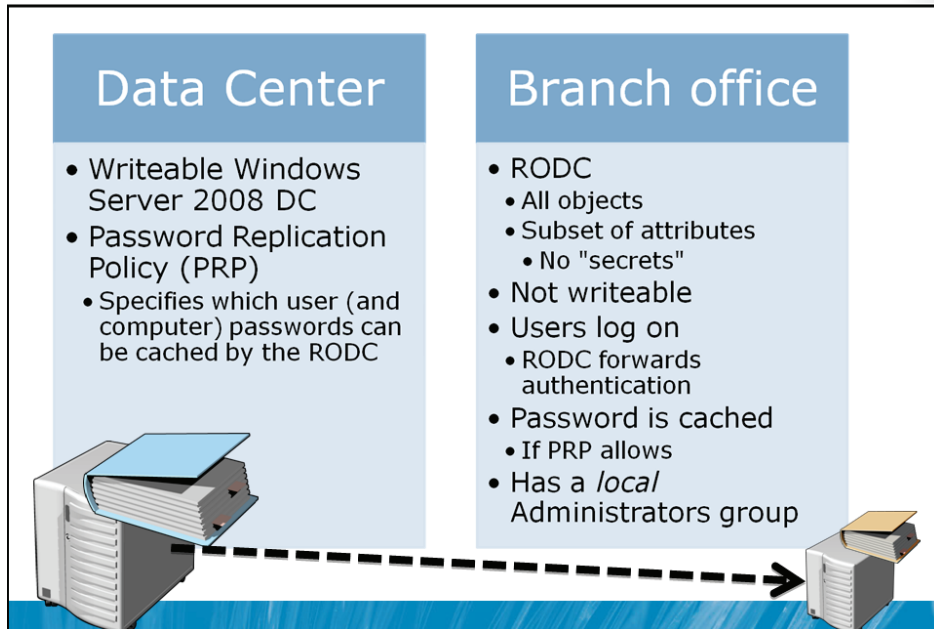
If a DC is not placed in the branch office, authentication and service ticket activities will be directed to the hub site over the WAN link. Authentication occurs when a user first logs on to his computer in the morning. Service tickets are a component of the Kerberos authentication mechanism used by Windows Server 2008 domains. You can think of a service ticket as a key issued by the domain controller to a user. The key allows the user to connect to a service, such as the File and Print service on a file server. When a user first tries to access a specific service, the user's client requests what is called a *service ticket* from the domain controller. Because users typically connect to multiple services during a work day, service ticket activity happens regularly. Authentication and service ticket activity over the WAN link between a branch office and a hub site can result in slow or unreliable performance.

If a DC is placed in the branch office, authentication is much more efficient but there are several potentially significant risks. A DC maintains a copy of all attributes of all objects in its domain, including secrets such as information related to user passwords. If a DC is accessed or stolen, it becomes possible for a determined expert to identify valid usernames and passwords, at which point the entire domain is compromised. At a minimum, you must reset the passwords of every user account in the domain. Because the security of servers at branch offices is often less than ideal, a branch office DC poses a considerable security risk.

A second concern is that changes to the Active Directory database on a branch office DC replicate to the hub site and to all other DCs in the environment. Therefore, corruption to the branch office DC poses a risk to the integrity of the enterprise directory service. For example, if a branch office administrator performs a restore of the DC from an outdated backup, there can be significant repercussions for the entire domain.

The third concern relates to administration. A branch office domain controller may require maintenance, for example a new device driver. In order to perform maintenance on a standard domain controller, you must log on as a member of the Administrators group on the domain controller, which means you are effectively an administrator of the domain. It may not be appropriate to grant that level of capability to a support team at a branch office.

Read-Only Domain Controllers



Key Points

These concerns—security, directory service integrity, and administration—left many enterprises with a difficult choice to make, and there was no best practice answer. Windows Server 2008 introduces the RODC, which is designed specifically to address the branch office scenario. An RODC is a domain controller, typically placed in the branch office, that maintains a copy of all objects in the domain and all attributes except for secrets such as password-related properties. When a user in the branch office logs on, the RODC receives the request and forwards it to a domain controller in the hub site for authentication.

You are able to configure a password replication policy (PRP) for the RODC that specifies user accounts the RODC is allowed to cache. If the user logging on is included in the PRP, the RODC caches that user's credentials, so the next time authentication is requested, the RODC can perform the task locally. As users who are included in the PRP log on, the RODC builds its cache of credentials so that it can perform authentication locally for those users.

Because the RODC maintains only a subset of user credentials, if the RODC is compromised or stolen, the effect of the security exposure is limited: Only the user accounts that had been cached on the RODC must have their passwords changed. The RODC replicates changes to Active Directory from DCs in the hub site. Replication is one way. No changes to the RODC are replicated to any other domain controller. This eliminates the exposure of the directory service to corruption resulting from changes made to a compromised branch office DC. Finally, RODCs have the equivalent of a local Administrators group. You can give one or more local support personnel the ability to fully maintain an RODC without granting them the equivalence of Domain Admins.

Deploy an RODC

1. Ensure the forest functional level is Windows Server 2003 or higher
 - All domain controllers running Windows Server 2003 or later
 - All domains functional level of Windows Server 2003 or higher
 - Forest functional level set to Windows Server 2003 or higher
2. If the forest has any DCs running Windows Server 2003, run **adprep /rodcprep**
 - Windows Server 2008 CD:\sources\adprep folder
3. Ensure that there is at least one writeable DC running Windows Server 2008
4. Install the RODC
 - Active Directory Domain Services Installation Wizard (dcpromo)
 - Stage delegated installation of an RODC: Domain Controllers OU

Key Points

The high-level steps to install an RODC are as follows:

1. Ensure the forest functional level is Windows Server 2003 or higher.
2. If the forest has any DCs running Windows Server 2003, run `adprep /rodcprep`.
3. Ensure there is at least one writable DC running Windows Server 2008.
4. Install the RODC.

Each of these steps are detailed in the following sections.

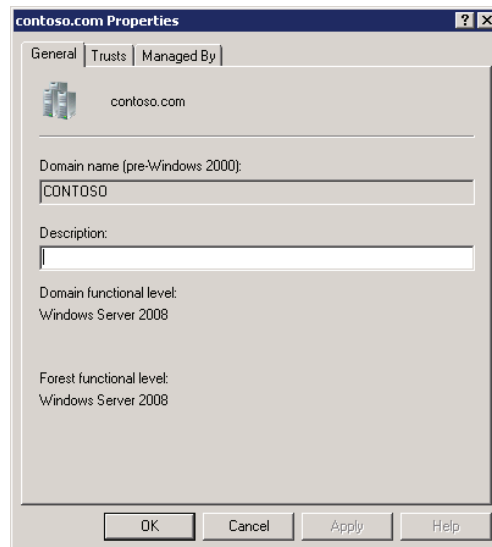
Verifying and Configuring Forest Functional Level of Windows Server 2003 or Higher

Functional levels enable features unique to specific versions of Windows, and are therefore dependent on the versions of Windows running on domain controllers. If all domain controllers are Windows Server 2003 or later, the domain functional level can be set to Windows Server 2003. If all domains are at Windows Server 2003 domain functional level, the forest functional level can be set to Windows Server 2003. Domain and forest functional levels are discussed in detail in another module.

RODCs require that the forest functional level is Windows Server 2003 or higher. That means that all domain controllers in the entire forest are running Windows Server 2003 or later.

To determine the functional level of your forest:

1. Open **Active Directory Domains and Trusts**.
2. Right-click the name of the forest, then click **Properties**.
3. Verify the forest functional level, as shown below. Any user can verify the forest functional level in this way. No special administrative credentials are required to view the forest functional level.



If the forest functional level is not at least Windows Server 2003, examine the properties of each domain to identify any domains for which the domain functional level is not at least Windows Server 2003. If you find such a domain, you must ensure that all domain controllers in the domain are running Windows Server 2003. Then, in Active Directory Domains and Trusts, right-click the domain and choose Raise Domain Functional Level. After you have raised each domain functional level to at least Windows Server 2003, right-click the root node of the Active Directory Domains And Trusts snap in, and choose Raise Forest Functional Level. In the Select An Available Forest Functional Level drop-down list, choose Windows Server 2003 and click Raise. You must be an administrator of a domain to raise the domain's functional level. To raise the forest functional level, you must be either a member of the Domain Admins group in the forest root domain or a member of the Enterprise Admins group.

Running ADPrep /RODCPrep

If you are upgrading an existing forest to include domain controllers running Windows Server 2008, you must run `adprep /rodcprep`. This command configures permissions so that RODCs are able to replicate DNS application directory partitions. DNS application directory partitions are discussed in another module. If you are creating a new Active Directory forest and it will have only domain controllers running Windows Server 2008, you do not need to run `adprep /rodcprep`.

The command is found in the `\sources\adprep` folder of the Windows Server 2008 installation DVD. Copy the folder to the domain controller acting as the schema master. The schema master role is discussed in another module. Log on to the schema master as a member of the Enterprise Admins group, open a command prompt, change directories to the `adprep` folder, and type `adprep /rodcprep`.

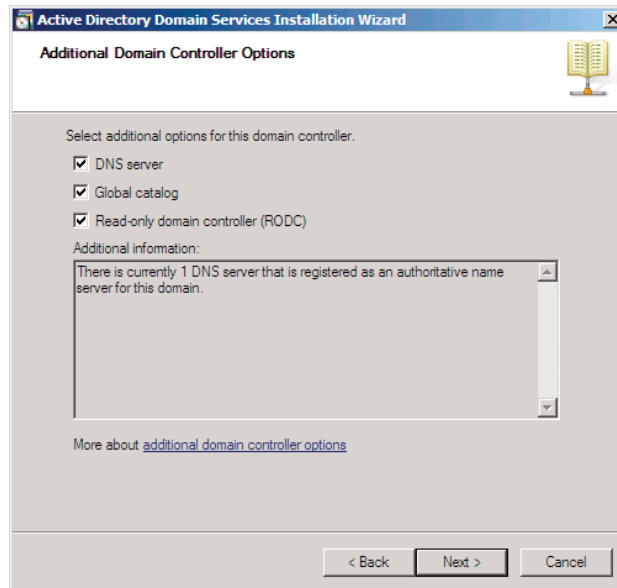
Before running `adprep /rodcprep`, you must run `adprep /forestprep` and `adprep /domainprep`. See Module 14 for more information about preparing a Windows Server 2003 domain and forest for the first Windows Server 2008 domain controller.

Placing a Writable Windows Server 2008 Domain Controller

An RODC must replicate domain updates from a writable domain controller running Windows Server 2008. It is critical that an RODC is able to establish a replication connection with a writable Windows Server 2008 domain controller. Ideally, the writable Windows Server 2008 domain controller should be in the closest site—the hub site. If you want the RODC to act as a DNS server, the writable Windows Server 2008 domain controller must also host the DNS domain zone.

Installing an RODC

After completing the preparatory steps, you can install an RODC. An RODC can be either a full or Server Core installation of Windows Server 2008. With a full installation of Windows Server 2008, you can use the Active Directory Domain Services Installation Wizard to create an RODC. Simply select Read-only Domain Controller (RODC) on the Additional Domain Controller Options page of the wizard, as shown below.



Alternatively, you can use the `dcpromo.exe` command with the `/unattend` switch to create the RODC. On a Server Core installation of Windows Server 2008, you must use the `dcpromo.exe /unattend` command.

It is also possible to delegate the installation of the RODC, which allows a user who is not a domain administrator to create the RODC, by adding a new server in the branch office and running `dcpromo.exe`. To delegate the installation of an RODC, pre-create the computer account for the RODC in the Domain Controllers OU and specify the credentials that will be used to add the RODC to the domain. That user can then attach a server running Windows Server 2008 to the RODC account. The server must be a member of a workgroup—not the domain—when creating an RODC using delegated installation.

Additional Reading

- For details regarding other options for installing an RODC, including delegated installation see <http://go.microsoft.com/fwlink/?LinkId=168763>

Demonstration: Password Replication Policy

In this demonstration, we will

- View an RODC's PRP
 - Allow List and Deny List
 - Deny takes precedence
- Configure domain-wide password replication policy
 - Using the Allowed RODC Password Replication Group and the Denied RODC Password Replication Group
 - The groups are added to all new RODCs PRPs by default
- Configure RODC-specific password replication policy
- Question: What would be the most *manageable* way to ensure that users in a branch have their credentials cached on an RODC?

Key Points

Open the properties of BRANCHDC01-A in the Domain Controllers OU. Click the Password Replication Policy tab.

Password Replication Policy (PRP) determines which users' credentials can be cached on a specific RODC. If PRP allows an RODC to cache a user's credentials, then authentication and service ticket activities of that user can be processed by the RODC. If a user's credentials cannot be cached on RODC, authentication and service ticket activities are referred by the RODC to a writable domain controller.

An RODC's PRP is determined by two multivalued attributes of the RODC's computer account. These attributes are commonly known as the Allowed List and the Denied List. If a user's account is on the Allowed List, the user's credentials are cached. You can include groups on the Allowed List, in which case all users who belong to the group can have their credentials cache on the RODC. If the user is both on the Allowed List and the Denied List, the user's credentials will not be cached—the Denied List takes precedence.

Configure Domain-Wide Password Replication Policy

To facilitate the management of PRP, Windows Server 2008 creates two domain local security groups in the Users container of Active Directory. The first, named Allowed RODC Password Replication Group, is added to the Allowed List of each new RODC. By default, the group has no members. Therefore, by default, a new RODC will not cache any user's credentials. If there are users whose credentials you want to be cached by all domain RODCs, add those users to the Allowed RODC Password Replication Group.

The second group is named Denied RODC Password Replication Group. It is added to the Denied List of each new RODC. If there are users whose credentials you want to ensure are never cached by domain RODCs, add those users to be Denied RODC Password Replication Group. By default, this group contains security sensitive accounts that are members of groups including Domain Admins, Enterprise Admins, and Group Policy Creator Owners.

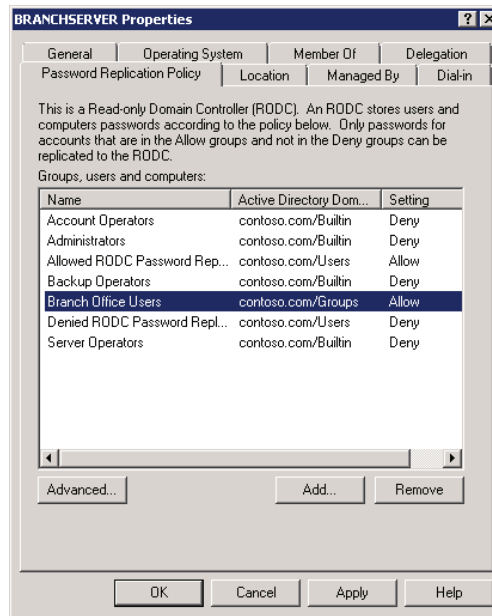


Note: Computers are people, too! Remember that it is not only users that generate authentication and service ticket activity. Computers in a branch office also require such activity. To improve performance of systems in a branch office, allow the branch RODC to cache computer credentials as well.

Configure RODC-Specific Password Replication Policy

The two groups described in the previous section provide a method to manage PRP on all RODCs. However, to best support a branch office scenario, you need to allow the RODC in each branch office to cache credentials of users in that specific location. Therefore, you need to configure the Allowed List and the Denied List of each RODC.

To configure any RODC's PRP, open the properties of the RODC's computer account in the Domain Controllers OU. On the Password Replication Policy tab, shown in the following screen shot, you can view the current PRP settings and add or remove users or groups from the PRP.



Demonstration Steps

1. Run **Active Directory Users and Computers** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
2. In the **Domain Controllers** OU open the properties of **BRANCHDC01**.
3. Click the **Password Replication Policy** tab and view the default policy.
4. Close the **BRANCHDC01** properties.
5. In the **Active Directory Users and Computers** console tree, click the **Users** container.
6. Double-click **Allowed RODC Password Replication Group**. Go to the **Members** tab and examine the default membership of **Allowed RODC Password Replication Group**.
7. Click **OK**.

8. Double-click **Denied RODC Password Replication Group** and go to the **Members** tab.
9. Click **Cancel** to close the **Denied RODC Password Replication Group** properties.

Demonstration: Administer RODC Credentials Caching

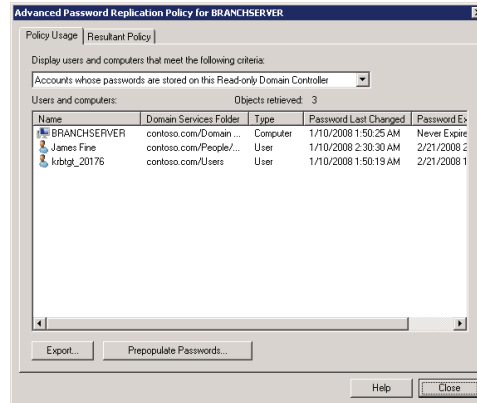
In this demonstration, we will see

- Policy Usage Reports
 - Accounts Whose Passwords Are Stored On This Read-Only Domain Controller
 - Accounts That Have Been Authenticated To This Read-Only Domain Controller
- Resultant Policy
- Prepopulating credentials in the RODC cache

Key Points

For this demonstration, use the virtual machine BRANCHDC01, and open Policy Usage.

When you click the Advanced button on the Password Replication Policy tab of an RODC, an Advanced Password Replication Policy dialog box appears. An example is shown in the following screen shot.



The drop-down list at the top of the Policy Usage tab allows you to select one of two reports for the RODC:

- Accounts Whose Passwords Are Stored On This Read-Only Domain Controller:** Display the list of user and computer credentials that are currently cached on the RODC. Use this list to determine whether credentials are being cached that you do not want to cache on the RODC, and modify the PRP accordingly.
- Accounts That Have Been Authenticated To This Read-Only Domain Controller:** Display the list of user and computer credentials that have been referred to a writable domain controller for authentication or service ticket processing. Use this list to identify users or computers that are attempting to authenticate with the RODC. If any of these accounts are not being cached, consider adding them to the PRP.

In the same dialog box, the Resultant Policy tab allows you to evaluate the effective caching policy for an individual user or computer. Click the Add button to select a user or computer account for evaluation.

You can also use the Advanced Password Replication Policy dialog box to prepopulate credentials in the RODC cache. If a user or computer is on the Allow list of an RODC, the account credentials can be cached on the RODC, but will not be cached until the authentication or service ticket events causes the RODC to replicate the credentials from a writable domain controller. By pre-populating credentials in the RODC cache, for users and computers in the branch office for example, you can ensure that authentication and service ticket activity will be processed locally by the RODC even when the user or computer is authenticating for the first time. To prepopulate credentials, click the Prepopulate Passwords button and select the appropriate users and computers.

Demonstration Steps

1. On HQDC01 in the **Active Directory Users and Computers** console tree, click the **Domain Controllers** OU and open the properties of BRANCHDC01.
2. Click the **Password Replication Policy** tab.
3. Click the **Advanced** button.

The Advanced Password Replication Policy for BRANCHDC01 dialog box appears.

The Policy Usage tab is displaying Accounts whose passwords are stored on this Read-Only Domain Controller.

4. From the drop-down list, select **Accounts Whose Passwords Are Stored On This Read-Only Domain Controller**.
5. From the drop-down list, select **Accounts that have been authenticated to this Read-only Domain Controller**.
6. Click the **Resultant Policy** tab, and then click the **Add** button.
The Select Users or Computers dialog box appears.
7. Type **Chris.Gallagher**, and then press ENTER.
8. Click the **Policy Usage** tab.
9. Click the **Prepopulate Passwords** button.

The Select Users or Computers dialog box appears.

10. Type the name of the account you want to pre-populate, and then click **OK**.
11. Click **Yes** to confirm that you want to send the credentials to the RODC.

A message appears: *Passwords for all accounts were successfully prepopulated.*

Administrative Role Separation

- Allows performing local administrative tasks on the RODC
- Each RODC maintains a local (SAM) database of groups for specific administrative purposes
- DSMgmt command allows you to manage the local roles
 - **dsmgmt [enter]**
 - **local roles [enter]**
 - **? [enter]** for a list of commands
 - **List roles [enter]** for a list of roles
 - **add *username* administrators [enter]**

Key Points

RODCs in branch offices may require maintenance, such as an updated device driver. Additionally, small branch offices may combine the RODC rolled with the file server role on a single system, in which case it will be important to be able to back up the system. RODCs support local administration through a feature called *administrative role separation*. Each RODC maintains a local database of groups for specific administrative purposes. You can add a domain user account to these local roles to allow support of a specific RODC.

You can configure administrative role separation using the dsmgmt.exe command. To add a user to the Administrators role on an RODC, follow these steps:

1. Open a command prompt on the RODC.
2. Type **dsmgmt**, and then press ENTER.

3. Type **local roles**, and then press ENTER.

At the local roles prompt, you can type ? and press ENTER for a list of commands. You can also type list roles and press ENTER for a list of local roles.

4. Type **add *username* administrators**, where *username* is the pre-Windows 2000 logon name of a domain user, and then press ENTER.

You can repeat this process to add other users to the various local roles on an RODC.

Additional Reading

- RODCs are a valuable new feature for improving authentication and security in branch offices. Be sure to read the detailed documentation on the Microsoft Web site at: <http://go.microsoft.com/fwlink/?LinkId=168764>

Lab C: Configure Read-Only Domain Controllers

- Exercise 1: Install an RODC
- Exercise 2: Configure Password Replication Policy
- Exercise 3: Manage Credential Caching

Logon information

Virtual machine	6425B-HQDC01-A	6425B-BRANCHDC01-A
Logon user name	Pat.Coleman	
Administrative user name	Pat.Coleman_Admin	Administrator
Password	Pa\$\$w0rd	Pa\$\$w0rd

Estimated time: 60 minutes

Scenario

Contoso's security team has tasked you with increasing the security and monitoring of authentication against the enterprise's AD DS domain. Specifically, you are to improve the security of domain controllers in branch offices.

Exercise 1: Install an RODC

In this exercise, you will configure the server BRANCHDC01 as an RODC in the distant branch office. In order to avoid travel costs, you decide to do the conversion remotely, with the assistance of Aaron Painter, the desktop support technician and only IT staff member at the branch. Aaron Painter has already installed a Windows Server 2008 computer named BRANCHDC01 as a server in a workgroup. You will stage a delegated installation of an RODC so that Aaron Painter can complete the installation.

The main tasks for this exercise are as follows:

1. Prepare for the lab.
2. Stage a delegated installation of an RODC.
3. Run the Active Directory Domain Services Installation Wizard on a workgroup server.

► Task 1: Prepare for the lab

1. Start 6425B-HQDC01-A.
2. Log on to HQDC01 as **Pat.Coleman** with the password **Pa\$\$w0rd**.
3. Open **D:\Labfiles\Lab09c**.
4. Run **Lab09c_Setup.bat** with administrative credentials. Use the account **Pat.Coleman_Admin** with the password **Pa\$\$w0rd**.
5. The lab setup script runs. When it is complete, press any key to continue.
6. Close the Windows Explorer window, **Lab09c**.

► Task 2: Stage a delegated installation of an RODC

1. Run **Active Directory Users and Computers** as an administrator, with the user name **Pat.Coleman_Admin** and the password **Pa\$\$w0rd**.
2. Right-click the **Domain Controllers** OU, and then click **Pre-create Read only Domain Controller** account.

3. Step through the Active Directory Domain Services Installation Wizard, accepting all defaults. Use the computer name BRANCHDC01 and, on the **Delegation of RODC Installation and Administration** page, delegate installation to **Aaron.Painter_Admin**.

Note that when the wizard is complete, the server appears in the Domain Controllers OU with the DC Type column showing Unoccupied DC Account (Read-only, GC).

► **Task 3: Run the Active Directory Domain Services Installation Wizard on a workgroup server**

1. Start 6425B-BRANCHDC01-A.
2. Log on to BRANCHDC01 as **Administrator** with the password **Pa\$\$w0rd**.
3. Click **Start**, and then click **Run**.
4. Type **dcpromo**, and then press ENTER.

A window appears that informs you that the Active Directory Domain Services binaries are being installed. When installation is completed, the Active Directory Domain Services Installation Wizard appears.

5. Click **Next**.
6. On the **Operating System Compatibility** page, click **Next**.
7. On the **Choose A Deployment Configuration** page, click the **Existing forest** option, then click **Add a domain controller to an existing domain**, and then click **Next**.
8. On the **Network Credentials** page, type **contoso.com**.
9. Click the **Set** button.

A Windows Security dialog box appears.

10. In the **User Name** box, type **Aaron.Painter_Admin**.
11. In the **Password** box, type **Pa\$\$w0rd**, and then press ENTER.
12. Click **Next**.

13. On the **Select a Domain** page, select **contoso.com**, and then click **Next**.

A message appears to inform you that your credentials do not belong to the Domain Admins or Enterprise Admins groups. Because you have pre-staged and delegated administration of the RODC, you are able to proceed with the delegated credentials.

14. Click **Yes**.

A message appears to inform you that the account for BRANCHDC01 has been prestaged in Active Directory as an RODC.

15. Click **OK**.

A warning message appears that indicates the computer has a dynamically assigned IP address. BRANCHDC01 has a dynamically assigned IPv6 address. However, the server does have a fixed IPv4 address. IPv6 addresses are not being used in this course, so you can ignore this message.

16. Click **Yes, the computer will use a dynamically assigned IP address (not recommended)**.

17. On the **Location For Database, Log Files, And SYSVOL** page, click **Next**.

18. On the **Directory Services Restore Mode Administrator Password** page, type **Pa\$\$w0rd12345** in the **Password** and **Confirm Password** boxes, and then click **Next**.

In a production environment, you should assign a complex and secure password to the Directory Services Restore Mode Administrator account.

Also note that we modified the minimum password length in Lab A and as such need to meet the new minimum password length requirements.

19. On the **Summary** page, click **Next**.

20. In the progress window, select the **Reboot On Completion** check box. Active Directory Domain Services is installed on BRANCHDC01, the server reboots.

Results: After this exercise, you will have a new RODC named BRANCHDC01 in the contoso.com domain.

Exercise 2: Configure Password Replication Policy

In this exercise, you will configure domain-wide password replication policy and the password replication policy specific to BRANCHDC01.

The main tasks for this exercise are as follows:

1. Configure domain-wide password replication policy.
2. Create a group to manage password replication to the branch office RODC.
3. Configure password replication policy for the branch office RODC.
4. Evaluate resultant password replication policy.

► Task 1: Configure domain-wide password replication policy

- Who are the default members of Allowed RODC Password Replication Group?
- Who are the default members of Denied RODC Password Replication Group?
- Add the DNSAdmins group as a member of the Denied RODC Password Replication Group.
- Examine the password replication property for BRANCHDC01.
- What is the password replication policy for the Allowed RODC Password Replication Group? For the Denied RODC Password Replication Group?

► Task 2: Create a group to manage password replication to the branch office RODC

1. In the **Groups\Role OU**, create a new global security group called **Branch Office Users**.
2. Add the following users to the Branch Office Users group:
 - **Anav.Silverman**
 - **Chris.Gallagher**
 - **Christa.Geller**
 - **Daniel.Roth**

- ▶ **Task 3: Configure password replication policy for the branch office RODC**
 - Configure BRANCHDC01 so that it caches passwords for users in the **Branch Office Users** group.

- ▶ **Task 4: Evaluate resultant password replication policy**
 - Open the **Resultant Policy** for BRANCHDC01's password replication policy.

Question: What is the resultant policy for Chris.Gallagher?

Results: After this exercise, you will have configured the domain-wide password replication policy to prevent the replication of passwords of members of DNSAdmins to RODCs. You will have also configured the password replication policy for BRANCHDC01 to allow replication of passwords of members of Branch Office Users.

Exercise 3: Manage Credential Caching

In this exercise, you will monitor credential caching.

The main tasks for this exercise are as follows:

1. Monitor credential caching.
2. Pre-populate credential caching.

► Task 1: Monitor credential caching

1. Log on to BRANCHDC01 as **Chris.Gallagher** with the password **Pa\$\$w0rd**, and then log off.
2. Log on to BRANCHDC01 as **Mike.Danseglio** with the password **Pa\$\$w0rd**, and then log off.

The contoso.com domain used in this course includes a Group Policy object (named 6425B) that allows users to log on to domain controllers. In a production environment, it is not recommended to give users the right to log on to domain controllers.

3. On HQDC01, in **Active Directory Users and Computers**, examine the password replication policy for BRANCHDC01.

Question: What users' passwords are currently cached on BRANCHDC01?

Question: What users have been authenticated by BRANCHDC01?

► Task 2: Pre-populate credential caching

- In the password replication policy for BRANCHDC01, pre-populate the password for **Christa Geller**.

Results: After this exercise, you will have identified the accounts that have been cached on BRANCHDC01, or have been forwarded to another domain controller for authentication. You will have also prepopulated the cached credentials for Christa Geller.

Lab Review Questions

Question: Why should you ensure that the PRP for a branch office RODC has, in its Allow list, the accounts for the *computers* in the branch office as well as the users?

Question: What would be the most manageable way to ensure that computers in a branch are in the Allow list of the RODC's PRP?

Question: What are the pro's and con's of prepopulating the credentials for all users and computers in a branch office to that branch's RODC?