
Module 6: Delegating Administrative Control

Contents

Overview	1
Object Security in Active Directory	2
Controlling Access to Active Directory Objects	13
Delegating Administrative Control of Active Directory Objects	21
Lab A: Delegating Administrative Control	27
Customizing MMC Consoles	36
Setting Up Taskpads	41
Lab B: Creating Custom Administrative Tools	45
Best Practices	50
Review	51



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, places or events 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.

© 2000 Microsoft Corporation. All rights reserved.

Microsoft, Active Directory, BackOffice, FrontPage, IntelliMirror, PowerPoint, Visual Basic, Visual Studio, Win32, Windows, Windows Media, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the U.S.A. and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Instructor Notes

Presentation:
105 Minutes

Labs:
60 Minutes

This module provides students with the knowledge and skills to efficiently delegate administrative control of Active Directory® directory service objects in Microsoft® Windows® 2000. Students will learn how to grant users access to Active Directory objects and to create customized tools to match specific administrative responsibilities. They will also learn the different methods and strategies to use when delegating administrative control in Active Directory.

At the end of this module, students will be able to:

- Manage object security in Active Directory.
- Control access to Active Directory objects.
- Delegate administrative control of Active Directory objects.
- Create and deploy customized consoles.
- Create and deploy customized taskpads.
- Apply best practices when delegating administrative control.

In the two hands-on labs in this module, students will have a chance to delegate administrative control in Active Directory. In the first lab, students will view permissions on Active Directory objects and delegate control of an organizational unit (OU). In the second lab, students will create custom administrative tools.

Materials and Preparation

This section provides you with the required materials and preparation tasks that are needed to teach this module.

Required Materials

To teach this module, you need the following materials:

- Microsoft PowerPoint® file 2154A_06.ppt

Preparation Tasks

To prepare for this module, you should:

- Read all of the materials for this module.
- Complete the labs.
- Study the review questions and prepare alternative answers to discuss.
- Anticipate questions that students may ask. Write out the questions and provide the answers.
- Read chapter 12, “Distributed Security” in the Distributed Systems book in the Microsoft Windows 2000 Server Resource Kit.
- Read the white paper, *Windows 2000 Kerberos Authentication*, on the Student Materials compact disc.
- Read the white paper, *Microsoft Management Console: Overview*, on the Student Materials compact disc.

Module Strategy

Use the following strategy to present this module:

- **Object Security in Active Directory**

In this topic, you will introduce delegating administrative control of Active Directory objects. Begin the module with a discussion of the security components that constitute the access control of all objects in Active Directory. Describe the concepts of discretionary and system access control lists and how access control information is passed down through inheritance. Illustrate the Windows 2000 logon process and explain how Windows 2000 uses access tokens to grant users access to resources.

- **Controlling Access to Active Directory Objects**

In this topic, you will introduce the permissions that are applied to objects in Active Directory. Illustrate how to control inheritance of permissions in Active Directory and demonstrate how to assign permissions. Describe the concept of object ownership and explain how to change the ownership of an object in Active Directory.

- **Delegating Administrative Control of Active Directory Objects**

In this topic, you will introduce how to delegate administrative control at the OU level in Active Directory. Demonstrate how to assign permissions at the OU level by using the Delegation of Control wizard and identify the guidelines for delegating administrative control of objects in Active Directory.

- **Lab A: Delegating Administrative Control**

Prepare students for the lab in which they will review the default security settings on Active Directory and delegate control over objects in an OU. Tell the students to note the different Active Directory permissions for the OU before and after they delegate control of the OU. After students have completed the lab, ask them if they have any questions concerning the lab.

- **Customizing MMC Consoles**

In this topic, you will introduce how to customize Microsoft Management Console (MMC) consoles. List the tasks for customizing an MMC console and demonstrate how to create and customize an MMC console. Illustrate the procedures for distributing customized MMC consoles and installing snap-ins in Windows 2000.

- **Setting Up Taskpads**

In this topic, you will introduce the setting up of taskpads. Describe a taskpad and show students what a completed taskpad looks like. Explain the procedures for creating and configuring a taskpad, and adding tasks in a taskpad.

- **Lab B: Creating Custom Administrative Tools**

Prepare students for the lab in which they will create a custom administrative tool by using MMC console and create a taskpad. After students have completed the lab, ask them if they have any questions concerning the lab.

- **Best Practices**

Present best practices for delegating administrative control of Active Directory objects. Emphasize the reason for each best practice.

Customization Information

This section identifies the lab setup requirements for a module and the configuration changes that occur on student computers during the labs. This information is provided to assist you in replicating or customizing Microsoft Official Curriculum (MOC) courseware.

Important The labs in this module are also dependent on the classroom configuration that is specified in the Customization Information section at the end of the Classroom Setup Guide for Course 2154, *Implementing and Administering Microsoft Windows 2000 Directory Services*.

Lab Setup

The labs in this module require that the student computers be configured as domain controllers. To prepare student computers to meet this requirement, perform one of the following actions:

- Complete Module 3, “Creating a Windows 2000 Domain,” in Course 2154, *Implementing and Administering Microsoft Windows 2000 Directory Services*.
- Run Autodc.vbs from the C:\Moc\Win2154A\Labfiles\Custom\Autodc folder.
- Run Dcpromo.exe on the student computers by using the following parameters:
 - A domain controller for a new domain.
 - A new domain tree.
 - A new forest of domain trees.
 - A full DNS domain name, which is *computerdom.nwtraders.msft* (where *computer* is the assigned computer name).
 - A NetBIOS domain Name, which is *COMPUTERDOM*.
 - Default location for the database, log files, and SYSVOL.
 - Permission compatible only with Windows 2000–based servers.
 - Directory Services Restore Mode administrator password, which is **password**.

Note Before you use Module 3, “Creating a Windows 2000 Domain,” in Course 2154, *Implementing and Administering Microsoft Windows 2000 Directory Services*, you must successfully complete Module 2, “Implementing DNS to Support Active Directory,” in Course 2154, *Implementing and Administering Microsoft Windows 2000 Directory Services*.

Lab Results

Performing the labs in this module introduces the following configuration changes:

- An OU called Security is created.
- The following user accounts are created in the Security OU:
 - Assistant User
 - Secretary User
 - Password Reset
- The Assistant User account is delegated control of user accounts in the Security OU.
- The Password Reset account is delegated Reset Password permissions for the entire domain.
- The Users group is granted the Log on Locally right.

Important You can run the Undel.vbs script in the C:\Moc\Win2154A\Labfiles\Custom\Undel folder to remove all configuration changes introduced during the course of the labs in this module.

Overview

Topic Objective

To provide an overview of the module topics and objectives.

Lead-in

In this module, you will learn how to delegate administrative control of Active Directory objects and properties.

- Object Security in Active Directory
- Controlling Access to Active Directory Objects
- Delegating Administrative Control of Active Directory Objects
- Customizing MMC Consoles
- Setting Up Taskpads
- Best Practices

*****ILLEGAL FOR NON-TRAINER USE*****

The Microsoft® Windows® 2000 Active Directory® directory service provides administrators with a high degree of control over who has access to information in Active Directory. By managing the permissions on directory objects and properties, administrators can precisely specify which accounts can gain access to Active Directory and the level of access that these accounts can have. This precision allows administrators to delegate specific authority over portions of Active Directory to groups of users, without making the information in Active Directory vulnerable to unauthorized access. The ability to delegate relieves the burden of centralized administration.

Controlling access and delegating administrative authority to Active Directory objects is important, especially when developing a decentralized administrative model.

At the end of this module, you will be able to:

- Manage object security in Active Directory.
- Control access to Active Directory objects.
- Delegate administrative control of Active Directory objects.
- Create and deploy customized consoles.
- Create and deploy customized taskpads.
- Apply best practices for delegating administrative control.

◆ Object Security in Active Directory

Topic Objective

To introduce how Windows 2000 ensures secure access to information in Active Directory.

Lead-in

Access to information in Active Directory can be controlled down to the object attribute level.

- Active Directory Security Components
- Discretionary and System Access Control Lists
- Access Control Entries
- Inheritance
- The Logon Process
- Access Tokens
- How Windows 2000 Grants Access to Resources

*****ILLEGAL FOR NON-TRAINER USE*****

Windows 2000 implements an object-based security model and access control for all objects in Active Directory. *Access control* is the process of authorizing users, groups, and computers to access objects on the network. Several security components in Active Directory make up access control and allow access control information to be passed down through inheritance. Inheritance enables the access control information defined at higher-level containers in Active Directory to flow down to sub-containers and their objects.

Windows 2000 requires users to log on, and then after Windows 2000 and Active Directory authenticate the user's unique identity, Windows 2000 grants or denies access to resources.

Active Directory Security Components

Topic Objective

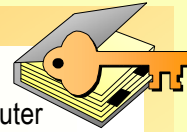
To describe the security components of Active Directory.

Lead-in

The basic components of access control in Active Directory include security descriptors, security principals, and security identifiers.

■ Security Principals

- User, security group, service, and computer
- Identified by a unique ID



■ Security Identifiers (SIDs)

- Uniquely identify security principals
- Are never reused

■ Security Descriptors

- Security information associated with an object
- Contains DACLs and SACLs

*****ILLEGAL FOR NON-TRAINER USE*****

DACLs and SACLs are mentioned on this page, but are discussed in detail in the Discretionary and System Access Control Lists page.

Key Points

Security principals, security identifiers, and security descriptors are the basic components of access control in Active Directory.

Security principals are user, security group, service, and computer accounts.

Security identifiers (SIDs) are alphanumeric structures that uniquely identify user, security group, and computer accounts within an organization.

Each object has a security descriptor that stores access control information associated with an object.

Each object in Active Directory is associated with a unique *security descriptor* that defines the access permissions that are required to read or update the object properties. Permissions are assigned at the property level. Security principals, security identifiers, and security descriptors are the basic components of the access control model.

Security Principals

A *security principal* is an account holder to which you can assign permissions. Examples of security principals are user, security group, and computer accounts. Each security principal within a Windows 2000 domain is identified by a unique security identifier.

Security Identifiers (SIDs)

A *security identifier (SID)* is a value that uniquely identifies a user, group, service, or computer account within an organization. Every account is issued a SID when it is created. Access control mechanisms in Windows 2000 identify security principals by SID rather than by name. After a SID is issued to an account, it is never reused on another account.

Security Descriptors

A security descriptor is a data structure containing the security information associated with a securable object. A security descriptor identifies an object's owner by SID. If permissions are configured for the object, its security descriptor contains a discretionary access control list (DACL) with SIDs for the users and groups who are allowed or denied access. If auditing is configured for the object, its security descriptor also contains a system access control list (SACL) that controls how the security subsystem audits attempt to access the object.

Discretionary and System Access Control Lists

Topic Objective

To illustrate discretionary and system access control lists (ACLs).

Lead-in

An access control list (ACL) is a list of security protections that apply to an object and its properties.

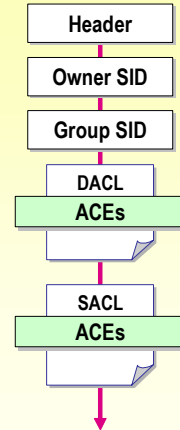
■ Discretionary Access Control List (DACL)

- Identifies the security principals that are allowed or denied access, and the level of access being allowed or denied

■ System Access Control List (SACL)

- Controls how object access will be audited

Security Descriptor



*****ILLEGAL FOR NON-TRAINER USE*****

Tell the class that because the rest of this module addresses security, the module will refer only to DACL and not to SACL.

Key Points

An access control list (ACL) is an ordered list of access control entries (ACEs) that define the security protections applicable to an object, a set of the object's properties, or an individual property of an object.

There are two types of ACLs in an object's security descriptor, DACLs and SACLs.

DACLs determine whether to allow or deny access to an object.

A security descriptor is a binary data structure of variable length that contains an access control list (ACL). An ACL is an ordered list of access control entries (ACEs) that define the security protections applicable to an object, a set of the object's properties, or an individual property of an object. The data structure of a security descriptor has the following parts:

- Header.** The header field contains a revision number and a set of control flags that describe characteristics of the security descriptor, such as the memory layout, which elements are present, and how particular elements were added or modified.
- Owner.** The Owner field contains the SID for the object's owner. The owner of an object can modify permissions and grant other users the right to take ownership.
- Primary Group.** The Primary Group field contains the SID for the owner's primary group. This information is used for services with Macintosh and by the POSIX subsystem but is ignored by the rest of Windows 2000.
- Discretionary access control list (DACL).** The DACL is a list of zero or more ACEs identifying who is allowed or denied access, and the level of access being allowed or denied.
- System access control list (SACL).** The SACL is similar to the DACL except that it is used to control how Windows 2000 audits access to objects. When an audited action occurs, the operating system records the event in the security log.

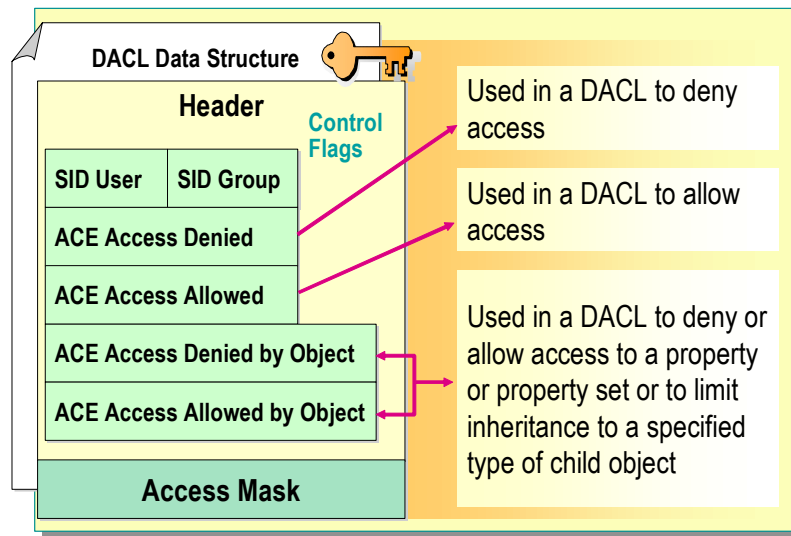
Access Control Entries

Topic Objective

To identify the purpose of access control entries (ACEs).

Lead-in

ACEs are used to determine which operations a security principal is allowed to perform on an object.



*****ILLEGAL FOR NON-TRAINER USE*****

Key Points

There are six different types of ACEs in Windows 2000.

An access control entry (ACE) is used to determine which operations a security principal is allowed to perform on an object, or in the case of ACEs in a SACL, which operations are audited.

ACEs in a DACL

Each ACE in a DACL consists of:

- A SID is a value that uniquely identifies a user, group, service, or computer account within an organization.
- A header that specifies whether the ACE allows or denies access.
- An access mask that lists the operations allowed or denied.
- A set of bit flags that determine whether child objects can inherit the ACE. The header for an ACE contains a set of inheritance flags that control how the ACE is inherited and how the ACE affects a child object that inherits it.
- A flag that indicates the type of ACE.

ACE Types

There are six different types of ACEs in Windows 2000. These six types of ACEs are divided into two groups, generic ACEs, which can be applied to all objects to which security can be applied, including file system objects and Active Directory objects, and object-specific ACEs, which can be applied only to Active Directory objects.

Generic and object-specific ACEs are fundamentally alike. The difference between them is the granularity of control they offer over inheritance and object access.

The following table lists the different ACE types:

Type	Description
Access denied	Used in a DACL to deny access.
Access allowed	Used in a DACL to allow access.
System audit	Used in a SACL to log access attempts.
Access denied, object specific	Used in a DACL to deny access to a property or property set, or to limit inheritance to a specified type of child object.
Access allowed, object specific	Used in a DACL to allow access to a property or property set, or to limit inheritance to a specified type of child object.
System audit, object specific	Used in a SACL to log attempts to access a property or property set, or to limit inheritance to a specified type of child object.

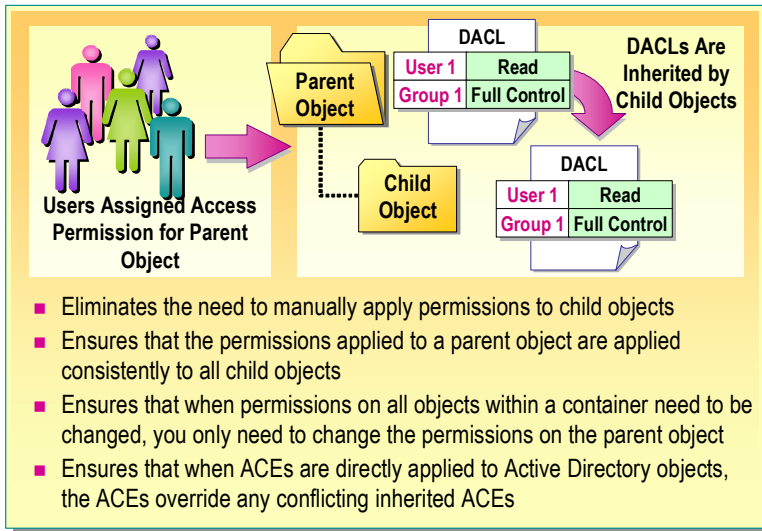
Inheritance

Topic Objective

To explain how inheritance flows from parent objects to child objects.

Lead-in

Child objects inherit parent rules.



*****ILLEGAL FOR NON-TRAINER USE*****

Key Points

Inheritance is propagated when:

A new child object is created.

The DACL on the parent object is modified.

The SACL on the parent object is modified.

Inheritance is the process that passes ACEs in a parent object's security descriptor to a child object's security descriptor. Inheritance means that access control information that is defined at higher-level containers in Active Directory flows down to sub-containers and their objects. In Windows 2000, inheritable ACEs are propagated from a parent object to a child object when one of the following three events takes place:

- A child object is created.
- The DACL on the parent object is modified.
- The SACL on the parent object is modified.

Any object can be the child of another object. Only container objects can be parents. The DACL and SACL for a container object can carry ACEs that are not needed on the container but are present only for the purpose of inheritance, so that the ACEs are passed down to subsequent generations of objects until they reach a non-container child object, where they become effective ACEs.

Permission inheritance in Windows 2000 simplifies the task of managing permissions by:

- Eliminating the need to apply permissions manually to child objects as they are created.
- Ensuring that the permissions applied to a parent object are applied consistently to all child objects.
- Ensuring that when you need to modify permissions on all objects within a container, you only need to change the permissions on the parent object, and the child objects automatically inherit those changes.
- Ensuring that when ACEs are directly applied to Active Directory objects, they are applied before any conflicting previously inherited ACEs.

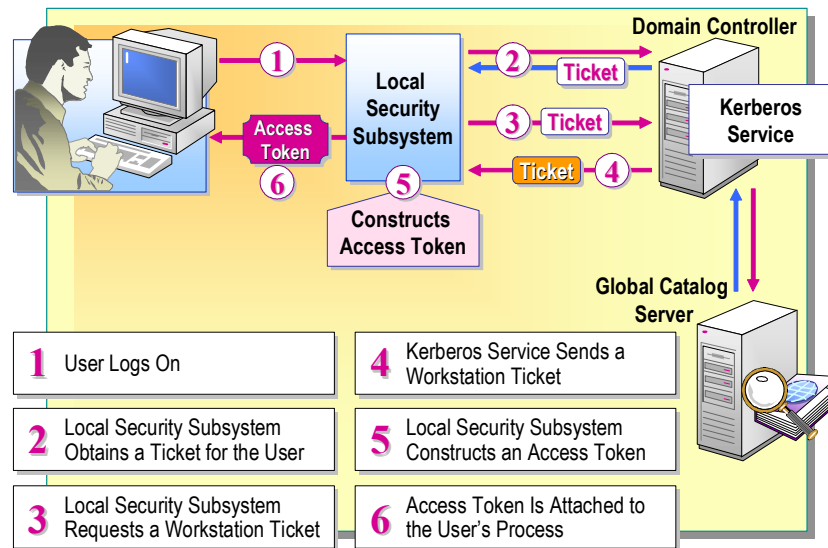
The Logon Process

Topic Objective

To describe the logon process.

Lead-in

The authentication process in Windows 2000 ensures that only valid users have access to network or computer resources.



*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Using the steps in the illustration, demonstrate the steps of the network and secondary logon process.

Key Points

A Key Distribution Center (KDC) is required to authenticate the logon process in a Windows 2000 native-mode domain. If a KDC is not available, domain authentication fails and the user is logged on using cached credentials.

A global catalog server is required to log on in a native-mode domain to determine universal group membership. If a global catalog server is not found, Active Directory is queried directly to determine universal group membership.

If a global catalog server is unavailable for logon in a multi-domain enterprise, the user will be logged on using cached credentials.

Windows 2000 controls access to resources by requiring a user to first log on to a computer. To log on to a computer, Windows 2000 requires each user to provide a unique user name and password. The logon process that occurs for a Windows 2000 computer includes the following steps:

1. A user logs on, providing the required security credentials, including user name, password, and domain name. These credentials are passed to the security subsystem on the local computer.
2. The local security subsystem uses DNS to locate a domain controller in the user's domain. The security subsystem then contacts the Kerberos service, called the Key Distribution Center, running on the domain controller, and requests a session ticket for the user to communicate with the Kerberos service. A *ticket* is a record that allows a client computer to authenticate itself to a server. The Kerberos service queries Active Directory to authenticate the user and contacts a global catalog server to obtain the user's universal group memberships. The Kerberos service then returns a session ticket to the client computer that contains the user's SID and the user's universal, global, and domain local group memberships, which are used for future transactions with the Kerberos service.

Note Every domain controller in the domain runs the Kerberos service and is capable of granting session tickets for users and computers. If a domain controller is not available, domain authentication fails and the user is logged on by cached logon credentials at the client computer. The client computer periodically attempts to locate the Kerberos service during the user's session, and will complete the domain authentication process if one is found.

Emphasize that for more detailed information on Kerberos, students should see *Windows 2000 Kerberos Authentication* under **Additional Reading** on the Web page on the Student Materials compact disc.

3. The local security subsystem again contacts the Kerberos service on the domain controller and requests another session ticket authorizing the user to gain access to the Workstation service on the client computer to complete the user logon process. This request includes a copy of the user's session ticket that the Kerberos service uses to identify the user.
4. The Kerberos service authenticates the user's ticket by querying Active Directory and the global catalog server to verify the information contained in the user's session ticket. The Kerberos service then constructs a Workstation session ticket for the user that contains the validated security credentials copied from the user's original ticket, and returns the session ticket to the client computer.
5. The local security subsystem on the client computer extracts the user's SID and universal, global, and domain local group memberships from the Workstation session ticket. The subsystem then constructs the user's access token by adding the SIDs for local groups to which the user belongs and a list of the local user rights assigned to the user.
6. The local computer creates a process with an access token attached. The access token is used to authenticate the user and serves as an identity card whenever the user attempts to use system resources.

The Network Logon Process

A *network logon* occurs when a user establishes a network connection to a remote computer running Windows 2000, for example, when connecting to a shared folder. The authentication process is very similar to that of an interactive logon process.

The client computer obtains a server session ticket from the Kerberos service running on a domain controller in the user's domain. The client computer then sends the server session ticket to the local security subsystem on the server, which extracts the user's security credentials and constructs an access token for the remote user. This access token is used to authenticate the user whenever a resource on the server is accessed.

The Secondary Logon Process

Secondary logon provides the ability to start and run an application by using the security credentials of another user without ending a session already in progress. For example, you can run administrative tools while logged on with a standard user account.

Important For more detailed information about the steps in the Kerberos V5 authentication process, see *Windows 2000 Kerberos Authentication* under **Additional Reading** on the Web page on the Student Materials compact disc.

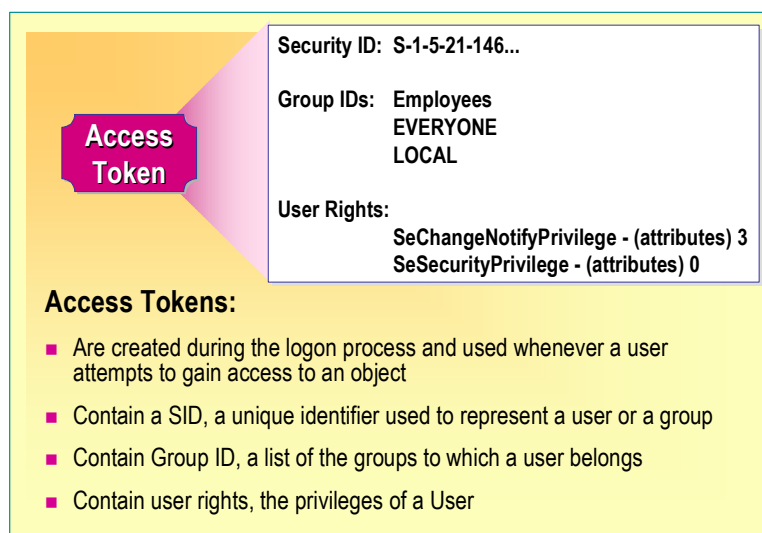
Access Tokens

Topic Objective

To explain the purpose and components of access tokens.

Lead-in

Access tokens are essential components of security in Windows 2000.



*****ILLEGAL FOR NON-TRAINER USE*****

Key Points

The main components of an access token are SIDs, the Group ID, and user rights.

An access token is permanently attached to a user's process.

Windows 2000 verifies the user SID and the list of group SIDs in the access token against the object's DACL before granting access to a resource.

To gain access to any resource on the network, a user must have an access token. An *access token* is created for the user during the logon process and contains attributes that establish the security credentials for that user on the local computer. The access token is used whenever a user attempts to gain access to an object. When the user runs an application, a new process is started that inherits the user's access token. The access token is permanently attached to each of the user's processes and serves as an identity card whenever the user attempts to use system resources. When a user's process attempts to gain access to any object, Windows 2000 verifies the user's SID and the list of Group IDs in the access token against the object's DACL. This verification determines whether the user is granted access to the object.

Security Identifier (SID)

The SID is the security identifier for the user who is logged on. A SID allows the operating system to uniquely identify each user and group account, even if that account is renamed or has the same name as another account. In this way, permissions assigned to an object can be used only by that object, regardless of what the user or group is named.

Group ID

The *Group ID* is a list of the groups to which the user belongs. For a domain logon process, the domain controller compiles a list of the SIDs for the global and domain local groups of which the user is a member. The domain controller contacts a global catalog server to obtain the SIDs of any universal groups of which the user is a member. This list is returned to the client computer, which then adds any local groups of which the user is a member.

User Rights

User rights are the privileges of the user. The local computer adds the list of user rights to the access token. User rights determine which administrative actions the user can perform on the local computer, such as shutting down the computer, logging on interactively, and taking ownership of objects.

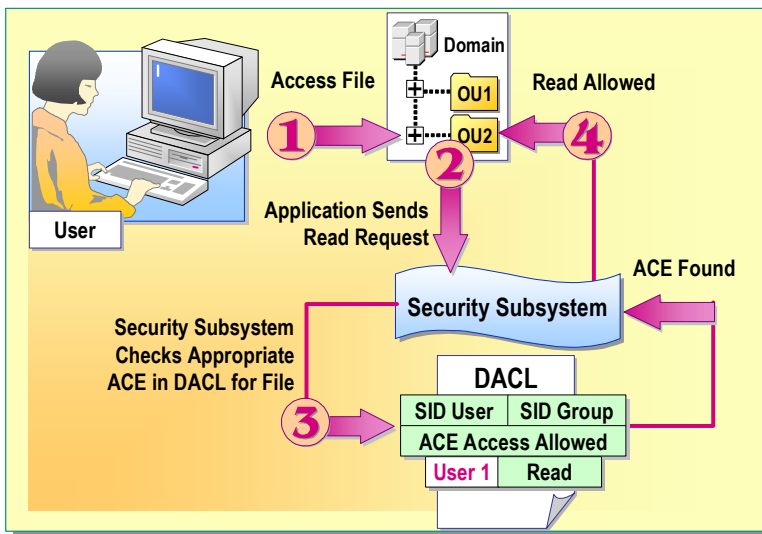
How Windows 2000 Grants Access to Resources

Topic Objective

To illustrate how Windows 2000 grants access to resources.

Lead-in

Windows 2000 uses access tokens and DACLs to grant access to resources.



*****ILLEGAL FOR NON-TRAINER USE*****

Key Points

The process of accessing an Active Directory object is identical to the process of accessing any NTFS file system object. This process is known as authorization.

Users gain access to resources when Windows 2000 checks the DACL list of allowed permissions against the user's requested access. This process is known as *authorization*. Windows 2000 controls access to resources in two ways:

- By requiring users to log on using a set of verifiable security credentials. These credentials are then compared against a set of permissions assigned to Active Directory objects and network resources, such as shared folders and NTFS file system files.
- By granting access to only those resources that the user has permission to use. After the user's unique identity has been authenticated by Windows 2000 and then by Active Directory, the user can receive access to specific resources on the network from any computer in any domain of the organization.

The user gains access to a resource through the following process:

1. The user requests access to an Active Directory object. For example, a user requests Read access to an object in an organizational unit (OU) by attempting to display the **Properties** dialog box for a user account.
2. By attempting to display the **Properties** dialog box, the user causes Active Directory Users and Computers to generate an input/output (I/O) request to Windows 2000, which validates the request through the security subsystem.
3. The security subsystem reads the DACL for an object, searching for ACEs that contain the user's SID or the SID of a group to which the user belongs. Each ACE that applies to the user is compared against the requested access until an ACE that denies or allows the requested access is located. If a denial is encountered, or no ACE exists for the requested access, the user's request fails.

ACEs that deny access are listed first in the DACL. The security subsystem first processes the ACEs that deny access, and grants access to the object as soon as an ACE that allows the requested access is encountered.

4. If access is granted, the resource is opened for only the requested access. If the user is denied access, an error message appears.

A DACL is checked only when the resource is initially opened. If a user's permissions for an object are changed while the user is accessing the object, the user retains the current access to the object. The permission assignment is updated the next time the user gains access to the object.

◆ Controlling Access to Active Directory Objects

Topic Objective

To introduce ways in which access to Active Directory objects is controlled.

Lead-in

You can use permissions to grant administrative privileges—for an OU, a hierarchy of OUs, or a single object—to a specific user or group.

- Active Directory Permissions
- Controlling Inheritance of Permissions
- Setting Active Directory Permissions
- Object Ownership
- Changing Object Ownership

*****ILLEGAL FOR NON-TRAINER USE*****

When you want to control which users have access to specific objects in Active Directory, consider the following:

- The permissions that you are allowed to attach to the object provide security for resources by allowing you to control which users can gain access to individual objects or object attributes, and the users' level of access. You can use permissions to grant administrative privileges for a single object, or to a specific user or group.
- Every object in Active Directory has an owner. The owner controls how permissions are set on an object and to whom permissions are assigned.

Active Directory Permissions

Topic Objective

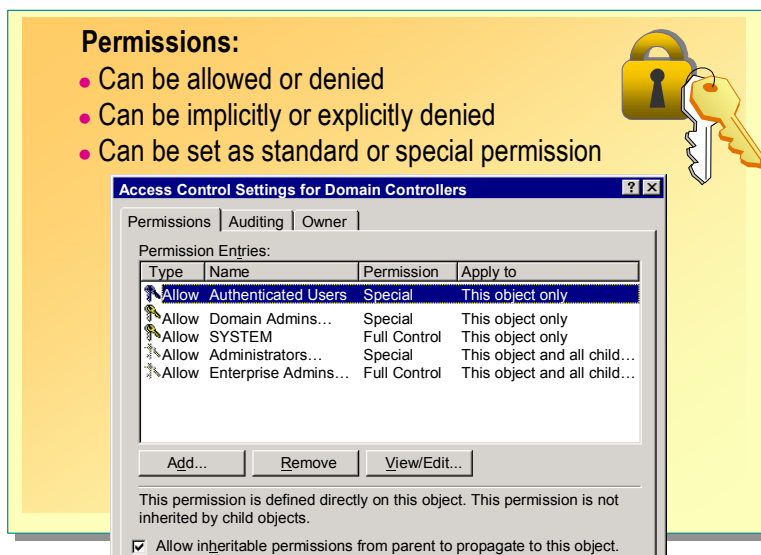
To describe how permissions are applied in Active Directory.

Lead-in

You control access to network resources by assigning permissions.

Permissions:

- Can be allowed or denied
- Can be implicitly or explicitly denied
- Can be set as standard or special permission



*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Demonstrate how to view the permissions for an object by using the **Access Control Settings** dialog box. Use the **Permission Entries** tab to show the assigned permissions.

Key Points

You can allow or deny permissions for every object in Active Directory.

Permissions can be implicitly or explicitly denied.

Permission is an authorization assigned by an owner so that users can perform an operation on a specific object, such as a user account. If you own an object, you can assign user or security group permission to perform some or all of the tasks that you are authorized to do. You can also assign permission to take ownership.

Every permission that an object's owner assigns to a particular user or group is stored as an ACE in a DACL that is part of the object's security descriptor. Users can view ACEs under **Permission Entries** in the **Access Control Settings** dialog box.

Allowing and Denying Permissions

You can allow or deny permissions. Denied permissions take precedence over any permissions that you otherwise allow for user accounts and groups. For example, if you deny permission for a user to gain access to an object, the user will not have that permission, even if you allow the permission for a group of which the user is a member. Deny permissions only when it is necessary to remove a permission that a user may have been assigned through a group membership.

Implicit or Explicit Permissions

You can implicitly or explicitly deny permissions as follows:

- When permission to perform an operation is not explicitly assigned, it is *implicitly denied*.

For example, if the Marketing group is allowed Read permission on a user object, and no other security principal is listed on the DACL for that object, users who are not members of the Marketing group are implicitly denied access. The operating system does not allow users who are not members of the Marketing group to read the properties of the user object.

- Permissions can also be *explicitly denied*.

For example, it may be necessary to prevent a user named Don from being able to view the properties of a user object, even though he is a member of the Marketing group that has permissions view the properties of the user object. You can prevent Don from accessing the user object properties by explicitly denying him Read permission. This example ideally illustrates the use of explicit denials, which is to exclude a subset, such as Don, within a larger group, such as Marketing, from performing a task that the larger group has permissions to perform.

Standard and Special Permissions

You can set *standard* and *special* permissions on objects. Standard permissions are the most frequently assigned permissions. Special permissions provide you with a finer degree of control for assigning access to objects.

The following table lists standard object permissions that are available for most objects, and the type of access that each permission allows the user to have.

Object permission	Allows the user to
Full Control	Change permissions and take ownership, plus perform the tasks that are allowed by all other standard permissions.
Read	View objects and object attributes, the object owner, and the Active Directory permissions.
Write	Change object attributes.
Create All Child Objects	Add any type of child object to an OU.
Delete All Child Objects	Remove any type of child object from an OU.

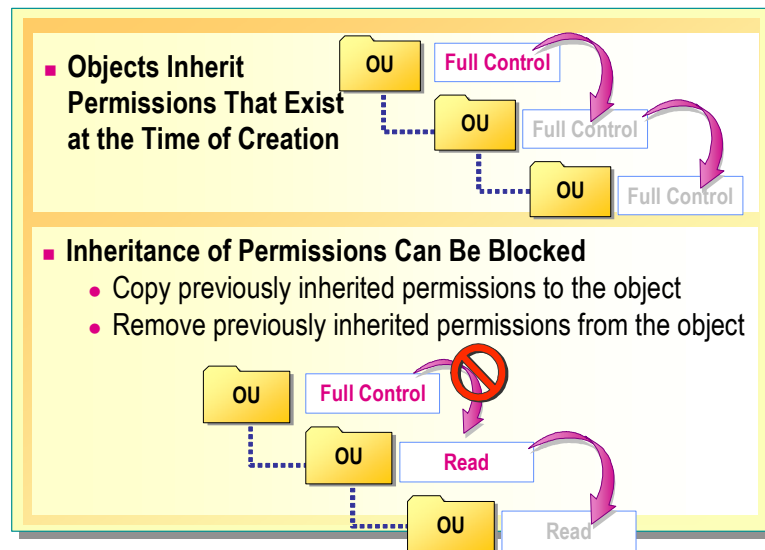
Controlling Inheritance of Permissions

Topic Objective

To illustrate how to control inheritance of permissions.

Lead-in

You can use permission inheritance to minimize the number of times you need to assign permissions for objects.



*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Explain that when you copy previously inherited permissions, you are starting with the same permissions that the object currently inherits from its parent object. However, any permission for the parent object that you modify after blocking inheritance no longer applies.

Demonstrate how to prevent inheritance by using the **Security** tab in the **Properties** dialog box for the User OU.

Permission inheritance in Active Directory automatically causes objects within a container to inherit the permissions of that container. For example, the files within a folder, when created, inherit the permissions of the folder. This inheritance minimizes the number of times that you need to assign permissions for objects. When an object is created within a container, the Active Directory schema defines a default DACL for the object.

Applying Permissions to Child Objects

When you assign permissions, you can have the permission apply to the object's child objects. For example, if you want a user to administer all objects in an OU, assign Full Control permissions to the user, and all child objects then inherit this permission. To indicate that permissions have been inherited, the check boxes in the **Permissions** dialog box for child objects are dimmed.

Preventing Child Objects from Inheriting Permissions

You can prevent permission inheritance so that a child object does not inherit permissions from its parent object. You prevent inheritance when you want to set more restrictive permissions on child objects than on a parent object. When you prevent inheritance, only the permissions that you explicitly assign to the object apply.

When you prevent permission inheritance, Windows 2000 allows you to:

- Copy previously inherited permissions to the object. Then, according to your needs, you can make any necessary changes to the permissions.
- Remove previously inherited permissions from the object. Then, according to your needs, you can assign new permissions for the object.

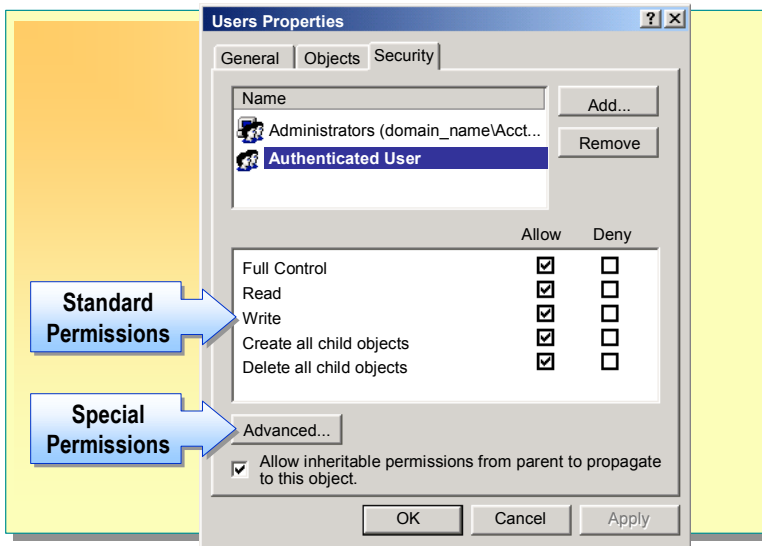
Setting Active Directory Permissions

Topic Objective

To explain how to assign permissions.

Lead-in

Windows 2000 verifies permissions before allowing access to an object.



*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Demonstrate the steps for adding or changing permissions for an object and viewing special permissions for an object.

Windows 2000 determines a user's authorization to use an object by checking the DACL for permissions assigned to the user on that object. These permissions are visible in Active Directory by viewing an object's **Properties** dialog box.

Standard Permissions

To add or change permissions for an object, perform the following steps:

1. In Active Directory Users and Computers, on the **View** menu, click **Advanced Features**.
2. Right-click the object, click **Properties**, and then in the **Properties** dialog box, click the **Security** tab.
3. Perform either or both of the following steps:
 - To add a new permission, click **Add**, click the user account or group to which you want to assign permissions, click **Add**, and then click **OK**.
 - To remove a permission, select the user account or group that you want to remove, click **Remove**, and then click **OK**.
4. In the **Permissions** box, select the **Allow** or **Deny** check box for each permission that you want to add or change.

Special Permissions

Standard permissions are sufficient for most administrative tasks. However, you may need to view the special permissions available within a standard permission to further refine the access permissions.

To view special permissions, perform the following steps:

1. On the **Security** tab in the **Properties** dialog box for the object, click **Advanced**.
2. In the **Access Control Settings** dialog box, on the **Permissions** tab, click the entry that you want to view, and then click **View/Edit**.
3. To view the permissions for specific attributes, click the **Properties** tab.

Caution Avoid assigning permissions for specific attributes of objects. Errors can result, such as objects in Active Directory not being visible, preventing users from completing tasks.

Object Ownership

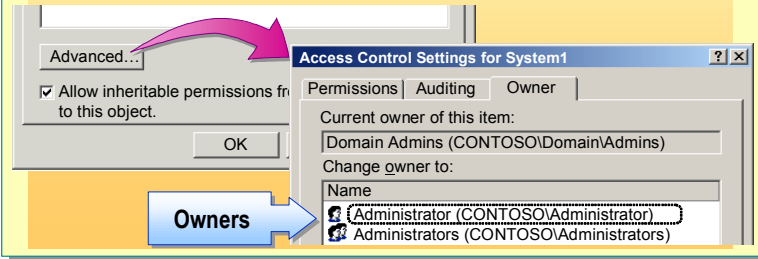
Topic Objective

To explain the concept of ownership.

Lead-in

Every object in Active Directory has an owner.

- Every Object Has an Owner
- The Owner Controls How Permissions Are Set on an Object, and to Whom Permissions Are Assigned
- If a Member of the Administrators Group Takes Ownership, the Default Owner Is the Group, Not the Individual User



*****ILLEGAL FOR NON-TRAINER USE*****

Key Points

Object owners have full control over the object by default.

Delivery Tip

Emphasize to students that unlike in Windows NT® version 4.0, when an administrator takes ownership, that person as well as the Administrators group are listed as the owners.

Every object in Active Directory has an owner. The person who creates the object automatically becomes the owner and, by default, has full control over the object. The owner controls how permissions are set on an object, and to whom permissions are granted, even if the owner is not listed in the DACL.

The default owner of an object is normally an individual user who is currently logged on. The only exceptions occur when the user is a member of either the Administrators group or the Domain Admins group. If a member of the Administrators group takes ownership, the default owner is the group, not the individual user. In both cases, the Owner field in the user's access token contains the SID for the group, not the SID for the individual user account.

To determine who owns which objects, perform the following steps:

1. Right-click the object, and then click **Properties**.
2. In the **Properties** dialog box, click the **Security** tab, and then click **Advanced**.
3. In the **Access Control Settings** dialog box, click the **Owner** tab.
4. The name of the object's owner is shown after **Current owner of this item** box.

Note For more information about object ownership, see chapter 12, "Distributed Security" in the Distributed Systems Guide in the Microsoft Windows 2000 Server Resource Kit.

Changing Object Ownership

Topic Objective

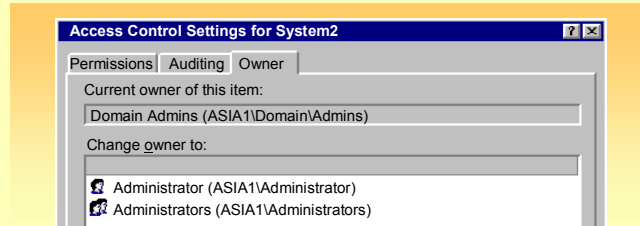
To explain how to change ownership of an object.

Lead-in

You can take ownership of any object, and then change the permissions for the object.

Ownership Changes When:

- The current owner assigns the Modify Ownership permission to other users
- Members of the Domain Admins group take ownership of any object in the domain



*****ILLEGAL FOR NON-TRAINER USE*****

As an administrator, you can take ownership of any object, and then change the permissions for the object.

You can change ownership in the following ways:

- The current owner, any user with Full Control permission, or any user with Change Permission assigns the Modify Owner permission to another user who takes ownership of the object.

For example, if an employee who owns an object leaves the company, you can let another user take ownership of that object, thereby reassigning responsibility for the object.

- A member of the Administrators group takes ownership of any object.

Note Although members of the Administrators group can take ownership of an object, members of the Administrators group cannot transfer ownership.

To change ownership of an object, perform the following steps:

1. In the **Properties** dialog box for an object, on the **Security** tab, click **Advanced**.
2. Click the **Owner** tab, and then click your user account. If you are a member of the Administrators group, you can also click **Administrators**. This makes the Administrators group the owner.
3. Click **OK**, and then click **OK** again to take ownership.

◆ Delegating Administrative Control of Active Directory Objects

Topic Objective

To introduce the topics related to delegating administrative control of Active Directory objects.

Lead-in

You delegate administrative control of Active Directory objects by assigning permissions to the objects to allow users or groups of users to administer the objects.

- Overview of Delegating Administrative Control
- Using the Delegation of Control Wizard
- Guidelines for Delegating Administrative Control

*****ILLEGAL FOR NON-TRAINER USE*****

Key Points

You can decentralize administration by delegating specific tasks to other administrators.

Delegation of administrative control at the OU level enables you to easily track permissions.

Delegation is the ability to assign responsibility of the management of Active Directory objects to another user, group, or organization.

You delegate by using the Delegation of Control wizard to set specific permissions on Active Directory objects. The Delegation of Control wizard allows you to select the user or group to which you want to delegate control, the organizational units and objects you want to grant those users the right to control, and the permissions to access and modify objects.

By delegating administrative control, you can eliminate the need for multiple administrative accounts that have broad authority, such as for an entire domain. Although you will most likely use the predefined Domain Admins group for administration of the entire domain, you can delegate the accounts that are members of the Domain Admins group to highly trusted administrative users.

Delivery Tip

If time permits, demonstrate how to delegate control by using the Active Directory Service Interfaces.

Important It is recommended that you delegate administrative control to domain local groups when assigning permissions to objects in a domain. It is recommended that you delegate administrative control to global or universal groups when assigning permissions to objects in the configuration naming context. It is recommended that you delegate administrative control to global or universal groups for attributes that are published to the global catalog for Read permission. For more information on delegating control see The Windows 2000 Deployment Planning Guide at <http://www.windows2000/library/resources/reskit/dpg/default.asp> and the Active Directory Service Interfaces Overview at <http://www.microsoft.com/windows2000/techinfo/howitworks/activedirectory/adsilinks.asp>.

Overview of Delegating Administrative Control

Topic Objective

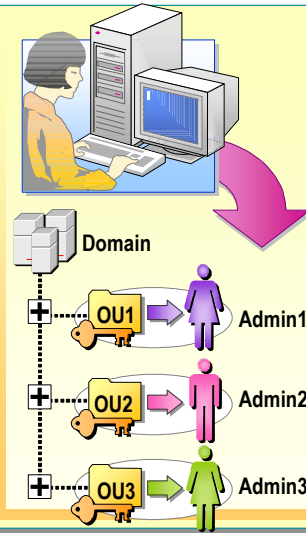
To introduce delegating of administrative control in Active Directory.

Lead-in

You can manage a network more efficiently by delegating administrative control to other administrators.

Delegation of Administration Means:

- Changing properties on a particular container
- Creating and deleting objects of a specific type under an organizational unit
- Updating specific properties on objects of a specific type under an organizational unit



*****ILLEGAL FOR NON-TRAINER USE*****

Key Points

The goal of delegating the ability to assign permissions wherever possible is to conserve administrative effort and cost.

You delegate administrative control by creating organizational units within a domain and delegating administrative control for specific organizational units.

Windows 2000 contains specific permissions and user rights that you can use to delegate administrative control. By using a combination of organizational units, groups and permissions, you can designate administrative rights to a particular user such that the user has an appropriate level of administration over an entire domain, all organizational units within a domain, or even a single organizational unit.

There are three ways to define the delegation of administration responsibilities:

- Change properties on a particular container.
- Create and delete objects of a specific type under an organizational unit, such as users, groups, or printers.
- Update specific properties on objects of a specific type under an organizational unit. For example, you can delegate the right to set a password on a User object.

Using the Delegation of Control Wizard

Topic Objective

To illustrate how to assign permissions at the OU level by using the Delegation of Control wizard.

Lead-in

Assigning permissions to objects and object attributes allows you very detailed control, but it can be cumbersome. Most of the time you can effectively assign permissions by using the Delegation of Control wizard.

Tasks for Delegating Control to Users or Groups

- 1** Start the Delegation of Control Wizard
- 2** Select Users or Groups to Which to Delegate Control
- 3** Assign Tasks to Delegate
- 4** Select Active Directory Object Type
- 5** Assign Permissions to Users or Groups

*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Demonstrate the Delegation of Control wizard.

Key Points

Always use the Delegation of Control wizard to assign permissions unless you need to assign permissions that are very detailed.

To assign permissions at the OU level, use the Delegation of Control wizard. You can assign permissions for managing objects, or you can assign permissions for specific attributes of those objects. Using the Delegation of Control wizard is the preferred method for delegating control because it reduces the possibility of unwanted effects from permission assignments.

To delegate administrative control to users or groups, perform the following tasks:

1. Start the Delegation of Control wizard.
 - a. In Active Directory Users and Computers, click the OU for which you want to delegate control; for example, *AUAdmins*.
 - b. On the **Action** menu, click **Delegate control** to open the wizard.
2. Select users or groups to which you want to delegate control.

After you have opened the Delegation of Control wizard, perform the following step to select users or groups:

- Click **Next** to open the **Users and Groups** page, select a user or group to which you want to assign permissions, and then click **Next** to assign tasks to delegate.

3. Assign tasks to delegate.

The Delegation of Control wizard allows you to either select common tasks to delegate or create custom tasks to delegate, by performing the following steps:

- a. To delegate an existing task from a list of tasks, click **Delegate the following common tasks**.

The following table describes the available tasks:

Common task	Description
Create, delete, and manage user accounts	Allows the user or group to create, delete, and modify user accounts and attributes of all user accounts in the selected OU.
Reset passwords on a user account	Allows the user or group to change the passwords of all user accounts in the selected OU.
Read all user information	Allows the user or group to view all attributes of the objects in the selected OU. The user or group cannot modify any information.
Create, delete, and manage groups	Allows the user or group to create, delete, and modify group accounts and attributes of all group accounts in the selected OU.
Modify the membership of a group	Allows the user or group to change the members of groups in the selected OU.
Manage Group Policy links	Allows the user or group to add, delete, or modify the Group Policy links of the selected OU.

- b. After you delegate a common task, end the wizard by clicking **Next** to display the **Completing the Delegation of Control Wizard** page.

Note You can delegate a custom task to users or groups by selecting **Create a custom task to delegate** and continuing forward to the next pages in the Delegation of Control wizard.

4. Select an Active Directory object type.

The Delegation of Control wizard allows you to select to delegate control of one of the following:

- A specific OU. The control of a specific OU gives you authority over all existing objects in the OU, and authority to create new objects in that OU.
- Specific objects within an OU. The wizard displays a list of object types that you can select to delegate control; for example, computer objects, group objects, and printer objects.

After you select an object type to control, click **Next** to continue.

-
5. Assign permissions to users or groups to which you want to delegate control.

The Delegation of Control wizard allows you to select the types of permissions you want to assign for the OU or its objects, by using the following filter options:

- *General*. Displays the most commonly used permissions available for the selected OU or the objects in the OU.
- *Property specific*. Displays all attribute permissions applicable to the type of object.
- *Creation/deletion of specific child object*. Displays permissions that are needed to create new objects in the OU.

After you select the permissions you want to assign, click **Next** to go to the **Completing the Delegation of Control** wizard page, and then click **Finish** to close the wizard.

Important It is recommended that you delegate control to groups rather than to users.

Guidelines for Delegating Administrative Control

Topic Objective

To identify guidelines for delegating administrative control of objects.

Lead-in

Here are some guidelines for delegating administrative control.



Assign Control at the OU Level



Use the Delegation of Control Wizard



Track the Delegation of Permission Assignments



Follow Organizational Guidelines for Delegating Control

*****ILLEGAL FOR NON-TRAINER USE*****

When you delegate administrative control of objects, you should follow these guidelines:

- Assign control at the OU level whenever possible to more easily track permission assignments. Tracking permission assignments becomes more complex when you assign permissions to specific objects and object attributes.
- Use the Delegation of Control wizard. The wizard simplifies the process of assigning object permissions by stepping you through the process.
- Track the delegation of permission assignments so that you can maintain records when you need to review security settings.
- Follow any guidelines that your organization uses for delegating control.

Lab A: Delegating Administrative Control

To introduce the lab.

Lead-in

In this lab, you will delegate control of an organizational unit.



*****ILLEGAL FOR NON-TRAINER USE*****

Explain the lab objectives.

Objectives

After completing this lab, you will be able to:

- View permissions on Active Directory objects.
- Delegate control of an organizational unit.

Prerequisites

Before working on this lab, you must have:

- An understanding of Active Directory permissions.
- Knowledge of how to create users.

Lab Setup

To complete this lab, you need the following:

- A computer running Microsoft Windows 2000 Server that is configured as a domain controller.
- The Log On Locally user right granted to all users. To grant this right, run the Lrights.bat batch file from C:\Moc\Win2154A\Labfiles folder.

Key Points

The lab does not reflect the real-world environment. It is recommended that you always use complex passwords for any administrator accounts, and never create accounts without a password.

Outside of the classroom environment, it is strongly advised that you use the most recent software updates that are necessary. Because this is a classroom environment, we may use software that does not include the latest updates.

Important The lab does not reflect the real-world environment. It is recommended that you always use complex passwords for any administrator accounts, and never create accounts without a password.

Important Outside of the classroom environment, it is strongly advised that you use the most recent software updates that are necessary. Because this is a classroom environment, we may use software that does not include the latest updates.

Estimated time to complete this lab: 30 minutes

Exercise 1

Reviewing Active Directory Permissions

Scenario

Northwind Traders is expanding, so you are preparing to modify security in Active Directory. Before you make any changes to security, you want to verify default security in Windows 2000, so that you do not make unnecessary changes.

Goal

In this exercise, you will review the default security settings on components in Active Directory.

Tasks	Detailed Steps
1. Create the following objects in Active Directory: - OU: Security - Users: - First Name: Assistant - Last Name: User - User logon name: assistant@domain.nwtraders.msft - Password: password - First Name: Secretary - Last Name: User - User logon name: secretary@domain.nwtraders.msft - Password: password - First Name: Password - Last Name: Reset - User logon name: passreset@domain.nwtraders.msft - Password: password	a. Open Active Directory Users and Computers from the Administrative Tools menu. b. Expand <i>domain.nwtraders.msft</i> (where <i>domain</i> is your assigned domain name), if necessary. c. Right-click <i>domain.nwtraders.msft</i>, point to New, and then click Organizational Unit. d. In the Name box, type Security and then click OK. e. Create a user account in the Security OU by using the following information: • First name: Assistant • Last name: User • User logon name: assistant@domain.nwtraders.msft • Password: password f. Create a second user account in the Security OU by using the following information: • First name: Secretary • Last name: User • User logon name: secretary@domain.nwtraders.msft • Password: password g. Create a third user account in the Security OU by using the following information: • First name: Password • Last name: Reset • User logon name: passreset@domain.nwtraders.msft • Password: password

Tasks	Detailed Steps																						
2. View default permissions for the OU you created in task 1. Note the results.	a. On the View menu, click Advanced Features. b. In the console tree, right-click Security, and then click Properties. c. Click the Security tab. d. In the following table, list the groups that have permissions for the Security OU, and list the permissions that each group has been assigned. If an account has special permissions, just record Special Permissions in the table. You will need to refer to these permissions in the next exercise. User account or group: Granted permissions Authenticated Users: Read System: Full Control Domain Admins: Full Control Enterprise Admins: Full Control Account Operators: Special Permissions Print Operators: Special Permissions Administrators: Special Permissions Pre-Windows 2000 Compatible Access: Special <table> <thead> <tr> <th>Group</th><th>Permission</th></tr> </thead> <tbody> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> </tbody> </table>	Group	Permission																				
Group	Permission																						
 Why are all permission check boxes for some groups cleared? Additional permissions are present, but you cannot view them in this dialog box. To see these additional permissions, click Advanced.																							

Tasks	Detailed Steps
 Are any of the default permissions inherited from the domain, which is the parent object? How can you tell? The permissions that are granted to Enterprise Admins are inherited from the parent object. The check boxes for inherited permissions are shaded.	
3. View special permissions for the Account Operators group for the Security OU. Note the results.	a. In the Security Properties dialog box, on the Security tab, click Advanced. b. In the Access Control Settings for Security dialog box, in the Permission Entries box, click each entry for the Account Operators group, and then click View/Edit.
 Which object permissions are granted to Account Operators? What can Account Operators do in this OU? The permissions that are granted to Account Operators are Create User Objects, Delete User Objects, Create Group Objects, Delete Group Objects, Create Computer Objects, and Delete Computer Objects. Account Operators can only create and delete user accounts, group accounts, and computer accounts.	
 Do any objects within this OU inherit the permissions that are granted to the Account Operators group? Why or why not? No. Objects within this OU do not inherit these permissions. The dialog box shows that permissions are applied on this object only.	
3. (continued)	c. Close all open dialog boxes, and then log off.

Exercise 2

Delegating Control

Scenario

You are the Information Technology (IT) Manager of Northwind Traders. The company is becoming larger, and it is becoming more challenging for you to perform the daily tasks of adding users and managing the OUs. You need to transfer some of the administrative tasks by delegating control of an OU to a user. You also need to delegate the ability to reset passwords for the entire domain to a user who works on the Help Desk.

Goal

In this exercise, you will delegate control over objects in an OU. Refer to the table that you completed in the previous exercise to answer the following questions.

Tasks	Detailed Steps
1. Log on as Assistant and attempt to view the contents of the Security OU.	a. Log on as Assistant with a password of password . b. Open Active Directory Users and Computers from the Administrative Tools menu. c. In the console tree, expand your domain if necessary, and then click Security .
 Which user objects appear in the Security OU? The Assistant User, Secretary User, and Password Reset user accounts.	
 Which permissions allow you to see these objects? (Hint: refer to your answers in the preceding exercise.) The Assistant user account automatically belongs to the Authenticated Users group, which has Read permission for the OU.	
2. Attempt to change the logon hours for the Secretary user account.	a. In the details pane, double-click Secretary User . b. In the Secretary User Properties dialog box, on the Account tab, attempt to change the allowed logon hours.

Tasks	Detailed Steps
 Were you successful? Why or why not? No. The Assistant user account does not have Write permission for this object.	
2. (continued)	c. Close Active Directory Users and Computers, and then log off.
3. Log on as Passreset and attempt to change the password for the TsInternetUser in the Users container.	a. Log on as Passreset with a password of password. b. Open Active Directory Users and Computers from the Administrative Tools menu. c. In the console tree, expand your domain if necessary. d. In the Users container, right-click TsInternetUser, and then click Reset Password.
 Were you successful? Why or why not? No. The Password Reset user account does not have the required permissions.	
3. (continued)	e. Close Active Directory Users and Computers, and then log off.
4. Use the Delegation of Control wizard to grant Active Directory permissions for creating, deleting and managing user accounts in the Security OU to the Assistant user account.	a. Log on as Administrator with a password of password. b. Open Active Directory Users and Computers from the Administrative Tools menu. c. In the console tree, expand your domain if necessary. d. Right-click Security, and then click Delegate Control. e. In the Delegation of Control wizard, click Next. f. On the Users or Groups page, click Add. g. In the Select Users, Computers, or Groups dialog box, click Assistant User, click Add, click OK, and then click Next. h. On the Tasks to Delegate page, click Create, delete and manage user accounts, and then click Next. i. On the Completing the Delegation of Control Wizard page, click Finish. j. Leave Active Directory Users and Computers open.

Tasks	Detailed Steps
5. Use the Delegation of Control wizard to grant Active Directory permissions for resetting passwords in the domain to the Password Reset user account.	- In the console tree, right-click <i>domain.nwtraders.msft</i>, and then click Delegate Control. - In the Delegation of Control wizard, click Next. - On the Users or Groups page, click Add. - In the Select Users, Computers, or Groups dialog box, click Password Reset, click Add, click OK, and then click Next. - On the Tasks to Delegate page, click Create a custom task to delegate, and then click Next. - On the Active Directory Object Type page, click Only the following objects in the folder, click User objects, and then click Next. - On the Permissions page, click Reset Password, and then click Next. - On the Completing the Delegation of Control Wizard page, click Finish. - Close all open windows, and log off.
6. Log on as Assistant and attempt to change the log on hours for the Secretary User account.	- Log on as Assistant with a password of password. - Open Active Directory Users and Computers from the Administrative Tools menu. - In the console tree, expand your domain if necessary, and then click Security. - Attempt to change the logon hours for the Secretary User account in the Security OU.
 Were you successful? Why or why not? Yes. The Assistant user account has been granted the Full Control permission for all user objects in the OU. This includes the permission to change the logon hours.	
6. (continued)	Attempt to change the logon hours for a user in the Users container.
 Were you successful? Why or why not? No. The Assistant user account has been granted the Full Control permission for user objects only in the Security OU. This does not extend to the Users container.	

Tasks	Detailed Steps
7. Log on as Passreset and attempt to reset the password for the TsInternetUser account in the Users container.	a. Log on as Passreset with a password of password. b. Open Active Directory Users and Computers from the Administrative Tools menu. c. In the console tree, expand your domain if necessary, and then click Users. d. Attempt to reset the password for the TsInternetUser account in the Users container.
 Were you successful? Why or why not? Yes. The Password Reset user account has been granted the required permissions for the entire domain.	
8. Attempt to change the logon hours for the Secretary User account in the Security OU.	a. In the console tree, click Security. b. Attempt to change the logon hours for the Secretary User account.
 Were you successful? Why or why not? No. The Password Reset user account has not been granted the required permissions to change logon hours.	
8. (continued)	c. Close Active Directory Users and Computers, and then log off.

◆ Customizing MMC Consoles

Topic Objective

To introduce the topics related to customizing MMC consoles.

Lead-in

To ensure that administrators have the tools required to perform administrative tasks, you can customize MMC consoles and distribute them to users.

- **Creating Customized MMC Consoles**
- **Distributing Customized MMC Consoles**
- **Installing Windows 2000 Snap-ins**

*****ILLEGAL FOR NON-TRAINER USE*****

Windows 2000 provides customized tools to administer Active Directory. You can create customized administrative tools to:

- Map to the permissions that have been assigned to a user for an administrative task.
- Simplify the user desktop for users with limited administrative privileges.

You can create customized MMC consoles that fit your administrative responsibilities or the responsibilities of other administrators. These customized consoles allow you to control the options available to groups to whom you have delegated administrative responsibilities by restricting access to specific operations within that customized console.

In addition to creating customized consoles, you may need to install the Windows 2000 Administration Tools, which contain the snap-ins needed to run the customized consoles you create.

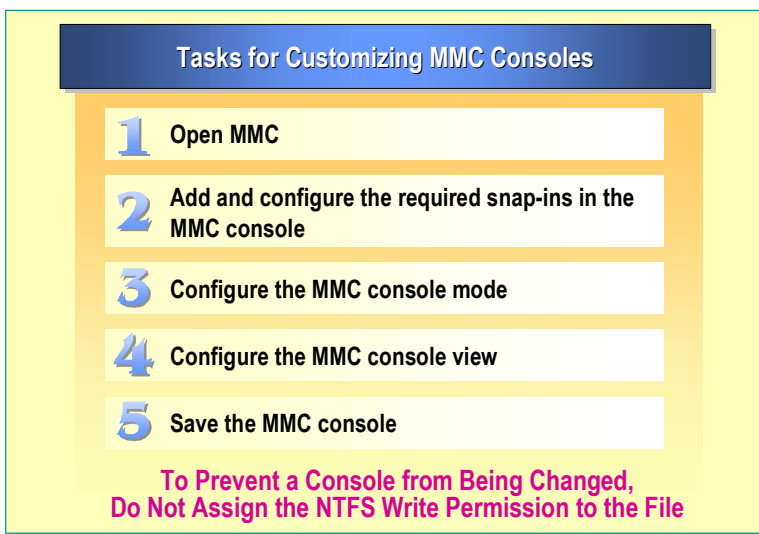
Creating Customized MMC Consoles

Topic Objective

To identify the tasks used to create a customized MMC console.

Lead-in

Customizing MMC consoles allows you to adapt the tools that you and others use to perform administrative responsibilities.



*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Demonstrate creating and customizing an MMC console. Add the Computer Management snap-in and the Event Viewer snap-in. Explain that you can point the snap-ins to the local computer or to another computer. Also, mention that Event Viewer can be both a stand-alone snap-in and an extension of Computer Management.

Key Points

An administrator or user can always open an MMC console in Author mode, even if it is saved in User mode.

Prevent changes to customized consoles by not assigning the NTFS Write permission to the .msc file.

Customizing an MMC console allows you create consoles that fit the responsibilities of administrators. Therefore, after you delegate administrative control, you can ensure that administrators have only the administrative tools that they require to perform their jobs. You can also customize a console to fit your own administrative needs. For example, you can create a console that contains all of the snap-ins that you typically use to perform one task or a set of related tasks, such as tasks for administering an OU.

Tasks for Customizing MMC Consoles

To customize an MMC console, perform the following tasks:

1. Open MMC by typing **mmc** in the **Run** box, and then clicking **OK**.
2. Add snap-ins and extensions that you require by using the **Add/Remove Snap-ins** dialog box.
3. Open the **Option** dialog box and click the **Console** tab. Select one of the following modes:
 - *Author mode*. Use this mode if you want other administrators to be able to easily modify the console as their administrative responsibilities change.
 - *User mode*. Use this mode when you want to limit the ability of other administrators to modify the console. You can determine the amount of the console tree to which the user can gain access and, therefore, the tasks that the user can perform.

To prevent MMC from prompting users to save the console after each use, open the **Option** dialog box, and then click **Do not save changes to this console** on the **Console** tab.
4. Configure the MMC console view. You can control the parts of the console to which the user can gain access by removing the console tree and toolbars. You can configure these on the **View** menu of the console.
5. Save the MMC console.

Modifying the Customized Console

Regardless of whether you save an MMC console in Author or User mode, you can still modify it by opening it in Author mode.

► **To open a console in Author mode:**

- Right-click the console, and then click **Author**.

The only way to prevent changes to customized consoles is to *not* assign the NTFS file system Write permission to the .msc file.

Tip If you configure a customized MMC console in User mode, ensure that you also have a version that you saved in Author mode so that you can easily make modifications if required.

Note For more information about MMC consoles, see *Microsoft Management Console: Overview* under **Additional Reading** on the Web page on the Student Materials compact disc.

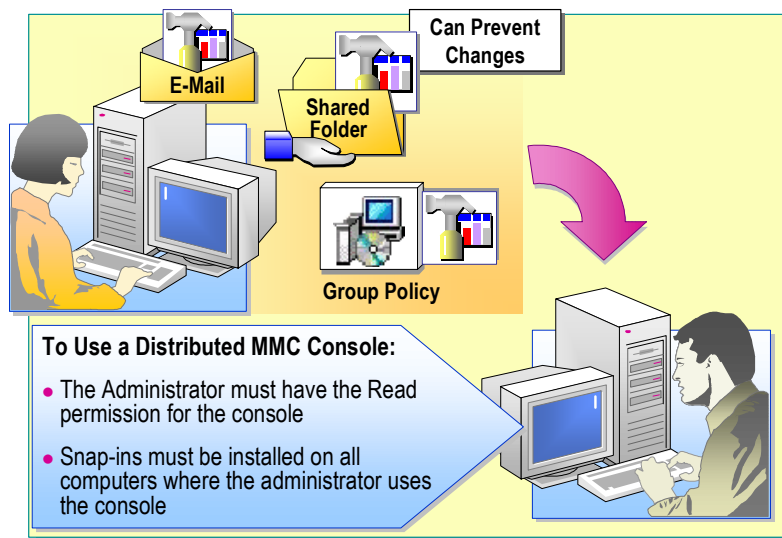
Distributing Customized MMC Consoles

Topic Objective

To illustrate the procedure for distributing customized MMC consoles.

Lead-in

After you have created customized MMC consoles for other administrators, you can choose one of several methods to distribute these consoles.



*****ILLEGAL FOR NON-TRAINER USE*****

Key Points

If you place the console file into a shared folder, the file is accessible from any computer in the network.

To use a console, you must have Read permission for the console file, and the appropriate snap-ins must be installed on the computer.

You can use the following methods to centrally distribute customized MMC consoles:

- Send the console file through e-mail as you would send any file. This is the quickest way to distribute a console file. However, you cannot prevent the administrator from changing the console file upon receiving it.
- Place the console file in a shared folder on a network server. The advantage of this method is that the console file is accessible from any computer in the network. Also, you can use NTFS file system permissions to prevent administrators from changing the console file.
- Package the console file for distribution so that you can distribute it by using Group Policy. This method guarantees that an administrator always has access to the customized console, but it does not prevent the administrator from changing the file upon receiving it.

For an administrator to successfully use an MMC console that you have distributed, the following conditions must be met:

- The administrator must have at least the Read permission for the console file.
- All snap-ins that the console references must be installed on the computers on which the administrator uses the console.

Note For more information about distributing software by using Group Policy, see Module 9, "Using Group Policy to Manage Software," in Course 2154, *Implementing and Administering Microsoft Windows 2000 Directory Services*.

Installing Windows 2000 Snap-ins

Topic Objective

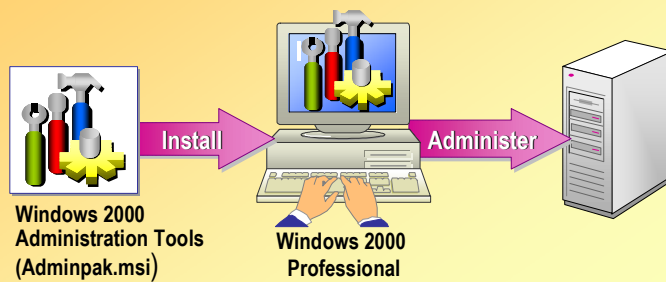
To illustrate how to install Windows 2000 snap-ins to run administrative tools on a client computer running Windows 2000 Professional.

Lead-in

After you have customized and distributed MMC consoles, you need to ensure that the computers on which you or other administrators use these consoles have the required snap-ins.

Snap-ins:

- Are contained in Windows 2000 administrative tools
- Are required for remote administration from a client computer running Windows 2000 Professional



*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Open Add/Remove Programs in Control Panel. Show students where they would install Windows 2000 Administration Tools on a computer running Windows 2000 Professional.

Key Points

Windows 2000 Professional does not have the required snap-ins for network administration; therefore, an administrator must install them.

Windows 2000 Administration Tools contain the complete set of Windows 2000 Advanced Server snap-ins, which is contained in the Windows Installer package, Adminpak.msi. If students do not know what a Windows Installer package is, provide a brief definition.

Windows 2000 Administration Tools are a complete set of all of the Windows 2000 snap-ins, which is a part of Windows 2000 Advanced Server. You must install these snap-ins to perform network administration remotely from a client computer running Windows 2000 Professional. Windows 2000 Professional has only the snap-ins required to perform routine administrative tasks on the client computer.

Even if an administrator has an MMC console to use on a client computer, the console is useless without the required snap-ins installed on this computer. To install Windows 2000 Administration Tools, use the Windows 2000 Administration Tools Setup wizard as follows:

1. In Control Panel, click **Add/Remove Programs**, and then click **Add**.

The Windows 2000 Administration Tools Setup wizard guides you through the process of selecting the installation mode.

2. You must provide the location of the Adminpak.msi file. You can get this file:

- From a computer running Windows 2000 Advanced Server in *systemroot\System32*.
- From the Windows 2000 Advanced Server compact disc.

◆ Setting Up Taskpads

Topic Objective

To introduce taskpads in Windows 2000.

Lead-in

Creating taskpads is another means of providing the appropriate administrative tools for novice administrators and users to whom you have delegated administrative responsibilities.

- What Is a Taskpad?
- Creating and Configuring a Taskpad
- Adding Tasks in a Taskpad

*****ILLEGAL FOR NON-TRAINER USE*****

A *taskpad* is an administrative tool that you set up for subadministrators and users whose primary job responsibilities do not include network administration. After you delegate administrative tasks, set up a taskpad that provides a subadministrator or user the ability to perform the specific tasks for which they are responsible.

To set up taskpads, you must create a customized MMC console, create a taskpad within the console, and then define the specific tasks that the user can perform.

What Is a Taskpad?

Topic Objective

To describe a taskpad.

Lead-in

A taskpad is a further customization of an MMC console.

A Taskpad:

- **Is a Customized Administrative Tool**
- **Contains Tasks That Are Shortcuts to Specific Commands in an MMC Console**
- **Provides Advantages:**
 - Makes it easier for novice users to perform their jobs
 - Makes complex tasks easier

*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Show students what a completed taskpad looks like. Point out the task buttons, the list of items, and the **Change** button. Explain the purpose of each of these items. Then create a user account.

Create a user account by using Active Directory Users and Computers to provide a comparison.

Key Point

Each task is a shortcut to a specific administrative task or command in an MMC snap-in.

A user must have the appropriate permissions to perform the task in the taskpad.

A taskpad is a further customization of an MMC console. A *taskpad* is created to add shortcuts to functions that exist both inside and outside a given console. Each task created in a taskpad is a shortcut to a specific administrative task or command in an MMC snap-in. You can use these shortcuts to run tasks such as starting wizards, opening property pages, performing menu commands, running command lines, and opening Web pages. You can configure a taskpad view so that it contains all of the tasks a given user might need. Also, you can create multiple taskpad views in a console, so that you can group tasks by function or user.

The advantages of creating a taskpad are:

- A taskpad makes it easier for novice users to perform their jobs. For example, you can add applicable tasks to a taskpad view and then hide the console tree, so that users can begin using tools before they are familiar with the location of particular items in the console tree or operating system.
- A taskpad makes complex tasks easier by hiding the complexity of MMC. For example, if a user must frequently perform a task that involves multiple snap-ins, in a single location, you can present tasks that open or run the necessary dialog boxes, property pages, command lines, and scripts.

The taskpad can contain several elements that make it easy to use:

- At least one button for each shortcut to a task.
- A list of items that appear in the details pane of the MMC console from which the taskpad was created. Users perform the task on these items.
- The **Change** button. This button allows the user to select different items in the console tree to administer.

Important To use a taskpad successfully, a user must have the appropriate permissions to perform the task for which the taskpad is designed.

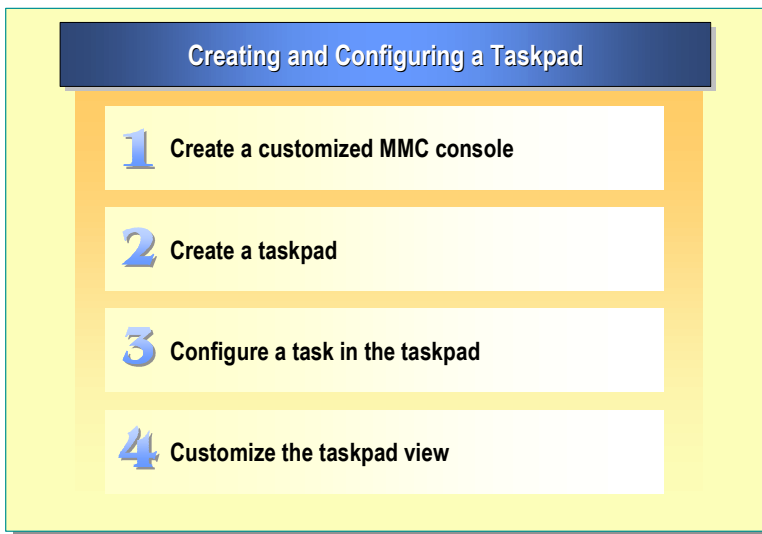
Creating and Configuring a Taskpad

Topic Objective

To describe the procedure for creating and configuring a taskpad.

Lead-in

You create a taskpad by using the Taskpad Creation wizard and then customize the tasks by using the Task Creation wizard.



*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Demonstrate creating a taskpad, but do not present all of the details on step 3. Explain to students that you will present the details in the next topic.

Key Points

To create a taskpad, you use two wizards, the Taskpad Creation wizard and the Task Creation wizard.

In the Taskpad Creation wizard, you determine the appearance, name, and descriptions of the taskpad.

You create a taskpad by using the Taskpad Creation wizard and then customize tasks by using the Task Creation wizard. To create a taskpad, perform the following steps:

1. Create an MMC console and add the required snap-in.
2. Create a taskpad by using the Taskpad Creation wizard. To open the Taskpad Creation wizard, right-click the item in the console tree to which the task applies, and then click **New Taskpad**.

The wizard guides you through the process of defining a taskpad's appearance, name, description, and the item in the console tree to which it applies.

3. Configure a task by using the Task Creation wizard. The Task Creation wizard is discussed in the Adding Tasks in a Taskpad topic.
4. Customize the taskpad by removing the console tree and changing the console to User mode. To remove the console tree, on the **View** menu, click **Customize**, and then clear the **Console Tree** check box.

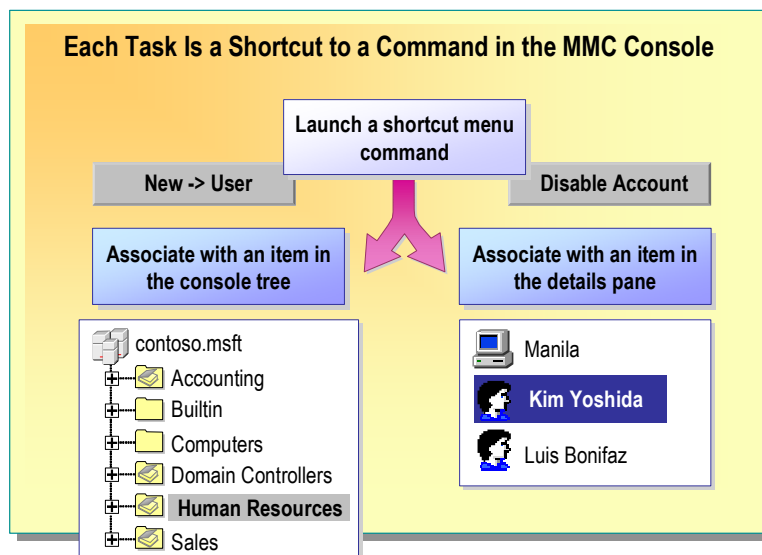
Adding Tasks in a Taskpad

Topic Objective

To illustrate the procedure for creating tasks in a taskpad.

Lead-in

A taskpad is not complete until you have created at least one task for it. This task is a shortcut to a command in the MMC console.



*****ILLEGAL FOR NON-TRAINER USE*****

Delivery Tip

Demonstrate adding an additional task to the taskpad that you created in the last section.

Show students how to create a shortcut associated with an item in the console tree and how to create a shortcut associated with an item in the details pane.

Illustrate the fact that the tasks correspond to commands on shortcut menus in a snap-in by right-clicking the item in the console tree for which you created a task in the Task Creation wizard. Perform the same procedure for the task that you created for an item in the details pane.

You can create tasks in taskpads for users who have limited administrative responsibilities. In this way, users do not have to use the MMC console interface to complete their administrative tasks. Each task in a taskpad is a shortcut to a command in the MMC console. These are the same commands that are available when you right-click an item in the console tree or the details pane.

To set up a task that corresponds to a shortcut menu command, you use the Task Creation wizard to start and select either of the following options:

- *Console tree.* This option allows you to select an item in the console tree and a command that a user can perform on that item.
For example, you can select an OU in the console tree and then select a command available for OUs, such as creating new user accounts. The user of the taskpad can then create user accounts only in that OU.
- *List in details pane.* This option allows you to select an item that appears in the details pane, and then select a command that the user can perform on the item.
For example, you can select a user account in an OU and then select a command available for that object type, such as resetting passwords. The users of the taskpad can then reset passwords for any user account that they select in the details pane for that OU.

This option is not available if you selected the stand-alone option in the Taskpad Creation wizard when creating the taskpad.

Note You can also set up a task that runs an application from the command prompt; for example, batch files and scripts. A user can then run an application by clicking a task icon rather than typing a command at the command line. You must provide the command and any arguments that are required.

Lab B: Creating Custom Administrative Tools

Topic Objective

To introduce the lab.

Lead-in

In this lab, you will create and configure a taskpad.



*****ILLEGAL FOR NON-TRAINER USE*****

Explain the lab objectives.

Objectives

After completing this lab, you will be able to:

- Create custom administrative tools by using the Microsoft Management Console (MMC).
- Create custom taskpads.

Prerequisites

Before working on this lab, you must have:

- Knowledge of how to use the snap-ins included with Windows 2000.
- Knowledge of MMC.
- Knowledge of how to use Active Directory and Computers to modify user account properties.

Lab Setup

To complete this lab, you need a computer running Windows 2000 Server configured as a domain controller.

Estimated time to complete this lab: 30 minutes

Exercise 1

Creating Custom Administrative Tools

Scenario

Now that you have delegated administrative control of the Security OU to the Assistant User, you need to create custom administrative tools that will allow this user to perform the delegated job functions.

Goal

In this exercise, you will create a custom administrative tool for the Assistant User that will allow the user to manage user accounts in the Security OU.

Tasks	Detailed Steps
1. Create a custom MMC console for the Assistant User account that contains the Active Directory Users and Computers snap-in.	- Log on as Administrator with a password of password. - In the Run box, type mmc.exe and then click OK. - On the Console menu, click Add/Remove Snap-in. - In the Add/Remove Snap-in dialog box, click Add. - In the Add Standalone Snap-in dialog box, click Active Directory Users and Computers, click Add, click Close, and then click OK.
2. Create a new window in the console that displays only the contents of the Security OU and close the original Console Root window.	- In the console tree, expand Active Directory Users and Computers, expand your domain, right-click Security, and then click New Window from Here. - On the Window menu, click 1 Console Root, and then close the Console Root window.
3. Create a new taskpad with the following properties: - Style for the details pane: Horizontal list. - Style for task descriptions: InfoTip. - List size: Small. - Apply this Taskpad view to: Selected tree item. - Name: Security OU User Management. - Description: Tool for Managing User Accounts.	- In the console tree, right-click Security and then click New Taskpad View. - On the Welcome to the New Taskpad View Wizard page, click Next. - On the Taskpad Display page, in the List size list, click Small, and then click Next. - On the Taskpad Target page, click Selected tree item, and then click Next. - On the Name and Description page, in the Name box, type Security OU User Management. - In the Description box, type Tool for Managing User Accounts and then click Next. - On the Completing the New Taskpad View Wizard page, ensure that Start the New Task wizard is selected, and then click Finish.

Tasks	Detailed Steps
4. Add the following menu commands to the new taskpad: • Disable Account • Reset password • Delete • Properties	- On the Welcome to the New Task Wizard page, click Next. - On the Command Type page, ensure that Menu command is selected, and then click Next. - On the Shortcut Menu Command page, in the Command source list, ensure that List in details pane is selected, under Available commands, click Disable Account, and then click Next. - On the Name and Description page, accept the default Task name and Description by clicking Next. - On the Task Icon page, select an icon, and then click Next. - On the Completing the New Task Wizard page, click Run this wizard again, and then click Finish. - Repeat steps a through f, changing step c as required to add the following tasks to the taskpad: • Reset password • Delete • Properties - After adding the Properties task, on the Completing the New Task Wizard page, ensure that Run this wizard again is not selected, and then click Finish.
5. Add a task to the taskpad that allows the creation of new user accounts.	- In the console tree, right-click Security, and then click Edit Taskpad View. - In the Security OU User Management Properties dialog box, on the Tasks tab, click New. - On the Welcome to the New Task Wizard page, click Next. - On the Command Type page, ensure that Menu command is selected, and then click Next. - On the Shortcut Menu Command page, in the Command source list, click Tree item task. - Under Available commands, click New->User, and then click Next. - On the Name and Description page, accept the default Task name and Description by clicking Next. - On the Task Icon page, select an icon, and then click Next. - On the Completing the New Task Wizard page, click Finish. - Click OK to close the Security OU User Management Properties box.

Tasks	Detailed Steps
6. Modify the following properties of the console: • Hide the console tree. • Remove the standard menus. • Remove the standard toolbar. • Remove the Status bar. • Remove the Taskpad navigation tabs. • Remove the snap-in menus. • Remove the snap-in toolbars. • Console name: Security OU User Management. • Console mode: User mode – limited access, single window. • Disable context menus on taskpads. • Do not allow changes to this console. • Do not allow users to customize views. • Save the console in the All Users Administrative Tools folder as Security OU User Management.	a. On the View menu, click Customize. b. In the Customize View dialog box, clear all of the check boxes, and then click OK. c. On the Console menu, click Options. d. Change the name of the console to Security OU User Management. e. In the Console mode list, click User mode – limited access, single window. f. Clear the Enable context menus on taskpads in this console and Allow the user to customize views check boxes. g. Select Do not save changes to this console, and then click OK. h. Save the console in the C:\Documents and Settings\All Users\Start Menu\Programs\Administrative Tools folder as Security OU User Management. i. Close the console, and then log off.

Exercise 2

Using a Custom Administrative Console

Scenario

The network administrator has created a custom administrative tool for the Assistant User to help that user perform user account management tasks in the Security OU.

Goal

In this exercise, you will log on as the Assistant User and use the custom administrative tool to manage user accounts in the Security OU.

Tasks	Detailed Steps
1. Log on as Assistant and perform the following user management tasks by using the Security OU User Management tool: - Create a new user. - Disable the Password Reset account. - Reset the password for the Secretary User account. - Enter an address for the Secretary user account. - Delete the new user account.	a. Log on as Assistant with a password of password. b. Open Security OU User Management from the Administrative Tools menu. c. Using the Security OU User Management tool, perform the following user management tasks: - Create a new user. - Disable the Password Reset account. - Reset the password for the Secretary User account. - Enter an address for the Secretary User account. - Delete the new user account.
2. Attempt to access the shortcut menu for one of the user accounts.	a. Right-click one of the user accounts and attempt to display the shortcut menu.
 Where you able to access the shortcut menu for the user account? Why or why not? No. Access to the shortcut menu was disabled when the Security OU User Management tool was created.	
2. (continued)	b. Close the Security OU User Management tool, and then log off.

Best Practices

Topic Objective

To describe the best practices for delegating administrative control.

Lead-in

Here are some best practices for delegating administrative control in Active Directory.



Use Deny Permissions Sparingly



Ensure That the Delegated User Completes the Delegated Tasks



Provide Training for Users Who Have Control of Objects



Add Frequently Used Customized Consoles to the Start Menu



Save Copies of Your Customized Consoles in a Shared Folder



Delegate to Groups and Add Specific Users to Those Groups

*****ILLEGAL FOR NON-TRAINER USE*****

Describe the best practices for delegating administrative control in Active Directory.

The following list provides best practices for delegating administrative control:

- Use Deny permissions sparingly. If you assign permissions correctly, you do not need to deny permissions. In most cases, denied permissions indicate mistakes that were made in assigning group membership. Denying permissions can also make it more difficult to track permissions.
- Ensure that delegated users take responsibility and that the tasks are completed. As an administrator, you are ultimately responsible for whether or not the delegated tasks are completed.
- Provide training for users who have delegated control of objects. Training ensures that the users to whom you delegate responsibility understand their responsibilities and know how to perform the administrative tasks.
- Add customized MMC consoles that you frequently use to the **Start** menu of the computer that you typically use for administration, so that you can quickly gain access to the consoles when you need to use them.
- If you perform administrative tasks from multiple computers, or administer remotely, save a copy of each of your customized MMC consoles in a shared folder. In this way, you can easily gain access to your consoles from any computer that has the necessary snap-ins installed.
- It is recommended to delegate to groups instead of to specific users. You can then add specific users to the groups with the required permissions.

Review

Topic Objective

To reinforce module objectives by reviewing key points.

Lead-in

The review questions cover some of the key concepts taught in the module.

- Object Security in Active Directory
- Controlling Access to Active Directory Objects
- Delegating Administrative Control of Active Directory Objects
- Customizing MMC Consoles
- Setting Up Taskpads
- Best Practices

*****ILLEGAL FOR NON-TRAINER USE*****

Give students time to read and answer the review questions on their own, and then discuss the answers as a group.

1. How does Windows 2000 determine who has permission to gain access to Active Directory objects and the type of access that is allowed?

Windows 2000 stores a security descriptor for every Active Directory object and attribute. The security descriptor contains a DACL that contains a list of security principals that can gain access to the object or attribute, and the specific actions that each security principal can perform on the object or attribute.

2. You want to assign an administrative assistant the permissions required to reset passwords for all user accounts in a single OU. What is the best way to do this?

Use the Delegation of Control wizard. Assign the administrative assistant the permission to reset passwords only, and assign this permission only for the OU.

3. When delegating administrative control of objects, why should an administrator assign control at the OU level whenever possible?

Assigning control at the OU level allows for easier tracking of permission assignments. Tracking permission assignments becomes more complex for objects and object attributes.

4. You customize an MMC console and give it to administrators to use on their client computers running Windows 2000 Professional. What must they do before they can successfully use the console?

Install Windows 2000 Administration Tools so that they have the required snap-ins.

5. An assistant in Marketing has only one administrative responsibility, which is resetting passwords for users. You do not want the assistant to use Active Directory User and Computers for this one task. What do you do?

Create a taskpad with the task to reset passwords for user accounts. With a taskpad, you can specifically define the task that the user performs and present the tasks in an interface that is easy to use.