NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

COMPUTER NETWORK SECURITY

Graduation Project
COM-400

AFIF S.ABDALLAH

Assoc.Prof.Dr RAHIB ABIYEV

Nicosia - 2003

ACKNOWLEDGEMENTS

First, I would like to thank *Assoc.Prof.Dr Rahib Abiyev* for his style, his support and his knowledge, which help me in my graduation project to make it success; *Assist.Prof.Dr Rahib Abiyev* he was my teacher for four year supporting me and helping me to success.

Second, I would like to thank my friends Dilek, Serdar, Mert, Ali, ghassan, Qais, Mohammad,Ahmad,Fawaz al-abadai, Tarik for their supports, which they help me to make my Graduation project and to make me happy

Third, I would like to thank every teacher in *NEAR EAST UNIVERSITY* who help me and who don't help me.

Finally, I would like to thank me family, specially my dad and mom, which they help me and supporting me every time and asking god for me, which they are the huge recourse of supporting, they made me happy and to reach to my goal to graduate and to be an engineer.

ABSTRACT

Maintenance and security of computer network it's a very important part in computers world or in communications world especially in net-communication.

Network it is a huge field which make the communication more easy and more fixable, security it is a huge field too which have many ways and many types, that made our communication, transfer important data or secret information more easy, faster, more trust, to transfer in every where in this world.

There are many aims in this project that's described and maintained the basic things about network and security, such as sharing folders in the network, and what it is benefit and how it's work, such as, to give every user in the network different user name and password or how to make server, which will help you to access your information or data or to save it from losing, when a small error has been done, or to give a permission to each user to access any folders or not, that's help you to keep away the people who are trying to tampering this folders, another aim which is an important aim which is about Windows 2000 that includes a native PKI that is designed to take full advantage of the Windows 2000 security architecture. Public key cryptography is an asymmetric scheme that uses a pair of keys for encryption and certificate-based processes use the X.509 standard. Certificate Services enables an organization to manage the issuance, renewal, and revocation of digital certificates without having to rely on external CAs; and bout Kerberos which is the default authentication provider in Windows 2000 and the primary security protocol, including principal, realm, secret key and TGT. The Kerberos authentication process involves the client computer negotiating exchanges between the target server and the KDC and a set of security configuration tools that allow configuring Windows 2000 security settings and at the end, which is about Auditing in Microsoft Windows 2000 that is the process of tracking both user activities and Windows 2000 activities.

This project have a unique style of description which provides by tables to make data clear to the readers, easy to understand and easy to use it or to make it which is also provided and described by steps one by one to make it clear and easy to do it or to understand it to even to imagine it, there is ,many photos which is familiar photo for every one he had use pc, which make every thing understood, the most useful part it is the summary after each chapter which is summary the key information in the chapter.

Table of Contents

INTRODUCTION

Computer network security, computer networks are every where, you find them in large businesses and small ones, schools, charitable institutions and in universities, because every business runs in information so it is very important to know and to understand how this information move and how to protect it.

Network mean many computers connected to each others, which make the information exchange easier and faster, so the information moves directly from computer to computer, rather than through a human intermediary, but when you use a computer not connected to a network (anther computers) that mean you are working in what is called stand-alone environment. There is two type of network (LAN) which mean local area network which is a number of computers connected to each other by cable in a single location, usually single floor, but the second type it (WAN) wide area network which is the set of connection links between local area networks. This links are made over telephone lines, every type of network have a security system to protect the information, to give each network user a different account name and password to allowing the network server to distinguish among those who need to access to have it and protecting the information from tampering by those who not.

Networks have many advantages as sharing the expensive equipment, protecting the information because small mistake will lose the information so we but the information in one computer which call server. But this level of security system it is not enough when there is information move from country to country there is a system of computer network.

In this project there is five chapters.

- ***First chapter*** describes File System Security, Share folders and assign permissions to those shares, Assign NTFS permissions to files and folders.
- ***Second chapter*** describes Public Key Infrastructure, Describe the fundamental concepts of public key cryptography and the Windows 2000 implementation of PKI, Process certificate requests and add certificate authorities (CAs) and Install Microsoft Certificate Services.
- ***Third chapter*** describes The Kerberos Protocol in Windows 2000, and how it works in Windows 2000.

- ***Fourth chapter*** describes Security Configuration Tools to understand how the security configuration tools are used to configure security settings and analyze system security in your Windows 2000 network.
- ***Fifth chapter*** describes Microsoft Windows 2000 Auditing, Plan an audit strategy and determine which events to audit, Set up auditing on Active Directory objects and on files, folders, and printers, Use Event Viewer to view a log and locate events.

CHAPTER 1

FILE SYSTEM SECURITY for WINDOWS

1.1 OVER VIEW

Sharing folders is the only way to make folders and their contents available over the network. Shared folders provide a way to secure file resources; they can be used on FAT16 and FAT32 partitions, as well as on NTFS (Network File Sharing) partitions. But NTFS supports more than just shared folders. NTFS permissions can be used to specify which users and groups can gain access to files and folders and what they can do with their content. However, NTFS permissions are not available on volumes that are formatted with FAT, Shared folders are used to provide network users with access to file resources. When a folder is shared, users can connect to the folder over the network and gain access to the files it contains. However, to gain access to the files, users must have permissions to access the shared folders.

1.2.1 Shared Folder Permissions

A shared folder can contain applications, data, or users' personal data (called home folders). Each type of data can require different shared folder permissions.
Shared folder permissions have the following characteristics in common:

1) Shared folder permissions apply to folders, not individual files. Since you can apply shared folder permissions only to the entire shared folder and not to individual files or subfolders in the shared folder, shared folder permissions provide less detailed security than NTFS permissions.
2) Shared folder permissions do not restrict access to users who gain access to the folder at the computer where the folder is stored. They apply only to users who connect to the folder over the network.
3) Shared folder permissions are the only way to secure network resources on a FAT volume. NTFS permissions are not available on FAT volumes.
4) The default shared folder permission is Full Control, and it is assigned to the Everyone group when you share the folder.

A shared folder appears in Microsoft Windows Explorer as an icon of a hand holding the shared folder (Figure 1.1).

Figure 1.1 *shared folders in Windows Explorer*

To control how users gain access to a shared folder, you must assign shared folder permissions. The following table explains what each of the shared folder permissions allows a user to do. The permissions are presented from most restrictive to least restrictive.

Permission	**Description**
Read	Users can display folder names, filenames, file data and attributes; run program files; and change folders within the shared folder
Change	Users can create folders, add files to folders, change data in files, append data to files, change file attributes, delete folders and files, and perform actions permitted by the Read permission.
Full Control	Users can change file permissions, take ownership of files, and perform all tasks permitted by the Change permission.

You can allow or deny shared folder permissions to individual users or to user groups. Generally, it is best to assign permissions to a group rather than to individual users. You should deny permissions only when it is necessary to override permissions that are otherwise applied. For example, it might be necessary to deny permissions to a specific user who belongs to a group that has been granted permissions. If you deny a shared folder permission to a user, the user will not have that permission.

1.2.2 Applying Shared Folder Permissions

Applying shared permissions to user accounts and groups affects access to a shared folder. Denying permission takes precedence over the permissions that you allow.

1.2.3 Multiple Permissions

A user can be a member of multiple groups, each with different permissions that provide different levels of access to a shared folder. When you assign permission to a user for a shared folder and that user is a member of a group to which you assigned a different permission, the user's effective permissions are the combination of the user and group permissions. For example, if a user has Read permission and is a member of a group with Change permission, the user's effective permission is Change, which includes Read.

1.2.4 Deny Overrides Other Permissions

Denied permissions take precedence over any permissions that you otherwise allow for user accounts and groups. If you deny shared folder permission to a user, the user will not have that permission, even if you allow the permission for a group of which the user is a member.

1.2.5 NTFS Permissions

Shared folder permissions are sufficient to gain access to files and folders on a FAT volume but are not the best solution for an NTFS partition. On a FAT partition, users can gain access to a shared folder in which they have permissions, as well as to all of the folder's contents. When users gain access to a shared folder on an NTFS partition, you should use either share rights or NTFS permissions but not both. NTFS permissions are preferred since permissions can be set on both files and folders. If share rights are configured for a folder and NTFS permissions are configured for folder or files within a folder, the most restrictive rights will become the user's effective rights to the resource. This significantly increases the complexity of resolving access permissions for network resources.

1.2.6 Copying or Moving Shared Folders

When you copy a shared folder, the original shared folder is still shared, but the copy is not shared. When you move a shared folder, it is no longer shared.

1.3 Guidelines for Shared Folder Permissions

The following list provides some general guidelines for managing your shared folders and assigning shared folder permissions:

1) Determine which groups need access to each resource and the level of access they require. Document the groups and their permissions for each resource.
2) Assign permissions to groups instead of user accounts to simplify access administration.
3) Assign to a resource the most restrictive permissions that still allow users to perform required tasks. For example, if users need only to read information in a folder, and they will never delete or create files, assign the Read permission.
4) Organize resources so that folders with the same security requirements are located within a folder. For example, if users require Read permission for several application folders, store the application folders within the same folder. Then share this folder instead of sharing each individual application folder.
5) Use intuitive share names so that users can easily recognize and locate resources, and use share names that all client operating systems can use.

NOTE:

MS-DOS, Windows 3.*x*, and WFW clients read up to 8.3 format share names; consequently, longer share names are not advisable in mixed environments.

Microsoft Windows 2000 provides 8.3-character equivalent names, but the resulting names might not be intuitive to users. For example, a Windows 2000 folder named Accountants Database would appear as Account~1 on client computers running MS-DOS, Windows 3.*x*, and Windows for Workgroups.

1.4 Sharing Folders

You can share resources with others by sharing folders containing those resources. To share a folder, you must be a member of one of several privileged groups, depending on the role of the computer where the shared folder resides. When you share a folder you can control access to the folder by limiting the number of users who can simultaneously gain access to it. You can also control access to the folder and its contents by assigning permissions to selected users and groups. Once you have shared a folder, users must connect to the shared folder and must have the appropriate permissions to gain access to it. After you have shared a folder, you may want to modify it. You can stop sharing it, change its share name, and change user and group permissions to gain access to it.

1.5 Requirements for Sharing Folders

In Windows 2000, members of the built-in Administrators, Server Operators, and Power Users groups are able to share folders. Which groups can share folders on which machines depends on whether the computers belong to workgroups or domains and on the type of computers on which the shared folders reside:

1) In a Windows 2000 domain, the Administrators group and Server Operators group can share folders residing on any machines in the domain. The Power Users group is a local group and can only share folders residing on the stand-alone server or on the computer running Windows 2000 Professional where the group is located.
2) In a Windows 2000 workgroup, the Administrators group and Power Users group can share folders on the Windows 2000 Server stand-alone server or the computer running Windows 2000 Professional on which the group exists.
3) Users that are granted the Create Permanent Shared Objects user right can also create shares on the computer where this right is assigned.

NOTE:

If the folder to be shared resides on an NTFS volume, users must also have at least the Read permission for that folder.

1.6 Administrative Shared Folders

Windows 2000 automatically shares folders for administrative purposes. These shares are appended with a dollar sign ($). The $ hides the shared folder from users who browse the computer. The root of each volume, the system root folder, and the location of the printer drivers are all hidden shared folders that you can gain access to across the network.

The following table describes the purpose of the administrative shared folders that Windows 2000 automatically generates:

Share	Purpose
C$, D$, E$, and so on	The root of each volume on a fixed disk is automatically shared, and the share name is the drive letter appended with a dollar sign ($). When you connect to this folder, you have access to the entire volume. You use the administrative shares to connect remotely to the computer to perform administrative tasks. Windows 2000 assigns the Full Control permission to the Administrators group. Removable drives like CD-ROM drives are not assigned the hidden share drive letter.
Admin$	The system root folder, which is C:\Winnt by default, is shared as Admin$. Administrators can gain access to this shared folder to administer Windows 2000 without knowing which folder it is installed in. Only members of Administrators have access to this share. Windows 2000 assigns the Full Control permission to the Administrators group.
Print$	When you install the first shared printer, the %systemroot%\System32\Spool\Drivers folder is shared as Print$. This folder provides access to printer driver files for clients. Only members of Administrators, Server Operators, and Print Operators have the Full Control permission. The Everyone group has the Read permission.

Hidden shared folders are not limited to those that the system automatically creates. You can share additional folders and append a $ to the share name. Only users who know the folder name can gain access to it, if they have also been granted the proper permissions.

1.7 Sharing a Folder Properties

When you share a folder, you can give it a share name, provide comments to describe the folder and its content, limit the number of users who have access to the folder, assign permissions, and share the same folder multiple times. To share a folder, right-click the folder you want to share and then click Properties. The share properties are set on the Sharing tab of the Properties dialog box (Figure 1.2).

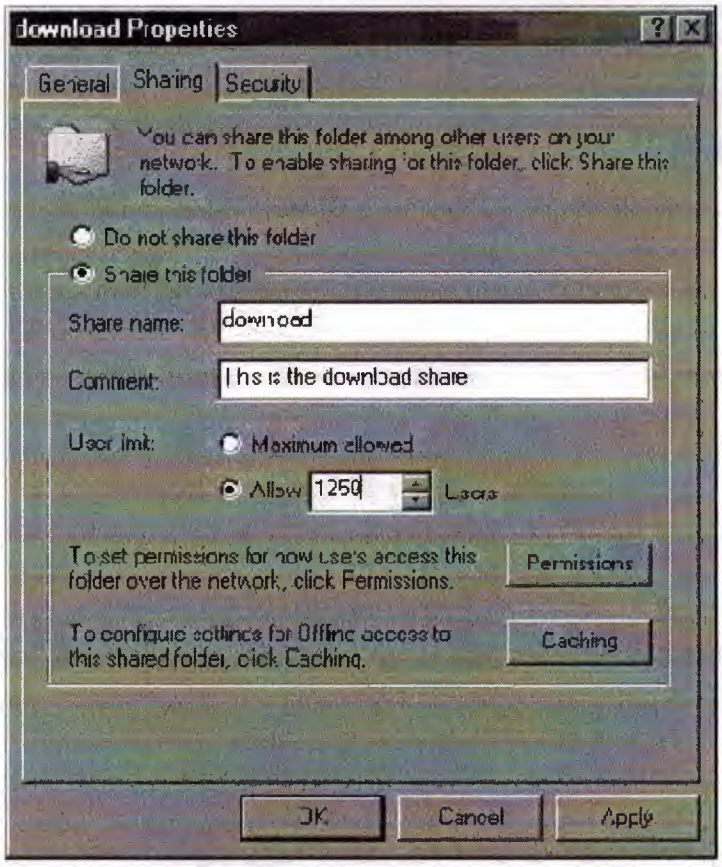


Figure 1.2 Sharing tab of a folder's Properties dialog box

The following table provides a description of the options on the Sharing tab:

Option	Description
Do Not Share This Folder	The option you should select if you do not want to share this folder. When this option is selected, all other options are grayed out.
Share This Folder	The option you should select if you want to share this folder. When this option is selected, all other options are active.
Share Name	The name that users from remote locations use to make a connection to the shared folder. You must enter a share name.
Comment	An optional description for the share name. The comment appears in addition to the share name when users at client computers browse the server for shared folders. This comment can be used to identify contents of the shared folder.
User Limit	The number of users who can concurrently connect to the shared folder. The Maximum Allowed option allows Windows 2000 Server to support an unlimited number of connections. However, the number of Client Access Licenses (CALs) that you purchased limits the connections.
Permissions	The shared folder permissions that apply only when the folder is accessed over the network. By default, the Everyone group is assigned Full Control for all new shared folders.
Caching	The settings to configure if and how files within the shared folder are cached locally when accessed by others.
New Share	The option that allows you to create a new share.
Remove Share	The option that allows you to remove a share. This option appears only after the folder has been shared more than once.

After you share a folder, the next step is to specify which users have access to the shared folder. This is done by assigning shared folder permissions to selected user accounts and groups. You can assign permissions by clicking the Permission button on the Sharing tab of the shared folder's Properties dialog box. From there, you can select the user accounts and groups to which you want to assign permissions.

1.8 Modifying Shared Folders

You can modify the properties of a shared folder. For example, you can stop sharing a folder, modify the share name, or modify shared folder permissions. To modify a shared folder, open the Properties dialog box for that folder. The following table provides the steps you should take to perform specific modifications:

Modification	Action
Stop sharing a folder	Click the option Do Not Share This Folder.
Modify the share name	First, stop sharing the folder by clicking the option Do Not Share This Folder. Click the Apply button to apply the change, and then click the option Share This Folder. Enter the new share name in the Share Name text box.
Modify shared folder permissions	Click the Permissions button. In the Permissions dialog box, click Add or Remove. To add a group or user account, select that group or user in the Select Users, Computers, Or Groups dialog box, which opens when you click Add.
Share folder multiple times	Click the New Share button to share a folder with an additional shared folder name. Do so to consolidate multiple shared folders into one while allowing users to continue to use the same shared folder name that they used before you consolidated the folders.
Remove a share name	Click the Remove Share button. This option appears only after the folder has been shared more than once.

NOTE:

If you stop sharing a folder while a user has a file open, the user might lose data. If you click the Do Not Share This Folder option and a user has a connection to the shared folder, Windows 2000 displays a dialog box notifying you that a user has a connection to the shared folder.

1.9 NTFS Permissions

NTFS permissions are a set of standard permissions that allow or deny access for each user or group. They provide security for resources by allowing administrators and users to control who can gain access to individual files and folders and to specify the kind of access users can gain. NTFS security is effective whether a file or folder is accessed interactively at a computer or over a network.

Windows NT provides the following standard NTFS permissions:

1) **NTFS folder permissions** Use these permissions to secure access to individual folders on NTFS formatted volumes.
2) **NTFS file permissions** Use these permissions to secure access to individual files on NTFS formatted volumes.

1.10.1 Assigning NTFS Permissions

When new files and folders are created, rules and priorities are associated with the ways permissions are assigned, combined, and inherited.

1.10.2 NTFS Full Control Permission

The Full Control permission grants all permissions to access a resource. It is assigned as follows by default:

1) When a user creates a file or folder, he or she becomes the Creator Owner and is assigned the Full Control permission.
2) When a volume is formatted with NTFS, Full Control is assigned to the Everyone group at the root of the drive.
3) When a FAT16 or FAT32 partition is converted to NTFS, Full Control is assigned to the Everyone group on all resources on that volume.

1.10.3 Multiple NTFS Permissions

Permissions to files and folders can be assigned to users and groups. It is possible for users to have multiple permissions assigned to them: those assigned to his or her user account and those assigned to groups the user is a member of. A user's effective permissions are the combination of NTFS permissions assigned to the individual user and the NTFS permissions assigned to all the groups the user belongs to. For example, if a user has Write permission to a folder and is also a member of a group

with Read permission to the same folder, the user has both Read and Write permission for that folder.

NTFS file permissions take priority over NTFS folder permissions. For example, if a user is assigned the Write permission to a folder and the Modify permission to a file in that folder, the user can both write to and modify the file. This is also true when a user has not been assigned access to a folder. A user can always gain access to the files for which he or she has permissions by using the full universal naming convention (UNC) or path to open the file from its application. For example, a user has no permissions for a folder that contains a file for which the user has Change permission. The user can open the file from the file's appropriate application by typing the full UNC or path to the file.

Denying permission for a user or group blocks that permission from the user, even if the permission has been granted to a group the user belongs to. For example, the Everyone group is assigned Full Control permission for a file for which a user has been denied Delete permission. The user will be able to read and modify the file, but will not be able to delete it.

1.10.4 Permission Inheritance

There are rules associated with the priority of file and folder permissions as you move down a directory tree from the parent folder to the subfolder and files. By default, permissions assigned to the parent folder are inherited and propagate to subfolders and files contained within the parent folder. However, inheritance can be prevented. When NTFS permissions are assigned or changed for a folder, permissions are assigned for the folder itself, for any existing files and subfolders, as well as for any new files and subfolders that might be created in the folder. A file or folder can be prevented from inheriting permissions from the parent folder, and permissions can be assigned explicitly to the file or folder. Also, permissions that have been inherited can be changed or removed.

1.11 Guidelines for Assigning NTFS Permissions

Administrators and the owner of a file or folder control which users and groups have permissions to the file or folder and what the permissions are. Use the following guidelines when assigning NTFS permissions:

1) To simplify administration, group resources into application, data, and home folders. Doing so provides three benefits:
   a) Permissions are assigned only to folders, not to individual files.
   b) Backup is less complex because it is typically a lower priority to backup application files.
   c) All home folders are in one location.
2) Use NTFS permissions to control access to files and folders. Assign the minimum level of permission required. This reduces the possibility of users accidentally modifying or deleting important documents and application files.
3) Whenever possible, assign permissions to groups rather than individual user accounts. Create groups according to the access they require for resources, and then assign the appropriate permissions to the group. Only when necessary, assign permissions to individual user accounts.
4) When assigning permissions to home folders, centralize home folders on a network volume separate from applications and the operating system to streamline backing up data and administration.
5) When assigning permissions to working data or application folders, remove the default Full Control permission from the Everyone group. Assign Read & Execute permission to the Users and Administrators groups. This prevents application files from being accidentally deleted or damaged by users or viruses. Administrators and users responsible for upgrading troubleshooting application software can be assigned Full Control permission, can complete their tasks, and then be assigned Read & Execute again.
6) When assigning permission to public data folders, assign Modify and Read & Execute to the Users group, and Full Control to Creator Owner. This gives users the ability to delete and modify only the files and folders they create, as well as the ability to read documents created by other users.

7) In general, it is better not to assign permissions than to deny permissions. Deny permissions only when it is essential to deny specific access to a specific user account or group.

8) Encourage and educate users to assign permissions to the files and folders they create and own. Provide them with guidelines for assigning appropriate permissions to the resources they control.

1.12.1 Configuring NTFS Permissions

The owners of files and folders can assign permissions to user accounts and groups. Administrators can also assign permissions to these resources.

To assign or modify NTFS permissions for a file or folder, open the Properties dialog box for that file or folder. NTFS permissions are configured on the Security tab of the Properties dialog box. The following table provides a description of the options on the Security tab:

Option	Description
Name	Lists the user accounts and groups with permissions for the file or folder. Click the user account or group to assign or change permissions, or to remove from the list.
Permissions	The permissions that you can allow or deny for the user account or group: Select the Allow check box to allow permission. Select the Deny check box to deny permission.
Add	Click this button to open the Select Users, Groups, or Computers dialog box where you can select user accounts and groups to add to the Name list.
Remove	Click this button to remove the selected user account or group and the associated permissions from the file or folder.
Allow Inheritable Permissions From Parent To Propagate To This Object	By default, this option is not selected for folders, which means subfolders do not inherit permissions assigned to their parent folder. Files are assigned this option, which means that files within a folder automatically receive the permissions assigned to their parent folder.
Advanced	Opens the Access Control Settings dialog box. From here you

	can configure special access permissions, auditing capability, and ownership control for files and folders.

1.12.2 Assigning Special Access Permissions

In general, the standard NTFS permissions provide all the permissions necessary to secure data. However, there are instances where the standard permissions do not provide the special access that might be needed. To create special access, use special NTFS permissions. Like standard permissions, special access permissions are either allowed or denied.

NOTE:

When special access permissions are assigned to a user or group, the permissions are indicated as Special on the Access Control Settings dialog box.

Special access permissions provide a finer degree of control for assigning access to resources. There are 13 special access permissions that, when combined, constitute the standard NTFS permissions, such as Read & Execute, Modify, and Full Control. For example, the standard NTFS Read permission includes the Read data, Read attributes, and Read extended attributes permissions.

Assigning special access permissions to folders and files requires three tasks:

1) Configuring more granular permissions
2) Transferring ownership
3) Auditing access

1.12.3 Changing Permissions

File and folder owners and other users with Full Control permissions can assign or change permissions. You can grant network administrators the ability to change permissions on a file or folder without giving them Full Control over the file or folder. In this way, the administrator can assign permissions but not have permission to delete a file or folder or write to it. To give network administrators the ability to change permissions, grant the Change Permissions special access permission on the file or folder to the network administrators' group account.

If a member of the Administrators group takes ownership, the Administrators group becomes the owner and any member can access and change the permissions for the file or folder.

1.12.4 Transferring Ownership

In addition to changing permissions, ownership can be transferred. There are several way to transfer ownership:

1) The current owner can assign the Full Control standard permission or the Take Ownership special access permission to other users, allowing those users to take ownership.

2) An administrator can take ownership of any folder or file under his or her administrative control. For example, if an employee leaves the company, an administrator can take ownership of the employee's files and change the permissions so that others can access the files or folders.

3) When assigned to a volume or folder, special access permissions are initially applied only where specified in the Apply Onto drop-down menu, which is discussed in more detail later in this lesson.

To transfer or take ownership of a file or folder, click the Owner tab in the Access Control Settings dialog box. The current owner of the file or folder is shown in the Current Owner Of This Item text box. You can select a new owner from the Change Owner To list. You can also select the Replace Owner On Subcontainers And Objects check box to change ownership for all subfolders and files contained within the folder.

1.12.5 Setting Special Access Permissions

To set special access permissions, access the Properties dialog box for a file or folder and click Advanced on the Security tab. In the Access Control Settings dialog box, click the Permissions tab, and then click Add to add a new user or group and modify the special access rights. Click View/Edit to modify the special access rights of an existing user or group. From here you can configure the options that allow you to set special access permissions. These options are described in the following table:

Option	Description
Name	The user account or group name. To select a different user account or group, click Change.
Apply Onto	The level of the folder hierarchy at which the special NTFS permissions are inherited. The default is This Folder, Subfolders, and Files.
Permissions	The individual special access permissions. To allow or deny

	an individual NTFS permission, select the Allow or Deny check box, respectively.
Apply These Permissions To Objects And/Or Containers Within This Container Only	This check box is available to folders and subfolders. Folders that are lower in the folder hierarchy can inherit the modified individual NTFS permissions from this folder. This option does not apply to files. Click to clear this check box to prevent permissions inheritance. Select this check box to propagate the modified individual NTFS permissions down the folder hierarchy.
Reset Permission On All Child Objects And Enable Propagation Of Inheritable permissions	This check box is only available to the partition From a partition, permissions on all folders, subfolders, and files can be reset. Select this check box to reset all permissions for folders and files located on the partition to the settings designated for the partition. This option also enables the Apply These Permissions To Objects And/or Containers Within This Container Only check box. This check box is described in the previous row.
Clear all	Clear all selected permissions and the level of folder hierarchy selected to inherit permissions

The following table provides an overview of the options available in the Apply Onto drop-down menu:

Option	**Objects that permissions apply to**
This Folder Only	Only to the folder.
This Folder, Subfolders, And Files	The folder, subfolder, and files. New files and folders created in this folder will inherit the permissions.
This Folder And	The folder and subfolders. New files and folders created in

Subfolders	this folder and subfolder will inherit the permissions.
This Folder And Files	The folder and files. New files and folders created in this folder will inherit the permissions.
Subfolders And Files Only	The subfolders and files. New files and folders created in the subfolder will inherit the permissions.
Subfolders Only	The subfolders. New folders created in the subfolder will inherit the permissions.
Files Only	Only to the files.

1.13.1 Copying and Moving Files and Folders

NTFS allows you to copy and move files and folders.

1.13.2 Copying Files and Folders

To copy files and folders within or between NTFS volumes, a user must have been granted Add permission for the destination folder. The user who performs the copy will become the owner of the new file or folder.

When files or folders are copied, permissions will be inherited or lost, depending on where the file or folder is copied:

1) When a folder or file is moved within an NTFS partition, the folder or file retain its permissions.

2) When a folder or file is copied within or between NTFS partitions, or moved to another partition, the folder or file inherit the permissions of the destination folder.

3) When folders or files are copied to FAT16 or FAT32 volumes, the folders and files lose their NTFS permissions because FAT16 and FAT32 volumes do not support NTFS permissions.

1.13.3 Moving Files and Folders

To move files and folders between NTFS partitions requires the Add permission for the destination folder or file and the Delete permission for the source folder or file. The Delete permission is required to move a folder or file because the folder or file is deleted from the source folder after it is moved to the destination folder. When a folder or file is moved to another partition, the user who performed the move will become Creator Owner.

Moving folders or files within and between NTFS volumes can affect the original permissions. The following table describes the results of what can occur when moving a file or folder:

Action	Result
Intra-volume move (within the same volume)	The folder or file retains the original permissions that are set for folder or file.
Inter-volume move (across different volumes)	The folder or file inherits the permissions that are set for the destination folder.

When folders or files are moved to FAT16 or FAT32 volumes, the folders and files lose their NTFS permissions because FAT16 and FAT32 volumes do not support NTFS permissions.

NOTE:

After you learn how to create users and groups in a later chapter, you will apply share rights and NTFS permissions to the users and groups you created.

1.14 Troubleshooting NTFS Permissions

The following table describes common permission problems and provides solutions:

Problem	Solution
A user cannot gain access to a file or folder.	Check the permissions assigned to the user account and to groups the user is a member of. If the user or a group that the user is a member of has been denied access to the file or folder, the user has no access to the resource. If the file or folder was copied within an NTFS partition, or copied or moved to another NTFS partition, the permissions may have changed by inheriting new permissions from the destination folder. If both share rights and NTFS permissions are configured for a folder, the most restrictive rights apply. Therefore, set the share rights to Everyone Full Control and control access exclusively through NTFS permissions.

A user account is added to a group to give that user access to a file or folder, but the user still cannot gain access to the file or folder.	An access token is created every time a user logs on and is authenticated by a computer running Windows NT or Windows 2000. The access token contains information about the groups the user belongs to. For the access token to be updated to include the new group, the user must log off and then log on again, or close all connections to the computer and then make new connections.
A user deletes a file, although that user does not have permission to delete the file.	Assign all permissions at the folder level, not at the file level. To deny users access, group files in a separate folder and then assign that folder-restricted access. If this problem is unavoidable, do not assign Full Control permission for a folder. Instead, assign all the permissions, that is, Modify, Read & Execute, List Folder Contents, Read, and Write. This assigns all the abilities for the Full Control permission for the folder and its contents, except that users cannot delete files in the folder.

1.15 Chapter Summary

Folders can be shared so that users can connect to a folder over the network and gain access to the files it contains. However, to gain access to the files, users must have permissions to access the shared folders. Shared folder permissions apply to folders, not individual files. When you share a folder, you can give it a share name, provide comments to describe the folder and its content, limit the number of users who have access to the folder, assign permissions, and share the same folder multiple times. Shared folder permissions are the only way to secure network resources on a FAT partition. NTFS permissions are not available on FAT volumes. NTFS permissions are a set of standard permissions that allow or deny access for each user or group. By default, permissions assigned to the partition and parent folder are not inherited and do not propagate to subfolders and files contained within the parent folder. The owner of a file or folder and administrators control which users and groups have permissions to the file or folder and what the permissions are. The owners of files and folders can assign

permissions to user accounts and groups. Administrators can also assign permissions to these resources.

CHAPTER 2

PUBLIC KEY INFRASTRUCTURE

2.1.1 OVER VIEW

Public key cryptography is a critical technology for e-commerce, intranets, extranets, and other Web-enabled applications. However, to take advantage of the benefits of public key cryptography, a supporting infrastructure is needed. The Windows 2000 operating system includes a native public key infrastructure (PKI) that is designed from the ground up to take full advantage of the Windows 2000 security architecture. This lesson provides an overview of the Windows 2000 PKI and includes discussions about security properties, cryptography, certificates, and Microsoft Certificate Services.

2.2.1 Security Properties

Computer security includes everything from the physical computing environment to the software environment. In a software environment, security should provide four functions: authentication, integrity, confidentiality, and anti-replay.

2.2.2 Authentication

Authentication is the process of reliably determining the genuine identity of the communicating computer (host) or user. Authentication is based on cryptography; it ensures that an attacker eavesdropping on the network cannot gain the information needed to impersonate a valid user or entity. It allows a communicating entity to prove its identity to another entity before unprotected data is sent across the network. Without strong authentication, any data and the host it is sent from is suspect.

2.2.3 Integrity

Integrity is the correctness of data as it was originally sent. Integrity services protect data from unauthorized modification in transit. Without data integrity, any data and the host it is sent from is suspect.

2.2.4 Confidentiality

Confidentiality ensures that data is disclosed only to intended recipients.

2.2.5 Anti-Replay

Anti-replay, also called *replay prevention,* ensures that datagrams are not retransmitted. Each datagram sent is unique. This uniqueness prevents attacks in which a message is intercepted and stored, then re-used later to attempt illegal access to information.

2.3.1 Cryptography

Cryptography is a set of mathematical techniques for encrypting and decrypting data so it can be transmitted securely and not be interpreted by unauthorized parties. Cryptography uses keys in conjunction with algorithms to secure data. A *key* is a value used to encrypt or decrypt information. Even if the algorithm is publicly known, security is not compromised because the data cannot be read without the key. For example, the algorithm of a combination lock is common knowledge: the dials are moved in a specific order to open the lock. However, the key to the lock—the numbers of the combination code—is secret and known only to the person with the combination. In other words, the key provides the security, not the algorithm. The algorithm provides the infrastructure in which the key is applied. Security systems can be based on public key or secret key cryptography, which are described later in this lesson.

There are a number of well-known cryptographic algorithms, each supporting different security operations. The following table describes several well-known cryptographic algorithms:

Algorithm	Description
Rivest, Shamir, Adleman (RSA)	A general purpose algorithm that can support digital signatures, distributed authentication, secret key agreement via public key, and bulk data encryption without prior shared secrets.
Digital Signature Standard (DSA)	A public key algorithm used for producing digital signatures.
Diffie-Hellman	A public key cryptography algorithm that allows two communicating entities to agree on a shared key without

	requiring encryption during the key generation.
Hash Message Authentication Code (HMAC)	A secret key algorithm that provides integrity, authentication, and anti-replay. HMAC uses hash functions combined with a secret key. A hash, also known as a message digest, is used to create and verify a digital signature.
HMAC-Message Digest function 5 (MD5)	A hash function that produces a 128-bit value known as a digital signature. This signature is used for authentication, integrity, and anti-replay.
HMAC-Secure Hash Algorithm (SHA)	A hash function that produces a 160-bit digital signature and that is used for authentication, integrity, and anti-replay.
Data Encryption Standard-Cipher Chaining (DES-CBC)	A secret key algorithm used for confidentiality. A random number is generated and used with the secret key to Block encrypt data.

2.3.2 Public Key Cryptography

Public key cryptography is an asymmetric scheme that uses a pair of keys for encryption. It is called asymmetric because it uses two encryption keys that are mathematically related. These related keys are called the public and private key pair. To use public key encryption, an object (such as a user) must generate a public and private key pair. The object will have only one private key (its own) but may obtain multiple public keys that pair to other private keys. Objects obtain public keys in one of two ways:

1) The owner of the private key sends the receiver the matching public key.
2) The receiver obtains the key from a directory service such as the Active Directory service or Domain Name System (DNS).

A public and private key pair is typically used for two purposes: data encryption and digital message signing.

1) Data Encryption

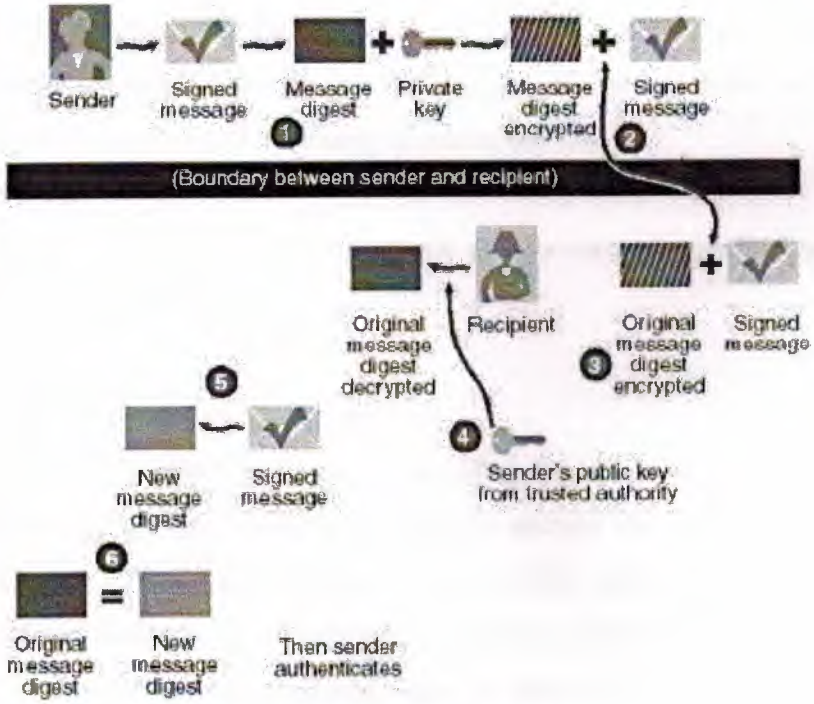
Data encryption provides confidentiality by ensuring that only the intended recipient is able to decrypt and view the original data. When secure data must be transmitted, the sender obtains the recipient's public key. The sender then uses the recipient's public key to encrypt data and then sends it. When the recipient receives the

data, the recipient uses his or her own private key to decrypt the data. Encryption is only secure if the sender uses the recipient's public key for encryption. If a sender uses his or her private key to encrypt data, anyone can capture the data and decrypt it by obtaining the sender's public key.

2) Digital Message Signing

Digital signing provides authentication and integrity but does not provide confidentiality. Digital signing allows a recipient to be certain of the identity of the sender and verifies the content has not been modified during transit. This is to prevent the originator of a message from attempting to send a message under the guise of another identity.

When a sender signs a message, a message digest is created. A message digest is a representation of the message and is similar to a cyclic redundancy check (CRC). The sender uses his or her private key to encrypt the message digest. When the recipient receives the message, the recipient obtains the sender's public key to decrypt the message digest. The recipient then creates a message digest from the message and compares the message digest to the decrypted message digest. If the message digests match, integrity is guaranteed (Figure 2.1).



using a signed message fig (2.1)

Authentication is provided through the key pair. Since the message digest was encrypted by using the sender's private key (and only the sender's public key will decrypt the message digest), the recipient can be certain that the message came from the owner of the key pair. The recipient, however, must have a mechanism for ensuring that the key pair belongs to the intended sender and not someone impersonating the sender. This is done through a certificate issued by a trusted third party, which confirms the identity of the owner of the public key. The trusted third party is known as a Certificate Authority (CA), which will be discussed later in the lesson.

2.3.3 Secret Keys

A *secret key* (also known as *shared secret* or *shared secret key*) is used in much the same way as a public key; however, there is only one key that provides security. Secret keys are generally used only for a particular session or for a short period of time before being discarded. This process holds an advantage over public keys. For example, if an unauthorized person became aware of the key, that person may be able to gain access to a session. However, the unauthorized person would not be able to impersonate either the user or computer outside of the session, and would not have access to other resources with the secret key.

In order to get the shared secret key to both parties, there must exist a mechanism for doing so without compromising security. If the key was sent over the network, an eavesdropper would have easy access to the key.

NOTE:

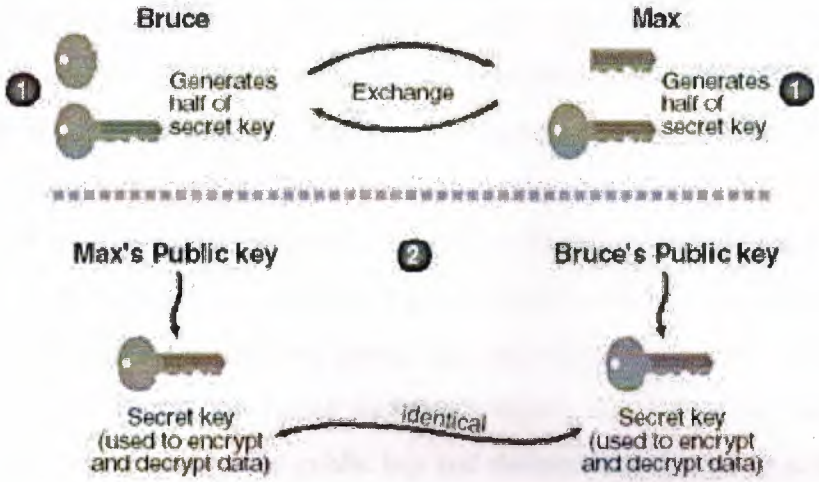
An *eavesdropper* is someone using a network-monitoring tool to capture packets on the network.

2.3.3.1 Secret Key Exchange

A common solution to providing the secret key to both parties is using public keys. Public keys make it possible to encrypt the secret key as it is sent across the network. Public keys ensure confidentiality, authentication, and integrity; therefore, security is not compromised when a secret key is sent.

For example, if Bruce wants to send data to Max by using a secret key, Bruce and Max will each generate half of the secret key. Bruce will obtain Max's public key to encrypt his half of the secret key and send it to Max. Likewise, Max will obtain Bruce's public key to encrypt his half of the secret key and send it to Bruce. Bruce and Max then

combine the halves of the secret key to generate the shared secret key to be used for encrypting the data to be sent (Figure 2.2). This secret key negotiation and the use of the secret key to encrypt the data provide authenticity, integrity, and confidentiality.



secret key exchange fig (2.2)

2.3.3.2 Data Encryption

In order to provide confidentiality, the data must be encrypted by using the shared secret key. Because there is only one key known to both the sender and the receiver, encryption is a straightforward process. The sender encrypts the data with the shared secret key and the receiver decrypts it with the shared secret key. Since no other entity on the network has knowledge of the secret key, the data is safe from attack. The sender and the receiver generally discard shared secret keys once the session has been terminated.

2.4.1 Certificates

Public key encryption assumes that the identity of the key pair owner is established beyond doubt. A *digital certificate*, also referred to simply as a *certificate*, is a set of data that completely identifies an entity. A trusted Certificate Authority (CA) issues certificates after the authority has verified the entity's identity. The CA provides a trusted third party for both communicating parties.

For example, if Tucker wants to send authenticated data to Max, Tucker sends his public key to Max. A trusted CA certifies Tucker's public key, thus certifying Tucker's identity. Because Max trusts the CA, he trusts Tucker.

This process is similar to that of a notary public. A person signs a document in front of a notary public and provides proof of identity. The notary public is a trusted entity so that anyone examining the document can be sure that the signature is authentic. Likewise, when the sender of a message signs the message with a private key, the recipient of the message can use the sender's public key, signed by a trusted CA, to verify that the sender is legitimate. Since the trusted CA certifies the public key, the recipient can be sure that the sender is the assumed sender. A trusted CA may be a third-party provider of certificates such as VeriSign or Microsoft Certificate Services.

A user, for example, can obtain a digital certificate for use with e-mail. The digital certificate includes the public key and information about the user. When the user sends e-mail, the e-mail includes a digital signature that uses the private key. The recipient obtains the public key and determines whether or not the sender of the mail message is the assumed sender. A private key is never sent to the recipient.

1) X.509

The term *X.509* refers to the International Telecommunication Union-Telecommunication (ITU-T) standard for certificate syntax and format. The Windows 2000 certificate-based processes use the X.509 standard. Because it is possible to use certificates for different applications (for example, secure e-mail, file system encryption), each certificate has different information contained within it. However, certificates should, at a minimum, contain the following attributes:

1) Version
2) Serial number
3) Signature algorithm ID
4) Issuer name
5) Validity period
6) Subject (user) name
7) Subject public key information
8) Issuer unique identifier
9) Subject unique identifier
10) Extensions
11) Signature on the above fields

2) Certificate Revocation Lists

Certificates, like most real-world forms of identification, can expire and become invalid. The CA can also revoke them for other reasons. In order to handle the existence of invalid certificates, the CA maintains a certificate revocation list (CRL). The CRL is available to network users to determine the validity of any given certificate.

2.4.2 CA Hierarchy

Rather than having one trusted CA provide authentication for the entire Internet or intranet, it is possible to have CAs certify other CAs. This hierarchical structure, called chaining, allows users to trust a single CA rather than having to trust all CAs. This chaining of CAs provides several benefits:

1) **Flexibility** It is easy to move, revoke, or chain CA's without affecting other parts of the organization.
2) **Distributed Administration** Administrators can be responsible for their own sites.
3) **Security Policies** Security policies can be different at each CA site.

The CA at the top of the chain is referred to as the root CA. CAs below the root are referred to as intermediate, subordinate, or issuing CAs.

2.5.1 Microsoft Certificate Services

Microsoft Certificate Services enables an organization to manage the issuance, renewal, and revocation of digital certificates without having to rely on external certificate authorities. In addition, Certificate Services allows an organization to fully control the policies associated with issuing, managing, and revoking certificates, as well as the format and contents of the certificates themselves. In addition, Certificate Services logs all transactions, enabling the administrator to track, audit, and manage certificate requests.

1) Certificate Services Features

Microsoft Certificate Services has a number of features that make it valuable to organizations that do not choose to rely upon external certificate authorities and who require a flexible tool that can be adapted to the needs of their organization.

2) Policy Independence

In order to obtain a certificate, requesters must meet certain criteria. This criteria is defined in certificate policies. For example, one policy may grant commercial certificates only if applicants present their identification in person. Another policy may grant credentials based on e-mail requests.

Policies are implemented in policy components that can be written in Java, Visual Basic, or Microsoft C/C++. The default policy for Certificate Services allows users to request certificates through an HTML page.

3) Transport Independence

Certificate Services can request and distribute certificates through any transport mechanism. That is, it can accept certificate requests from an applicant and post certificates to the applicant through Hypertext Transfer Protocol (HTTP), remote procedure call (RPC), disk file, or custom transport.

4) Adherence to Standards

Microsoft Certificate Services can perform the following services:

1) Accept standard Public Key Cryptography Standards (PKCS) #10 requests.
2) Support PKCS #7 cryptographically signed data.
3) Issue X.509 version 1.0 and 3.0 certificates.

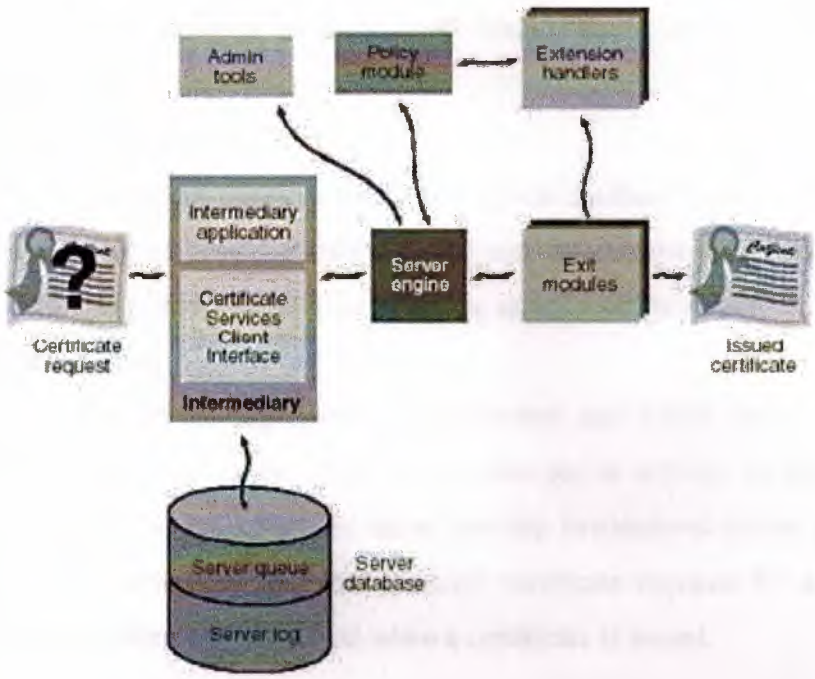
Support for additional certificate formats can be added to Certificate Services. Certificate Services includes an LDAP component so that Certificate Services can integrate with the Active Directory service.

5) Key Management

The security of a certification system depends on the protection of private keys. The design of Certificate Services ensures that individuals cannot access private key information without authorization. Certificate Services relies on Microsoft CryptoAPI to provide key management functionality and other cryptographic capabilities for building a secure store, with certificates kept in a certificate store.

2.5.2 Certificate Services Architecture

Certificate Services architectural elements include the server engine that handles certificate requests and other modules that perform tasks by communicating with the server engine. Figure 2.3 illustrates how the components communicate with the server engine.



server engine fig (2.3)

1) Server Engine

The server engine is the core component of Certificate Services. The engine acts as a broker for all requests it receives from the entry modules, driving the flow of information between components during the processing of a request and generation of a certificate. At each processing stage, the engine interacts with the various modules to ensure appropriate action is taken based on the state of the request.

2) Intermediary

The intermediary is the architectural component that receives new certificate requests from clients and submits them to the server engine. The intermediary is composed of two parts: the intermediary application that performs actions on behalf of clients and the Certificate Services Client Interface that handles communications between the intermediary application and the server engine.

Intermediary applications can be written to handle certificate requests from different types of clients, across multiple transports, or according to policy-specific criteria. Microsoft Internet Information Services (IIS) is an intermediary application that provides support for clients over HTTP. Intermediaries can also check on the status of a previously submitted request and obtain the Certificate Services' configuration information.

previously submitted request and obtain the Certificate Services' configuration information.

3) Server Database

Certificate Services includes a server database that maintains status information and a log of all issued certificates and certificate revocation lists (CRLs). The database is composed of two parts: the server log and the server queue.

4) Server Log

The server log stores all certificates and CRLs issued by the server so that administrators can track, audit, and archive server activity. In addition, the server log is used by the server engine to store pending revocations before publishing them in the CRL. The server log also stores recent certificate requests for a configurable period in case a problem is encountered when a certificate is issued.

5) Server Queue

The server queue maintains status information (receipt, parsing, authorization, signing, and dispatch) as the server processes a certificate request.

6) Policy Module

The policy module contains the set of rules governing issuance, renewal, and revocation of certificates. All requests received by the server engine are passed to the policy module for validation. Policy modules are also used to parse any supplemental information provided within a request and set properties on the certificate accordingly.

7) Extension Handlers

Extension handlers work in tandem with the policy module to set custom extensions on a certificate. Each extension handler acts as a template for the custom extensions that should appear in a certificate. The policy module must load the appropriate extension handler when it is needed.

8) Exit Modules

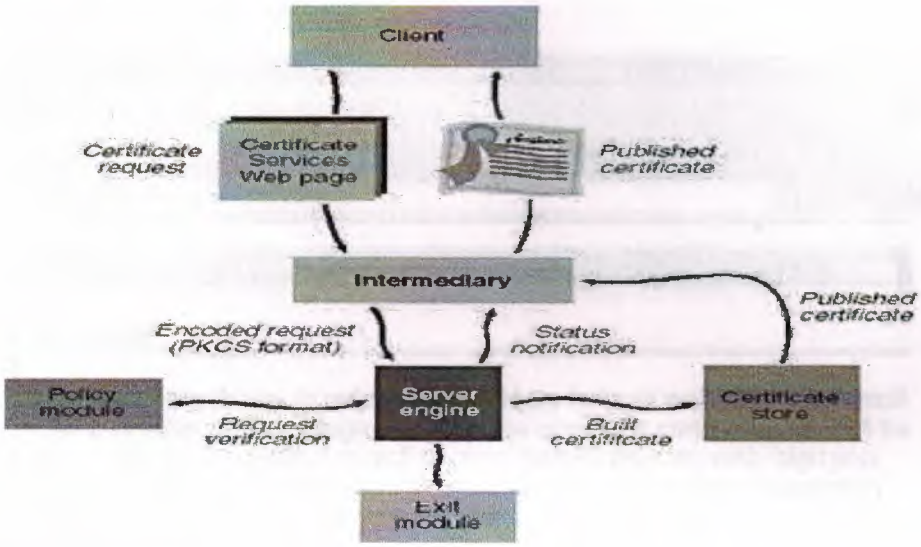
Exit modules publish completed certificates and CRLs through any number of transports or protocols. By default, the server notifies each exit module installed on the server whenever a certificate or CRL is published.

Certificate Services provides a Component Object Model (COM) interface for writing custom exit modules for different transports and protocols or for custom delivery options. For example, an LDAP exit module might be used to publish only client certificates in a directory service and not server certificates. In this case, the exit module

can use the COM interface to determine the type of certificate that the server is issuing and filter out any that are not client certificates.

2.5.3 Processing Certificate Requests

Certificate Services provides services for processing certificate requests and issuing digital certificates (Figure 2.4).



processing certificate fig (2.4)

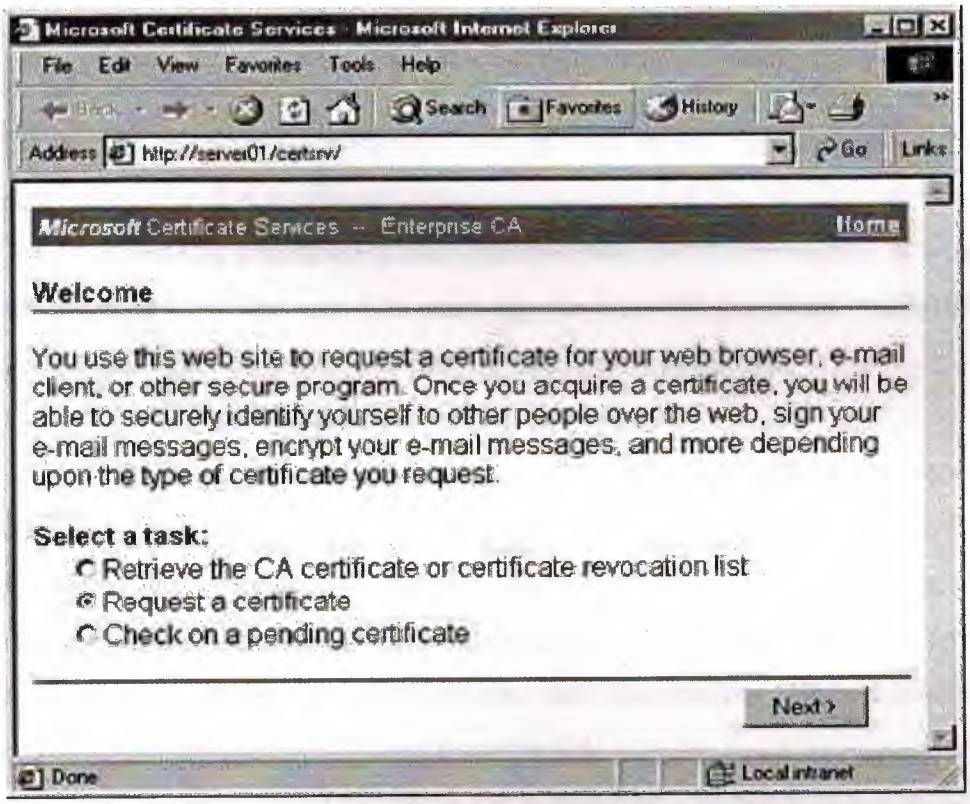
Certificate Services performs the following steps when processing a certificate request:

1. The certificate request is sent by the client to an intermediary application. The intermediary application formats it into a PKCS #10 format request and submits it to the server engine.
2. The server engine calls the policy module, which queries request properties, decides whether or not the request is authorized, and sets optional certificate properties.
3. If the request is approved, the server engine takes the request and builds a complete certificate.
4. The server engine stores the completed certificate in the certificate store and notifies the intermediary application of the request status. If the exit module has so requested, the server engine notifies it of a certificate issuance event. This allows the exit module to perform further operations, such as publishing the certificate to a directory service.
5. The intermediary gets the published certificate from the certificate store and passes it back to the client.

1) Enrolling Certificates

The process of obtaining a digital certificate is called *certificate enrollment*. This process begins with a client submitting a certificate request and ends with the installation of the issued certificate in the client application.

The enrollment control and its forms are accessed through the Certificate Services Enrollment Page. This page is available from the Certificate Services Web page at *http://server_name/certsrv/*.



enrollment control fig (2.5)

2.5.4 CA Certificates

In the process of issuing a digital certificate, the CA validates the identity of the individual requesting the certificate and then signs the certificate with its own private key.

A client application, such as Microsoft Internet Explorer, checks the CA signature before accepting a certificate. If the CA signature is not valid or if it comes from an unknown source, Internet Explorer warns the user by displaying a security message and may prevent the user from accepting the certificate.

NOTE:

If Internet Explorer is set to the low security level, it will not warn the user of invalid certificates. This setting is appropriate for highly trusted intranet environments and is inappropriate for Internet access.

In addition to the server and client authentication certificates issued by Certificate Services, there are certificates that identify CAs.

The CA certificate is a signature certificate that contains a public key used to verify digital signatures. It identifies the CA that issues authentication certificates to the servers and clients that request these certificates. Clients use the CA certificate of the CA issuing the server certificate to validate the server certificate. Servers use the CA certificate of the CA issuing the client certificate to validate the client certificate.

A self-signed CA certificate is also called a root certificate because it is the certificate for the root CA. The root CA must sign its own CA certificate because by definition there is no higher certifying authority to sign its CA certificate.

1) Distribution and Installation of CA Certificates

CA certificates are not requested and issued in the same manner as server and client authentication certificates. Server and client authentication certificates are unique for each requesting server and client, and are not shared—they must be generated and issued by a CA upon demand. In contrast, the CA certificate does not require issuance upon demand. Instead, it is created once and then made readily available to all servers or clients who request certificates from the CA.

A commonly used technique for distributing CA certificates is to place them in a location known and accessible to anyone who requests certificates from the CA.

2.5.5 Installing Certificate Services

You can install Certificate Services by using the Add/Remove Programs utility in Control Panel or optionally during the installation of Windows 2000 Server. Administrators familiar with creating CAs can choose a custom setup by using the advanced options available when installing Certificate Services. Those unfamiliar with creating CAs can select the default settings.

A) Certificate Authority Type

The CA type allows selection of how the CA will be utilized in a CA hierarchy and whether or not the CA will rely upon Active Directory services. The following certificate authority types are available:

1) **Enterprise Root CA** This CA becomes the root CA for the hierarchy and requires Active Directory services.
2) **Enterprise Subordinate CA** This CA becomes a subordinate CA to an Enterprise Root CA. It requires Active Directory services. It will request a certificate from the Enterprise Root CA.
3) **Stand-alone Root CA** This CA becomes the root CA for the hierarchy but does not require Active Directory services.
4) **Stand-alone Subordinate CA** This CA becomes a subordinate CA to a Stand-alone Root CA. It does not require Active Directory services. It requests a certificate from the Stand-alone Root CA.

When installing the Certificate Services as an Enterprise CA, Certificate Services copies the certificates into Active Directory services. Security support providers such as Kerberos can query Active Directory services to get the certificate, which contains the public key.

B) CA Information

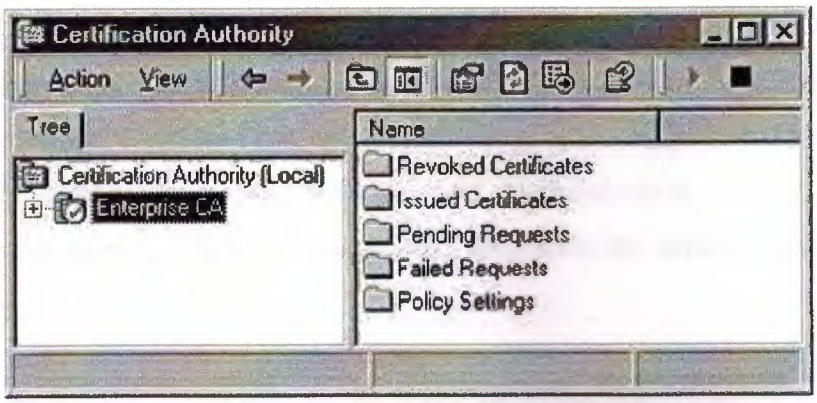
You must supply information about the initial CA that is created when you install Certificate Services. This information includes the CA name and other necessary information. None of this information can be changed after the CA setup is complete.

C) Advanced Configuration

The advanced configuration contains options for the type of cryptography algorithms to be used for the CA that you are creating. The advanced configuration options include the name of the cryptographic provider, the hash algorithm, the option to use existing public keys and private keys, and the key length.

2.5.6 Administering Certificate Services

The main tool used to administer Certificate Services is the Certification Authority snap-in (Figure 2.6).



certificate authority fig (2.6)

The snap-in allows you to perform a variety of administrative tasks:

1) Start or stop the CA service
2) Set security permissions and delegate control of a CA
3) View a CA certificate
4) Back up a CA
5) Restore a CA from a backup copy
6) Renew a root CA
7) Renew a subordinate CA
8) Manage certificate revocation
9) Manage certificate requests
10) Manage certificate templates
11) Change policy settings
12) Map certificate to user accounts
13) Modify the Policy Module or Exit Module

You can use the Certification Authority snap-in to administer a certification authority on the local computer or on a another computer. The snap-in is installed when Certificate Services are installed or when installing the Administration Pack (Adminpak.msi).

Certutil.exe is a command-line utility used for administering certificate services. Running certutil without any command-line switches displays summary information about the local certificate authority. Certutil is used to dump and display CA configuration information, configure Certificate Services, back up and restore CA components, and verify certificates, key pairs, and certificate chains.

If you need to set security for the CA Web pages, you should use the Internet Information Services snap-in. Expand the Default Web Site from the console tree and then select CertSrv. From the Action menu, select Properties. On the Directory Security tab, under Anonymous access and authentication control, click Edit. In the Authentication Methods dialog box, configure the security settings for the CA Web pages.

2.5.7 Installing and Configuring Certificate Services

In this exercise you install an Enterprise Root CA and use this CA to issue, install, and revoke certificates. Note that the secure way to configure Certification Services is to create a root CA that only issues certificates to subordinate CA types. The subordinate CA types then issue certificates for specific purposes such as application services and authentication. Using a root CA for this purpose is not secure because if the root CA security is breached, all certificates issued are compromised. However, for the purpose of learning how to install and configure certificate services, a root CA can be used.

A) Installing Certificate Services and configuring the Certificate Authority

In this procedure, you install Certificate Services on Server01. Server01 acts as an Enterprise Root CA.

1. Log on to Server01 as Administrator with a password of "password."
2. Click Start, point to Settings and then click Control Panel.
   Control Panel appears.
3. Double-click the Add/Remove Programs application.
   The Add/Remove Programs window appears.
4. In the left pane, click the Add/Remove Windows Components icon.
   The Windows Components wizard appears.
5. Click the Certificate Services check box.
   A Microsoft Certificate Services message box appears stating that once Certificate Services is installed, the computer cannot be renamed and it cannot join or be removed from a domain.
6. Click Yes.
7. On the Windows Components screen, click Details.
   The Certificate Services window appears.

Notice that Certificate Services subcomponents include both the service used to create a certificate authority and a Web enrollment form for submitting requests and retrieving certificates from the computer running as a CA.

8. Click OK.
9. On the Windows Components screen, click Next.

   The Certification Authority Type screen appears.
10. Select each radio button and read the text appearing in the Description box.

    Notice that the Enterprise CA types can only be used if Active Directory services is running. The stand-alone CA types run independently of Active Directory services. Thus, they can be used in the presence or absence of Active Directory services. If Active Directory services is present, the stand-alone CA types will use it. Subordinate CA types are dependent on the presence of a CA higher up in the CA hierarchy.
11. Click the Enterprise Root CA radio button and click the Advanced options check box.
12. Click Next.

    The Public and Private Key Pair screen appears.

    Notice that there are a number of Cryptographic Service Providers (CSPs), each having one or more associated hash algorithms used to generate key pairs. From this screen you can also specify the key length or use existing keys installed on the computer, import keys, and view certificates.
13. In the CSP list box, verify that Microsoft Base Cryptographic Provider v1.0 is selected. In the Hash Algorithm list box, verify that the SHA-1 hash algorithm is selected. In the Key Length drop-down list box, verify that Default is selected. Click Next.

    The CA Identifying Information screen appears.
14. Type the information in the table into the text boxes on the CA Identifying Information screen.

Label	**Value to type**
CA name	Enterprise CA
Organization	Microsoft Corporation
Organizational unit	Microsoft Press
City	Redmond

State or province	Washington
E-mail	ca-mp@microsoft.com
CA description	Root CA for self-study training only

15. Notice that this certificate is configured to be valid for two years.
16. Click Next.

    The Data Storage Location screen appears.

    Notice that the certificate database and log file folder, CertLog, is stored on the boot partition. If disk capacity on the boot partition is limited, consider specifying another secure partition for the certificate database and log folder.

    The Store configuration information in a shared folder is not necessary if Active Directory services is running and the computer operating as the certificate authority is a member of a domain. Configuration information about the CA is automatically published to the Active Directory store.
17. Click Next.

    A Microsoft Certificate Services message box appears stating that Internet Information Services is running on the computer and warning you that it must be stopped in order for you to be able to continue.
18. Click OK.

    The Configuring Components screen appears as the software is installed and configured, and then the Completing the Windows Components Wizard screen appears.
19. Click Finish and then on the Add/Remove Programs window, click Close.
20. Close Control Panel.

B) Running Certificate Services

In this procedure you will generate, install, and revoke a certificate on Server01. You will use the Certificate Enrollment URL and the Certificate Authority snap-in to complete this procedure.

1. Open Certification Authority from the Administrative Tools program group.

   The Certification Authority snap-in appears.
2. In the console tree, expand the Enterprise CA node.
3. In the console tree, select the Issued Certificates folder and then minimize the Certification Authority snap-in.

4. Click the Start menu and then choose run.

   The Run dialog box appears.

5. In the Open text box, type **http://server01/certsrv** and then click OK.

   The Internet Connection wizard appears.

6. Click the I Want To Setup My Internet Connection Manually, or I Want To Connect Through A Local Area Network (LAN) radio button.

7. Click Next.

   The Setting Up Your Internet Connection screen appears.

8. Click the I Connect Through A Local Area Network (LAN) radio button.

9. Click Next.

   The Local Area Network Internet Configuration screen appears.

10. Clear the Automatic Discovery Of Proxy Server (Recommended) check box.

11. Click Next.

    The Set Up Your Internet Mail Account screen appears.

12. Click the No radio button and then click Next.

    The Completing The Internet Connection wizard appears.

13. Click Finish.

    Internet Explorer appears and displays the certificate services enrollment page.

14. Read the information on this page and then verify that the Request A Certificate radio button is selected.

15. Click Next.

    The Choose Request Type page appears and the User Certificate Request radio button is selected.

16. Click Next.

    The User Certificate—Identifying Information page appears.

17. Click More Options.

    Notice that the CSP selected was the CSP type you specified during installation of Certificate Services.

18. Click Submit.

    The Certificate Issued page appears.

19. Minimize Internet Explorer and restore the Certification Authority snap-in.

    The Certification Authority snap-in appears and one certificate is listed in the details pane. If you don't see the certificate request, press F5 to refresh the details pane.

20. Double click the certificate appearing in the details pane.

    The Certificate dialog box appears with three tabs.

21. Click the Details tab.

22. In the top box below the Show drop-down list box, click Issuer.

    Notice that the information appearing in the bottom box is the information you typed into the CA Identifying Information screen.

23. Click OK.

24. Minimize the Certification Authority snap-in and restore Internet Explorer.

25. Click the Install This Certificate hyperlink.

    The Certificate Installed page appears stating that you have successfully installed a certificate.

26. Close Internet Explorer.

27. Restore the Certification Authority snap-in and select the certificate in the details pane.

28. Click the Action Menu, point to All Tasks and then click Revoke certificate.

    The Certificate Revocation dialog box appears.

29. In the Reason Code drop-down list box choose Key Compromise and then click Yes.

30. In the console tree, click the Revoked Certificates folder.

    The revoked certificate appears in the details pane.

31. Click the Action menu, point to All Tasks and then click Publish.

    The Certificate Revocation List dialog box appears stating that the previous list is still valid.

32. Click Yes.

33. Close the Certification Authority snap-in.

34. Click the Start menu, and then click Run.

    The URL to the Certsrv directory appears.

35. Click OK.

    Internet Explorer appears and displays the certificate services enrollment page.

36. Click the Retrieve The CA Certificate Or Certificate Revocation List radio button and then click next.

37. Click the Download Latest Certificate Revocation List hyperlink.

    The File Download dialog box appears.

38. Click the Open This File From Its Current Location radio button and then click OK.

    The Certificate Revocation List dialog box appears.

39. Click the Revocation List tab.

40. In the Revoked Certificates box, click the item that appears.

    In the Revocation entry box, the Serial number of the revoked certificate, the date of revocation, and the reason for revocation appear.

41. Click OK.

42. Close Internet Explorer.

2.6 Chapter Summary

Windows 2000 includes a native PKI that is designed to take full advantage of the Windows 2000 security architecture. Public key cryptography is an asymmetric scheme that uses a pair of keys for encryption. To use public key encryption, a user must generate a public and private key pair. Public key encryption uses digital certificates to completely identify the key pair owner. The Windows 2000 certificate-based processes use the X.509 standard. Certificate Services enables an organization to manage the issuance, renewal, and revocation of digital certificates without having to rely on external CAs. Certificate Services supports policy independence, transport independence, adherence to standards, and key management. Certificate Services architectural elements include the server engine that handles certificate requests and other modules that perform tasks by communicating with the server engine. Certificate Services provides services for processing certificate requests and issuing digital certificates. You can install Certificate Services by using the Add/Remove Programs utility in Control Panel or optionally during Windows 2000 Server installation. The tools used to administer Certificate Services once it is installed are the Certification Authority snap-in, the Certutil utility, and the Certificate Services enrollment Web page.

CHAPTER 3

THE KERBEROS PROTOCOL IN WINDOWS 2000

3.1 OVER VIEW

A standard process within computer security is to include a function that requires users to prove that they are who they claim to be. This affirmation of identity is accomplished when the user supplies the correct password for the user account. For example, when User1 attempts to connect to a server to access a file, the server must be sure that it is really User1 sending the request. Traditionally, the server assumes that it is User1 because the correct password was supplied when the connection was established. Stronger security is accomplished by having a trusted third party verify the identity of the user. This is a core function of the Kerberos authentication protocol.

3.2.1 The Kerberos Protocol

The Kerberos protocol is the default authentication provider in Windows 2000 and the primary security protocol. It allows users to use a single logon to access all resources. The Kerberos protocol verifies both the identity of the user and the integrity of the session data. This is accomplished by having a Kerberos service installed on each domain controller and a Kerberos client installed on all computers running Windows 2000.

NOTE:

The Active Directory client for Windows 95 and Windows 98 allows users to log on by using the Kerberos V5 authentication protocol.

When the Kerberos authentication protocol is used, a trusted Kerberos service on a server verifies the user's identity. Before connecting to the server the user requests a ticket from the Kerberos service, called the Kerberos Key Distribution Center service, to confirm the user's identity. The user then sends this ticket to the target server. Because the server trusts the Kerberos service to vouch for user identities, the server accepts the ticket as proof of the authenticity of the user.

When using the Kerberos authentication protocol, users can no longer log on and then access resources simply by providing a valid user ID and the correct password. Instead of trusting the source, the resource must contact the Kerberos service to obtain a ticket that vouches for the user. The Kerberos service operates as a trusted third party to generate session keys and grant tickets for specific client/server sessions.

When the Kerberos service issues a ticket, it contains the following components:

1) Session key
2) Name of the user to whom the session key was issued
3) Expiration period of the ticket
4) Any additional data fields or settings that may be required

The expiration period of a ticket is defined by the domain policy. If a ticket expires during an active session, the Kerberos service notifies the client and the server to refresh the ticket. The Kerberos service then generates a new session key and the session is resumed.

3.2.2 Kerberos Protocol Terms

To better understand the Kerberos protocol, you should review the following terms used to describe the various components of Kerberos.

1) Principal

A *principal* is a uniquely named user, client, or server that participates in a network communication.

2) Realm

A *realm* is an authentication boundary, which can be compared to a Windows 2000 domain. Each organization wishing to run a Kerberos server establishes its own realm. A Windows 2000 domain is a Kerberos realm but is named domain to maintain naming conventions established previously for Windows NT.

3) Secret Key

A *secret key* is an encryption key that is shared by a client or a server and a trusted third party to encrypt the information that is to be moved between them. In the case of Kerberos, the trusted third party is the Kerberos service. In the case of a principal, the secret key is typically based upon a hash or encryption of the principal's password. Secret keys are never transmitted on the network; only the encrypted information is transmitted.

4) Session Key

The *session key* is a temporary encryption key used between two principals, with a lifetime limited to the duration of a single login session. The session key is exchanged between the communication partners and is therefore known as a shared secret. The session key is always sent encrypted.

5) Authenticator

An *authenticator* is a record that is used to verify that a request actually originated from the principal. An authenticator contains information that verifies the identity of the sender and the time the request was initiated. This information is encrypted with the shared session key that is known only by the communicating principals. An authenticator is typically sent along with a ticket to allow the receiver to verify that the intended client recently initiated a request.

6) Key Distribution Center

The *key distribution center* (KDC) provides two functions: the authentication server (AS) and the ticket granting service (TGS). The TGS distributes tickets to clients that wish to connect to services on the network. However, before a client can use the TGS to obtain tickets, it must first obtain a special ticket (the ticket granting ticket [TGT]) from the AS.

7) Privilege Attribute Certificate

The *privilege attribute certificate* (PAC) is a structure that contains the user's security ID (SID).

8) Tickets

In a basic Kerberos exchange, the client will contact the TGS and request a ticket for the target server before contacting the target server. A *ticket* is a record that allows a client to authenticate itself to a server; it is simply a certificate issued by the Kerberos service. The ticket is encrypted so that only the target server is able to decrypt and read it. Tickets contain the identity of the requesting client, the timestamp, the servers session key, the lifetime of the ticket, and other information (such as the PAC) that will help verify the identity of the client to the target server. Tickets are reusable within their life span, which is usually 8 hours.

9) Ticket Granting Tickets

One method for using Kerberos is to simply request a ticket for each target server from the TGS portion of the Kerberos service whenever the user wants to access the specified target server. Using this method, the response from the request would contain a session key and other information that is encrypted with the user's secret key. This method results in a component of the user's secret key being exposed on the network every time a new ticket request is made.

In Windows 2000, Kerberos protects the secret key by initially authenticating the user and then requesting a ticket granting ticket (TGT). A *ticket granting ticket* is a request for a ticket and a random session key to be used with the TGS portion of the Kerberos service. After obtaining the ticket, the user can contact a service at any time; the requested ticket does not come from the AS, but from the TGS. The reply is encrypted not with the user's secret key, but with the session key that the AS provided for use with the TGS.

3.2.3 Features of the Kerberos Protocol

The Kerberos protocol has several advantages over traditional challenge/response authentication systems.

1) Mature Open Standard

The Windows 2000 implementation of the Kerberos protocol complies with RFC 1510 and RFC 1964. It can interoperate with other implementations of Kerberos that also comply with the RFCs. Therefore, Kerberos clients on other platforms, such as UNIX, can be authenticated by Windows 2000. In some cases, however, implementation-dependent values will not exist or will be unavailable. In the absence of required data, the Windows 2000 Kerberos service attempts to match the principal name in the ticket either to a Windows 2000 user account or to a default account created for this purpose.

2) Faster Connection Authentication

When using the Kerberos protocol, servers do not need to do pass-through authentication. A server running Windows 2000 can verify the client credentials by using the client-supplied ticket, without having to query the Kerberos service. This is because the client will have already obtained a Kerberos ticket from a domain controller, which the server can then use to build the client's access token. Since the

server is required to do less work when establishing a connection, it can more easily accommodate a large number of simultaneous connection requests.

3) Mutual Authentication

The Kerberos protocol provides mutual authentication of both the client and server. The Windows NTLM authentication protocol provides only client authentication, and it assumes that all servers are trusted. It does not verify the identity of the server that a client connects to. The assumption that all servers can be trusted is no longer valid. Mutual authentication of both client and server is an important foundation for secure networks.

4) Delegation of Authentication

Delegation of authentication allows a user to connect to an application server, which in turn can connect to one or more additional servers on the client's behalf, by using the client's credentials.

5) Transitive Trusts

Authentication credentials issued by one Kerberos service are accepted by all Kerberos services within the domain.

3.2.4 Kerberos Authentication Process

The Kerberos authentication process involves the client computer negotiating exchanges between the target server and the KDC. Figure 3.1 provides an overview of the authentication process. The numbered steps in the diagram are described below.

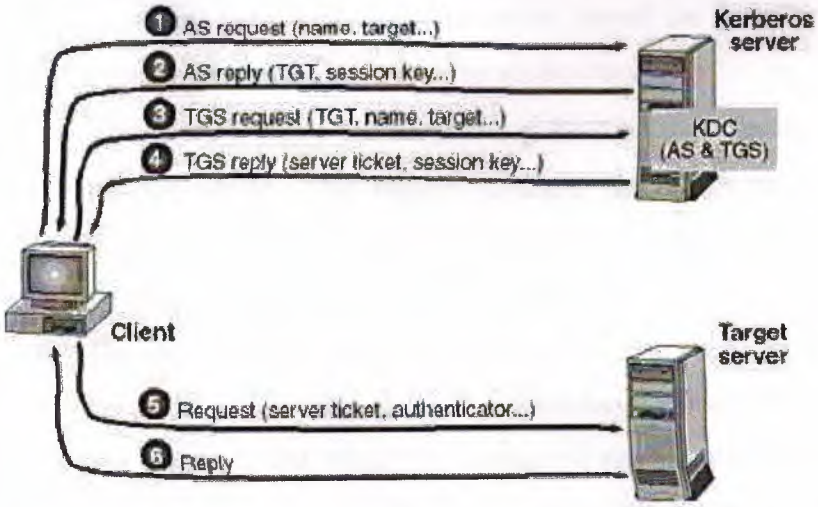


Figure 3.1 *Kerberos authentication process*

The Kerberos authentication process works as follows:

1. The client sends an initial AS request to the AS portion of the Kerberos service. The AS includes the client's principal name and the principal name of the target server for which it is requesting a ticket.
2. The Kerberos service generates an AS reply and sends it to the client. The reply contains the following:
    1. A TGT for the TGS portion of the Kerberos service. The TGT is encrypted with the TGS secret key. The TGT contains the user's SID. By encrypting the TGT with the TGS secret key, the client is unable to change the SID properties.
    2. A session key for exchanges with the TGS portion of the Kerberos service. The session key is encrypted with the client's secret key. The client's secret key is a computation of the client's password. It is similar to the session key used in NTLM challenge/response. The encryption here makes it difficult for someone to steal the session key.
3. The client generates and sends a TGS request that contains the client's and target server's principal names, realms, and the TGT that identifies the client.
4. The TGS portion of the Kerberos service generates and sends a TGS reply to the client. This reply contains a ticket for the target server. The ticket is encrypted with the server's secret key. The server's secret key is a computation of the password generated when the server joined the domain. The reply also includes other information, including the session key.
5. The client extracts the session key for the target server and generates a request for the server. This request contains the target server and an authenticator encrypted with the session key. The client sends this request to the target server by using an established transport path.
6. The target server decrypts the ticket by using its secret key to obtain the session key. The server then uses the session key to decrypt the authenticator to verify the client. If the client has requested mutual authentication, the target server generates a reply encrypted with the session key and send it to the client. Mutual authentication not only authenticates the client to the target server, but also authenticates the target server to the client.

NOTE:

The AS and TGS exchanges with the Kerberos service operate over User Datagram Protocol (UDP) port 88. The exchanges between the client and target server are dependent on the protocol in use between the two principals.

3.2.5 Kerberos Delegation

Occasionally, it is necessary for an application server to connect to another server on behalf of a client. Like impersonation, delegation is used to ensure that proper security permissions are applied against the application server's request.

The Kerberos authentication protocol supports delegated authentication. This type of authentication is used when a client transaction involves multiple servers. In this case, each of the verifying servers obtains another ticket and authenticates the ticket to the requested server on behalf of the client. There is no restriction on the number of consecutive servers that can delegate authentication. This is different than impersonation, in that the server accesses remote resources on the behalf of the client instead of local resources.

Figure 3.2 provides an overview of the Kerberos delegation process. The numbered steps in the diagram are described below.

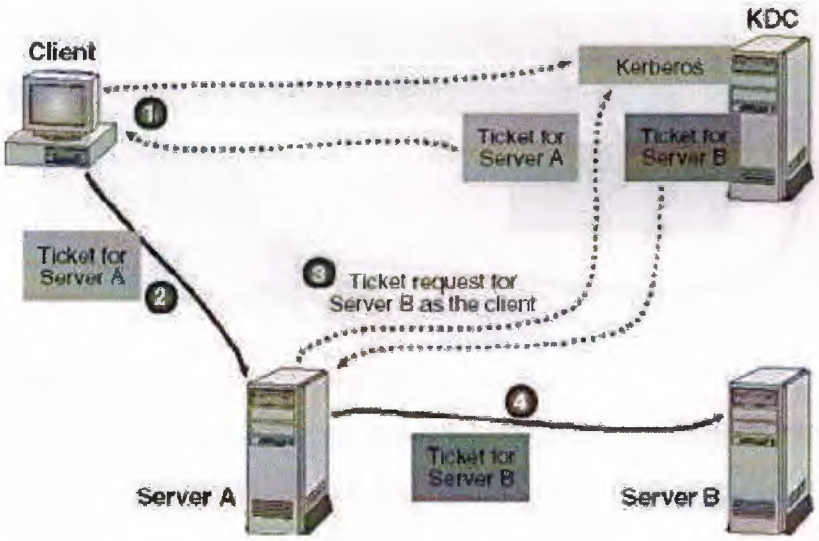


Figure 3.2 *Kerberos delegation process*

The following steps describe the access of resources involving two servers:

1. The client requests and receives a ticket for target Server A from the Kerberos service.
2. The client sends the ticket directly to Server A.

3. Server A sends a request, impersonating the client, to the Kerberos service for a ticket for target Server B. The Kerberos service responds with a ticket that allows the client to access Server B.
4. Server A can then send the ticket to Server B, accessing Server B as the client.

3.3.1 Kerberos Logon Processes

The addition of Kerberos as an authentication package in Windows 2000 affects various aspects of the logon process. However, the portions of the logon process that run before an authentication package becomes involved remain unchanged in Windows 2000.

3.3.2 Local Interactive Logon

When a local interactive logon occurs, the user logs on with a user account that exists on the local computer rather than with a domain user account. Figure 11.14 provides an overview of the local interactive logon process in Windows 2000. The numbered steps in the diagram are described below.

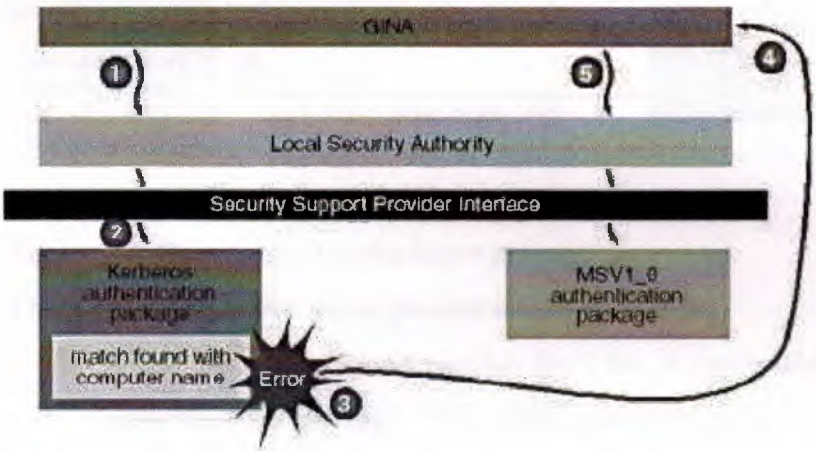


Figure 3.3 *Local interactive logon process*

For local user accounts, the following occurs in Windows 2000:

1. When the Graphical Identification and Authentication DLL (GINA) receives the logon request, it forwards the request to the Local Service Authority (LSA). This request specifies Kerberos as the authentication package to use because this is the default package in Windows 2000.
2. LSA processes the request and sends it to the Kerberos authentication package.

3. When Kerberos receives the logon request. Kerberos returns an error because it is used only when authenticating logon requests for domain user accounts, not local user accounts.
4. LSA receives the error and returns an error to the GINA.
5. The GINA resubmits the logon request to LSA specifying the "MSV1_0" authentication package. The logon process then occurs as it would for a local interactive logon under Windows NT 4.0.

3.3.3 Domain Interactive Logon

The exchange that occurs when a user logs on to Windows 2000 with a domain user account is similar to the basic Kerberos exchange. Figure 3.4 provides an overview of this logon process. The number steps in the diagram are described below.

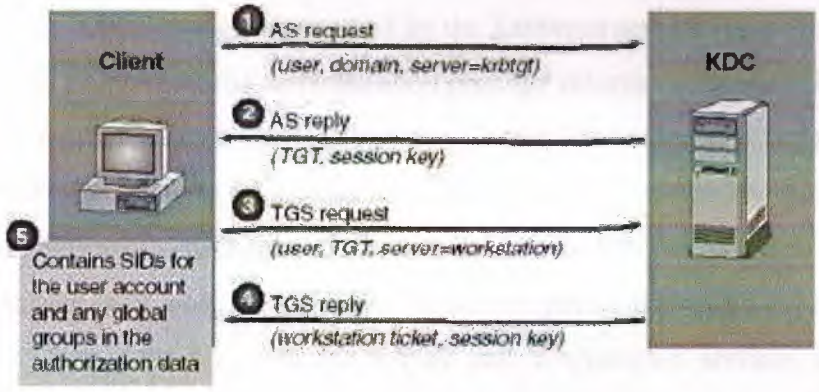


Figure 3.4 *Domain interactive logon process*

The domain interactive logon process occurs as follows:

1. When the logon request reaches the LSA, it passes the request to the Kerberos authentication package. The client sends an initial AS request to the Kerberos service, providing the user name and domain name. This is a request for authentication and a TGT. The request is made by using the principal name of krbtgt@<*domain_name*>, where <*domain_name*> is the name of the domain in which the user account is located. The first domain controller in the domain automatically generates the krbtgt@<*domain_name*> account.
2. The Kerberos service generates an AS reply containing a TGT (encrypted with the Kerberos secret key) and a session key for the TGS exchanges (encrypted with the client's secret key). This response is sent back to the client. The authorization data portion of the TGT contains the SID for the user account and

SIDs for any global groups to which the user belongs. The SIDs are returned to the LSA for inclusion in the user's access token. The SIDs are copied by the Kerberos service from the TGT into subsequent tickets obtained from the Kerberos service.

3. The client then generates and sends a TGS request containing the client's principal name and realm, the TGT to identify the client, and the local workstation name as the target server. This is done to request access to the local computer for the user.
4. The Kerberos service generates and sends a TGS reply. This reply contains a ticket for the workstation and other information, including the session key (encrypted by using the session key from the TGT). Also included in the authorization data portion of the TGS reply are the SIDs for the user account and any global groups copied by the Kerberos service from the original TGT.
5. The Kerberos authentication package returns the list of SIDs to the LSA.

Windows 2000 services use the Kernel Mode Security Support Provider Interface (SSPI) to perform authentication. Instead of communicating directly with the Kerberos authentication package, both services access Kerberos through an authentication package built into LSA. This authentication package is called the Negotiate package.

During startup, both the Server and Workstation services initialize their interface with the Negotiate package in LSA by using SSPI. During this process, the server service obtains a credential handle for its default credentials.

The network communication occurs in two segments: protocol negotiation and session setup. Before a user can establish a session with the server, the client computer and the server must agree on the security protocol to use by determining which version of security they both support. Once the client has been authenticated and has a ticket, it can establish a session with the server.

3.3.4 Kerberos Public Key Support

Windows 2000 extends the functionality of Kerberos to allow it to interact with the Active Directory service. Windows 2000 includes extensions to the Kerberos V5 authentication protocol to support public key-based authentication. The public key extensions allow clients to request an initial TGT by using a private key. The Kerberos service verifies such a request by using the user's public key that is obtained from the user's X.509 certificate published to the Active Directory store. In order to obtain a

ticket, the user's X.509 certificate must be stored in their user object. If the Kerberos service finds the certificate, the Kerberos service issues a ticket for the client and the standard Kerberos procedure is followed thereafter. This replaces the secret key that is known only to the principal and the KDC. Smart cards, for example, use public key extensions provided by Kerberos.

3.4 Chapter Summary

Kerberos is the default authentication provider in Windows 2000 and the primary security protocol. To better understand the Kerberos protocol, you should be familiar with the terms common to Kerberos, including principal, realm, secret key, session key, authenticator, KDC, AS, TGS, PAC, ticket, and TGT. The Kerberos authentication process involves the client computer negotiating exchanges between the target server and the KDC. The Kerberos authentication protocol supports delegated authentication. When a local interactive logon occurs, the user logs on with a user account that exists on the local computer rather than with a domain user account. The exchange that occurs when a user logs on to Windows 2000 with a domain user account is similar to the basic Kerberos exchange. Windows 2000 services use the Kernel Mode SSPI to perform authentication. In addition, Windows 2000 extends the functionality of Kerberos to allow it to interact with Active Directory services. Windows 2000 includes extensions to the Kerberos V5 authentication protocol to support public key-based authentication.

CHAPTER 4

SECURITY CONFIGURATION TOOLS

4.1 OVER VIEW

Windows 2000 provides a set of security configuration tools that are designed to reduce the costs associated with security configuration and analysis of Windows 2000 networks. These tools are MMC snap-ins that allow you to configure Windows 2000 security settings and perform periodic analyses of the system to ensure that the configuration remains intact or to make necessary changes over time. Security settings include security policies (account and local policies), access control (services, files, and the registry), event logs, group membership (restricted groups), IPSec security policies, and public key policies. The security configuration tools include three snap-ins: the Security Configuration And Analysis snap-in, the Security Templates snap-in, and the Group Policy snap-in.

4.2.1 Security Configuration And Analysis Snap-In

The Security Configuration And Analysis snap-in allows you to configure and analyze local system security.

4.2.2 Security Configuration

The Security Configuration And Analysis snap-in can also be used to directly configure local system security. You can import security templates created with the Security Templates snap-in, and apply these templates to the group policy object (GPO) for the local computer. This immediately configures the system security with the levels specified in the template.

4.2.3 Security Analysis

The state of the operating system and applications on a computer is dynamic. For example, security levels may be required to change temporarily to enable immediate resolution of an administration or network issue; this change can often go unreversed.

This means that a computer may no longer meet the requirements for enterprise security.

Regular analysis enables an administrator to track and ensure an adequate level of security on each computer as part of an enterprise risk management program. Analysis is highly specified; information about all system aspects related to security is provided in the results. This enables an administrator to tune the security levels and, most importantly, detect any security flaws that may occur in the system over time.

The Security Configuration And Analysis snap-in enables quick review of security analysis results. Recommendations are presented along with current system settings, and icons or remarks are used to highlight any areas where current settings do not match the proposed level of security. The Security Configuration And Analysis snap-in also allows you to resolve any discrepancies revealed by analysis.

If frequent analysis of a large number of computers is required, as in a domain-based infrastructure, the Secedit command-line tool may be used as a method of batch analysis. However, analysis results still must be viewed by using the Security Configuration And Analysis snap-in. For more information about the Secedit utility, see Windows 2000 Help.

4.2.4 Using the Security Configuration And Analysis Snap-In

The Security Configuration And Analysis snap-in (Figure 4.1) reviews and analyzes your system security settings and recommends modifications to the current system settings. Administrators can use the snap-in to adjust the security policy and detect security flaws that arise in the system.

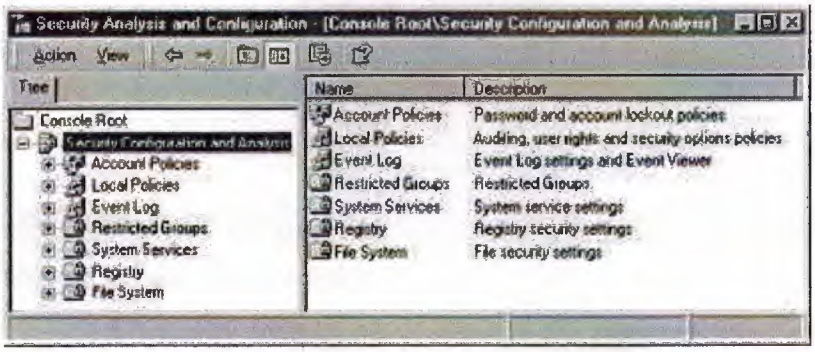


Figure 4.1 *Security Configuration And Analysis snap-in*

The Security Configuration And Analysis snap-in allows you to perform a variety of tasks:

1) Set a working database

2) Import a security template
3) Analyze system security
4) Review security analysis results
5) Configure system security
6) Edit the base security configuration
7) Export a security template

For details about how to perform each of these tasks, see Windows 2000 Help.

4.3.1 Security Templates Snap-In

A security template is a physical representation of a security configuration; it is a file where a group of security settings may be stored. Windows 2000 includes a set of security templates, each based on the role of a computer. The templates range from security settings for low security domain clients to highly secure domain controllers. They can be used as provided, modified, or serve as a basis for creating custom security templates.

4.3.2 Using the Security Templates Snap-In

The Security Templates snap-in (Figure 4.2) is a tool for creating and assigning security templates for one or more computers.

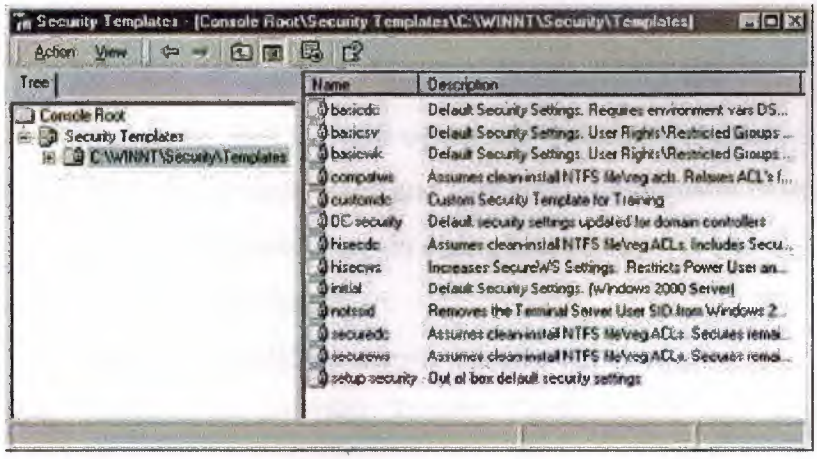


Figure 4.2 *Security Templates snap-in*

A security template is a physical file representation of a security configuration, and can be applied to a local computer or imported to a Group Policy Object (GPO) in the Active Directory service. When you import a security template to a GPO, Group Policy processes the template and makes the corresponding changes to the members of that GPO, which may be users or computers.

The Security Templates snap-in allows you to perform a variety of tasks:

1) Customize a predefined security template.
2) Define a security template.
3) Delete a security template.
4) Refresh the security template list.
5) Set a description for a security template.

C) Creating and Using the Security Analysis And Configuration Snap-In

In this you will create a custom snap-in containing the Security Analysis And Configuration snap-in and the Security Templates snap-in. You then customize a template and open a new database using the custom template. You will then analyze the security settings of Server01 against the template and then you will apply the template's configuration to the security settings of Server01. Complete this exercise on Server01.

1) Creating a Security Analysis And Configuration snap-in:

You will run the MMC and add the Security Analysis And Configuration snap-in. MMC version 1.2, included with Windows 2000, allow you to add multiple snap-ins to an existing console. For the purpose of clarity, you will create a new console rather than adding to an existing consoles running other snap-ins.

1. Log on to Server01 as administrator with a password of "password."
2. Click Start and then click Run.
   The Run dialog box appears.
3. In the Open text box, type **mmc** and then click OK.
   An empty MMC console opens and is named Console1.
4. Click the Console menu and then click Add/Remove Snap-in.
   The Add/Remove Snap-in dialog box appears.
5. Click the Add button.
   The Add Standalone snap-in window appears.
6. Scroll down and click Security Configuration And Analysis and then click the Add button.
7. Click Close.
   The Add/Remove Snap-in dialog box appears.
8. Click OK.
9. Click the Console menu and then click Save.
   The Save As dialog box appears.
10. In the File Name text box, type **Security** and then click Save.

2) Adding and configuring security using the Security Template snap-in to the Security console:

Before analyzing Server01 and applying new security settings, you install the Security Template snap-in to the Security console.

1. Click the Console menu and then choose Add/Remove Snap-in.
   The Add/Remove Snap-in dialog box appears.
2. Click the Add button.
   The Add Standalone Snap-in window appears.
3. Scroll down and click Security Templates and then click the Add button.
4. Click Close.
   The Add/Remove Snap-in dialog box appears.
5. Click OK.
6. Click the Console menu and then click Save.
7. Expand the Security Templates node then expand the C:\WINNT\Security\Templates folder.
   All of the defined templates appear in the console tree and in the details pane.
8. Expand the securedc template.
   This is an incremental security template usually used after a basic security template is applied. For the purpose of this exercise, this template is sufficient.
9. Expand the Account Policies node and then click Password Policy.
   Password policy settings appear in the details pane.
10. In the details pane, double-click Minimum Password Length.
    The Template Security Policy Setting dialog box appears.
11. In the Password Must Be At Least box, change the value to **5** characters and then click OK.
12. In the console tree, click securedc.
13. Click the Action menu and then click Save As.
    The Save As window appears.
14. In the File Name text box, type **customdc** and then click Save.
15. In the console tree, click customdc.
16. Click the Action menu and click Set Description.
    The Security Template Description box appears.
17. In the Description box, type **Custom Security Template for Training** and click OK.

18. In the console tree, click the C:\WINNT\Security\Templates folder.

    Notice in the details pane that customdc now has a description associated with it.

19. Read the other template descriptions to familiarize yourself with the templates included with Windows 2000 Server.

3) Creating a new security database:

In this procedure you create a new security database.

1. In the console tree, click Security Configuration And Analysis and read the text in the details pane.
2. Click the Action menu and then click Open Database.

   The Open Database dialog box appears.
3. In the File Name text box, type **training** and then click Open.

   The Import Template dialog box appears.
4. Click customdc.inf and then click Open.

   This is the custom template you created in the previous procedure.

4) Analyzing current security settings:

In this procedure you analyze the current settings of Server01 against the custom template you created in Procedure 2.

1. In the console tree, verify that the Security Configuration And Analysis node is selected.
2. Click the Action menu and then click Analyze Computer Now.

   The Perform Analysis dialog box appears and shows the path and name of the error log as C:\Documents and Settings\Administrator\Local Settings\Temp\training.log.
3. Click OK.

   The Analyzing System Security status box appears as various aspect of Server01's security configurations are checked against the template.
4. When the analysis is complete, expand the Security Configuration And Analysis node.
5. Expand the Account Policies node and then click the Password Policy node.

   In the details pane, both template settings and the computer's settings are displayed for each policy. Discrepancies appear with a red circle with a white "X" in the center. Consistencies appear with a white circle and a green check mark in the center. If there is no flag or check mark, the security setting is not specified in the template.

6. In the console tree, click the Security Configuration And Analysis node.
7. Click the Action menu and then click Configure Computer Now.
   The Configure System dialog box appears.
8. Click OK.
9. Click the Action menu and then click Analyze Computer Now.
   The Perform Analysis dialog box appears.
10. Click OK.
11. Review the policy settings to verify that the Database Settings column is equivalent to the Computer Setting column.
12. Close the Security snap-in.
    The Microsoft Management Console message box appears.
13. Click Yes.
14. If a Save Security Templates window appears, click Yes.

4.4 Group Policy Snap-In

Security settings define the security-relevant behavior of the system. Through the use of GPOs in Active Directory services, administrators can centrally apply the security levels required to protect enterprise systems.

When determining settings for a GPO that contains multiple computers, the organizational and functional character of that given site, domain, or organizational unit (OU) must be considered. For example, the security levels necessary for an OU containing computers in a sales department would be very different from that for an OU containing finance department computers.

The Group Policy snap-in allows you to configure security centrally in the Active Directory store. A Security Settings folder is located on the Computer Configuration node and the User Configuration node. The security settings allow group policy administrators to set policies that can restrict user access to files and folders, set how many incorrect passwords a user can enter before the user is locked out, and control user rights, such as which users are able to log on at a domain server.

For details about how to use the Group Policy snap-in and how to administer group policies.

4.5 Chapter Summary

Windows 2000 provides a set of security configuration tools that allow you to configure Windows 2000 security settings and perform periodic analyses of the system to ensure that the configuration remains intact or to make necessary changes over time. The Security Configuration And Analysis snap-in allows you to configure and analyze local system security. It reviews and analyzes your system security settings and recommends modifications to the current system settings. The Security Templates snap-in allows you to create and assign security templates for one or more computers. The Group Policy snap-in allows you to configure security centrally in the Active Directory store.

CHAPTER 5

MICROSOFT WINDOWS 2000 AUDITING

5.1 OVER VIEW

Auditing in Microsoft Windows 2000 is the process of tracking both user activities and Windows 2000 activities, called events, on a computer. Through auditing, you can specify that Windows 2000 writes a record of an event to the security log. The security log maintains a record of valid and invalid logon attempts and events related to creating, opening, or deleting files or other objects. An audit entry in the security log contains the following information:

1) The action that was performed
2) The user who performed the action
3) The success or failure of the event and then the event occurred

5.1.1 Using an Audit Policy

An audit policy defines the types of security events that Windows 2000 records in the security log on each computer. The security log allows you to track the events that you specify.

Windows 2000 writes events to the security log on the computer where the event occurs. For example, you can configure auditing so that any time someone tries to log on to the domain by using a domain user account and the logon attempt fails, Windows 2000 writes an event to the security log on the domain controller. The event is recorded on the domain controller rather than on the computer at which the logon attempt was made, because it is the domain controller that attempted to and could not authenticate the logon attempt.

You can set up an audit policy for a computer to do the following:

1) Track the success and failure of events, such as logon attempts by users, an attempt by a particular user to read a specific file, changes to a user account or to group memberships, and changes to your security settings.
2) Eliminate or minimize the risk of unauthorized use of resources.

You can use Event Viewer to view events that Windows 2000 has recorded in the security log. You can also archive log files to track trends over time—for example, to

determine the use of printers or files or to verify attempts at unauthorized use of resources.

5.2 Planning an Audit Policy

When you plan an audit policy, you must determine the computers on which to set up auditing. Auditing is turned off by default. As you are determining which computers to audit, you must also plan what to audit on each computer. Windows 2000 records audited events on each computer separately.

The types of events that you can audit include the following:

1) Access to files and folders
2) Users logging on and off
3) Shutting down and restarting a computer running Windows 2000 Server
4) Changes to user accounts and groups
5) Attempts to make changes to Active Directory objects

After you have determined the types of events to audit, you must determine whether to audit the success and/or failure of events. Tracking successful events can tell you how often Windows 2000 users or services gain access to specific files, printers, or other objects. You can use this information for resource planning. Tracking failed events can alert you to possible security breaches. For example, if you notice a lot of failed logon attempts by a certain user account, especially if these attempts are occurring outside normal business hours, an unauthorized person might be attempting to break into your system.

Consider the following guidelines in determining your audit policy:

1) Determine if you need to track trends of system usage. If so, plan to archive event logs. Archiving these logs allows you to view how usage changes over time and allows you to plan to increase system resources before they become a problem.
2) Review security logs frequently. You should set a schedule and regularly review security logs because configuring auditing alone does not alert you to security breaches.
3) Define an audit policy that is useful and manageable. Always audit sensitive and confidential data. Audit only those events that will provide you with meaningful information about your network environment. This minimizes usage of server

resources and makes essential information easier to locate. Auditing too many types of events can create excess overhead for Windows 2000.

4) Audit resource access by the Everyone group instead of the Users group. This ensures that you audit anyone who can connect to the network, not just the users for whom you create user accounts in the domain.

5.3.1 Implementing an Audit Policy

Auditing is a powerful tool for tracking events that occur on computers in your organization. To implement auditing, you must consider auditing requirements and set the audit policy. After you set an audit policy on a computer, you can implement auditing on files, folders, printers, and Active Directory objects.

5.3.2 Configuring Auditing

You can implement an audit policy based on the role of the computer in the Windows 2000 network. Auditing is configured differently for the following types of computers running Windows 2000:

1) For member or stand-alone servers or computers running Windows 2000 Professional, an audit policy is set for each individual computer. For example, to audit user access to a file on a member server, you set the audit policy on that computer.
2) For domain controllers, an audit policy is set for all domain controllers in the domain. To audit events that occur on domain controllers, such as changes to Active Directory objects, you configure a group policy for the domain, which applies to all domain controllers.

NOTE:

The types of events that you can audit on a domain controller are identical to those you can audit on a computer that is not a domain controller. The procedure is similar as well, but you use a group policy for the domain to control auditing for domain controllers.

1) Auditing Requirements

The requirements to set up and administer auditing are as follows:

1) You must have the Manage Auditing And Security Log permission for the computer where you want to configure an audit policy or review an audit log. Windows 2000 grants these rights to the Administrators group by default.
2) The files and folders to be audited must be on NTFS volumes.

2) Setting Up Auditing

Setting up auditing is a two-part process:

1) **Setting the audit policy** The audit policy enables auditing of objects but does not activate auditing of specific objects.
2) **Enabling auditing of specific resources** You identify the specific events to audit for files, folders, printers, and Active Directory objects. Windows 2000 then tracks and logs the specified events.

5.3.3 Setting an Audit Policy

The first step in implementing an audit policy is selecting the types of events that Windows 2000 audits. For each event that you can audit, the configuration settings indicate whether to track successful or failed attempts. You can set audit policies by using the Group Policy snap-in.

The following table describes the types of events that Windows 2000 can audit.

Event	Description
Account logon events	A domain controller received a request to validate a user account.
Account management	An administrator created, changed, or deleted a user account or group. A user account was renamed, disabled, or enabled, or a password was set or changed.
Directory service access	A user gained access to an Active Directory object. You must configure specific Active Directory objects for auditing to log this type of event.
Logon events	A user logged on or logged off, or a user made or canceled a network connection to the computer.
Object access	A user gained access to a file, folder, or printer. You must configure specific files, folders, or printers for auditing. Directory service access

	is auditing a user's access to specific Active Directory objects. Object access is auditing a user's access to files, folders, and printers.
Policy change	A change was made to the user security options, user rights, or audit policies
Privilege use	A user exercised a right, such as changing the system time. (This does not include rights that are related to logging on and logging off.)
Process tracking	A program performed an action. This information is generally useful only for programmers who want to track details of program execution.
System	A user restarted or shut down the computer, or an event occurred that affects Windows 2000 security or the security log. (For example, the audit log is full and Windows 2000 discards entries.)

To set an audit policy on a computer that is not a domain controller, create a custom MMC console and add the Group Policy snap-in. In the console tree, select Audit Policy from the Computer Configuration node, as shown in Figure 5.1 The console displays the current audit policy settings in the details pane.

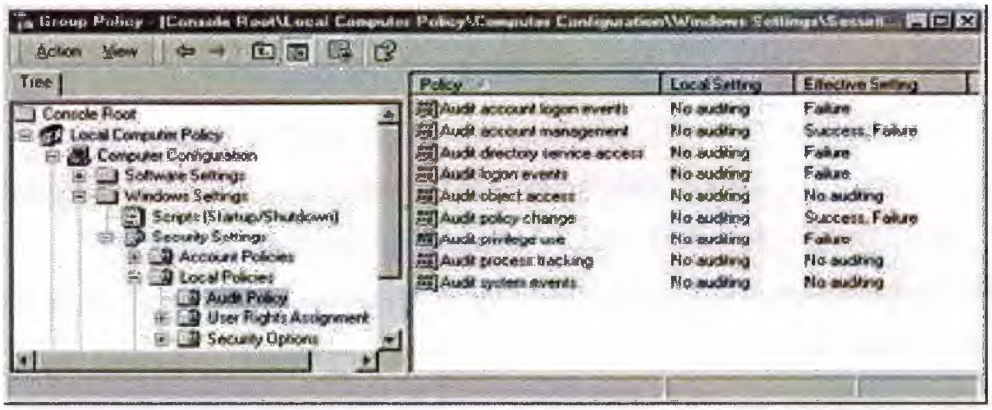


Figure 5.1 *Group Policy snap-in with the Audit Policy folder selected*

Changes that you make to your computer's audit policy take effect when one of the following events occurs:

1) You initiate policy propagation by typing **secedit /RefreshPolicy machine_policy** at the command prompt and then pressing Enter.
2) You restart your computer. Windows 2000 applies changes that you made to your audit policy the next time that you restart your computer.
3) Policy propagation occurs. Policy propagation is a process that applies policy settings, including audit policy settings, to your computer. Automatic policy

propagation occurs at regular, configurable intervals. By default, policy propagation occurs every eight hours.

5.3.4 Auditing Access to Files and Folders

If security breaches are an issue for your organization, you can set up auditing for files and folders on NTFS partitions. To audit user access to files and folders, you must first enable the Audit object access policy, which includes files and folders.

Once you have set your audit policy to audit object access, you enable auditing for specific files and folders and specify which types of access, by which users or groups, to audit. To enable auditing for a specific file or folder, open the Properties dialog box for that file or folder, select the Security tab, and then click Advanced. Select the Auditing tab and configure auditing for the selected file or folder.

5.3.5 Auditing Access to Active Directory Objects

To audit Active Directory object access, you must configure an audit policy and then set auditing for specific objects, such as users, computers, organizational units (OUs), or groups by specifying which types of access and access by which users to audit.

To enable auditing of access to Active Directory objects, enable the Audit Directory Service Access policy in the Group Policy snap-in.

To enable auditing for specific Active Directory objects, open the Active Directory Users And Computers snap-in and select Advanced Features from the View menu. Open the Properties dialog box for the object that you want to audit. On the Security tab, click Advanced. Select the Auditing tab and configure auditing for that object.

5.3.6 Auditing Access to Printers

You can audit access to printers in order to track access to sensitive printers. To audit access to printers, enable the Audit Object Access policy, which includes printers. Then enable auditing for specific printers, and specify which types of access and access by which users to audit. After you select the printer, you use the same steps that you use to set up auditing on files and folders.

To set up auditing on a printer, open the Properties dialog box for the printer that you want to audit. On the Security tab, click Advanced. Select the Auditing tab and configure auditing for the printer.

5.4.1 Using Event Viewer

You can use Event Viewer to perform a variety of tasks, including viewing the audit logs that are generated as a result of setting audit policies and auditing events. You can also use Event Viewer to view the contents of security log files and find specific events within log files.

5.4.2 Windows 2000 Logs

You can use Event Viewer to view information contained in Windows 2000 logs. By default there are three logs available to view in Event Viewer. These logs are described in the following table.

Log	Description
Application log	Contains errors, warnings, or information that programs, such as a database program or an e-mail program, generate. The program developer presets which events to record.
Security log	Contains information about the success or failure of audited events. The events that Windows 2000 records are a result of your audit policy.
System log	Contains errors, warnings, and information that Windows 2000 generates. Windows 2000 presets which events to record.

NOTE:

If additional services are installed, they might add their own event log. For example, the Domain Name System (DNS) service logs DNS events in the DNS Server log.

5.4.3 Viewing the Security Log

The Security log contains information about events that are monitored by an audit policy, such as failed and successful logon attempts. You can view the Security log in the Event Viewer snap-in, as shown in Figure 5.2.

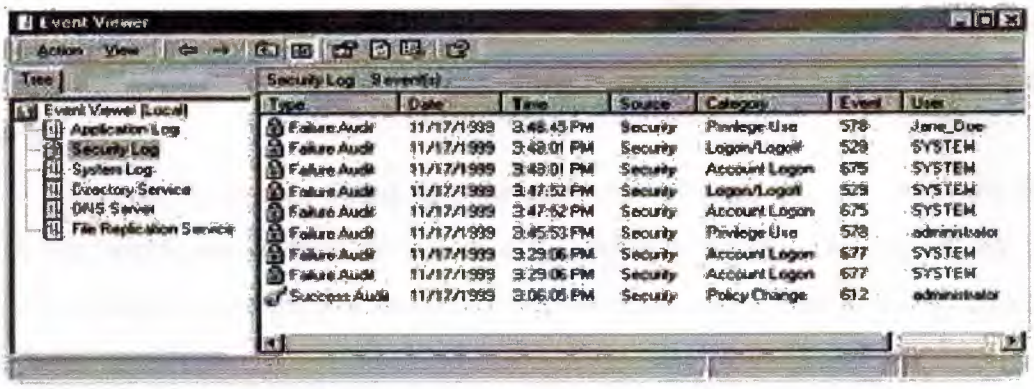


Figure 5.2 *Event Viewer snap-in with the Security Log selected*

In the details pane, Event Viewer displays a list of log entries and summary information for each item.

Successful events appear with a key icon, and unsuccessful events appear with a lock icon. Other important information includes the date and time that the event occurred, the category of the event, and the user who generated the event. The category indicates the type of event, such as object access, account management, directory service access, or logon events.

Windows 2000 records events in the Security log on the computer at which the event occurred. You can view these events from any computer as long as you have administrative privileges for the computer where the events occurred. To view the security log on a remote computer, point Event Viewer to a remote computer when you add this snap-in to a console.

5.4.4 Locating Events

When you first start Event Viewer, it automatically displays all events that are recorded in the selected log. To change what appears in the log, you can locate selected events by using the Filter command. You can also search for specific events by using the Find command. To filter or find events, start Event Viewer, and click Filter or click Find on the View menu.

5.4.5 Managing Audit Logs

You can track trends in Windows 2000 by archiving event logs and comparing logs from different periods. Viewing trends helps you determine resource use and plan for growth. If unauthorized use of resources is a concern, you can also use logs to determine patterns of usage. Windows 2000 allows you to control the size of the logs and to specify the action that Windows 2000 takes when a log becomes full.

You can configure the properties of each individual audit log. To configure the settings for logs, select the log in Event Viewer, and then display the Properties dialog box for the log.

Use the Properties dialog box for each type of audit log to control the size of each log, which can be from 64 KB to 4,194,240 KB (4 GB). The default size is 512 KB. You can also use the log properties to control the action that Windows 2000 takes when the log fills up.

TIP:

Use the Security Configuration And Analysis snap-in to configure settings for Event Viewer.

5.4.6 Archiving Logs

Archiving security logs allows you to maintain a history of security-related events. Many companies have policies on keeping archive logs for a specified period to track security-related information over time. If you want to save the log file, clear all events, or open a log file, select the log from the Event Viewer console tree and then select the appropriate option from the Action menu.

5.5 Chapter Summary

Auditing in Microsoft Windows 2000 is the process of tracking both user activities and Windows 2000 activities, called events, on a computer. Through auditing, you can specify that Windows 2000 writes a record of an event to the Security log. An audit policy defines the types of security events that Windows 2000 records in the Security log on each computer. The Security log allows you to track the events that you specify. When you plan an audit policy you must determine the computers on which to set up auditing. As you are determining which computers to audit, you must also plan what to audit on each computer. To implement auditing, you need to consider auditing requirements and set the audit policy. After you set an audit policy on a computer, you can implement auditing on files, folders, printers, and Active Directory objects. You can use Event Viewer to view the audit logs that are generated as a result of setting the audit policy and auditing events. You can also use Event Viewer to view the contents of Security log files and find specific events within log files.

CONCLUSION

This project which was about computer network security, in every part of this world we need communications, which should be strong, fast, easy to use and secure to be trusted, as a sharing folders Sharing folders are the only way to make folders and their contents available over the network.

Shared folders provide a way to secure file resources; they can be used on FAT16 and FAT32 partitions, as well as on NTFS partitions. But NTFS supports more than just shared folders. NTFS permissions can be used to specify which users and groups can gain access to files and folders and what they can do with their content. Applying shared permissions to user accounts and groups affects access to a shared folder, or a public key cryptography which is a critical technology for e-commerce, intranets, extranets, and other Web-enabled applications, computer security includes everything from the physical computing environment to the software environment.

In a software environment, security should provide four functions, which they are authentication, integrity, confidentiality, and anti-replay.

Computer network security in every time in ever day developed, such as the first network was LAN (local area network) which is for a short distance, then the network have been developed to make WAN (wide area network) for a far distance, or in the security level, such as, in the LAN or in the sharing folders the security level was just to give each user of the network different username and password, but then it is developed to make a server which will distinguish between the user, so the server can give a permission to some user but not to others whose they are not allow to access this folders, then security have been developed to far distance as the internet which is more complex and more strong and fast.

REFERENCES

[1] Jeff madden, *IT Professional*, Microsoft Company, 2000.

[2] Lynn Finnel, *MCSE Training Kit--Microsoft Windows 2000 Server / Microsoft Corporation*, Published By: Microsoft Company, 2000.

[3] Microsoft Press, *A Division of Microsoft Corporation*, Redmond, Washington 98052-6399 Distributed in Canada by Penguin Books Canada Limited, 2000.

[4] Microsoft Company, *One Microsoft Way*, Microsoft Press, Redmond, Washington, Distributed in Canada by Penguin Books Canada Limited, 2000.

[5] Microsoft 2000, *Library of Congress Cataloging-in-Publication Data*, Microsoft Company, 2000.

[6] Security research center, 2000, from the World Wide Web "http//www. mspress.microsoft.com".

[7] Encyclopedia Search, 2000, from the World Wide Web "http//www.britanica.com".

[8] Maintenance and security, 1999, from the World Wide Web "http//www.networksecurity.gov".