

Microsoft® Windows® 2000 Guide to Unattended Setup

Answer File Parameters for Unattended Installation of the Windows 2000 Family of Operating Systems

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

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

Unpublished work. © 2000 Microsoft Corporation. All rights reserved.

Microsoft, MS-DOS, Windows, Windows NT, Active Desktop, Active Directory, and FrontPage are either registered trademarks or trademarks of Microsoft Corporation in the U.S.A. and/or other countries.

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

Microsoft Corporation
April 2000

Contents

Introduction	
Answer File Format	
Description of Answer File Parameters	
Chapter 1 Unattend.txt File Parameters	
[Unattended]	
ComputerType	
DriverSigningPolicy	
ExtendOemPartition	
FileSystem	
KeyboardLayout	
NtUpgrade	
OemFilesPath	
OemPnPDriversPath	
OemPreinstall	
OemSkipEula	
OverwriteOemFilesOnUpgrade	
Repartition	
TargetPath	
UnattendMode	
Win9xUpgrade	
[MassStorageDrivers]	
<mass storage driver description>	
[OEMBootFiles]	
<HAL file name>	
<SCSI driver file name>	
Txtsetup.oem	
[OEM_Ads]	
Background	
Logo	
[GuiUnattended]	
AdminPassword	
AdvServerType	
Arguments	
AutoLogon	
AutoLogonAccountCreation	
AutoLogonCount	

DetachedProgram
OEMSkipRegional
OEMSkipWelcome
ProfilesDir
TimeZone
[UserData]
ComputerName
FullName
OrgName
ProductID
[Proxy]
FTP_Proxy_Server
Gopher_Proxy_Server
HTTP_Proxy_Server
Proxy_Enable
Proxy_Override
Secure_Proxy_Server
Socks_Proxy_Server
Use_Same_Proxy
[URL]
AutoConfig
AutoConfigJSURL
AutoConfigURL
Help_Page
Home_Page
Quicklink
[FavoritesEx]
[Branding]
BrandIEUsingUnattended
IEBrandingFile
[LicenseFilePrintData]
AutoMode
AutoUsers
[GuiRunOnce]
[Display]
BitsPerPel
Vrefresh
Xresolution
Yresolution
[RegionalSettings]

InputLocale	
Language	
LanguageGroup	
SystemLocale	
UserLocale	
[TapiLocation]	
AreaCode	
CountryCode	
Dialing	
LongDistanceAccess	
[Fax]	
ArchiveFolderName	
ArchiveOutgoing	
Csid	
FaxNumber	
FaxPrinterName	
Rings	
RouteFolderName	
RoutePrinterName	
RouteToFolder	
RouteToPrinter	
Tsid	
[Win9xUpg]	
Boot16	
DefaultPassword	
DomainJoinText	
ForcePasswordChange	
ForceWorkgroup	
IgnoreNetworkErrors	
KeepBadLinks	
MigrateDefaultUser	
MigrateUsersAsAdmin	
MigrateUsersAsPowerUser	
MigrationDlls	
ReportOnly	
SafeMode	
SaveReportTo	
ScanDrives	
UseLocalAccountOnError	
UserDomain	

UserPassword	
[SystemFileProtection]	
SFCDllCacheDir	
SFCShowProgress	
SFCQuota	
[Components]	
accessopt	
calc	
cdplayer	
certsrv	
certsrv_client	
certsrv_server	
charmap	
chat	
cluster	
deskpaper	
dialer	
fp_extensions	
freecell	
hypertrm	
iis_common	
iisdbg	
iis_doc	
iis_ftp	
iis_htmla	
iis_inetmgr	
iis_nntp	
iis_nntp_docs	
iis_pwmgr	
iis_smtp	
iis_smtp_docs	
iis_www	
indexsrv_system	
LicenseServer	
media_clips	
media_utopia	
minesweeper	
mousepoint	
mplay	
msmq	

mwordpad
netcis
netoc
objectpkg
paint
pinball
rec
reinst
rstorage
solitaire
templates
TSClients
TSEnable
vol
WMS
WMS_Admin
WMS_SERVER
[Networking]
[Identification]
DomainAdmin
DomainAdminPassword
JoinDomain
JoinWorkgroup
MachineObjectOU
[NetAdapters]
<adapter instance>
[<params.adapter instance>]
ConnectionName
DMA
DoNotDetectLegacyCards
InfID
IOAddr
IRQ
MEM
NetCardAddress
PCIBusNumber
PCIDeviceNumber
PCIFunctionNumber
[NetProtocols]
<protocol name>

[<protocol parameters section>]
AdapterSections
[<adapter specific protocol section>]
SpecificTo
[<MS_TCPIP parameters>]
DeadGWDetectDefault
DNSDomain
DNSSuffixSearchOrder
DontAddDefaultGatewayDefault
EnableICMPRedirect
EnableLMHosts
EnableSecurity
ScopeID
SyncDomainWithMembership
UseDomainNameDevolution
DefaultGateway
DHCP
DHCPClassId
DNSDomain
DNSServerSearchOrder
IPAddress
NetBIOSOptions
SubnetMask
WINS
WINSServerList
[<MS_NWIPX parameters>]
DedicatedRouter
EnableWANRouter
NetworkNumber
PktType
VirtualNetworkNumber
[<MS_PPTP parameters>]
NumberLineDevices
[<MS_L2TP parameters>]
WanEndpoints
[<MS_DLC parameters>]
[<MS_AppleTalk parameters>]
DefaultPort
DefaultZone
DesiredZone

EnableRouter	
NetworkRangeLowerEnd	
NetworkRangeUpperEnd	
SeedingNetwork	
ZoneList	
[<MS_NetBEUI parameters>]	
[<MS_NetMon parameters>]	
[<MS_ATMLANE parameters>]	
[<MS_ATMUni parameters>]	
[<MS_ATMArps parameters>]	
[<MS_STREAMS parameters>]	
[NetClients]	
<network client name>	
[<MS_MSClient parameters>]	
BrowseDomains	
NameServiceNetworkAddress	
NameServiceProtocol	
[<MS_NWClient parameters>]	
DefaultTree	
DefaultContext	
LogonScript	
PreferredServer	
[NetServices]	
<network service name>	
[<MS_Server parameters>]	
BroadcastsToLanman2Clients	
Optimization	
[<MS_RasSrv parameters>]	
ParamsSection	
[<MS_RAS parameters>]	
AssignSameNetworkNumber	
AutomaticNetworkNumbers	
ClientCanRequestIPAddress	
ClientsCanRequestIpxNodeNumber	
DialinProtocols	
IpAddressEnd	
IpAddressStart	
IPXClientAccess	
Multilink	
NetBEUIClientAccess	

NetworkNumberFrom	
RouterType	
TcpIpClientAccess	
UseDHCP	
[<MS_Psched parameters>]	
[<MS_NwSapAgent parameters>]	
[<MS_WLBS parameters>]	
AliveMsgPeriod	
AliveMsgTolerance	
ClusterIPAddress	
ClusterModeOnStart	
ClusterName	
ClusterNetworkMask	
DedicatedIPAddress	
DedicatedNetworkMask	
DescriptorsPerAlloc	
HostPriority	
MaskSourceMAC	
MaxDescriptorAllocs	
MulticastSupportEnable	
NetmonAliveMsgs	
NumActions	
NumAliveMsgs	
NumPackets	
Ports	
RemoteControlEnabled	
RemoteControlPassword	
RemoteControlUDPPort	
[NetBindings]	
Disable	
Enable	
Demote	
[NetOptionalComponents]	
ACS	
DHCPServer	
DNS	
IAS	
ILS	
LPDSVC	
MacPrint	

MacSrv
Netcm
NETMONTTOOLS
SimpTcp
SNMP
WINS
[SNMP]
 Accept_CommunityName
 Any_Host
 Community_Name
 Contact_Name
 Limit_Host
 Location
 Send_Authentication
 Service
 Traps
[InternetServer]
 PathFTPRoot
 PathWWWRoot
[Cluster]
 Account
 Action
 Domain
 ExcludeDrive
 IPAddr
 LocalQuorum
 Name
 Network
 Password
 Quorum
 Subnet
[TerminalServices]
 ApplicationServer
 PermissionsSetting
[Certsrv_Client]
 CAMachine
 CAName
[CertSrv_Server]
 CAType
 Country

CSPPProvider	
Description	
Email	
ExistingKey	
HashAlgorithm	
KeyLength	
Locality	
Name	
Organization	
OrganizationUnit	
ParentCAMachine	
ParentCAName	
PreserveDB	
RequestFile	
SharedFolder	
State	
UseExistingCert	
ValidityPeriod	
ValidityPeriodUnits	
[RemoteInstall]	
Repartition	
UseWholeDisk	
[OsChooser]	
Description	
Help	
ImageType	
LaunchFile	
Version	
[DCInstall]	
AdministratorPassword	
AllowAnonymousAccess	
AutoConfigDNS	
ChildName	
CreateOrJoin	
CriticalReplicationOnly	
DatabasePath	
DNSOnNetwork	
DomainNetBiosName	
IsLastDCInDomain	
LogPath	

NewDomainDNSName
ParentDomainDNSName
Password
RebootOnSuccess
ReplicaDomainDNSName
ReplicaOrMember
ReplicaOrNewDomain
ReplicationSourceDC
SafeModeAdminPassword
SiteName
SysVolPath
TreeOrChild
UserDomain
UserName
[Data]
AutoPartition
MsDosInitiated
UnattendedInstall
UseBIOSToBoot

Chapter 2 Sample Unattend.txt Files

 Sample 1

 Sample 2

 Sample 3

Chapter 3 Sysprep.inf File Parameters

 [Unattended]

 ExtendOemPartition

 InstallFilePath

 KeepPageFile

 OemPnPDriversPath

 OemSkipEula

 UpdateHAL

 UpdateUPHAL

 [Oem_Ads]

 [GuiUnattended]

 AdminPassword

 AutoLogon

 AutoLogonAccountCreation

 AutoLogonCount

 OEMDuplicatorString

 OEMSkipRegional

 OEMSkipWelcome

 TimeZone

 [UserData]

 [LicenseFilePrintData]

 [GuiRunOnce]

 [Display]

 [RegionalSettings]

 [TapiLocation]

 [Networking]

 [Identification]

Chapter 4 Sample Sysprep.inf Files

 Sample 1

 Sample 2

Introduction

Unattended Setup mode is a hands-free method of installing Microsoft® Windows® 2000 that is convenient for Original Equipment Manufacturers (OEMs), Administrators in corporations, Value Added Resellers (VARs), and other users.

Note For a complete listing of Winnt.exe and Winnt32.exe commands, see Windows 2000 Help. Or, at a command prompt, switch to the \i386 folder on the Windows 2000 product CD, and then type: **winnt /?** or **winnt32 /?**

- To start Windows 2000 Setup in unattended Setup mode from MS-DOS® or Windows 3.1/Windows for Workgroups, a user must specify:

```
winnt /u:<answer file> /s:<install source> /t:<target drive>
```

- To start Windows 2000 Setup in unattended Setup mode from Windows 95 or Windows 98, Windows NT®, or Windows 2000, a user must specify:

```
winnt32 /unattend:<answer file> /s:<install source> [/syspart:<target drive>] [/tempdrive:<target drive>]
```

where:

- *<answer file>* is a file that contains answers to questions that should be automated during installation.
- *<install source>* is the location of the Windows 2000 installation files.
- **/syspart** copies all boot files to the target drive and marks it as active. This option is only valid for Winnt32.exe.
- **/tempdrive** copies all the installation files to a temporary folder on the target drive. This option implies that Windows 2000 must be installed in the specified drive and is only valid for Winnt32.exe.

If the temporary folder is full, the user is prompted to choose another directory during setup. After this is done, unattended setup continues.

Note The **/tempdrive** parameter, like the **/t** switch for the Winnt.exe command, is useful to ensure the correct drive is chosen without user intervention when the hard disk has multiple partitions or hard disks.

- **/t** (Optional) copies all the installation files to a temporary directory on the target drive. This option assumes that Windows 2000 is installed in the specified drive and is only valid for Winnt.exe.

If the temporary directory is full, the user is prompted to choose another directory during setup. After this is done, unattended setup continues.

Note Use the **/t** or **/tempdrive** switches to install Windows 2000 to a partition other than the boot partition on a computer.

A combination of **/syspart** and **/tempdrive** is useful if the target drive is used as the primary drive on another computer. The **/syspart** and **/tempdrive** parameters are optional. By using them together, you can create a custom image that can be used across non-identical computers. GUI-mode Setup then completes the installation on those computers.

Important If you direct **/syspart** to a non-boot drive on your computer, remove that drive before restarting your computer. Otherwise, your computer cannot boot.

To start Windows 2000 Setup in unattended Setup mode from the Windows 2000 product CD

1. The computer must support booting from the CD-ROM drive—El Torito No Emulation CD boot support.
2. The answer file must be named Winnt.sif and must be placed on a floppy disk to be inserted as soon as the computer boots from the CD.

The Winnt.sif answer file may also be placed on and run from a bootable CD. For more information about creating a bootable CD, see KB article Q167685, “How to Create an El Torito Bootable CD-ROM” at the Microsoft Product Support Services Web site:

<http://support.microsoft.com/support/>

3. The answer file must contain a [Data] section with the required keys specified. For more information, see “[Data],” later in this document.

Note When using the Remote Installation Service (RIS) to install Windows 2000 on a computer that has a bootable network card, specify the network card as the first bootable device and demote all others. You then provide the maximum possible number of methods for successfully deploying Windows 2000 to computers in your environment. For more information about using RIS, see “[RemoteInstall],” later in this document.

One method to simplify this process is to modify the BIOS to specify the CD-ROM as the first bootable device, the floppy drive as the second, and the hard drive as the third.

Answer File Format

In general, an answer file, such as Unattend.txt or Sysprep.inf, consists of section headers/keys, parameters, and the values for each parameter. Most of the section headers are pre-defined, but some may be user-defined. You don't need to specify all of the possible parameters in the Unattend.txt file if the installation does not require them.

The file format is as follows:

```
[section1]
;
; Section contains keys and the corresponding
; values for those keys/parameters.
; keys and values are separated by '=' signs
; Values that have spaces in them usually require double quotes
; "" around them
;
key = value
.
.
[section2]
key = value
```

Invalid parameter values generate errors or may cause incorrect behavior after setup.

Description of Answer File Parameters

Bold	for a section or a key name. The exact name specified must be used.
<i><user-def></i>	for a key or a section name when specified and enclosed in brackets (< >) and italicized. The name used can be specified by the creator of the answer file.
Key values	are string type, unless otherwise mentioned. Wherever Type = Numeric is specified, the value is written as a decimal number unless expressly mentioned otherwise.
Optional	indicates that the key is optional. Unless indicated, all keys are required for the given section.
Default	indicates the default value assumed if a key is not present.

CHAPTER 1

Unattend.txt File Parameters

This section lists the various keys, and their parameters, which can be included in the Unattend.txt file. Many keys/parameters are interdependent; please note the interdependencies where specified.

A fully unattended Setup would require, at a minimum, the following keys and parameters:

```
[Unattended]
UnattendMode = FullUnattended
TargetPath = WINNT
```

```
[UserData]
ComputerName = "COMPUTER_NAME"
FullName = "Your User Name"
```

```
[GuiUnattended]
AdminPassword = *
TimeZone = "004"
```

```
; For Server installs
[LicenseFilePrintData]
AutoMode = "PerServer"
AutoUsers = "5"
```

```
[Networking]
```

```
[Identification]
JoinWorkgroup = Workgroup
```

Note To ensure a fully unattended Setup on hard disks with multiple partitions, include `/t` or `/tempdrive` on Winnt and Winnt32 command lines, respectively, to indicate the destination partition.

[Unattended]

This section header identifies that an unattended installation is being performed. This section is required in the Unattend.txt file; otherwise, the answer file is ignored. Parameters that can exist in this section are discussed below.

ComputerType

Values: <HAL description> [, **Retail** | **OEM**]

Specifies the type of custom Hardware Abstraction Layer (HAL) to be loaded by the Setup Loader and installed by text-mode Setup. This parameter is only valid when OemPreinstall = Yes. If this key is not present, Setup attempts to detect the type of computer and install the appropriate retail HAL.

Note This parameter is only used if your vendor has supplied a custom HAL.

The <HAL description> string identifies the HAL to be installed. It must match one of the strings in the [Computer] section of Txtsetup.sif (for a retail HAL), or Txtsetup.oem (for an OEM HAL).

where:

- **Retail** informs Setup that the HAL to be installed is part of Windows 2000.
- **OEM** indicates that the HAL to be loaded is OEM-supplied. If this is the case, the driver name must also be listed in the [OEMBootFiles] section of the Unattend.txt file.

DriverSigningPolicy

Value: **Ignore** | **Warn** | **Block**

Default: **Warn**

Specifies how non-signed drivers are processed during unattended Setup. Signed drivers have gone through the Microsoft driver testing and signing process to ensure they are Windows 2000 compatible.

For the most up-to-date list of supported hardware, see the Hardware Compatibility List by visiting the Microsoft Web site at:

<http://www.microsoft.com/>

Your Windows 2000 CD includes a copy of this list (*drive:\Support\Hcl.txt*) that was accurate as of the date Windows 2000 was released.

where:

- **Block** Setup does not install the non-signed device driver.
- **Warn** Setup stops the installation and prompts the user for input before accepting the non-signed device driver.
- **Ignore** Setup continues even if the driver is not signed.

Important Microsoft strongly advises against using `DriverSigningPolicy = Ignore` unless you have fully tested the device driver in your environment and are sure that it works properly. Using non-signed drivers increases the risk of device driver problems that can affect the performance or stability of your computer.

If you are using `DriverSigningPolicy = Ignore` and you attempt to install a newer, unsigned copy of a driver that ships with Windows 2000, Setup will install the signed Windows 2000 driver instead of the unsigned drivers in accordance with the ranking process used by the operating system.

For more information about driver ranking and how Setup chooses the correct driver, see the Microsoft Driver Development Kits Web site at:

<http://www.microsoft.com/ddk/>

ExtendOemPartition

Value: `0 | 1 | <extra size in MB>`

The `ExtendOemPartition` key is used to extend the partition on which you are installing Windows 2000. This key causes Setup to extend this destination partition into any available unpartitioned space that physically follows it on the disk.

where:

- `0` Setup does not extend the partition.
 - `1` Setup extends the partition to fill out the hard disk.
 - `<extra size in MB>` Setup increases the current partition size by this amount.
-

Important

- Only NTFS partitions can be extended. If the destination partition you plan to extend is FAT or FAT32, set `FileSystem = ConvertNTFS` to convert the partition but *only* during text-mode Setup.
 - `ExtendOemPartition` can be used with both the `Unattend.txt` and `Sysprep.inf` Setup files.
 - When used in `Sysprep.inf` for imaged computers, the destination computer's hard disk must be the same size or larger than the master computer's hard disk.
-

FileSystem

Value: **ConvertNTFS** | **LeaveAlone**

Specifies whether the primary partition should be converted to NTFS or left alone.

Note If you plan to use `ExtendOemPartition` during Setup, set `FileSystem = ConvertNTFS`. For more information, see “ExtendOemPartition,” earlier in this document.

KeyboardLayout

Value: *<layout description>*

Specifies the type of keyboard layout to be installed during text-mode Setup. If this key does not exist, Setup detects and installs a keyboard layout.

This key must match one of the right-hand strings (in quotes) in the `KeyboardLayout` section of `Txtsetup.sif`.

Note This parameter is provided for backward compatibility only and will be removed in future releases.

NtUpgrade

Values: **Yes** | **No**

Determines whether a previous version of Windows NT 3.51, Windows NT 4.0, Windows 2000 Professional, Windows 2000 Server, Windows 2000 Advanced Server, or Windows 2000 Datacenter Server should be upgraded. This parameter is only valid for `Winnt32.exe`.

Set `NtUpgrade = Yes` to upgrade the previous Windows NT or Windows 2000 installation. If `OemPreinstall = Yes`, `NtUpgrade` should *not* be set to `Yes`.

Note

- This key upgrades your previous version of Windows 2000, Windows NT 3.51, or Windows NT 4.0. All user settings are taken from the previous installation, so no user intervention is required during Setup.
 - To upgrade from Windows 95 or Windows 98, use the `Win9xUpgrade` key.
-

OemFilePath

Value: *<path to \OEM\$ folder>*

Specifies the path to the `\OEM$` folder (containing OEM files) if it does not exist under the `i386` folder of the distribution share point. The path can be a UNC name.

For more information about the `\OEM$` folder, see the *Microsoft Windows 2000 OEM Preinstallation Kit (OPK) User Guide* if you are a computer manufacturer. Otherwise, see the *Microsoft Windows 2000 Deployment Guide*.

OemPnPDriversPath

Value: "<folder 1 on system drive>;<folder 2 on system drive>; ..."

Specifies the path to folders that contain Plug and Play (PnP) drivers that do not ship on the Windows 2000 CD. The folders must contain all the files necessary to install the particular devices—catalog files, .inf files, and drivers.

For example, if you have a folder called \Drivers with subfolders called \Audio and \Net, you would specify OemPnPDriversPath = "drivers\audio;drivers\net" in the answer file. Setup adds:

- %systemdrive% to each of the folder names
- the path for each subfolder to the PnP device search path

Note When using this parameter, be sure that the folders are available during GUI-mode Setup or Mini-Setup—you can use the \\${OEM}\\$1 folder structure mechanism for this. For best results, make sure your drivers are signed.

OemPreinstall

Values: **Yes** | **No**

Determines whether an installation from distribution folders is being performed. When the value is Yes, any other existing subfolders are copied. When set to No, a regular unattended installation is performed.

OemSkipEula

Values: **Yes** | **No**

Determines whether the user should be prompted to accept the Microsoft License Agreement (previously known as the End User License Agreement or EULA) included with Windows 2000. Setting this parameter to Yes implies that the person performing the installation has read and agreed to the contents of the License Agreement included with the product. It also implies that the end user on whose behalf Windows 2000 is being installed has agreed to the License Agreement.

Important OEMs must not use this key to bypass the Microsoft License Agreement screen, because end users are required to see and accept it.

OverwriteOemFilesOnUpgrade

Values: **Yes** | **No**

Determines whether OEM-supplied files that have the same name as Windows 2000 system files should be overwritten during an unattended upgrade.

Yes means overwrite the files. No means do not overwrite the files if found. The default behavior is to overwrite OEM-supplied files.

Note This key is provided for backward compatibility only and will be removed from future versions.

Repartition

Value: **Yes** | **No**

Specifies whether or not all partitions on the first drive on the client computer should be deleted and the drive reformatted with the NTFS file system.

Important Repartition is only valid when performing an unattended installation by booting the computer from the Windows 2000 CD.

TargetPath

Values: * | *<path name>*

Determines the installation folder in which Windows 2000 should be installed.

* indicates that Setup should generate a unique folder name for the installation. The folder name given is usually Winnt, unless that folder already exists. In that case, Setup installs into Winnt.x (where x is 0, 1, ... 999) if that folder does not already exist.

<path name> is the user-defined installation folder and should not include the drive letter. If you want to specify the target drive, you must use the **/tempdrive** parameter with Winnt32.exe or the **/t** switch with Winnt.exe.

UnattendMode

Values: **GuiAttended** | **ProvideDefault** | **DefaultHide** | **ReadOnly** | **FullUnattended**

UnattendMode defines the unattended Setup mode to be used during GUI-mode Setup. The default value is DefaultHide when the key is not specified. When this key is specified, text-mode Setup is fully automated with or without the necessary answers. where:

- **GuiAttended** specifies that the GUI-mode section of Setup is attended. When specified, the end user is required to answer all questions in the GUI-mode portion of Setup before Setup finishes. This mode is useful in preinstallation scenarios in which the OEM or administrator wants to automate only text-mode Setup.
- **ProvideDefault** specifies that answers in the answer file are defaults. In this case, Setup displays these default answers to the user, who may change them if desired. This approach is useful in preinstallation scenarios where the OEM or administrator wants to give the person setting up the computer the option to change the predefined default answers (especially network options).
- **DefaultHide** specifies that answers in the answer file are defaults. Unlike the ProvideDefault value, Setup does not display the user interface to end users if all the answers relating to a particular wizard page are specified in the answer file. If only subsets of the answers on a page are specified, the page is displayed with the provided answers. The user can modify any of the answers on the displayed page. This approach is useful in deployment scenarios where an administrator may only want end users to provide the administrator password on the computer. This behavior is the default if unattended Setup mode is not specified.
- **ReadOnly** specifies that answers in the answer file are read-only if the wizard pages containing these answers are displayed to the end user. Just like the DefaultHide parameter, no user interface appears if all answers on a page are

supplied in the answer file. Unlike the DefaultHide parameter, however, the user can only specify new answers on a displayed page. This approach is useful in scenarios where an administrator wants to force specific answers on one page but not others.

- **FullUnattended** specifies that GUI-mode Setup is fully unattended. If a required Setup answer is not specified in the answer file, an error is generated. This behavior is useful in deployment scenarios where a complete hands-off installation is required.

Win9xUpgrade

Values: **Yes** | **No**

Default: **No**

Instructs Setup to upgrade previous installations of Windows 95 or Windows 98 to Windows 2000.

where:

- **Yes** instructs Setup to upgrade the Windows installation, if found.
- **No** instructs Setup *not* to upgrade the Windows installation, if found.

This parameter is only necessary when using an answer file to upgrade an existing Windows 95 or Windows 98 computer to Windows 2000. This parameter is only valid for Winnt32.exe.

[MassStorageDrivers]

This section contains a list of SCSI drivers to be loaded by the Setup Loader and installed during text-mode Setup. If this section is missing or empty, Setup attempts to detect the SCSI devices on the computer and install the corresponding retail drivers.

<mass storage driver description>

Value: **Retail** | **OEM**

Identifies the driver to be installed. It must match one of the strings defined in the right-hand side of the [SCSI] section of Txtsetup.sif (for a retail driver) or Txtsetup.oem (for an OEM driver). Multiple instances of *<mass storage driver description>* may be specified.

where:

- **Retail** indicates that the driver is part of the retail Windows 2000 product.
- **OEM** indicates that the driver is OEM-supplied. If the value is OEM, the driver must also be listed in the [OEMBootFiles] section of the Unattend.txt file.

[OEMBootFiles]

Specifies OEM-supplied boot files. This parameter is only valid if OemPreinstall = Yes and the files listed here have been placed in the \%OEM%\Textmode folder of the OEM's distribution share point.

Note The parameters of this key are necessary if you are using ComputerType (under [Unattended]) and/or [MassStorageDrivers].

<HAL file name>

Maps to a HAL description that has been defined by the ComputerType key in the [Unattended] section of the Unattend.txt file.

<SCSI driver file name>

Maps to a mass storage device driver description defined in the [MassStorageDriver] section of the Unattend.txt file. There may be multiple instances of <SCSI driver file name> listed in the [OEMBootFiles] section.

Txtsetup.oem

This file contains descriptions of all the OEM-supplied drivers listed in this section and includes instructions on how to install them. This file must exist if this section is listed.

[OEM_Ads]

This section instructs Setup to modify the default end user background bitmap and add a logo at the top right corner of the screen.

Background

Values: <file name> [,<resource ID>]

Specifies a background bitmap to be displayed. If this line has only one field, then it refers to a .bmp file located in the distribution share point in the \SOEM\$ folder. However, if two fields are specified, then the first field is the name of a .dll and the second is a base-10 number that represents the resource identification (ID) of the bitmap in the .dll. The .dll specified should be located in the \SOEM\$ folder.

Logo

Values: <file name> [,<resource ID>]

Specifies a bitmap to be displayed in the upper-right corner of the screen. If this line has only one field, then it refers to a .bmp file located in the distribution share point in the \SOEM\$ folder. However, if two fields are specified, then the first field is the name of a .dll and the second is a base-10 number that represents the resource ID of the bitmap in the .dll. The .dll specified should be located in the \SOEM\$ folder.

[GuiUnattended]

AdminPassword

Value: <password> | *

Sets up the Administrator account password. If the value is *, Setup sets the administrator password to NULL.

Note If a password is specified in the Administrator account, you cannot use AdminPassword in the Sysprep.inf file to change it—the administrator password remains the same. However, if the administrator password was initially blank (either manually or through an unattended installation), you can use the AdminPassword key to change it to a non-blank password.

However, if a non-blank password is used, end users cannot change or specify their own passwords in the Mini-Setup wizard. A new password may be entered in the appropriate dialog box, but the password is not changed.

Also, security breaches may occur if you use a common, non-blank administrator password for all computers provided to end users. Prior to running Sysprep, you should use an automation process to set the administrator password to blank. End users can then specify their own passwords upon receipt of the computer.

AdvServerType

Value: **ServerNT**

This key and its value are valid only when upgrading one of the members of the Windows 2000 Server family.

Arguments

Value: *<string>*

Indicates that arguments or parameters accompany the custom program that runs concurrently with the Setup program.

Note This parameter is required if you are using DetachedProgram.

AutoLogon

Value: **Yes | No**

Sets up the computer to automatically log on once with the Administrator account if set to Yes. The default behavior is No. The key is not valid on upgrades.

Note

- If you specify a password by using AdminPassword, that password is used when you automatically log on. After the installation is complete, the password is deleted from the copy of the answer file left on the computer.
 - If AdminPassword = * (is blank) and AutoLogon = Yes, the computer logs on only one time. However, if AdminPassword = *<password>*, the computer logs on repeatedly and the non-blank password is saved in the registry.
-

AutoLogonAccountCreation

Value: **Yes | No**

Default: **Yes**

Specifies whether a computer account should be created automatically for the user whose name is specified by FullName. For more information about this parameter, see "FullName," later in this document. AutoLogonAccountCreation also allows the generated user account to automatically log on to the computer.

This parameter is only valid on computers that are configured with the Workgroup option. This will not work for computers that are configured as members of a domain.

Note

- The user's account is only configured to log on automatically by default when the option to join a workgroup is specified. This does not apply to computers that are members of a domain.
 - This parameter is not related to AutoLogon or AutoLogonCount; these parameters apply to the Administrator account only.
-

AutoLogonCount

Value: *<integer>*

Specifies the number of times that the computer automatically logs on using the Administrator account and password specified. The value decrements after each logon and the feature is disabled after the specified number of logon attempts.

Note For the count to be decremented, you must reboot the computer.

This parameter is only useful when AutoLogon = Yes and AdminPassword = *<password>* are specified in the answer file. If AdminPassword = * (is blank) and AutoLogon = Yes, the computer logs on only one time, regardless of the value given for AutoLogonCount.

DetachedProgram

Value: *<detached program string>*

Indicates the path of the custom program that should run concurrently with the Setup program. If the program requires any arguments, the Arguments key must be specified.

Note Make sure to include Arguments in [GuiUnattended] if you are using this parameter.

OEMSkipRegional

Values: 0 | 1

Allows unattended Setup to skip the Regional Settings page in GUI-mode Setup and Mini-Setup.

Note When specifying OemPreinstall = Yes and providing values for the [RegionalSettings] section, set OEMSkipRegional to 1 to ensure that Setup completes without prompting the user for regional information.

OEMSkipWelcome

Value: **1** | **0**

Allows unattended Setup to skip the Welcome page in GUI-mode Setup and Mini-Setup.

Note If OemPreinstall = Yes, unattended Setup automatically stops at the Welcome page. To avoid this pause, set OEMSkipWelcome to a value of 1. However, OEMs must not ship computers with this setting enabled.

ProfilesDir

Value: "<path to profile folder>"

Default: "%systemdrive%\Documents and Settings"

Specifies the location of Windows 2000 profiles. This parameter is only valid on clean installations of Windows 2000 and is ignored during upgrades.

Important The directory specified can contain an environment variable such as %systemdrive% or %systemroot%.

The ProfilesDir parameter is useful in scenarios that require new installations to use the same profile folder as Windows NT 4.0. For example:

```
ProfilesDir = "%systemroot%\Profiles"
```

TimeZone

Value: <index>

Specifies the time zone of the computer. If the key is not present, the user is prompted to select a time zone.

The following table lists the valid TimeZone indices:

Index	Zone	Index	Zone	Index	Zone
000	Int'l Dateline	090	GMT Greenwich	200	Sri Lanka
001	Samoa	095	Central Europe	201	N. Central Asia
002	Hawaii	100	Central European	203	Myanmar: Rangoon
003	Alaskan	105	Romance	205	S.E. Asia
004	Pacific	110	W. Europe	207	N. Asia
010	Mountain (U.S. and Canada)	113	W. Central Africa	210	China
015	U.S. Mountain: Arizona	115	E. Europe	215	Singapore
020	Central (U.S. and Canada)	120	Egypt	220	Taipei
025	Canada Central	125	EET (Helsinki, Riga, Tallinn)	225	W. Australia
030	Mexico	130	EET (Athens, Istanbul, Minsk)	227	N. Asia East
033	Central America	135	Israel: Jerusalem	230	Korea: Seoul
035	Eastern (U.S. and Canada)	140	S. Africa: Harare, Pretoria	235	Tokyo
040	U.S. Eastern: Indiana (East)	145	Russian	240	Sakha Yakutsk
045	S.A. Pacific	150	Arab	245	A.U.S. Central: Darwin
050	Atlantic (Canada)	155	E. Africa	250	Central Australia
055	S.A. Western	160	Iran	255	A.U.S. Eastern
056	Pacific S.A.	165	Arabian	260	E. Australia
060	Newfoundland	170	Caucasus Pacific (U.S. and Canada)	265	Tasmania
065	E. South America	175	Afghanistan	270	Vladivostok
070	S.A. Eastern	180	Russia Yekaterinburg	275	W. Pacific
073	Greenland	185	W. Asia	280	Central Pacific
075	Mid-Atlantic	190	India	285	Fiji
080	Azores	193	Nepal	290	New Zealand
083	Cape Verde Islands	195	Central Asia	300	Tonga
085	GMT (Greenwich Mean Time)				

[UserData]

ComputerName

Value: <string>

Specifies the computer name. If the ComputerName key is empty or missing, the user is prompted to enter a computer name. If the value is *, Setup generates a random computer name based on the organization name specified.

Note The computer name specified should contain no more than 63 characters. If more are specified, the name is truncated to 63 characters.

FullName

Value: <string>

Specifies the user's full name. If the key is empty or missing, the user is prompted to enter a name.

Note This parameter is used to ensure that the Setup mode is completely unattended.

OrgName

Value: <string>

Specifies an organization's name.

ProductID

Value: <string>

Specifies the Microsoft Product Identification (Product ID) number.

Important

- This parameter is not required for customers who are using Select CDs. Also, be aware that this parameter assigns the same Product ID to all computers; this may cause some difficulty when you try to receive support on multiple computers with the same Product ID.
 - If you are using the Windows 2000 Setup Manager wizard and select **Fully Automated** from the User Interaction screen, you *must* manually add the ProductID to the [UserData] section of the answer file. This is necessary to fully automate the installation of Windows 2000.
-

[Proxy]

This section contains proxy settings for the Windows 2000 browser. If these settings are not present, the default for Windows 2000 is used.

Important To make sure your [Proxy] settings take effect, you need to include the [Branding] section header and the BrandIEUsingUnattended key in the answer file to enable the browser-related settings:

```
[Branding]
BrandIEUsingUnattended = Yes
```

FTP_Proxy_Server

Value: <URL or IP address>

Specifies the Internet Protocol (IP) address or URL of the FTP proxy on the network.

For example:

```
FTP_Proxy_Server = http://proxyserver:80
```

Gopher_Proxy_Server

Value: <URL or IP address>

Specifies the IP address or URL of the Gopher proxy on the network. For example:

```
Gopher_Proxy_Server = http://proxyserver:80
```

HTTP_Proxy_Server

Value: <URL or IP address>

Specifies the IP address or URL of the HTTP proxy on the network. For example:

```
HTTP_Proxy_Server = http://proxyserver:80
```

Note This parameter is required if you are using Use_Same_Proxy.

Proxy_Enable

Value: 1 | 0

Specifies whether a proxy server is used to connect to the Internet. A value of 1 is Yes and 0 is No.

Proxy_Override

Value: <list of IP addresses>

Specifies a semicolon-separated list of IP addresses to be used to bypass the proxy. The list must be enclosed in quotes (for more than one address) and may contain the string <local> to override local addresses. For example:

```
Proxy_Override = <local>
```

Secure_Proxy_Server

Value: <URL or IP address>

Specifies the IP address or URL of the Secure proxy on the network. For example:

```
Secure_Proxy_Server = http://proxyserver:80
```


Socks_Proxy_Server

Value: <URL or IP address>

Specifies the IP address or URL of the Socks proxy on the network. For example:

```
Socks_Proxy_Server = http://proxyserver:80
```

Use_Same_Proxy

Value: 1 | 0

Specifies whether the same proxy server should be used for all protocols. A value of 1 is Yes and 0 is No. If this key is specified, the value of the HTTP proxy is used for all other transports.

[URL]

This section contains default URL settings for the Windows 2000 browser. If not present, the default for Windows 2000 is used.

Important To make sure your [URL] settings take effect, you need to include the [Branding] section header and the BrandIEUsingUnattended key in the answer file to enable the browser-related settings:

```
[Branding]  
BrandIEUsingUnattended = Yes
```

AutoConfig

Value: 1 | 0

Specifies whether the browser should be auto-configured from a server. A value of 1 is Yes and 0 is No.

AutoConfigJSURL

Value: <URL>

Specifies the URL of a JavaScript file that auto-configures the proxy settings for the browser. For example:

```
AutoConfigJSURL = http://configserver/autoconfig.js
```

AutoConfigURL

Value: <URL>

Specifies the URL of an .INS file that auto-configures the proxy settings for the browser. For example:

```
AutoConfigURL = http://configserver/autoconfig.ins
```

Help_Page

Value: <URL>

Specifies the URL for HTML-based Help. For example:
Help_Page = http://configserver

Home_Page

Value: <URL>
Specifies the URL for the browser's default home page. For example:
Home_Page = http://www.msn.com/

Quicklink

Value: <site name, URL>
Specifies shortcuts in the link folder of [FavoritesEx]. This key is a set of "Quick_Link_N_Name" and "Quick_Link_N," where N is a numerical designation for the site and URL. For example:
Quick_Link_1_Name = "MS HomePage"
Quick_Link_1 = http://www.microsoft.com/
Quick_Link_2_Name = "MS Japan HomePage"
Quick_Link_2 = http://www.microsoft.com/Japan

[FavoritesEx]

Value: Title<#> = "<Site Name>.url"
URL<#> = "<Site Address>"

This section contains default favorite settings for the Windows 2000 browser. If not present, the default for Windows 2000 is used. For example:

```
[FavoritesEx]
Title1 = "MSN.url"
URL1 = "http://www.microsoft.com/"
Title2 = "Cable News Network.url"
URL2 = "http://www.cnn.com"
Title3 = "MSNBC.url"
URL3 = "http://www.msnbc.com"
Title4 = "USA Today.url"
URL4 = "http://www.usatoday.com"
```

Important To make sure your [FavoritesEx] settings take effect, you need to include the [Branding] section header and the BrandIEUsingUnattended key in the answer file to enable the browser-related settings:

```
[Branding]
BrandIEUsingUnattended = Yes
```

[Branding]

This section contains parameters to "brand" Microsoft Internet Explorer during an unattended Setup. If not present, the default for Windows 2000 is used.

BrandIEUsingUnattended

Value: **Yes** | **No**

Specifies which file is to be used to brand Internet Explorer during an unattended Setup. If set to **Yes**, the Unattend.txt file is used to brand the browser settings using the parameters specified in the browser-specific sections of the answer file: [Proxy], [URL], and [FavoritesEx]. If set to **No**, the .ins file specified in IEBrandingFile is used.

Note This parameter is not required if you are not configuring the browser. If you are providing an .ins file that you created using the Internet Explorer Administration Kit (IEAK), add the following to your answer file:

```
[Branding]
BrandIEUsingUnattended = No
IEBrandingFile = <filename>
```

For more information, see "IEBrandingFile".

IEBrandingFile

Value: <file name>

Specifies the name of the .ins file (created by IEAK) to be used for branding Internet Explorer. (The full path to the file is not required.) If this key is specified:

- All other sections relating to Internet Explorer branding are ignored.
- Set BrandIEUsingUnattended = No.

Note The file specified must exist at the root of the %\$OEM\$ folder of the distribution share point.

[LicenseFilePrintData]

This section is only valid when installing one of the members of the Windows 2000 Server family.

AutoMode

Values: **PerSeat** | **PerServer**

Determines whether Windows 2000 Server, Windows 2000 Advanced Server, or Windows 2000 Datacenter Server is to be installed in a per-seat or a per-server license mode. If AutoMode = PerServer, the AutoUsers key must also be specified.

where:

- **PerSeat** indicates that a client access license has been purchased for each computer that accesses the server.
- **PerServer** indicates that client access licenses have been purchased for the server to allow a certain number of concurrent connections to the server.

If AutoMode is empty or missing, the user is prompted to select the license mode.

AutoUsers

Value: <integer>

This key is only valid if AutoMode = PerServer. The integer value indicates the number of client licenses purchased for the server being installed.

Important The number specified must be greater than five, otherwise unattended Setup mode fails and the user is prompted to pick a value through the user interface.

[GuiRunOnce]

This section contains a list of commands to be executed the first time a user logs on to the computer after GUI-mode Setup has completed. Each line specifies a command to be executed by the GuiRunOnce registry entry. For example:

To install	Add this line to [GuiRunOnce]
Domain Controller	" depromo/answer :<answer file>"
Cluster Service	"%windir%\cluster\cluscfg.exe -unattend"

Important Each command line must be in quotes.

For more information about these examples, see their respective sections in this document.

Commands run using the GuiRunOnce key run in the context of the currently logged-in user. If the user does not have the permissions necessary to run the command completely, then the application fails. Because this is run in the context of a logged-in user rather than as a service, the registry entries that the application creates are written for the current user rather than the default user. (Default user registry settings are propagated to new users.) If you want any settings and updates to show only for the specifically logged-in user, then this may be appropriate. Otherwise, Cmdlines.txt is a better approach to running commands and installing applications because it runs as a system service.

[Display]

This section specifies display settings for the particular graphics device being installed. For this to work properly, the user must know what settings are valid for the graphics device under consideration.

If the previously specified settings are not valid for the particular graphics device, Setup finds the closest match to the selected settings to configure the device.

BitsPerPel

Value: <valid bits per pixel>

Specifies the <valid bits per pixel> for the graphics device being installed. For example, a value of 8 (2⁸) implies 256 colors, and a value of 16 implies 65,536 colors.

Vrefresh

Value: <valid refresh rate>

Specifies a <valid refresh rate> for the graphics device being installed.

Xresolution

Value: <valid x resolution>

Specifies a <valid x resolution> for the graphics device being installed.

Yresolution

Value: <valid y resolution>

Specifies a <valid y resolution> for the graphics device being installed.

[RegionalSettings]

This optional section specifies regional options.

Note

- To use this section, you must add, as a minimum, the **/copysource:lang** command line to Winnt32.exe or the **/rx:lang** command line to Winnt.exe to get the appropriate language files copied to the hard disk. For example, if you are only interested in Korean settings while installing a U.S. version of Windows 2000, you may specify **/copysource:lang\kor** if starting from Winnt32.exe.

For a complete listing of Winnt.exe and Winnt32.exe commands, see Windows 2000 Help. Or, at a command prompt, switch to the \i386 folder on the Windows 2000 product CD, and then type: **winnt /?** or **winnt32 /?**

- When specifying OemPreinstall = Yes and not providing values for the [RegionalSettings] section, set OEMSkipRegional = 1 in the [GuiUnattended] section of the Unattend.txt file to ensure Setup completes without prompting for regional option information.

InputLocale

Value: <locale ID>:<keyboard layout ID>, <locale ID>:<keyboard layout ID>, ...

Specifies the input locale and keyboard layout combinations to be installed on the computer (the first keyboard layout specified is the default layout for the installation). The specified combinations must be supported by one of the language groups defined using the LanguageGroup key or the default language group for the language version of Windows 2000 being installed. If an available language group does not support the combination specified, the default combination is used for the installation. This key is ignored if the Language key is specified.

Note For a list of valid locale ID and keyboard layout combinations, visit the Microsoft Global Software Development Web site at:

<http://www.microsoft.com/globaldev/>

Language

Value: <locale ID>

Specifies the language/locale to be installed. This language must be supported by one of the language groups specified using the LanguageGroup key. If an available language group does not support the locale, the default language for the Windows 2000 version being installed is used.

Important If this key is specified, the SystemLocale, UserLocale, and InputLocale keys are ignored.

For a list of valid locales and their language group, visit the Microsoft Global Software Development Web site at:

<http://www.microsoft.com/globaldev/>

LanguageGroup

Value: <language group ID>, <language group ID>, ...

Specifies the language group support to be installed on the computer. Supported language group IDs are:

ID	Language Group
1	Western Europe and United States
2	Central Europe
3	Baltic
4	Greek
5	Cyrillic
6	Turkic
7	Japanese
8	Korean
9	Traditional Chinese
10	Simplified Chinese
11	Thai
12	Hebrew
13	Arabic
14	Vietnamese
15	Indic
16	Georgian
17	Armenian

SystemLocale

Value: <locale ID>

Specifies the system locale to be installed on the computer. (*System locale* enables localized applications to run and display menus and dialogs in their native language.) The specified system locale must be supported by one of the language groups specified

using the LanguageGroup key or the default language group for the language version of Windows 2000 being installed. If an available language group does not support the locale specified, the default system locale is installed. This key is ignored if the Language key is specified.

Note For a list of valid locales and their language groups, visit the Microsoft Global Software Development Web site at:
<http://www.microsoft.com/globaldev/>

UserLocale

Value: <locale ID>

Specifies the user locale to be installed on the computer. (The user locale controls the settings for numbers, time, currency, and dates.) The specified user locale must be supported by one of the language groups specified using the LanguageGroup key or the default language group for the language version of Windows 2000 being installed. If an available language group does not support the locale specified, the default user locale is installed. This key is ignored if the Language key is specified.

Note For a list of valid locales and their language groups, visit the Microsoft Global Software Development Web site at:
<http://www.microsoft.com/globaldev/>

[TapiLocation]

AreaCode

Value: <area code string>

Specifies the area code that the computer is located in, for example, 425.

CountryCode

Value: <country code number>

Specifies the country code to use for telephony, such as 1 for the United States. For more information, search on the Internet for "ISO 3166" for a list of valid country codes, or visit the International Telecommunication Union Survey Web site at:

<http://www.itu.int/net/cctlds/index.html>

Note Web addresses can change, so you may be unable to connect to the Web site mentioned here.

Dialing

Value: **Tone** | **Pulse**

Specifies the type of dialing to use for the telephony device in the computer.

LongDistanceAccess

Value: <number to get outside line>

Specifies the number to dial to gain access to an outside line, such as 9.

[Fax]

This is an optional section that can be used to specify custom installation information for the fax service in Windows 2000. If the parameters are not specified, the defaults are used for the installation.

ArchiveFolderName

Value: *<folder to save faxes to>*

Specifies the folder in which copies of outbound faxes should be saved or archived.

ArchiveOutgoing

Value: **True** | **False**

Default: **True**

Specifies whether outbound faxes should be archived or saved into a folder. If set to True, a folder must be specified using the ArchiveFolderName parameter.

Csid

Value: *<called subscriber ID string>*

Default: **"Fax"**

Specifies the Called Subscriber ID (CSID) transmitted by the receiving fax machine when receiving an inbound fax. This string is usually a combination of the fax or telephone number and the name of the business. It is often the same as the Transmitter Subscriber ID.

FaxNumber

Value: *<fax number>*

Default: **"Fax"**

Specifies the fax phone number.

FaxPrinterName

Value: *<fax printer name string>*

Default: **"Fax"**

Specifies the fax printer name for outbound faxes. The value for this setting *cannot* be the same as the value of the RoutePrinterName parameter.

Rings

Value: *<number of rings>*

Default: **2**

Specifies the number of rings before the phone is answered.

RouteFolderName

Value: *<folder to route faxes to>*

Specifies the destination folder for inbound faxes.

RoutePrinterName

Value: *<printer to route faxes to>*

Specifies the printer name to which faxes should be routed automatically. If the printer is remote, the value must be the UNC name of the fax printer, for example, \\server\printername.

RouteToFolder

Value: **True** | **False**

Default: **True**

Specifies whether faxes should be directly routed to a folder. If set to True, a folder can be specified using the RouteFolderName parameter.

RouteToPrinter

Value: **True** | **False**

Default: **False**

Specifies whether inbound faxes should be directly routed to a printer. If set to True, the custom printer can be specified using the RoutePrinterName parameter.

Tsid

Value: *<transmitter subscriber ID string>*

Default: **"Fax"**

Specifies the Transmitter Subscriber ID (TSID) sent by the fax machine when sending a fax to a receiving fax machine. This string is usually a combination of the fax or telephone number and the name of the business. It is often the same as the Called Subscriber ID.

[Win9xUpg]

This section specifies the parameters for an unattended upgrade from Windows 95 or Windows 98. These parameters are not valid on any other upgrade path.

Boot16

Value: **Yes** | **No**

Default: **No**

Enables MS-DOS boot mode.

DefaultPassword

Value: *<password>*

Optional

Provides a default password for all local accounts created during a migration process. Since Setup cannot migrate the Windows passwords of users when upgrading a system, it must assign passwords for those created accounts. When Setup needs to assign one of these passwords, it first checks to see if a UserPassword (see "UserPassword," later in this document) entry for that user exists. If not, it uses the value of this key if specified.

Note There are some security concerns about using this key, since the password is stored as plain text within the Unattend.txt file. All passwords in the answer file are automatically deleted after the upgrade.

If a local account needs to be created for a user without a UserPassword entry and no DefaultPassword is specified, Setup creates a random password. After the first reboot, the user is prompted to enter a password for all users created during Setup. For more information, see "UserPassword," later in this document.

DomainJoinText

Value: *<"HTML text">*

Optional

Inserts the HTML text on the Domain Join page of the GUI-mode Setup wizard.

Note The only supported HTML tags are anchor (<A>) and bold ().

ForcePasswordChange

Value: **Yes** | **No**

Default: **Yes**

Optional

Instructs Setup to automatically require a password change on all local accounts it creates during the migration process. When a user first logs on using one of these accounts, the user is informed that the current password has expired. The user is forced to select a new password before logging on.

ForceWorkgroup

Value: **Yes** | **No**

Default: **No**

Instructs Setup to skip domain checks and to run the computer in Workgroup mode after the upgrade.

IgnoreNetworkErrors

Value: **Yes** | **No**

Default: **No**

Instructs Setup not to retry a domain that fails. If this parameter is set to **Yes**, the setup is completely unattended; however, if this parameter is set to **Yes**, the domain account migration may fail.

KeepBadLinks

Value: **Yes** | **No**

Default: **Yes**

Instructs Setup to keep all links. Setting this parameter to **No** removes incompatible links.

MigrateDefaultUser

Value: **Yes** | **No**

Default: **Yes**

Optional

Causes Setup to migrate the default Windows 95 or Windows 98 user account settings to the default Windows 2000 user account.

Important

- If this parameter is set to No, the Win9x default user settings are *not* maintained. Subsequent user accounts created on the computer will inherit Windows 2000 defaults.
 - Some applications rely on settings migrated to the default user. These applications may not work unless default user migration is enabled.
 - This key should be used in special cases only (for example, when it is known that no applications require default user migration).
-

MigrateUsersAsAdmin

Value: **Yes** | **No**

Default: **Yes**

Optional

Causes Setup to add all accounts that it creates during migration to the Local Administrators group, giving those users full control over the computer. If users are not migrated as administrators, some migrated applications and settings may not work properly after the upgrade is completed.

Note For more information about group-level security settings, see Windows 2000 Help.

MigrateUsersAsPowerUser

Value: **Yes** | **No**

Default: **Yes**

Optional

Causes Setup to add all accounts that it creates during migration to the Power Users group, giving the new users in the Power Users group more permissions than users in the Users group, but fewer permissions than users in the Administrators group. Power Users can perform any operating system task except tasks reserved for the Administrators group.

Note For more information about group-level security settings, see Windows 2000 Help.

MigrationDlls

Value: *<migration dll path>* [*,<migration dll path>, ...*]

Optional

Specifies the location of upgrade packs that Setup needs to copy and process during an upgrade to Windows 2000. If multiple paths are specified, commas must separate the paths. Setup searches each of these paths (including each path's subfolders) for upgrade packs. Multiple upgrade packs can be located at a single location, but each upgrade pack must exist in its own subfolder of that single path. Do not put more than one upgrade pack in a single folder.

Note An upgrade pack consists of a migration .dll file (Migrate.dll) and any additional files that may be required to properly upgrade a particular software component from Windows 95 or Windows 98 to Windows 2000.

ReportOnly

Value: **Yes** | **No**

Default: **No**

Optional

Instructs Setup to generate an upgrade report and then exit without making any changes to the current Windows 95 or Windows 98 installation. The upgrade report contains a list of hardware and software incompatibilities and is saved to the root of the system drive if the SaveReportTo key is not specified.

For more information, see "SaveReportTo," later in this document.

SafeMode

Value: **Yes** | **No**

Default: **No**

When set to Yes, this parameter instructs Setup to maintain a list of settings during the upgrade. If any setting causes Win9x to hang or crash, that setting is skipped during the next setup. Using this parameter slows Setup, but it allows Setup to skip certain types of malfunctioning software installed on Win9x.

Important Safe mode is started automatically if a problem is detected. If SafeMode = Off, Setup *must* be run three times. If SafeMode = Yes, Setup need only be run two times.

SaveReportTo

Value: *<path to report>*

Optional

Specifies the full path and file name of the upgrade report (text file) generated by Setup. If the Windows 95 or Windows 98 installation has network connectivity, the *<path to report>* can be a UNC path.

Note System environment variables enclosed in percent signs (%) can be used in the path. If the special environment variable %*computername*% is specified in the path, the resulting file name contains the actual computer name. This is useful when upgrade reports are being collected from several Windows 95 or Windows 98 computers to a central location.

For more information, see “ReportOnly,” earlier in this document.

ScanDrives

Value: [*drive*][,*drive,drive*]

Optional

Enables scanning of drives that are normally skipped. This parameter is used with ReportOnly and allows software on network drives to be checked.

For more information, see “ReportOnly,” earlier in this document.

UseLocalAccountOnError

Value: **Yes** | **No**

Default: **No**

Optional

Directs Setup to create a local account if a network account cannot be automatically determined or resolved. This is only valid on computers with the Microsoft Networking Client software installed.

Windows 95 and Windows 98 only keep the domain of the last logged-in user in the registry. Because neither Windows 95 nor Windows 98 keeps the domains of other users who may have logged on to the computer, Windows 2000 Setup searches all trusted domains on the network by default. Windows 2000 Setup automatically uses a domain account when an exact match is found.

If a user is not found on any trusted domain or if the user account is found on two or more domains on the network, a dialog box appears prompting the person performing the upgrade to resolve the conflict. This dialog box also appears if network errors occur.

Specifying UseLocalAccountOnError = Yes in the Unattend.txt file ensures a complete unattended installation. This causes Setup to create a local account whenever a network account cannot be automatically resolved.

Note A user with a local account may not have his or her original network privileges.

In addition, if a computer cannot be added to the computer domain during installation of the network on Windows 2000, all user accounts change to local accounts.

For more information, see “UserDomain.”

UserDomain

Value: <*domain name*>, <*user name*> [, <*user name*>, <*user name*>, ...]

Optional

Specifies the user domain for a user. Multiple UserDomain lines can be used to specify different domains for different users. When specified, this key prevents Setup from searching all trusted domains on the network for a matching user account. (The search process can be time-consuming if a large number of trusted domains exist on the network.)

If the account is not found in the specified domain, either because the account does not exist or the domain is not accessible, a dialog box appears prompting the user to resolve the account unless the UseLocalAccountOnError key is set to Yes.

For more information, see "UseLocalAccountOnError," earlier in this document.

UserPassword

Value: <user>,<password> [,<user>,<password>,...]

Optional

Provides Setup with the names of the passwords to create for specific local accounts. Because Setup cannot migrate the Windows passwords of users when upgrading a system, it must create passwords for non-domain accounts during the migration process. Using this key, an administrator can predetermine what those passwords are for specific users.

Note There are some security concerns about using this key, since the password is stored as plain text within the answer file. However, after the upgrade is completed, all the password keys are deleted from the copy of the answer file left on the computer. The original copy of the answer file you started Setup with is not deleted.

If a local account needs to be created for a user without a UserPassword entry and no DefaultPassword is specified, Setup creates a random password. After the first reboot, the user is prompted to enter a password for all users created during Setup.

For more information, see "DefaultPassword," earlier in this document.

[SystemFileProtection]

In Windows 2000, a new feature, called Windows (or System) File Protection (WFP), prevents the replacement of certain monitored or protected system files. By preventing the replacement of essential system files, file version mismatches can be avoided. WFP runs in the background on a Windows 2000 system. All .sys, .dll, .exe, and .ocx files that ship on the Windows 2000 CD are protected. TrueType fonts Micros.ttf, Tahoma.ttf, and TahomaBd.ttf are also protected.

Note In earlier pre-release versions of Windows 2000, WFP was known as System File Checker. This nomenclature is being maintained for the parameter naming conventions.

WFP is activated when:

- Attempts to replace a protected system file are made.
- Attempts to change a protected folder or a file in a protected folder are made. (WFP then determines which file was changed.)

Windows File Protection also maintains version control of protected files. It looks up the file signature in a catalog file to determine if the new file is the correct Microsoft version. If the new file is not the correct Microsoft version, Windows File Protection looks for the replaced file(s) in the following order:

1. Search the Dllcache folder.
2. Search the network install path (if the system was installed via network install).
3. Search the CD.

The following parameters are used by the Windows File Protection service. If this section is missing or empty, Setup installs WFP using default values.

SFCDllCacheDir

Value: <location of Dllcache folder>

Default: "%systemroot%\System32\Dllcache"

Specifies the folder that is used by Windows File Protection to cache the system files (the Dllcache folder location). This path must be a local path. Environment variables are also allowed for this parameter. For example:

```
SFCDllCacheDir = "%systemroot%\Winnt\System32\Dllcache"
```

Important Quotation marks are required around the value you provide.

SFCShowProgress

Value: **0** | **1**

Default: **0**

Specifies if System File Checker displays a progress meter during scans. If SFCShowProgress = 0, no progress indicator is displayed. If SFCShowProgress = 1, a progress meter is displayed during file system scans.

SFCQuota

Value: <size in MB (hex)>

Default: 0x32h (50 MB for Windows 2000 Professional;

FFFFFFFF for Windows 2000 Server, Windows 2000 Advanced Server, and Windows 2000 Datacenter Server)

Specifies the size of the Dllcache file cache stored on the system hard drive. If FFFFFFFF is specified, all system files will be cached in the Dllcache folder. For example:

```
SFCQuota = FFFFFFFF
```

[Components]

This section contains parameters for installing the components of Windows 2000. A value of On installs the component, and a value of Off prevents the component from being installed.

accessopt

Value: **On** | **Off**

Default: **On**

Installs the Accessibility wizard.

calc

Value: **On** | **Off**

Default: **On**

Installs the Calculator.

cdplayer

Value: **On** | **Off**

Default: **On**

Installs the CD Player program.

Note If cdplayer = Off, the CD player is not installed. However, if there is a DVD-ROM device present, the computer detects it and installs the DVD player by default.

certsrv

Value: **On** | **Off**

Default: **Off**

Installs the Certificate Services components.

certsrv_client

Value: **On** | **Off**

Default: **Off**

Installs the Web client components of the Certificate Services feature. These components allow you to publish Web pages on your server for submitting and requesting certificates from a Certification Authority. Options for the Web client services must be specified in the [Certsrv_Client] section.

certsrv_server

Value: **On** | **Off**

Default: **Off**

Installs the server components of the Certificate Services features for the Windows 2000 Server family only. These components allow you to create a Certification Authority on your server for issuing digital certificates. When certsrv_server is set to On, the certsrv_client parameter must also be set to On. Options for the server services must be specified in the [Certsrv_Server] section.

charmap

Value: **On** | **Off**

Default: **On**

Installs the Character Map program that inserts symbols and characters into documents.

chat

Value: **On** | **Off**

Default: **Off**

Installs the Chat program.

cluster

Value: **On** | **Off**

Default: **Off**

Installs the Cluster service Node and Administration components. This is only valid when installing Windows 2000 Advanced Server or Windows 2000 Datacenter Server. When set to On, the parameters in the [Cluster] section are processed.

Important

- The Cluster service and Terminal Services are mutually exclusive in certain situations. If you intend to use the server with Terminal Services installed and Remote Administration enabled, you *can* enable the Cluster service. On the other hand, if you intend to use the server with Terminal Services installed and Application Server enabled, you *cannot* enable the Cluster service. For more information, see “[TerminalServices],” later in this document.
 - This parameter requires that iis_common be installed.
-

desktoppaper

Value: **On** | **Off**

Default: **On**

Installs desktop wallpaper on your computer desktop.

Note If desktoppaper = Off, only the standard Windows 2000 wallpaper bitmaps are not installed. Wallpaper files for Active Desktop™ are not affected and are still installed.

dialer

Value: **On** | **Off**

Default: **On**

Installs the Phone Dialer program.

fp_extensions

Value: **On** | **Off**

Default: **On** (Windows 2000 Server family);

Off (Windows 2000 Professional)

Installs the FrontPage® server extensions.

Note To use this parameter, the parameters iis_common, iis_inetmgr, and iis_www must all be installed.

freecell

Value: **On** | **Off**

Default: **On**

Installs the Freecell game.

hypertrm

Value: **On** | **Off**

Default: **On**

Installs the HyperTerminal program.

iis_common

Value: **On** | **Off**

Default: **On** (Windows 2000 Server family);

Off (Windows 2000 Professional)

Installs the common set of files needed by Internet Information Services (IIS).

iisdbg

Value: **On** | **Off**

Default: **On** (Windows 2000 Server family);

Off (Windows 2000 Professional)

Installs the Microsoft Script Debugger.

iis_doc

Value: **On** | **Off**

Default: **On** (Windows 2000 Server family);

Off (Windows 2000 Professional)

Installs documentation for IIS.

Note Installing this parameter also installs the following prerequisite parameters: `iis_common`, `iis_inetmgr`, and `iis_www`.

iis_ftp

Value: **On** | **Off**

Default: **Off**

Installs the FTP service.

Note Installing this parameter also installs the following prerequisite parameters: `iis_common` and `iis_inetmgr`.

iis_htmla

Value: **On** | **Off**

Default: **On** (Windows 2000 Server family);

Off

Installs the HTML-based administration tools for IIS. This parameter is only valid for the Windows 2000 Server family.

Note Installing this parameter also installs the following prerequisite parameters: `iis_common`, `iis_inetmgr`, and `iis_www`.

iis_inetmgr

Value: **On** | **Off**

Default: **On** (Windows 2000 Server family);

Off (Windows 2000 Professional)

Installs the Microsoft Management Console–based administration tools for IIS.

Note Installing this parameter also installs the following prerequisite parameter:
iis_common.

iis_nntp

Value: **On** | **Off**

Default: **Off**

Installs the Network News Transfer Protocol (NNTP) service for the Windows 2000 Server family.

Note Installing this parameter also installs the following prerequisite parameters:
iis_common, iis_inetmgr, and iis_www.

iis_nntp_docs

Value: **On** | **Off**

Default: **Off**

Installs the documentation for the NNTP service.

Note Installing this parameter also installs the following prerequisite parameters:
iis_common, iis_inetmgr, and iis_www.

iis_pwmgr

Value: **On** | **Off**

Default: **Off**

Installs the Personal Web Manager. This parameter is valid only on Windows 2000 Professional.

Note Installing this parameter also installs the following prerequisite parameters:
iis_common, iis_inetmgr, and iis_www.

iis_smtp

Value: **On** | **Off**

Default: **On** (Windows 2000 Server family);

Off (Windows 2000 Professional)

Installs the Simple Mail Transfer Protocol (SMTP) service for the Windows 2000 Server family.

Note Installing this parameter also installs the following prerequisite parameters: iis_common, iis_inetmgr, and iis_www.

iis_smtp_docs

Value: **On** | **Off**
Default: **On** (Windows 2000 Server family);
Off (Windows 2000 Professional)
Installs documentation for the SMTP service.

Note Installing this parameter also installs the following prerequisite parameters: iis_common, iis_inetmgr, and iis_www.

iis_www

Value: **On** | **Off**
Default: **On** (Windows 2000 Server family);
Off (Windows 2000 Professional)
Installs the World Wide Web (WWW) service.

Note Installing this parameter also installs the following prerequisite parameters: iis_common and iis_inetmgr.

indexsrv_system

Value: **On** | **Off**
Default: **On** (Windows 2000 Server family);
Off (Windows 2000 Professional)
Installs the Indexing Service files.

Note Installing this parameter also installs the following prerequisite parameters: iis_common, iis_inetmgr, and iis_www.

LicenseServer

Value: **On** | **Off**
Turns Terminal Services licensing on or off.

Important Terminal Services licensing can only be run on domain controllers or on a server in a workgroup. For more information, see Terminal Services licensing in Windows 2000 Server Help.

media_clips

Value: **On** | **Off**
Default: **On**
Installs sample sound clips on your computer.

media_utopia

Value: **On** | **Off**

Default: **Off**

Installs the Utopia Sound Scheme on your computer.

minesweeper

Value: **On** | **Off**

Default: **On**

Installs the Minesweeper game on your computer.

mousepoint

Value: **On** | **Off**

Default: **On**

Installs all the available mouse pointers that ship with Windows 2000 on your computer.

mplay

Value: **On** | **Off**

Default: **On**

Installs the Media Player program on your computer.

msmq

Value: **On** | **Off**

Default: **Off**

Installs the Message Queuing Services (MSMQ) components.

mwordpad

Value: **On** | **Off**

Default: **On**

Installs the WordPad program on your computer.

netcis

Value: **On** | **Off**

Default: **On** (Windows 2000 Server family);

Off (Windows 2000 Professional)

Installs the Microsoft COM Internet Services.

Note This parameter requires that iis_common, iis_inetmgr, and iis_www are all installed.

netoc

Value: **On** | **Off**

Default: **On**

Installs additional optional networking components. To modify the default components installed, see the [NetOptionalComponents] section. If the value of netoc is Off, the [NetOptionalComponents] section is not processed.

objectpkg

Value: **On** | **Off**

Default: **On**

Installs the Object Packager program on your computer.

paint

Value: **On** | **Off**

Default: **On**

Installs the Paint program on your computer.

pinball

Value: **On** | **Off**

Default: **On**

Installs the Pinball game on your computer.

rec

Value: **On** | **Off**

Default: **On**

Installs the Sound Recorder program on your computer.

reminst

Value: **On** | **Off**

Default: **Off**

Installs the Remote Installation Service, which provides the ability to remotely install an operating system onto a computer that contains either a new PXE-based remote boot ROM or a network card that is supported by the remote installation boot floppy.

rstorage

Value: **On** | **Off**

Default: **Off**

Installs the Remote Storage Service feature that allow the use of tape libraries as extensions of NTFS volumes.

solitaire

Value: **On** | **Off**

Default: **On**

Installs the Solitaire game on your computer.

templates

Value: **On** | **Off**

Default: **On**

Installs Document Templates on your computer.

TSClients

Value: **On** | **Off**

Default: **Off**

Installs the files (approximately 10 MB) required to create Terminal Services client disks.

TSEnable

Value: **On** | **Off**

Default: **Off**

Installs Terminal Services on Windows 2000. This key is only valid for the Windows 2000 Server family.

Important Cluster service and Terminal Services are mutually exclusive in certain situations. If you intend to use the server with Terminal Services installed and Remote Administration enabled, you *can* enable the Cluster service. On the other hand, if you intend to use the server with Terminal Services installed and Application Server enabled, you *cannot* enable the Cluster service.

To fully automate the installation of a server with Terminal Services enabled, include the [TerminalServices] section and appropriate ApplicationServer parameter. For more information, see “[TerminalServices],” later in this document.

vol

Value: **On** | **Off**

Default: **On**

Installs the Volume Control program on your computer.

WMS

Value: **On** | **Off**

Default: **On**

Must be ON to install either of the Windows Media Services components: administrative or server.

Valid for the Windows 2000 Server family only.

WMS_Admin

Value: **On** | **Off**

Default: **On**

Installs the Windows Media Services administrative components. Requires a WMS value of ON.

Valid for the Windows 2000 Server family only.

WMS_SERVER

Value: **On** | **Off**

Default: **On**

Installs the Windows Media Services server components. Requires a WMS value of ON and installation of the WMS administrative components.

Valid for the Windows 2000 Server family only.

[Networking]

This section header must be present for unattended installation of networks.

[Identification]

Specifies the network identification parameters of a computer. If this key is not present, the computer is added to the default workgroup. If the information for this key is not complete, the user is prompted to provide this information.

DomainAdmin

Value: *<account name>*

Specifies the name of a user account in the domain that has permission to create a computer account in that domain. This key is required if you set the value of the JoinDomain key to Yes even though the computer account may already exist on the domain. For example:

```
DomainAdmin = accountname
```

DomainAdminPassword

Value: *<password of user account>*

Specifies the password of the user account as defined by the DomainAdmin parameter. This key is required if you set the value of the JoinDomain key to Yes even though the computer account may already exist on the domain. For example:

```
DomainAdminPassword = abcdef123
```

Note This is a potential security issue, since the password is in plain text. For a deployment, you should create a special account that can be used to create computer accounts in the domain, or—better still—create computer accounts in the domain before starting the deployment. The password is deleted from the copy of the Unattend.txt file left on the computer after Setup is complete.

JoinDomain

Value: *<domain name>*

Specifies the name of the domain in which the computer participates. You can specify either this key or the JoinWorkgroup key, but you cannot specify both. For example:

```
JoinDomain = MYDOMAIN
```

JoinWorkgroup

Value: *<workgroup name>*

Specifies the name of the workgroup in which the computer participates. You can specify either this key or the JoinDomain key, but you cannot specify both. For example:

```
JoinWorkgroup = MYUSERGROUP
```

MachineObjectOU

Value: "<Domain Name System (DNS) name, ldap path>"

Optional

Specifies the full LDAP path name of the organizational unit (OU) in which the computer belongs. For example:

```
MachineObjectOU = "OU = myou,OU = myparentou,DC = mydom,DC = mycompany,DC = com"
```

[NetAdapters]

This section specifies the network adapters to be installed. Each adapter present in the computer must be listed as a key = value pair.

The key name is user-defined and acts as an identifier for the adapter to be installed.

The value of this key (also user-defined) specifies a section that contains parameters for the adapter under consideration.

<adapter instance>

Value: <adapter parameters section>

This parameter is a user-defined name for the adapter for which specific settings need to be defined. Each adapter needs a parameter created, with each one having a unique <adapter parameters section> specified for the value. For example:

```
Adapter1 = params.Adapter1
```

[<params.adapter instance>]

This section describes the parameters of a particular adapter's instance on a computer. To create an unattended answer file for identical computers with the same hardware configurations, including the same type of network adapters in the same PCI slots:

1. Install Windows 2000 on one of the computers.
2. View each network adapter's properties on the General tab in Device Manager and make a note of the PCI location information.
3. This information (PCI Bus, Device, and Function numbers) should be written in the Unattend.txt file as specified later in this document.
4. The file is now ready to install the operating system on the other computers.

Note The PCI location information is composed of three parameters: PCIBusNumber, PCIDeviceNumber, and PCIFunctionNumber. All of these parameters must be used to ensure that networking settings are applied based on PCI location.

If only adapters of different types are on the computers (that is, the PnP IDs are not the same), the answer file can use the explicit PnP IDs of each adapter instead of PCI location information.

Note Adding devices after the operating system is installed can cause bus numbers to change. As a result, the Unattend.txt file may not work on other computers. If only adapters of different types are on the computers (that is, the PnP IDs are not the same), the unattended file can use the explicit PnP IDs of each adapter instead of PCI location information.

ConnectionName

Value: "<network connection name>"

Optional

Specifies the name for the network connection associated with the network adapter being installed. If this key is not specified or its value is empty, the default connection name of "Local Area Connection" is used for the first enumerated network adapter and "Local Area Connection x" is used for subsequent adapters where *x* starts from 2. For example:

```
ConnectionName = "Corporate Network"
```

DMA

Value: <DMA channel number>

Optional

Specifies the direct memory access (DMA) channel setting of a network adapter. For example:

```
DMA = 1
```

DoNotDetectLegacyCards

Value: <InfID>[, <InfID>, ...]

Specifies legacy (non-PnP) adapters that Setup should ignore when trying to detect the network adapters on a computer. These cards are not detected and/or installed on the computer. For example:

```
DoNotDetectLegacyCards = *PNP030b,*PNP8132
```

Note

- This key does not prevent PnP adapters from being enumerated and installed.
 - The remaining keys in this section are optional and are only meaningful for non-PnP adapters that require resources to be explicitly specified.
-

InfID

Value: <PnP ID of adapter>

This key uniquely identifies a network adapter. Its value is the same as the adapter's PnP ID.

If InfID = "*", Setup matches the first enumerated/detected network adapter on the computer with the settings in this section. Therefore, on a computer with a single

network card, you can specify adapter-specific parameters without knowing the name or type of the network adapter.

For example:

```
InfID = *PNP030b
```

Note For computers with more than one network adapter, the correct PnP ID of each of the adapters must be specified. For computers with multiple adapters of the same type (that is, the PnP ID for each is the same), the NetCardAddress or PCI location information must be specified. If neither NetCardAddress nor PCI location is specified, only the first adapter enumerated/detected that matches the InfID receives the answer file parameters.

If NetCardAddress or PCI location is specified, this parameter is not used because it is the least specific parameter that can be provided.

IOAddr

Value: <numeric IO address>

Optional

Specifies the IO address (hexadecimal number) of a network adapter. For example:

```
IOAddr = 330
```

IRQ

Value: <IRQ number>

Optional

Specifies the interrupt number setting of a network adapter. For example:

```
IRQ = 5
```

MEM

Value: <memory range>

Optional

Specifies the memory base address setting of a network adapter. For example:

```
MEM = 0xC123456
```

NetCardAddress

Value: <network card MAC address>

Specifies the media access control (MAC) address for the network adapter being configured during unattended Setup mode. This key is required when installing multiple network cards of the same type (that is, each card has the same PnP ID) on a computer and non-default parameters need to be applied to the adapters. For example:

```
NetCardAddress = 0x123456789AB
```

Note This parameter is not required for PCI adapters if the PCI location information is specified.

PCIBusNumber

Value: <PCI bus number of adapter>

Indicates on which PCI bus the network card resides. This parameter is one of the three parameters that are required to specify the PCI location information.

PCIDeviceNumber and PCIFunctionNumber must also be specified.

These PCI location keys are required if:

- NetCardAddress is not specified.
- Multiple network cards of the same type (that is, cards with the same PnP ID) are installed on a computer.
- Non-default parameters need to be applied to the adapter.

For example:

```
PCIBusNumber = 0
PCIDeviceNumber = 11
PCIFunctionNumber = 0
```

Note If NetCardAddress is specified, the PCIBusNumber, PCIDeviceNumber, and PCIFunctionNumber parameters are not used, because NetCardAddress provides more specific information than the PCI location information.

PCIDeviceNumber

Value: <PCI device number of adapter>

Indicates the PCI device number for the network card and approximately relates to the particular slot number in which the network card resides on the given PCI bus. This parameter is one of the three parameters that are required to specify the PCI location information. PCIBusNumber and PCIFunctionNumber must also be specified.

These PCI location keys are required if:

- NetCardAddress is not specified.
- Multiple network cards of the same type (that is, cards with the same PnP ID) are installed on a computer.
- Non-default parameters need to be applied to the adapter.

For example:

```
PCIBusNumber = 0
PCIDeviceNumber = 11
PCIFunctionNumber = 0
```

Note If NetCardAddress is specified, the PCIBusNumber, PCIDeviceNumber, and PCIFunctionNumber parameters are not used, because NetCardAddress provides more specific information than the PCI location information.

PCIFunctionNumber

Value: *<PCI function number of adapter>*

Defines which function on the card provides the networking function—for example, if there is a combination modem/network card. The function number begins with 0. This parameter is one of the three parameters that are required to specify the PCI location information for the network adapter being configured during unattended Setup mode. PCIbusNumber and PCIdeviceNumber must also be specified.

These PCI location keys are required if:

- NetCardAddress is not specified.
- Multiple network cards of the same type (that is, cards with the same PnP ID) are installed on a computer.
- Non-default parameters need to be applied to the adapter.

For example:

```
PCIBusNumber = 0
PCIDeviceNumber = 11
PCIFunctionNumber = 0
```

Note If NetCardAddress is specified, the PCIBusNumber, PCIdeviceNumber, and PCIFunctionNumber parameters are not used, because NetCardAddress provides more specific information than the PCI location information.

[NetProtocols]

This section specifies the network protocols being installed.

<protocol name>

Value: *<protocol parameters section>*

Specifies the section where parameters of this protocol are defined. The protocol name must be selected from the Protocol ID column of the following table. This table describes each protocol and the components that are automatically installed with it. For example:

```
MS_TCPIP = params.TCPIP
```

Protocol ID	Description	Components Installed
MS_TCPIP	TCP/IP Protocol	MS_NetBT
MS_NWIPX	IPX Protocol	MS_NWSPX, MS_NWNB
MS_PPTP	Point-to-Point Tunneling Protocol	
MS_L2TP	Layer 2 Tunneling Protocol	
MS_DLC	DLC Protocol	
MS_AppleTalk	AppleTalk Protocol	

Protocol ID	Description	Components Installed
MS_NetBEUI	NetBEUI Protocol	
MS_NetMon	Network Monitor Agent	
MS_ATMLANE	ATM LAN Emulation Client	
MS_ATMUni	ATM Call Manager Protocol	
MS_ATMArps	ATM ARP Server Protocol	
MS_STREAMS	STREAMS Protocol	

[<protocol parameters section>]

This section contains parameters for the protocol specified by the <protocol name> key specified in the [NetProtocols] section.

AdapterSections

Value: <section name>[,<section name>, ...]

Optional

Specifies a list of sections that contain device-specific network protocol parameters.

Each of the adapter sections listed here has a SpecificTo value specified. The value of the SpecificTo key is the name of a network adapter defined in the [NetAdapters] section. For example:

```
AdapterSections = params.TCPIP.Adapter1, params.TCPIP.Adapter2
```

For more information, see "SpecificTo," later in this document.

[<adapter specific protocol section>]

SpecificTo

Value: <network component name>

Specifies that the parameters in this section are specific to the network component defined by the <network component name> value. If SpecificTo is not present, the parameters in the section are global to the network protocol being configured during Setup. For example:

```
SpecificTo = Adapter1
```

Note If a parameter is specified in both the Protocol parameters section and the adapter-specific Protocol section (that is, a section where the SpecificTo key is used), the parameters specified in the adapter-specific section are ignored.

[<MS_TCPIP parameters>]

This section contains parameters for configuring the TCP/IP protocol.

Important The parameters in the first part of this section are global; that is, they apply to all network adapters on the computer.

DeadGWDetectDefault

Value: **Yes** | **No**

Default: **No**

Global

Optional

This option should only be used if the computer is going to be a Routing and Remote Access service of any type. If this is the case, this value must be set to **No**.

For more information, see “[<MS_RAS parameters>],” later in this document. For example:

```
DeadGWDetectDefault = No
```

DNSDomain

Value: <primary DNS suffix>

Global

Optional

Specifies the primary DNS suffix of the computer. For example:

```
DNSDomain = reskit.com
```

Requires **SyncDomainWithMembership** = 0.

DNSSuffixSearchOrder

Value: <suffix>[, <suffix>[, ...]]

Global

Optional

Specifies the search order for DNS servers. For example:

```
DNSSuffixSearchOrder = server1.bar.com, bar.com
```

DontAddDefaultGatewayDefault

Value: **Yes** | **No**

Default: **Yes**

Global

Optional

This option should only be used if the computer is going to be a Routing and Remote Access service of any type, in which case this value must be set to **Yes**. For example:

```
DontAddDefaultGatewayDefault = Yes
```

For more information, see “[<MS_RAS parameters>],” later in this document.

EnableICMPRedirect

Value: **Yes** | **No**

Default: **No**

Global

Optional

This option should only be used if the computer is going to be a Routing and Remote Access service of any type. If this is the case, this value must be set to No. For example:

```
EnableICMPRedirect = No
```

For more information, see “[<MS_RAS parameters>],” later in this document.

EnableLMHosts

Value: **Yes** | **No**

Default: **Yes**

Global

Optional

Specifies whether LMHosts lookup should be used on this computer. For example:

```
EnableLMHosts = Yes
```

EnableSecurity

Value: **Yes** | **No**

Default: **No**

Global

Optional

Enables TCP/IP filtering. For example:

```
EnableSecurity = No
```

ScopeID

Value: <scope ID>

Global

Optional

Specifies the computer's scope identifier if it is required on a network that uses NetBIOS over TCP/IP. Do not use a scope ID if DNS is enabled on the network in which the computer participates. For example:

```
ScopeID = REDMOND
```

SyncDomainWithMembership

Value: **0** | **1** or **not present**

Default: **not present**

Global

If not present, or if non-zero, the primary DNS suffix value will be changed when the computer joins an Active Directory (Windows 2000) domain. The name will be the DNS name of the Active Directory domain the computer joins.

To specify a primary DNS suffix, set `SyncDomainWithMembership` to 0 and set `DNSSDomain` = *<primary DNS suffix>*.

UseDomainNameDevolution

Value: **Yes** | **No**

Default: **Yes**

Global

Optional

Enables domain name devolution when the DNS caching resolver is given an unqualified query to resolve. For example:

```
UseDomainNameDevolution = No
```

Important The remaining keys in this section are network adapter-specific. Therefore, they must only be used in a TCP/IP parameters section that has the `SpecificTo` key specified. For more information, see “`SpecificTo`,” earlier in this chapter.

DefaultGateway

Value: *<gateway address>* [, *<gateway address>* [, ...]]

Adapter-specific

Specifies the default gateway address(es). For example:

```
DefaultGateway = 193.1.1.1
```

DHCP

Value: **Yes** | **No**

Default: **Yes**

Adapter-specific

Optional

Specifies whether Dynamic Host Configuration Protocol (DHCP) should be used on the network connection that uses this adapter. For example:

```
DHCP = Yes
```

Note If `DHCP` = `No`, the `IPAddress` and `SubnetMask` keys must be specified. The `DefaultGateway` key may also need to be specified, if necessary, for your network.

DHCPClassId

Value: *<string>*

Adapter-specific

Optional

Specifies the DHCP class ID. For example:

```
DHCPClassId = foo
```

DNSSDomain

Value: <DNS domain name>

Adapter-specific

Optional

Specifies the suffix of the network connection that uses this adapter. For example:

DNSSDomain = bar_interface.com

DNSServerSearchOrder

Value: <server address>[, <server address> [, ...]]

Adapter-specific

Optional

Specifies a list of addresses to use when searching for the DNS server on the network.

For example:

DNSServerSearchOrder = 193.1.1.3,193.1.1.44

IPAddress

Value: <IP address>[, <IP address>[, ...]]

Adapter-specific

Specifies the IP address(es) for the network connection that uses this adapter. For example:

IPAddress = 193.1.1.33,193.1.1.34

Note

- If more than one IP address is specified, multiple subnet mask addresses must be specified in a one-to-one relationship. The ordering of the addresses is used to pair up with the subnet mask addresses.
 - If IPAddress and SubnetMask are specified, you must specify DHCP = No in the Unattend.txt file.
-

NetBIOSOptions

Value: 0 | 1 | 2

Default: 0

Adapter-specific

Optional

Specifies the NetBIOS option to use on the network connection that uses this adapter.

NetBIOSOptions	Effect
0	Use NetBIOS setting from the DHCP Server
1	Enable NetBIOS over TCP/IP
2	Disable NetBIOS over TCP/IP

SubnetMask

Value: <subnet address>[, <subnet address>[, ...]]

Adapter-specific

Specifies the subnet mask address(es). For example:

```
SubnetMask = 193.1.1.255
```

Note If IPAddress and SubnetMask are specified, you must specify DHCP = No in the Unattend.txt file.

WINS

Value: **Yes** | **No**

Default: **No**

Adapter-specific

Optional

Specifies whether WINS should be used on the network connection that uses this adapter. For example:

```
WINS = Yes
```

WINSServerList

Value: <IP address>[, <IP address>, ...]

Adapter-specific

Optional

Specifies the IP addresses of the WINS servers on the network. For example:

```
WINSServerList = 193.1.1.95,193.1.1.97
```

Important Set WINS = Yes to make sure the WINSServerList IP addresses are applied.

[<MS_NWIPX parameters>]

This section contains parameters for the IPX protocol. The following keys are network adapter-independent.

DedicatedRouter

Value: **Yes** | **No**

Default: **No**

Optional

Specifies that the computer is being installed as a dedicated router and other network services are not running on it. For example:

```
DedicatedRouter = No
```

EnableWANRouter

Value: **Yes** | **No**

Default: **Yes**

Optional

Enables the RIP router. For example:

```
EnableWANRouter = No
```

NetworkNumber

Value: <hexadecimal number>

Default: **0**

Optional

For example:

```
NetworkNumber = 0x1234
```

Important This parameter is network adapter–specific. Therefore, it must only be used in an IPX parameters section that has the SpecificTo key specified. For more information, see “SpecificTo,” earlier in this chapter.

PktType

Value: **0** | **1** | **2** | **3** | **FF**

Default: **FF**

Optional

Specifies the packet type/form to use for the IPX protocol. For example:

```
PktType = FF
```

Value	Packet Type	Adapter Type
0	Ethernet_II	Ethernet
1	Ethernet_802.3	Ethernet
2	802.2	Ethernet, Token Ring, FDDI
3	SNAP	Ethernet, Token Ring, FDDI
FF	Auto-detect	

Important This parameter is network adapter–specific. Therefore, it must only be used in an IPX parameters section that has the SpecificTo key specified. For more information, see “SpecificTo,” earlier in this chapter.

VirtualNetworkNumber

Value: *<hexadecimal number>*

Default: **0**

Optional

Specifies the internal network number for configuring the IPX protocol. For example:

`VirtualNetworkNumber = 0`

[<MS_PPTP parameters>]

This section contains parameters for the Point-to-Point Tunneling Protocol (PPTP).

NumberLineDevices

Value: *<integer value from 0 to 16384>*

Default: **5**

Optional

Specifies the number of virtual private networks to be supported by PPTP. For

example:

`NumberLineDevices = 10`

Note This parameter and its value are valid only for the Windows 2000 Server family.

[<MS_L2TP parameters>]

This section contains parameters for the Layer 2 Tunneling Protocol (L2TP).

WanEndpoints

Value: *<integer value from 0 to 30000>*

Default: **5**

Optional

Specifies the maximum number of virtual private network ports available to the L2TP.

For example:

`WanEndpoints = 3`

Note This parameter and its value are valid only for the Windows 2000 Server family.

[<MS_DLC parameters>]

Since no configuration parameters are required for Data Link Control (DLC), this section should be left empty when specifying that DLC be installed on the computer.

[<MS_AppleTalk parameters>]

This section contains parameters for configuring AppleTalk.

DefaultPort

Value: <adapter name>

Optional

Specifies the network on which the Services for Macintosh (SFM) service names are registered. If the AppleTalk protocol is not routing, only Macintosh workstations connected to this network can access the file and print services on the computer.

Normally, the default port is automatically set to the first Ethernet adapter, Token Ring adapter, or LocalTalk adapter found on the computer (in that order). For example:

```
DefaultPort = PCI\VEN_10B7&DEV_5900\1&5070
```

DefaultZone

Value: <zone name>

Optional

Specifies the default zone for the network if the network adapter is seeding the network. For example:

```
DefaultZone = Zone2
```

DesiredZone

Value: <zone name>

Optional

Specifies the zone in which the SFM service is present. If this value is not set, the SFM service for the default zone on that network is used. For example:

```
DesiredZone = Zone1
```

EnableRouter

Value: **Yes** | **No**

Default: **No**

Optional

Starts routing for the AppleTalk protocol on this computer. If routing is started, Macintosh workstations connected to any of the networks that this computer is on are able to use the file and print servers for Macintosh. For example:

```
EnableRouter = No
```

Important The remaining keys in this section are network adapter-specific. Therefore, they must only be used in an AppleTalk parameters section that has the SpecificTo key specified. For more information, see “SpecificTo,” earlier in this chapter.

NetworkRangeLowerEnd

Value: <integer from 1 to 65279>

Optional

Specifies the lower network number of the network range if the network adapter is seeding the network. For example:

NetworkRangeLowerEnd = 3

NetworkRangeUpperEnd

Value: <integer from 1 to 65279>

Optional

Specifies the upper network number of the network range if the network adapter is seeding the network. For example:

NetworkRangeUpperEnd = 33

SeedingNetwork

Value: 0 | 1

Default: 0

Optional

Used by the AppleTalk protocol during startup to determine if the computer is a seed router for a specific physical AppleTalk network segment. If this value is 0, this adapter is not seeding the network and will ignore seeding information. If this value is 1, the AppleTalk protocol reads all seeding information and seeds the network to provide the network address to clients, servers, or other routers on the AppleTalk network. For example:

SeedingNetwork = 1

Important When specifying SeedingNetwork = 0 for a computer acting as a router on an AppleTalk network, another router that is seeding the network needs to be available so that the router will start.

ZoneList

Value: <zone name>[, <zone name>, ...]

Optional

Specifies the list of zones with which to seed the network. This parameter is only valid if the network adapter is seeding the network. For example:

ZoneList = zone1, zone2

[<MS_NetBEUI parameters>]

Since no configuration parameters are required for NetBEUI, this section should be left empty when specifying that NetBEUI be installed on the computer.

[<MS_NetMon parameters>]

There are currently no settings required for this section.

[<MS_ATMLANE parameters>]

There are currently no settings required for this section.

[<MS_ATMUni parameters>]

There are currently no settings required for this section.

[<MS_ATMArps parameters>]

There are currently no settings required for this section.

[<MS_STREAMS parameters>]

There are currently no settings required for this section.

[NetClients]

This section specifies the network clients to be installed.

<network client name>

Value: <client parameters section>

Specifies the section where parameters for a network client are defined. The network client name must be included in the following table. The table describes each client and the components that are automatically installed with it. For example:

MS_MSClient = params.MS_MSClient

Client ID	Description	Component Installed
MS_MSClient	Client for Microsoft Networks, which represents MS_Browser, MS_Workstation, MS_RPC	MS_NetBIOS
MS_NWClient	Client for NetWare Networks	

[<MS_MSClient parameters>]

This section contains parameters for the Client for Microsoft Networks client.

BrowseDomains

Value: <domain name>[, <domain name>[, ...]]

Optional

Valid for Windows 2000 Server only. Specifies a list of domains that can be browsed by the computer. For example:

```
BrowseDomains = Sales, Research
```

NameServiceNetworkAddress

Value: *<IP address>*

Optional

Specifies the network address of the name service provider. This value is only used if NameServiceProtocol = ncacn_ip_tcp. For example:

```
NameServiceNetworkAddress = 193.1.1.1
```

NameServiceProtocol

Value: **ncacn_np** | **ncacn_ip_tcp**

Optional

Specifies the protocols used by the name service. For example:

```
NameServiceProtocol = ncacn_np
```

[<MS_NWClient parameters>]

This section contains parameters for the Client Service for NetWare client. The settings are applied to the default user account on the computer.

DefaultTree

Value: *<tree name>*

Optional

Specifies the name of the preferred tree. For example:

```
DefaultTree = NDSSERVER
```

DefaultContext

Value: *<NetWare context name>*

Optional

Specifies default logon context. For example:

```
DefaultContext = SALES.NWSERVER1
```

LogonScript

Value: **Yes** | **No**

Optional

Specifies whether to execute a logon script. For example:

```
LogonScript = No
```

PreferredServer

Value: *<server name>*

Optional

Specifies the name of the preferred server. This key and the DefaultTree and DefaultContext keys are mutually exclusive. For example:

```
PreferredServer = NWSERVER1
```

[NetServices]

This section contains the network services to be installed.

<network service name>

Value: <service parameters section>

Specifies the section where parameters for a network service are defined. For example:

```
MS_Server = params.MS_Server
```

The network service name must be selected from the Service ID column in the table below. The table describes each client and the components that are automatically installed with it.

Service ID	Description
MS_Server	File and Print Services
MS_RasSrv	Dial-Up Server
MS_PSched	QoS Packet Scheduler
MS_NwSapAgent	SAP Agent
MS_WLBS	Windows Load Balancing Service

[<MS_Server parameters>]

This section contains parameters for the File and Print Service.

BroadcastsToLanman2Clients

Value: **Yes** | **No**

Default: **No**

Optional

Valid for Windows 2000 Server only. Specifies whether the Windows 2000 Server service makes browser broadcasts to LAN Manager 2.x clients. For example:

```
BroadcastsToLanman2Clients = Yes
```

Optimization

Value: **MinMemoryUsed** | **Balance** | **MaxThroughputForFileSharing** |

MaxThroughputForNetworkApps

Default: **MaxThroughputForFileSharing**

Optional

Valid for Windows 2000 Server only. Controls the optimization method used by the Windows 2000 Server. For example:

`Optimization = Balance`

[<MS_RasSrv parameters>]

This section contains parameters for the Routing and Remote Access Service for RAS servers.

ParamsSection

Value: <RAS parameters section>

Specifies a section that contains Routing and Remote Access Service parameters as listed in the <MS_RAS parameters> section below. For example:

`ParamsSection = params.MS_RAS`

Important The extra level of indirection is required.

[<MS_RAS parameters>]

This section contains parameters for Routing and Remote Access Service.

Important The parameter RouterType is required for all [<MS_RAS parameters>] to take effect. For more information, see "RouterType," later in this document.

AssignSameNetworkNumber

Value: **Yes** | **No**

Default: **Yes**

Optional

Specifies whether the same network number should be assigned to all clients. For example:

`AssignSameNetworkNumber = No`

AutomaticNetworkNumbers

Value: **Yes** | **No**

Default: **Yes**

Optional

Specifies whether network numbers should be automatically allocated. For example:

```
AutomaticNetworkNumbers = No
```

If the value of AutomaticNetworkNumbers is No, then the NetworkNumberFrom key must be specified.

ClientCanRequestIPAddress

Value: **Yes** | **No**

Default: **No**

Optional

Specifies whether a TCP/IP client can request a specific IP address. For example:

```
ClientCanRequestIPAddress = Yes
```

ClientsCanRequestIpxNodeNumber

Value: **Yes** | **No**

Default: **No**

Optional

Specifies whether a client can request an IPX node number. For example:

```
ClientsCanRequestIpxNodeNumber = Yes
```

DialinProtocols

Value: **All** | **TCP/IP** | **IPX** | **NetBEUI** | **AppleTalk**

Specifies the dial-in protocols to support. All indicates all installed protocols on the computer. Multiple protocols can be specified, but they must be comma-separated as shown in the following example:

```
DialinProtocols = IPX,TCP/IP
```

IpAddressEnd

Value: *<IP address>*

Specifies the ending IP address to use when assigning addresses to clients. For example:

```
IpAddressEnd = 193.1.1.200
```

Note At this time, only one range of IP addresses can be added during the unattend Setup.

IpAddressStart

Value: *<IP address>*

Specifies the beginning IP address to use when assigning addresses to clients. For example:

```
IpAddressStart = 193.1.1.100
```

Note At this time, only one range of IP addresses can be added during the unattend Setup.

IPXClientAccess

Value: **Network** | **ThisComputer**

Default: **Network**

Optional

Specifies the access scope of an IPX client. This key should be present only if one of the values of the key DialinProtocols is IPX. For example:

```
IPXClientAccess = ThisComputer
```

Multilink

Value: **Yes** | **No**

Default: **Yes**

Optional

Specifies whether multilinking of modems should be enabled for increased bandwidth. This parameter is only valid if multiple modems are installed. For example:

```
Multilink = No
```

NetBEUIClientAccess

Value: **Network** | **ThisComputer**

Default: **Network**

Optional

Specifies the access scope of a NetBEUI client. This key should be present only if one of the values of the key DialinProtocols is NetBEUI. For example:

```
NetBEUIClientAccess = ThisComputer
```

NetworkNumberFrom

Value: *<0x00000001 to 0xFFFFFFFF>*

Specifies the start address of allowed IPX network numbers. For example:

```
NetworkNumberFrom = 0x000005ab
```

RouterType

Value: **1** | **2** | **3** | **6** | **7**

Required

Specifies how to configure the Routing and Remote Access service. For example:

```
RouterType = 3
```

where:

- 1 Enable remote access
- 2 Enable local routing only (LAN-only router)

- 3 Enable remote access and local routing
- 6 Enable local/remote routing (LAN and WAN routers)
- 7 Enable remote access and local/remote routing

Important

- This parameter is required for all [*<MS_RAS parameters>*] to take effect.
 - This parameter and its value are valid only for the Windows 2000 Server family.
-

TcpIpClientAccess

Value: **Network** | **ThisComputer**

Default: **Network**

Optional

Specifies the access scope of a TCP/IP client. This key should be present only if one of the values of the key DialinProtocols is TCP/IP. For example:

`TcpIpClientAccess = ThisComputer`

UseDHCP

Value: **Yes** | **No**

Default: **Yes**

Optional

Specifies whether DHCP should be used to assign IP addresses to TCP/IP clients. For example:

`UseDHCP = Yes`

Important If UseDHCP = No, then IpAddressStart and IpAddressEnd must be specified.

[<MS_Psched parameters>]

This section contains parameters for the Quality of Service (QoS) Packet Scheduler service. At this time, there are no parameters required for this service.

[<MS_NwSapAgent parameters>]

This section contains parameters for the NetWare SAP Agent service in the future. There are currently no settings required for this service.

[<MS_WLBS parameters>]

This section contains parameters for configuring the Windows Load Balancing service (WLBS). Only one binding to a network adapter can be enabled at a time. If unique binding is not specified in the NetBindings section, the install process will pick a random binding.

Note

- [<MS_WLBS parameters>] are *only* available for Windows 2000 Advanced Server and Windows 2000 Datacenter Server.
- By default, all networking components *except* WLBS are installed with all of their bindings enabled. These can be selectively disabled using the Disable command in the [NetBindings] section. In addition, the Enable command is supported in the [NetBindings] section for explicit binding enabling.

WLBS is installed with all of its bindings disabled. You must then explicitly enable a binding to a SINGLE adapter via the Enable command in the [NetBindings] section. For example:

```
[Networking]

[NetServices]
MS_WLBS=params.MS_WLBS

[params.MS_WLBS]
HostPriority           = 1
ClusterModeOnStart   = 1
ClusterIPAddress      = 172.31.240.176
ClusterNetworkMask    = 255.255.248.0
DedicatedIPAddress    = 172.31.240.165
DedicatedNetworkMask  = 255.255.248.0
ClusterName           = cluster.microsoft.com
MulticastSupportEnable = 1
RemoteControlPassword =
RemoteControlEnabled  = 1
Ports                 = 80,80,Both,Multiple,None,Equal,443,443,
                       Both,Multiple,Single,Equal

[params.MS_TCPIP]
```

For more information about this parameter, see “<MS_TCPIP parameter>,” earlier in this chapter.

```
[NetBindings]
Enable = MS_WLBS,A1
Disable = MS_WLBS,A2

[NetAdapters]
A1 = params.A1
A2 = params.A2

[params.A1]
NetCardAddress = 0x2bfc01f388d

[params.A2]
NetCardAddress = 0x2bfc01f3890
```


AliveMsgPeriod

Value: *<integer between 100 and 10,000>*

Default: **1000**

Optional

Specifies a period between sending WLBS cluster heartbeat messages in milliseconds.

For example:

```
AliveMsgPeriod = 5000
```

AliveMsgTolerance

Value: *<integer between 5 and 100>*

Default: **5**

Optional

Specifies number of heartbeat messages that can be lost before WLBS cluster host is considered dead and convergence is initiated. For example:

```
AliveMsgTolerance = 50
```

ClusterIPAddress

Value: *<IP address>*

Default: **0.0.0.0**

Specifies the WLBS cluster's primary IP address. This address is a virtual IP address and it must be set identically for all hosts in the WLBS cluster. If you use one alias for several IP addresses to the WLBS cluster, then only the primary (main) IP address needs to be specified. For example:

```
ClusterIPAddress = 10.192.45.7
```

ClusterModeOnStart

Value: **0** | **1**

Default: **1**

Optional

Specifies whether a host should join the WLBS cluster upon startup or wait until given an explicit command to do so. For example:

```
ClusterModeOnStart = 0
```

ClusterName

Value: *<Fully qualified domain name of the cluster>*

Default: **cluster.domain.com**

Specifies a full Internet name for the WLBS cluster. For example:

```
ClusterName = cluster.microsoft.com
```

ClusterNetworkMask

Value: *<network mask>*

Default: **0.0.0.0**

Specifies the subnet mask for the WLBS cluster IP address. For example:
`ClusterNetworkMask = 255.255.255.0`

DedicatedIPAddress

Value: *<IP address>*
Default: **0.0.0.0**
Optional
Specifies host's unique IP address. For example:
`DedicatedIPAddress = 10.192.45.1`

DedicatedNetworkMask

Value: *<network mask>*
Default: **0.0.0.0**
Optional
Specifies the subnet mask for the dedicated IP address. For example:
`DedicatedNetworkMask = 255.255.255.0`

DescriptorsPerAlloc

Value: *<integer between 16 and 1024>*
Default: **512**
Optional
Specifies number of connection descriptors that are created per each allocation. Change this value only when instructed to do so by an event log message. For example:
`DescriptorsPerAlloc = 1024`

HostPriority

Value: *<integer between 1 and 32>*
Default: **1**
Specifies host's unique priority ID in the WLBS cluster. For example:
`HostPriority = 5`

MaskSourceMAC

Value: **0 | 1**
Default: **1**
Optional
Specifies whether source Media Access Control (MAC) address masking should be used, allowing WLBS cluster hosts to be connected to different switch ports. Set the value to 0 if WLBS cluster is connected to a hub to re-enable learning in the upstream switch and conserve its bandwidth. For example:
`MaskSourceMAC = 0`

MaxDescriptorAllocs

Value: *<integer between 1 and 1024>*

Default: **512**

Optional

Specifies maximum number of connection descriptor allocations. Change this value only when instructed to do so by an event log message. For example:

```
MaxDescriptorAllocs = 1024
```

MulticastSupportEnable

Value: **0** | **1**

Default: **0**

Optional

Specifies whether a multicast MAC address should be used for WLBS cluster operations. For example:

```
MulticastSupportEnable = 1
```

NetmonAliveMsgs

Value: **0** | **1**

Default: **0**

Specifies if WLBS forwards heartbeat messages up to the bound protocols, allowing heartbeats to be captured by the Network Monitor utility. For example:

```
NetmonAliveMsgs = 0
```

NumActions

Value: *<integer between 5 and 500>*

Default: **50**

Optional

Specifies number of actions that are created per each allocation. Change this value only when instructed to do so by an event log message. For example:

```
NumActions = 100
```

NumAliveMsgs

Value: *<integer between 66 and 660>*

Default: **66**

Optional

Specifies number of heartbeat packet buffers that are created per each allocation.

Change this value only when instructed to do so by an event log message. For example:

```
NumAliveMsgs = 100
```

NumPackets

Value: *<integer between 5 and 500>*

Default: **100**

Optional

Specifies number of packets that are created per each allocation. Change this value only when instructed to do so by an event log message. For example:

```
NumPackets = 100
```

Ports

Value: *<integer from 1 to 65535>*,*<integer from 1 to 65535>*,

<Both | TCP | UDP>,*<Multiple | Single | Disabled>*,

<mode parameter>,[...]

Mode Parameter for Multiple: *<None | Single | ClassC>*,

<Equal | integer from 1 to 100>

Mode Parameter for Single: *<integer from 1 to 32>*

Mode Parameter for Disabled: None

Default: **1,65535,Both,Multiple,Single,Equal**

Optional

Specifies rules for distributing traffic among WLBS cluster hosts according to protocol and destination ports. For example:

```
Ports = 20,21,TCP,Multiple,Single,Equal,80,80,Both,Multiple,  
None,Equal,443,443,Both,Multiple,Single,Equal,1024,65535,TCP,  
Multiple,Single,Equal
```

RemoteControlEnabled

Value: **0** | **1**

Default: **0**

Optional

Specifies whether remote-control operations are enabled. For security reasons, it is vital that the WLBS UDP control port be placed behind a firewall. For example:

```
RemoteControlEnabled = 1
```

RemoteControlPassword

Value: *<password>*

Default: **NULL**

Optional

Specifies a password to be used for restricting access to the WLBS cluster from remote, networked computers running Windows 2000, using the Wlbs.exe control program. For example:

```
RemoteControlPassword = BigCluster
```

RemoteControlUDPPort

Value: *<integer between 1 and 65,535>*

Default: **2504**

Optional

Specifies a list of addresses to use when searching for the DNS server on the network.
For example:

```
RemoteControlUDPPort = 1717
```

[NetBindings]

This section contains information about how to disable, enable, and demote binding rules for network components such as services, protocols, and adapters. Each key parameter can be listed more than once in this section. The network components that make up the binding order/path must be listed in the following order: <network service>, <network protocol>, <network adapter>

The binding order/path must always start with a client/service (if one exists) and end with an adapter. Incorrectly specified binding orders/paths are ignored.

Disable

Value: <network component>[, <network component>[, ...]]

Optional

Disables bindings between network components in the specified path. Examples:

```
Disable = Adapter1 ; wrong
Disable = MS_Server, MS_NetBEUI, Adapter2 ; correct
Disable = Adapter2, MS_TCPIP ; wrong
```

Enable

Value: <network component>[, <network component>[, ...]]

Optional

Enables bindings between components in the specified path. If a single component is specified as the path, all bindings to it are enabled. For example:

```
Enable = MS_Server, MS_NetBT, MS_TCPIP, Adapter1
```

Demote

Value: <network component>[, <network component>[, ...]]

Optional

Demotes the specified binding path/order to the bottom of the list of binding paths that start with the first specified <network component>. For example:

```
Demote = MS_Server, MS_NetBT, MS_TCPIP, Adapter1
```

[NetOptionalComponents]

This section contains a list of the optional network components to install.

Note Each component listed in this section uses the same values—1 or 0—where:

- 1 Installs the component during the unattended installation.
- 0 Ignores the component during the unattended installation.

ACS

Value: **1 | 0**

Specifies that QoS Admission Control Service be installed on the computer.

Note This parameter and its value are valid only for the Windows 2000 Server family.

DHCPServer

Value: **1 | 0**

Specifies that DHCP Server be installed on the computer.

Note This parameter and its value are valid only for the Windows 2000 Server family.

DNS

Value: **1 | 0**

Specifies that DNS be installed on the computer.

Note This parameter and its value are valid only for the Windows 2000 Server family.

IAS

Value: **1 | 0**

Installs the Internet Authentication Services (IAS) and creates a shortcut to the IAS MMC snap-in under the Administrative Tools section of the Start menu.

Note This parameter and its value are valid only for the Windows 2000 Server family.

ILS

Value: **1 | 0**

Specifies that the Internet Location Service (ILS) be installed on the computer.

Important

- This parameter requires that iis_common, iis_inetmgr, iis_www, and com are all installed.
 - This parameter and its value are valid only for the Windows 2000 Server family.
-

LPDSVC

Value: **1 | 0**

Specifies that Print Services for UNIX be installed on the computer.

MacPrint

Value: **1 | 0**

Specifies that Print Services for Macintosh be installed on the computer.

Note This parameter and its value are valid only for the Windows 2000 Server family.

MacSrv

Value: **1 | 0**

Specifies that Services for Macintosh (SFM) be installed on the computer.

Note This parameter and its value are valid only for the Windows 2000 Server family.

Netcm

Value: **1 | 0**

Installs the Microsoft Connection Manager Administration Kit and Phone Book Service.

Note

- This parameter and its value are valid only for the Windows 2000 Server family.
 - This parameter requires that iis_common, iis_ftp, iis_inetmgr, and iis_www are all installed.
-

NETMONTTOOLS

Value: **1 | 0**

Specifies that Network Monitor tools be installed on the computer.

Note This parameter and its value are valid only for the Windows 2000 Server family.

SimpTcp

Value: **1 | 0**

Specifies that Simple TCP/IP Services be installed on the computer.

SNMP

Value: **1 | 0**

Specifies that SNMP be installed on the computer.

WINS

Value: **1 | 0**

Specifies that WINS be installed on the computer.

Note This parameter and its value are valid only for the Windows 2000 Server family.

[SNMP]

Accept_CommunityName

Value: *<community name 1>:<privilege>, ...*

Default: **public:Read_Only**

Defines the community names from which the computer running the SNMP service can accept traps. Each community name must have a privilege associated with it. Valid privileges are: **None**, **Notify**, **Read_Only**, **Read_Write**, and **Read_Create**.

If no privilege is assigned to a community name, the default of Read_Only is applied.

Commas separate community names.

Any_Host

Value: **Yes** | **No**

Specifies whether the computer on which the SNMP service is being installed should accept SNMP packets from any host or not.

Community_Name

Value: *<community name>*

Indicates the *<community name>* for the computer.

Contact_Name

Value: *<name>*

Specifies the contact name for this managed node, as well as information about how to contact this person.

Limit_Host

Values: *<host names>*

Specifies a maximum of three *<host names>* separated by commas. This key is valid when Any_Host = No.

Location

Value: *<computer location>*

Specifies the physical location of the computer.

Send_Authentication

Value: **Yes** | **No**

Indicates whether an authentication trap should be sent when an unauthorized community or host requests information.

Service

Values: **Physical, Applications, Datalink, Internet, End-to-End**

Default: **Applications, Internet, End-to-End**

Any combination of the five SNMP services listed here can be specified as values. Commas must separate these values.

Traps

Values: *<IP addresses>* | *<IPX addresses>*

Specifies a maximum of three IP or IPX addresses to which traps should be sent.

Note For the values specified in the Traps parameter to take effect, a valid value must be specified in the Community_Name parameter. For more information, see "Community_Name," earlier in this document.

[InternetServer]

For this section to be processed, IIS must be turned on in the [Components] section.

Note At this time, to change the default installation paths for the FTP and WWW roots, you must include the [InternetServer] parameters in the Unattend.txt file.

PathFTPRoot

Value: *<path to FTProot>*

Specifies the installation folder for the FTP service. This parameter is only useful if the FTP service is installed. The default path is %systemdrive%\Inetpub\Ftproot.

For example:

```
PathFTPRoot = "C:\Inetpub\Ftproot"
```

PathWWWRoot

Value: *<path to wwwroot>*

Specifies the installation folder for the WWW service. This parameter is only useful if the WWW service is installed. The default path is %systemdrive%\Inetpub\Wwwroot. For example:

```
PathWWWRoot = "C:\Inetpub\Wwwroot"
```

[Cluster]

This section contains parameters for the Cluster service component. This section need not be included in the Unattend.txt file to install the Cluster service files to the computer. Enabling cluster in the [Components] section of the answer file is sufficient to install the necessary files. However, to make sure that the Cluster service is installed completely during unattended Setup, an answer file can be created and run after Setup has been completed and a user has logged on to the system.

To perform a full cluster installation

1. In the [Components] section of the Unattend.txt file, set

```
cluster = on
```
2. In the [Cluster] section, add all parameters.
3. In the [GuiRunOnce] section of the Unattend.txt file, add

```
"%windir%\cluster\cluscfg.exe -UNATTEND"
```

Any parameters specified are saved on the computer and are only processed after rebooting to configure the Cluster service. They are not processed during unattended Setup mode.

For more information about configuring the Cluster service and Clustering requirements, see the *Microsoft Windows 2000 Server Resource Kit*.

Account

Value: *<account name>*

Specifies the name of the account under which Cluster service runs. This key is required only if Action = Form. For example:

```
Account = adminname
```

Action

Values: **Form** | **Join**

Specifies whether a cluster is to be formed or joined.

where:

- **Form** specifies that the cluster is to be created. If this is the first node in a cluster, you are creating a new cluster. When selected, the Account and Domain keys must be specified.
- **Join** specifies that cluster is to be joined. If at least one other node already exists, you are joining a cluster. When selected, the Account and Domain keys should not be specified.

For example:

```
Action = Form
```

Domain

Value: *<domain name>*

Specifies the domain to which the cluster belongs. This key is required only if Action = Form. For example:

```
Domain = domainname
```

ExcludeDrive

Value: *<drive letter>*[, *<drive letter>* [, ...]]

Optional

Specifies a drive to be excluded from the list of possible quorum devices. For example:

```
ExcludeDrive = q, r
```

IPAddr

Value: *<IP address>*

Specifies the IP address of the cluster. For example:

```
IPAddr = 193.1.1.95
```

LocalQuorum

Value: **Yes** | **No**

Optional

Normally, only disks that are on a shared SCSI bus not used by the system disk can be selected as the quorum device. This key specifies that a system drive should be used as the quorum device. For example:

```
LocalQuorum = Yes
```

Note This parameter should only be used for demo, testing, and development purposes. The local quorum resource cannot failover.

Name

Value: *<cluster name>*

Specifies the name of the cluster. The value can contain a maximum of 15 characters.

For example:

```
Name = MyCluster
```

Network

Value: *<connection name string>*, *<role>*[, *<priority>*]

Specifies the connection name associated with a network adapter and the role that adapter is to fulfill in the cluster. The first two parameters, *<connection name string>* and *<role>*, are required. The third parameter, *<priority>*, should be supplied only for network connections configured for internal communications.

The *<role>* parameter specifies the type of cluster communication for the network connection. Valid parameters are All | Internal | Client. To use the network connections for communication with clients and between the nodes, select All. To use the network connections only for internal communication between the nodes, select Internal. To use the network connections only for communication with clients, select Client.

The *<priority>* parameter specifies the order in which the network connections are used for internal communication.

For example:

```
Network="Local Area Connection 2", INTERNAL, 1
```

Password

Value: *<password>*

Specifies the password of the account under which the Cluster service runs. For example:

```
Password = MyPassword
```

Note There are some security concerns about using this key, since the password is stored as plain text within the answer file. However, the password key is deleted after the upgrade.

Quorum

Value: *<drive letter>*

Specifies the drive to be used as the quorum device. For example:

```
Quorum = Q:
```

Subnet

Value: *<IP subnet mask>*

Specifies the IP subnet mask of the cluster. For example:

```
Subnet = 255.255.0.0
```

[TerminalServices]

This section contains parameters for the Windows Terminal Services component. This section is needed if TSEnable = On in the [Components] section of the Unattend.txt file to ensure that the installation of Windows 2000 with Terminal Services is fully automated.

For more information about configuring Windows Terminal Services and about Windows Terminal Services requirements, see the *Microsoft Windows 2000 Server Resource Kit*.

ApplicationServer

Value: **0** | **1**

Terminal Services offers two modes: Application Server and Remote Administration. By setting ApplicationServer = 1, you are specifying that the Terminal Server Services will function as an Application Server. By setting ApplicationServer = 0, you are specifying that the Terminal Services will function as a Remote Administration server.

Important Specifying ApplicationServer = 1 for [TerminalServices] is mutually exclusive with the Cluster service. The server can use *either* Cluster service support *or* Terminal Services with Application Server support enabled.

If you specify ApplicationServer = 1 for [TerminalServices] and Cluster = On in [Components], Windows 2000 automatically defaults to the Cluster service and Remote

Administration for Terminal Services instead of enabling Terminal Services as an Application Server.

If you specify `ApplicationServer = 0` for `[TerminalServices]`, you can enable the Cluster service, because Terminal Services with Remote Administration is not mutually exclusive with the Cluster service.

PermissionsSetting

Value: **0** | **1**

Default: **1**

This parameter allows administrators to choose the user permissions required to run applications using Terminal Services. To run some non-Windows 2000-certified applications, Windows 2000 must grant write access to certain parts of HKLM and to certain system directories. However, granting these additional permissions may allow users to delete or change vital settings under HKLM and/or system files.

The default choice is to use Terminal Server 4.0-compatible security, which allows non-Windows 2000-certified applications to run with the same level of compatibility as in Terminal Server 4.0. This allows users to continue running previous versions of applications that may require additional user permissions.

If the applications in your environment are Windows 2000 certified or do not require additional permissions for the users to run the applications, you can choose to use Windows 2000-compatible security.

where:

- **0** grants Terminal Services users the same permissions as Windows 2000 users.
- **1** grants Terminal Services users the same permissions as Terminal Server 4.0 users.

[Certsrv_Client]

This section contains the parameters for the Web client component of the Certificate Service. This section need not be included in the Unattend.txt file to install the Web client component of Certificate Services files to the computer. Enabling `certsrv_client` service in the `[Components]` section of the answer file is sufficient to install the necessary files.

Any parameters specified are saved on the computer and are only processed after rebooting to configure the Web client Certificate Services. They are not processed during unattended Setup mode. For more information about configuring Web client Certificate Services and about Web client Certificate Services requirements, see the *Microsoft Windows 2000 Server Resource Kit*.

CAMachine

Value: `<certificate authority computer name>`

Specifies the full DNS name of the computer with an installed Certification Authority. The value of this parameter is required. The computer must be running and connected to the network.

CAName

Value: <certificate authority name>

Specifies the name of the Certification Authority.

The value of this parameter is required. Certificate Services on the computer must be running. Certificate Services can be called through DCOM from the current computer.

Note When installing only the Web client component, CAMachine and CAName refer to the Certification Authority that the Web enrollment pages work with. When installing the server components, these two parameters are the attributes for the Certification Authority.

[CertSrv_Server]

This section contains the parameters for the server components of the Certificate Services. The attributes that are required depend on the installation type and configuration. The "Name" parameter is required in all cases.

Any parameters specified are saved on the computer and are only processed after rebooting to configure Certificate Services. They are not processed during Setup. For more information about configuring Certificate Services and Certificate Services requirements, see the *Microsoft Windows 2000 Server Resource Kit*.

CAType

Value: **EnterpriseRoot** | **EnterpriseSubordinate** | **StandaloneRoot** | **StandaloneSubordinate**

Default: **Determined Programmatically**

For example:

```
If the Active Directory is available and writable then:
    If any CA is in the Active Directory then
        Default is Enterprise Subordinate CA
    Else
        Default is Enterprise Root CA
Else
    Default is Standalone Root CA
```

Specifies the type of Certification Authority to be installed. This parameter requires a value.

Country

Value: <certificate authority country code>

Default: **determined using GetLocaleInfo**

Optional

Specifies the country code for the Certification Authority being installed. This value is not case-sensitive and it must contain a maximum of two characters: A-Z (or a-z).

For more information, search on the Internet for "ISO 3166" for a list of valid country codes, or visit the International Telecommunication Union Survey Web site at:

<http://www.itu.int/net/cctlds/index.html>

Note Web addresses can change, so you may be unable to connect to the Web site mentioned here.

CSPProvider

Value: <CSP name>

Default: "Microsoft Base Cryptographic Provider v1.0"

Specifies the name of the Cryptography Service Provider (CSP). The value is case-sensitive.

Description

Value: <description of certificate authority>

Optional

Specifies a comment or description of the Certification Authority being installed. The value is case-sensitive and it can contain a maximum of 2048 characters.

Email

Value: <certificate authority e-mail address>

Optional

Specifies the e-mail address for the Certification Authority being installed. The value is case-sensitive and it can contain a maximum of 128 characters.

ExistingKey

Value: <name>

Specifies the name of an existing key to be used by the Certification Authority.

HashAlgorithm

Value: <hash algorithm string or algorithm ID>

Default: "SHA1"

Specifies the hash algorithm used by the Certification Authority to sign certificates. The specified CSP must support the algorithm. The value is not case-sensitive.

KeyLength

Value: <key length>

Specifies the key length for the Certification Authority. The default key length for the CSP is used if no value is specified.

Locality

Value: <certificate authority locality>

Specifies the locality of the Certification Authority being installed. The value is required, is case-sensitive, and can contain a maximum of 128 characters.

Name

Value: <certification authority name>

Specifies the name of the Certification Authority being installed. The value is required, is case-sensitive, and can contain a maximum of 64 characters.

Note No default value for Name is provided. Because this is a required field, if Certificate Service is being installed through a fully unattended Setup mode and if the Name value is not provided in the Unattend.txt file, then Setup fails with E_INVALIDARG.

Organization

Value: <certificate authority organization name>

Specifies the organization unit name of the Certification Authority being installed. The value is required, is case-sensitive, and can contain a maximum of 64 characters.

OrganizationUnit

Value: <certificate authority organization unit name>

Specifies the organization unit name of the Certification Authority being installed. The value is required, is case-sensitive, and can contain a maximum of 32 characters.

ParentCAMachine

Value: <parent computer name for subordinate CA>

Optional

Specifies the computer name that has a CA that works as a parent CA with the current subordinate CA installation. The value is ignored if the current CA installation type is not subordinate CA. The value is not case-sensitive. If the attribute is not defined when setting up a subordinate CA, Setup saves the CA certificate request to a file. This parameter is used in combination with ParentCAName.

For more information about certificate requests, see "RequestFile," later in this document.

ParentCAName

Value: <parent CA name for subordinate CA>

Optional

Specifies the parent CA name for the current subordinate CA installation. The value is ignored if CA type is not subordinate CA. The value is not case-sensitive. If the attribute is not defined but ParentCAMachine is defined when setting up a subordinate CA, Setup calls Certificate Services on the parent computer to get the CA name.

PreserveDB

Value: **Yes** | **No**

Default: **Yes**

Used together with ExistingKey and UseExistingCert. If existing key and CA certificate are not used, this parameter is ignored.

RequestFile

Value: <request file name/path>

Default: *The file path is always in the format:*

\$\$SharedFolder\$\$MachineName\$_\$CAName\$.req.

For more information about this default setting, see "SharedFolder," later in this document.

Specifies the name (full path) of the file into which certificate requests should be saved. This parameter is ignored if the CAType is a root-type CA (EnterpriseRoot or StandaloneRoot).

SharedFolder

Value: *<path to folder>*

Specifies the path to the folder containing the configuration information for the Certification Authority. The default value is the registered shared folder path, if it exists. If a registered shared folder does not exist, the value is in the format of %systemdrive%\CAConfig.

State

Value: *<certificate authority state/province>*

Specifies the state/province for the Certification Authority being installed. The value is required, is case-sensitive, and can contain a maximum of 128 characters. This value should be the entire name rather than an abbreviation. For example, use "Washington" rather than "WA."

UseExistingCert

Value: **Yes** | **No**

Default: **Yes**

Used together with ExistingKey. If Setup finds a matched CA certificate for the existing key, then the matching certificate is used.

ValidityPeriod

Value: *<numeric value>*

Default: **2**

Specifies the validity period for the Certification Authority being installed. This key works in conjunction with the ValidityPeriodUnits parameter. The value must be greater than 0 and less than or equal to 1000.

ValidityPeriodUnits

Value: **Years** | **Months** | **Weeks** | **Days**

Default: **Years**

Specifies the units for validity period for the Certification Authority being installed.

[RemoteInstall]

This section is only used by clients being installed using the Remote Installation Service. The RIS administration tools generate these parameters.

Repartition

Value: **Yes** | **No**

Specifies whether or not all partitions on the first drive on the client computer should be deleted and reformatted with NTFS. If the value is set to No, the default parameters in the client answer file are used.

Note You cannot run unattended Setup on a disk that does not contain a file system. Make sure that your destination disk is formatted with a file system.

UseWholeDisk

Value: **Yes** | **No**

Default: **Yes**

Valid only for Riprep images. Setting this value to **NO** stops the Setup program from extending the volume to the end of the drive.

[OsChooser]

This section is only used by clients being installed using the Remote Installation Service. In general, the RIS administration tools generate this parameter.

Description

Value: *<string>*

Specifies the description of the operating system image. This text appears on the client computer within the Client Installation wizard (CIW).

Help

Value: *<string>*

When users select the operating system description within the CIW, the Help topic is displayed. The Help topic provides a more detailed formal description of the operating system image.

ImageType

Value: **Flat** | **Sysprep**

Specifies the type of operating system image being installed. RIS supports either a flat CD-based image or a Sysprep image created using the Remote Installer Preparation (RIPrep) wizard.

Important There is a known issue when using Sysprep or Riprep to create system images, and then installing a 16-bit application on the destination computer.

If you are using Windows 2000 Gold (pre-Service Pack 1), see the KB article, "Q248257 - Program Installation Problems on Sysprep or Riprep Installed Systems," for more information about this issue and possible resolutions, and how to obtain an important hotfix for Winlogon.

If you are using Windows 2000 Service Pack 1 or later, this fix is already included in the service pack.

LaunchFile

Value: *<path/file name>*

Specifies the file to be executed when the user selects an operating system image from within the CIW. This path is automatically defined when using RIS to install an

operating system on a client computer from within the CIW. Use this option to execute pre-boot maintenance and troubleshooting tools for use with RIS. This option accepts either a literal path or a relative path to the tool binary being executed as follows:

Literal Path:

```
Setup\English\Tools\\i386\Vendortool.bin
```

Relative Path:

```
"%Installpath%\%MachineType%\Vendortool.bin"
```

Version

Value: *<string>*

Specifies the version of the operating system or tool being executed when the user selects an image from within the CIW.

[DCInstall]

This optional section contains parameters for installing a domain controller after the initial Setup of Windows 2000 is complete. The parameters in this section are only supported for the Windows 2000 Server family.

An answer file can be created containing only this section, and it can be run after Setup has been completed and a user has logged on to the system. The command line for this scenario is **dcpromo/answer:***<answer file>*

AdministratorPassword

Value: *<admin password>*

Default: **blank password**

Sets the local administrator password for the computer during the demotion of a domain controller to a member server. This parameter is only valid during a demotion. If it is not specified or has no value, a blank administrator password is used.

Note The value is deleted from the answer file after the demotion operation is completed.

AllowAnonymousAccess

Value: **Yes** | **No**

Default: **Yes**

Used when any pre-Windows 2000 servers will be authenticating users from this domain or any trusting domain.

where:

- **Yes** is used with pre-Windows 2000 servers. Causes the permissions to be set to permit anonymous access to user and group information.
- **No** is used when all servers in the domain are running a Windows 2000 Server product. Makes the default permissions more restrictive.

AutoConfigDNS

Value: **Yes** | **No**

Default: **Yes**

Used to indicate whether the wizard should configure DNS for the new domain if it has detected that dynamic DNS updates are not available.

ChildName

Value: *<child domain name>*

Specifies the DNS label to be appended at the beginning of the name of an existing directory service domain when a child domain is being installed. For example, if the parent name is "parentdom.mydomain.com" and the ChildName is "childdom," then the name of the new domain is "childdom.parentdom.mydomain.com."

For more information, see "TreeOrChild," later in this document.

Note This new domain name must not be in use and DNS services must be properly configured on the computer. This value must be specified.

CreateOrJoin

Value: **Create** | **Join**

Default: **Join**

Specifies that the new tree domain created is part of an existing forest of domains or causes the creation of a new forest of domains.

where:

- **Join** places the new domain as the root of a new domain tree in an existing forest of domains.
- **Create** creates a new forest of domains.

CriticalReplicationOnly

Value: **Yes** | **No**

Default: **None**

Used to specify that the promotion operation should perform only critical replication, then continue, skipping the non-critical portion, which can be very lengthy. Set to **Yes** to skip non-critical replication.

In the case that non-critical replication is skipped, that replication will automatically and silently resume when the computer is rebooted and assumes its new role as a domain controller.)

DatabasePath

Value: *<path to database files>*

Default: "%systemroot%\NTDS"

Specifies the fully qualified, non-UNC path to a directory on a fixed disk of the local computer that contains the domain database. If the directory exists, it must be empty. If the directory does not exist, it is created.

The disk must have enough free disk space available—at least 20 MB for new domains—and must have room to grow if you plan to add numerous objects to the domain. For replica domains, the space required is a function of domain size.

Note For optimal performance, the domain database should be placed on a different volume than the domain log files.

DNSSOnNetwork

Value: **Yes** | **No**

Default: **Yes**

Used when a new forest of domains is being installed and no DNS client is configured on the computer. No skips DNS client configuration, and DNS auto-configuration for the new domain is created. Yes allows the DNS client to be configured and auto-configuration of DNS to be offered.

DomainNetBiosName

Value: <domain NetBIOS name>

Assigns a NetBIOS name to the new domain. The value is required, and the name specified must not already be in use as a domain or computer name.

IsLastDCInDomain

Value: **Yes** | **No**

Default: **No**

Indicates whether the computer on which depromo.exe is running is the last domain controller in the domain.

Note This parameter is only valid when demoting an existing domain controller to a member server.

LogPath

Value: <path to log files>

Default: "%systemroot%\NTDS"

Specifies the fully qualified, non-UNC path to a directory on a fixed disk of the local computer that contains the domain log files. If the directory exists, it must be empty. If the directory does not exist, it is created.

The disk must have enough free disk space available—at least 10 MB for new domains—and must have room to expand if you plan to add numerous objects to the domain. For replica domains, the space required is a function of domain size.

Note For optimal performance, the log files should be placed on a different volume than the database files.

NewDomainDNSName

Value: <DNS name of domain>

Specifies the required name of a new tree in an existing domain or when a new forest of domains is being installed. For example, this DNS name could be "newdom.mydomain.com."

ParentDomainDNSName

Value: *<DNS name of domain>*

Specifies the DNS domain name of an existing directory service domain when a child domain is being installed.

For more information, see "TreeOrChild," later in this document.

Note When specifying this parameter, make sure that the current user has administrative privileges to the specified domain, and that the DNS services are properly configured. The domain name must refer to an existing directory service domain.

Password

Value: *<password>*

Specifies the password for the user name (account credentials) to be used for promoting the member server to a domain controller.

Note The value is deleted from the Unattend.txt file after the promotion operation is completed.

RebootOnSuccess

Value: **Yes** | **No**

Default: **No**

Specifies whether the computer should be rebooted upon successful completion.

Note The server must be rebooted for the directory service to be started.

ReplicaDomainDNSName

Value: *<DNS name of domain>*

Specifies the DNS domain name of the domain to be replicated from. This parameter is only valid for backup domain controller (BDC) upgrades and replica domain controller installations. In such situations, a value must be specified or the installation fails.

Note Normally, the user who is currently logged on has administrative privileges to the specified domain, and DNS services are properly configured. The domain name must refer to an existing directory service domain.

ReplicaOrMember

Value: **Replica** | **Member**

Default: **Member**

Specifies whether a Windows NT 3.51 / Windows NT 4.0 BDC being upgraded should be converted to a replica domain controller or be demoted to a regular member server in the domain. This parameter is only valid when upgrading a BDC.

ReplicaOrNewDomain

Value: **Replica** | **Domain**

Default: **Replica**

Specifies whether a new domain controller should be installed as the first domain controller in a new directory service domain, or whether the new domain controller should be installed as a replica directory service domain controller. If the value is set to Domain, the TreeOrChild parameter must be specified with a valid value.

ReplicationSourceDC

Value: <DNS name of DC> | **none**

Default: **none**

Used to indicate the full DNS name of the domain controller from which the domain information is to be replicated. Only applies in new replica or BDC upgrade cases. If no value is supplied, the closest domain controller for the domain being replicated will be selected.

SafeModeAdminPassword

Value: <password> | **None**

Default: **None**

Used to supply the password to be set on the administrator account used when the computer is started in safe mode or a variant of safe mode, such as DS repair mode.

SiteName

Value: <site name>

Default: "**Default-First-Site**"

Optional.

Specifies the name of an existing site to place the new domain controller. If not specified, a suitable site is selected using the current site and subnet configuration of the forest.

SysVolPath

Value: <path to database file>

Default: "%systemroot%\sysvol"

Specifies the fully qualified, non-UNC path to a directory on a fixed disk of the local computer. If the directory exists, it must be empty. If the directory does not exist, it is created.

Note The disk must be formatted with NTFS version 5.0.

TreeOrChild

Value: **Tree** | **Child**

Default: **Child**

Specifies that the new domain is the root of a new tree or a child of an existing domain. If the value is set to Tree, the CreateOrJoin parameter must be specified with a valid value.

UserDomain

Value: <domain name>

Specifies the domain name for the user name (account credentials) to be used for promoting the member server to a domain controller.

UserName

Value: <user name>

Specifies the user name (account credentials) to be used for promoting the member server to a domain controller.

[Data]

This optional section is *only* required when performing an unattended installation by booting directly from the Windows 2000 installation CD.

AutoPartition

Allows Windows 2000 unattended Setup mode to choose a partition to install into. The value must be set to 1. If you do not set the value, text-mode Setup stops and waits for user input. For example:

```
AutoPartition = "1"
```

MsDosInitiated

Informs the Windows 2000 Setup Loader that an unattended installation is being performed directly from the CD. The value must always be set to 0. If you do not set the value to 0, Setup fails at the beginning of GUI-mode Setup. For example:

```
MsDosInitiated = "0"
```

UnattendedInstall

Informs the Windows 2000 Setup Loader that an unattended installation is being performed directly from the CD. The value must always be set to Yes. For example:

```
UnattendedInstall = Yes
```

UseBIOSToBoot

Value: 0 | 1

Default: 0

Instructs Setup to always use the BIOS to start the computer, even though Windows 2000 Setup may have detected that it is best to use a device miniport driver to start the computer. On computers with large drives that support extended int13 BIOS calls, this helps the computers to start up faster by eliminating the delays involved in using a miniport driver.

Important

- Do *not* use this parameter unless you are sure that the extended int13 functions are supported by BIOS.

- Setup will always use the BIOS to start if you include UseBIOSToBoot in your Unattend.txt file, regardless of the value assigned. If this key is not present, Setup invokes its logic to determine whether or not to use the BIOS to start Windows 2000.
-

CHAPTER 2

Sample Unattend.txt Files

Sample 1

Installs Windows 2000 Professional with default settings.

```
[Unattended]
    FileSystem = LeaveAlone
    TargetPath = WINNT

[UserData]
    ComputerName = OEM_Computer
    FullName = "User Name"
    OrgName = "Microsoft"

[Display]
    BitsPerPel = 8
    Vrefresh = 60
    Xresolution = 640
    Yresolution = 480

[Networking]

[Identification]
    DomainAdmin = admin
    DomainAdminPassword = adminpassword
    JoinDomain = ntwksta
```

Sample 2

Installs Windows 2000 Server with two NICs—one using DHCP and the other using static information.

```
[Unattended]
    FileSystem = LeaveAlone
    TargetPath = WINNT

[GuiUnattended]
    AdminPassword = *
    AdvServerType = Servernt
    OemSkipWelcome = 1
    TimeZone = 04

[LicenseFilePrintData]
    AutoMode = PerServer
    AutoUsers = 8000

[UserData]
    ComputerName = computer2
    FullName = "User Name"
    OrgName = "Microsoft"

; high color 1024 X 768
[Display]
    BitsPerPel = 16
    Vrefresh = 60
    XResolution = 1024
    YResolution = 768

[Networking]

[Identification]
    JoinWorkgroup = ntdev

; We'll have two NICs - Adapter01 and Adapter02.
; Note: the adapter we specify here as #1 is not
; always LAN connection #1 in the UI.
[NetAdapters]
    Adapter01 = params.Adapter01
    Adapter02 = params.Adapter02
; specify what card NIC #1 is
[params.Adapter01]
    INFID = "pci\ven_0ell&dev_ae32"
; specify NIC #2
[params.Adapter02]
    INFID = "pci\ven_8086&dev_1229&subsys_00018086"
```

```
; install client for MS networks
[NetClients]
MS_MSClient = params.MS_MSClient
[params.MS_MSClient]

; only install TCP
[NetProtocols]
    MS_TCPIP = params.MS_TCPIP
[params.MS_TCPIP]

; TCP/IP properties
    AdapterSections =
        params.MS_TCPIP.Adapter01,params.MS_TCPIP.Adapter02
; card 1 using DHCP server info
[params.MS_TCPIP.Adapter01]
    DHCP = yes
    SpecificTo = Adapter01
; card 2 using static info
[params.MS_TCPIP.Adapter02]
    DefaultGateway = 2.2.2.2
    DHCP = no
    IPAddress = 1.1.1.1
    SpecificTo = Adapter02
    SubnetMask = 255.255.248.0
    WINS = no

; file and print services
[NetServices]
    MS_Server = params.MS_Server
[params.MS_Server]
BroadcastsToLanman2Clients = No
Optimization = Balance
```

Sample 3

Installs Windows 2000 Server with Routing and Remote Access Service.

```
[Unattended]
    FileSystem = LeaveAlone
    TargetPath = WINNT

[GuiUnattended]
    AdvServerType = Servernt
    OemSkipWelcome = 1
    TimeZone = 04

[LicenseFilePrintData]
    AutoMode = PerServer
    AutoUsers = 8000

[UserData]
    ComputerName = computer2
    FullName = "User Name"
    OrgName = "Microsoft"

; high color 1024 X 768
[Display]
    BitsPerPel = 16
    Vrefresh = 60
    XResolution = 1024
    YResolution = 768

[Networking]
[NetClients]
    MS_MSClient = params.MS_MSClient

[NetProtocols]
    MS_NetBeui = params.MS_NetBeui
    MS_NWIPX = params.MS_NWIPX
    MS_TCPIP = params.MS_TCPIP

[NetServices]
; install RAS parameters
    MS_RasSrv = params.MS_RasSrv

[params.MS_TCPIP]
    AdapterSections = params.MS_TCPIP.A1,params.MS_TCPIP.A2
    DeadGWDetectDefault = No
    DontAddDefaultGatewayDefault = Yes
    EnableICMPRedirect = No
```

```
[params.MS_TCPIP.A1]
  DHCP = Yes
  SpecificTo = A1

[params.MS_TCPIP.A2]
  DHCP = Yes
  SpecificTo = A2

[params.MS_RasSrv]
  ParamsSection = params.MS_RAS
[params.MS_RAS]
  AssignSameNetworkNumber = Yes
  AutomaticNetworkNumbers = No
  ClientCanRequestIPAddress = No
  ClientsCanRequestIpxNodeNumber = No
  DialinProtocols = All
  ForceEncryptedPassword = 0
  IPXClientAccess = Network
  Multilink = No
  NetBEUIClientAccess = Network
  NetworkNumberFrom = 00000003
  RouterType = 7
  TcpIpClientAccess = Network
  UseDHCP = Yes

[NetAdapters]
  A1 = params.A1
  A2 = params.A2
[params.A1]
  NetCardAddress = 0x2bfc01f388d
[params.A2]
  NetCardAddress = 0x2bfc01f3890
```

Sysprep.inf File Parameters

This chapter lists the various keys and their parameters that can be included in the Sysprep.inf file. Many keys/parameters are interdependent; please note the interdependencies where specified.

A sample Sysprep.inf file may include:

```
[GuiUnattended]
    TimeZone = 20

[UserData]
    FullName = "Authorized User"
    ComputerName = W2B32054
    OrgName = "Company Name, Inc."
```

[Unattended]

ExtendOemPartition

Value: **0** | **1** | **<extra size in MB>**

The ExtendOemPartition key is used to extend the partition on which you are installing Windows 2000. This key causes Setup to extend this destination partition into any available unpartitioned space that physically follows it on the disk.

where:

- 0 Setup does not extend the partition.
- 1 Setup extends the partition to fill out the hard disk.
- <extra size in MB> Setup increases the current partition size by this amount.

Important

- Only NTFS partitions can be extended. If the destination partition you plan to extend is FAT or FAT32, set FileSystem = ConvertNTFS to convert the partition but only during text-mode Setup. Setup will not extend FAT and FAT32 partitions.
- ExtendOemPartition can be used with both the Unattend.txt and Sysprep.inf Setup files.

- When used in Sysprep.inf for imaged computers, the destination computer's hard disk must be the same size or larger than the master computer's hard disk.

InstallFilesPath

Value: <path to installation files on computer>

Only valid for Sysprep.inf. Specifies the location of files that may be necessary for installation during Mini-Setup, such as the language files, so that the installation is not interrupted to prompt for user input. For example, for multilingual support, you can:

- Use Setup Manager
- Use Sysprep.inf

To use Sysprep.inf for multilingual support

1. Create a subfolder named \i386 under \%OEM%\\$1\Sysprep\ in the distribution folder.
2. Copy the following files from \i386 of the Windows 2000 product CD to the \i386 directory in the Sysprep folder:

```
*.nl?  
kbd*.dl?  
*.fo?  
agt*.dl?  
agt*.hl?  
conime.ex?  
wbcache.*  
noise.*  
wdbbase.*  
infosoft.dl?  
f3ahvoas.dl?  
sylfaen.tt_  
c_is*.dl_  
\Lang\...
```

3. In Sysprep.inf, add:

```
[Unattended]  
InstallFilesPath = C:\Sysprep\i386
```

Note

- InstallFilesPath is only valid when used in the Sysprep.inf file in conjunction with Sysprep.exe to indicate the location of files needed during Mini-Setup, such as the localization files for installing additional code pages.

- InstallFilesPath *does not* support environment variables (i.e., %systemdrive%). Instead, a valid path with an actual drive letter must be used.
-

KeepPageFile

Value: **0** | **1**

Default: **0**

Optional

Only valid for Sysprep.inf. When Sysprep is run, the Mini-Setup wizard automatically regenerates the system's pagefile to accommodate differences in the amount of RAM between the master and destination computers. The pagefile is set to regenerate by default (KeepPageFile = 0). This ensures that the pagefile is large enough to accommodate the amount of RAM on the computer.

If the amount of RAM between the master computer and the destination computers is the same, the pagefile does not need to be regenerated. The pagefile is regenerated automatically only when you do not include this parameter and a value in Sysprep.inf.

Note

- KeepPageFile is only valid when used in the Sysprep.inf file in conjunction with Sysprep.exe.
 - **Known Issue:** If you add KeepPageFile to your Sysprep.inf file, regardless of the value assigned, the pagefile is not regenerated.
-

OemPnPDriversPath

Value: "<folder 1 on system drive>;<folder 2 on system drive>; ..."

Specifies the path to folders that contain PnP drivers that do not ship on the Windows 2000 CD. The folders must contain all the files necessary to install the particular devices—catalog files, .inf files, and drivers.

For example, if you have a folder called \Drivers with subfolders called \Audio and \Net, you would specify OemPnPDriversPath = "drivers\audio;drivers\net" in the answer file. Setup adds:

- %systemdrive% to each of the folder names
 - the path for each subfolder to the PnP device search path
-

Note When using this parameter, be sure that the folders are available during GUI-mode Setup or Mini-Setup; you can use the \\${OEM}\\$1 directory structure mechanism for this.

OemSkipEula

Value: **Yes** | **No**

Determines whether the user should be prompted to accept the Microsoft License Agreement (previously known as the End User License Agreement or EULA) included

with Windows 2000. Writing this key and setting it to Yes implies that the person performing the installation has read and agreed to the contents of the License Agreement included with the product. It also implies that the end user on whose behalf Windows 2000 is being installed has agreed to the License Agreement.

UpdateHAL

Value: "*hwid*,%windir%\inf\hal.inf"

(where *hwid* is either MPS_MP or ACPIAPIC_MP)

Only valid for Sysprep.inf. Required if you are using compatible HALs that are based on an APIC uni-processor-system (UP) master computer image that is to be placed on a compatible multiprocessor-system (MP) destination computer.

Important A Sysprep.inf file containing this parameter should not be used on destination UP computers, because UpdateHAL loads the MP HAL on the destination computer whether it is a UP or an MP computer. An MP kernel on a UP computer will noticeably reduce the computer's performance.

To add the modified Sysprep.inf file to the destination computer, you can:

- Save the .inf file to a floppy disk and insert the disk before Mini-Setup starts.
- Update the image on the destination computer to include the .inf file.

UpdateUPHAL

Value: "*hwid*,%systemdrive%\Sysprep\i386\Uniproc\Mp2up.inf"

(where *hwid* is either MPS_UP or ACPIAPIC_UP)

Only valid for Sysprep.inf. Required if you are using compatible HALs that are based on an APIC MP master computer image that is to be placed on compatible APIC UP or MP destination computers.

In this situation, Windows 2000 automatically identifies the processor type and loads the appropriate kernel.

[Oem_Ads]

All keys are supported (Logo and Background). For more information about these keys, see "[Oem_Ads]," in Chapter 1.

[GuiUnattended]

Only the following keys are supported by Sysprep.inf.

AdminPassword

Value: <*password*> | *

Sets up the Administrator account password. If the value is set to *, the administrator password is blank or NULL. (The maximum password length is 127 characters.)

Note If a password is specified in the Administrator account, you cannot use AdminPassword in the Sysprep.inf file to change it—the administrator password remains the same. However, if the administrator password was initially blank (either

manually or through an unattended installation), you can use the AdminPassword key to change it to a non-blank password.

However, if a non-blank password is used, end users cannot change or specify their own passwords in the Mini-Setup wizard. A new password may be entered in the appropriate dialog box, but the password is not changed.

In addition, security breaches may occur if you use a common, non-blank administrator password for all computers provided to end users. Prior to running Sysprep, you should use an automation process to set the administrator password to blank. End users can then specify their own passwords upon receipt of the computer.

AutoLogon

Value: **Yes** | **No**

Sets up the computer to automatically log on once with the Administrator account if set to Yes. The default behavior is No. This key is not valid on upgrades.

Note

- If you specify a password by using AdminPassword, that password is used when you automatically log on. After the installation is complete, the password is deleted from the copy of the answer file left on the computer.
 - If AdminPassword = * (is blank) and AutoLogon = Yes, the computer logs on only one time. However, if AdminPassword = <password>, the computer logs on repeatedly and the non-blank password is saved in the registry.
-

AutoLogonAccountCreation

Value: **Yes** | **No**

Default: **Yes**

Specifies whether a computer account should be created automatically for the user whose name is specified by FullName. For more information about this parameter, see "FullName," in Chapter 1. AutoLogonAccountCreation also allows the generated user account to automatically log on to the computer.

This parameter is only valid on computers that are configured with the Workgroup option. This parameter will not work for computers that are configured as members of a domain.

Note

- The user's account is only configured to log on automatically by default when the option to join a workgroup is specified. This does not apply to computers that are members of a domain.
 - This parameter is not related to AutoLogon or AutoLogonCount; these parameters apply to the Administrator account only.
-

AutoLogonCount

Value: *<integer>*

Specifies the number of times that the computer automatically logs on using the Administrator account and password specified. The value decrements after each logon, and the feature is disabled after the specified number of logon attempts.

Note For the count to be decremented, you must reboot the computer.

This parameter is only useful when AutoLogon = Yes and AdminPassword = *<password>* are specified in the answer file.

Important

- Make sure that the password for the master computer (the computer being used for disk duplicating) is blank.
 - If AdminPassword = * (is blank) and AutoLogon = Yes, the computer logs on only one time, regardless of the value given for AutoLogonCount.
-

OEMDuplicatorString

Value: *<user defined string>*

Only valid for Sysprep.inf. This string contains a description of the duplication utility used, as well as any other information an OEM or administrator wants to store in the registry. The value can have a maximum of 255 characters and is stored in the HKLM\System\Setup\OemDuplicatorString registry key.

OEMSkipRegional

Values: **0** | **1**

Allows unattended Setup to skip the Regional Options page in GUI-mode Setup and Mini-Setup.

Note When specifying OemPreinstall = Yes and providing values for the [RegionalSettings] section, set OEMSkipRegional to 1 to ensure that Setup completes without prompting the user for regional information.

OEMSkipWelcome

Value: **1** | **0**

Allows unattended Setup to skip displaying the Welcome page in GUI-mode Setup and Mini-Setup.

Note If OemPreinstall = Yes, unattended Setup automatically stops at the Welcome page. To avoid this pause, set OEMSkipWelcome to a value of 1. However, OEMs must *not* ship computers with this setting enabled.

TimeZone

Value: <index>

Specifies the time zone where the computer is located. If the key is not present, the user is prompted to select a time zone. For a list of valid time zone indices, see “TimeZone,” in Chapter 1.

[UserData]

All keys are supported. For more information about these keys, see “UserData,” in Chapter 1.

[LicenseFilePrintData]

All keys are supported. For more information about these keys, see “[LicenseFilePrintData],” in Chapter 1.

[GuiRunOnce]

All keys are supported. For more information about these keys, see “[GuiRunOnce],” in Chapter 1.

[Display]

All keys are supported. For more information about these keys, see “[Display],” in Chapter 1.

Note Sysprep uses the video settings in the [Display] section of Sysprep.inf. If there are no display settings in the [Display] section of Sysprep.inf or if Sysprep.inf is not used, Sysprep uses the video settings in the registry. If the display settings were configured manually, set in the answer file, or used as the defaults, then Sysprep retains those settings.

[RegionalSettings]

All keys are supported, provided the files are available on the hard disk. For more information about [RegionalSettings], see “InstallFilePath,” earlier in this chapter and “[RegionalSettings],” in Chapter 1.

[TapiLocation]

All keys are supported. The values are only valid if a modem is present on the computer. For more information about these keys, see “[TapiLocation],” in Chapter 1.

Note If you use the Sysprep.inf file with Sysprep, the modem information (Telephony Application Programming Interface (TAPI)) and the Networking screens are not displayed during Mini-Setup. The default networking components are used if the networking components are not configured and if they are not specified in Sysprep.inf.

[Networking]

No keys are required.

[Identification]

All keys are supported. For more information about these keys, see “[Identification],” in Chapter 1.

C H A P T E R 4

Sample Sysprep.inf Files

Sample 1

Customizes the Windows 2000 installation for a standard departmental desktop.

```
[Unattended]
    OemSkipEula = Yes

[GuiUnattended]
    AdminPassword = Admin password
    OemSkipWelcome = 1
    TimeZone = 20

[UserData]
    FullName = "Company Department Name"
    OrgName = "Company Name, Inc."

[Identification]
    DomainAdmin = "CORPDOM\AcctAddID"
    DomainAdminPassword = Domain password
    JoinDomain = "CORPDOM"

[Networking]
```


Sample 2

Customizes the Windows 2000 installation for remote office servers.

```
[Unattended]
    OemSkipEula = Yes

[GuiUnattended]
    AutoLogon = Yes
    AdminPassword = *
    OEMSkipRegional = 1
    OemSkipWelcome = 1
    TimeZone = 20

[UserData]
    FullName = "Corporate Server Name"
    OrgName = "Company Name, Inc."
    ComputerName = 005-REGSRV-1

[LicenseFilePrintData]
    AutoMode = PerSeat

[GuiRunOnce]
    Command0 = "dcpromo /answer:ansfile.txt"

[Identification]
    DomainAdmin = "CORPDOM\IDAcct"
    DomainAdminPassword = Domain password
    JoinDomain = "REGDOM"

[Networking]
```