

Windows NT System Policies

Darren Mar-Elia

(Reprinted from WindowsItPro Magazine)

At a recent Microsoft briefing for enterprise customers, one presenter asked attendees whether they were using Windows NT 4.0's system policy features. A small percentage raised their hands, and an even smaller percentage kept their hands up when the speaker asked whether they had an inhouse resource person who understood how system policies work.

NT system policies are useful for managing user and machine Registry changes in the enterprise. They help systems administrators centralize configuration control in large and small NT environments. They also ease problems associated with desktop configuration management, such as delivering icons to your users' desktops. However, NT system policies can be difficult to configure, can cause widespread damage, and can become unmanageable if you're not careful.

System policies in NT 5.0 will be more powerful than those in NT 4.0. To take advantage of NT system policies, you need to understand how to efficiently implement this tool in your NT infrastructure--what works and what doesn't. In this article, I'll focus on the benefits and challenges of implementing system policies in real-world enterprise NT environments.

The Power of Policy

NT system policies let you deliver user- and machine-specific Registry changes each time a user logs on. You can use the System Policy Editor (SPE) and templates that define which Registry keys your policies affect to create policy files that perform various functions. The default implementation for a system policy is to use the SPE to create an ntconfig.pol file, and copy the file to the replication directory in your domain controller infrastructure. The replicator service then replicates this directory to all other domain controllers, and makes it available via a Netlogon share. If you implement a single or multiple master domain model, you must replicate ntconfig.pol in the master account, or authenticating domain. The ntconfig.pol file has no effect in the resource domain. Even if the computer is registered in the resource domain, the authentication or master account domain delivers policies when a user logs on. You can change the policy file's name and the location where NT workstations look for policy files. You can also have multiple policy files that various NT workstations in a domain can use, which I'll discuss later.

System policies are powerful and can solve many problems. For example, in your work to solve the Year 2000 compliance problem in your NT environment, you might realize that the default configuration on your 5000 NT workstations is to display dates with only two characters to represent the year (e.g., MMDDYY). You can use the user's profile in the Registry (the HKEY_CURRENT_USER Registry hive) to change this date representation to MMDDYYYY, and you can easily create a special policy template that includes this Registry key. Then you can build a new policy file that incorporates the new template, assign it to Default User, and replicate it to your domain controllers. The next time your users log on, NT will update their default date display.

Security Through Obscurity

System policies can secure the user's desktop to reduce the costs of maintaining your NT environment. Microsoft provided the Zero Administration Kit (ZAK) to help reduce the total cost of ownership (TCO) for your NT environment. (For more information about ZAK, see "Zero Administration Kit: The Answer to Your TCO Woes?" January 1998.) ZAK provides extra system policy templates that let you augment NT 4.0's standard templates--common.adm and winnt.adm. You can use these default templates, ZAK's templates, and custom templates that you develop to lock down desktops and prevent users from performing unsafe tasks.

Most restrictions that system policies enforce merely hide features from the errant user rather than provide true security measures. For example, you can use a system policy to remove the Run

Windows NT System Policies

Darren Mar-Elia

(Reprinted from WindowsItPro Magazine)

command from the Explorer shell's Start menu. But a clever user can run a command shell via several back doors, including File Manager (which doesn't respect system policy settings), the MSInfo utility that comes with Microsoft Office, macros, and Visual Basic for Applications (VBA). Likewise, you can use a system policy to prevent a user from interactively running regedit or regedt32, but the user can write a Registry script and use regedit to run it from a command line or association (e.g., regedit myfile.reg).

When you consider implementing policies to secure the desktop against tampering, you need to remember that policies affect user and machine Registry settings only: They don't modify file or Registry access control lists (ACLs) or change a user's NT security context. Policy restrictions work because the Explorer shell checks the user or machine Registry for Registry keys, and then permits or denies user actions based on these keys. Most restrictions related to the Explorer shell (e.g., removal of Start menu elements, access to Network Neighborhood) are in one key in a user's profile--HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies--and Registry settings in this key control most user-specific policies that you set. Typical Registry permissions give users full control over their profiles. But the Policies key is read only, so users can't tamper with policy restrictions.

Policy-based desktop restrictions have inherent limitations, but policies are valuable in managing the enterprise desktop. For example, you can use a policy to remove multiple users' desktop elements at once. Shared folders provide another powerful policy feature. With shared folders, you can use policies to deliver file folders and application shortcuts to a user's desktop similarly to how you deliver desktop restrictions.

Share and Share Alike

One advantage of the NT 4.0 Explorer shell is that you can represent most desktop elements, such as application icons and file folders, as simple file system objects--shortcut or .lnk files. NT 4.0's Explorer gives you better control over desktop objects than NT 3.51's Program Manager did (i.e., when program groups were binary files or binary Registry values that were difficult to manage). Explorer's default system policy templates let you create customized user- and machine-specific shared folders that you can centrally manage and deliver to the user's desktop in one step. Instead of sending shortcuts to hundreds or thousands of desktops when you implement a new application, you can use system policies to point users to a centralized folder on a server where these shortcuts reside. When you deliver customized user- or machine-specific folders via a system policy, the policy file redirects the user- or machine-specific path of various Explorer shell elements to the location you specify in the policy file.

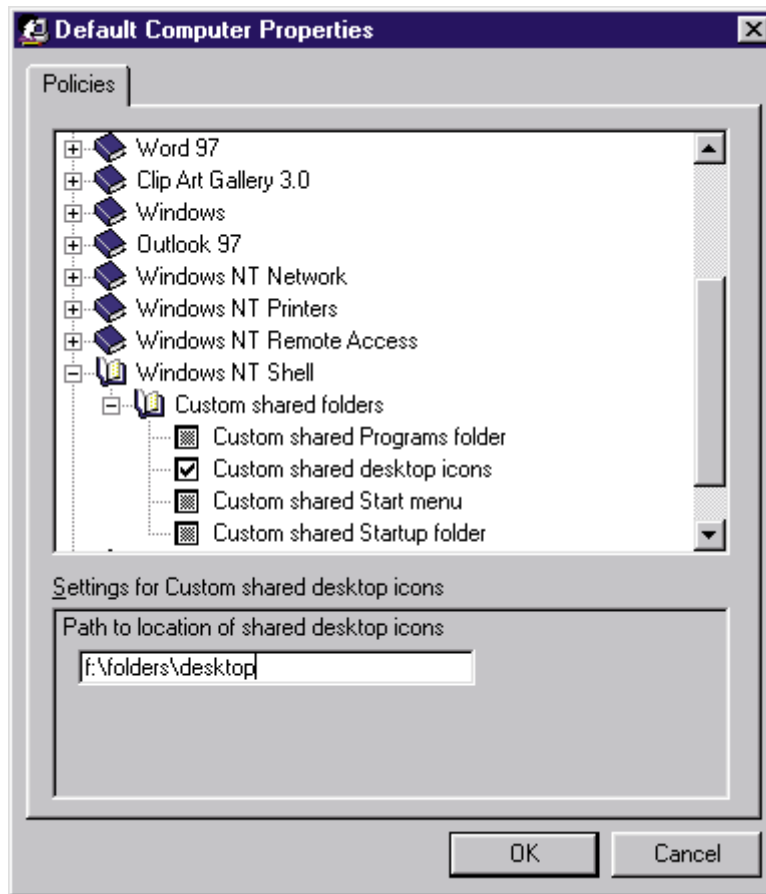
For example, the default location for a machine-specific (or common-group) desktop icon is %systemroot%\profiles\All Users\desktop. But you can use a machine-specific setting in the NT shell policy, as Screen 1 shows, to redirect this path to a server share (e.g., f:\folders\desktop). The NT shell policy is part of the winnt.adm policy template file. When users log on to their workstations and receive the system policy, the policy redirects the machine-based, or common, desktop folder from the default location in the All Users directory to the server share you specified in the policy.

Users who log on to local machines receive desktop icons from the server-based folder, so you can modify the contents of one server-based folder to make changes to hundreds or thousands of user desktops.

Windows NT System Policies

Darren Mar-Elia

(Reprinted from WindowsItPro Magazine)



The machine-specific shared folders model also applies to user-specific shared folders. In a system policy file, you can specify custom folders for a user or user group. From the same NT shell policy, you can set server-based folder paths for the desktop, Programs folder, Startup folder, Network Neighborhood, and Start menu. These policy folders redirect user profile folders to other locations. For example, the default location for desktop icons on a per-user basis is %userprofile%\desktop. But you can create a policy for a user-specific custom folder, telling the Explorer shell to replace the user profile path with the path you provide in the policy. After the policy file is in place, you can redirect many user desktops to a shared area on a server, where you control the content. Give read-only access on this shared folder so that users can't write or delete icons that can affect other users, or place extraneous shortcuts, folders, or files on their desktops. This restriction is helpful if you use roaming profiles, because users who place large documents on their desktops and in their user profiles might adversely affect network performance at logon and logoff times.

NT System Policy Challenges and Benefits

Implementing NT 4.0 system policies can be complicated, and the current system policy architecture has limitations. But system policies provide significant benefits, and they will play an important role in NT 5.0.

You can distribute policies on a user, machine, and global group basis. If you plan to provide global group-based policy distinctions, limit the number of global groups in your policy--perhaps three to five groups. Managing global group-based policies can be difficult because of the nature of policy application. NT applies policies cumulatively (assuming you don't specify user-specific policies, in which case NT ignores all group policies), based first on Default User, then defined global groups,

Windows NT System Policies

Darren Mar-Elia

(Reprinted from WindowsItPro Magazine)

then machine-specific policies. You can prioritize NT's application of global group-based policies in SPE so that users who belong to multiple policy groups will receive a predictable policy. Every user gets the Default User policy, unless you define a user-specific policy or remove the Default User policy. Keep in mind that when a user gets an effective policy, this policy includes the information you defined in the Default User policy. If you retain the Default User policy, you must decide how you will use this policy group. Users who are not members of another defined policy group will receive only the information Default User gives them. You can lock down Default User and remove restrictions with other policy groups, or you can give an open desktop to users who aren't in a policy group.

Each policy item can be included (selected), unchanged (grayed out), or removed (blank). You can easily interpret these states independently, but when you apply multiple policy groups on top of each other, you might have trouble determining the resulting policy. A typical user gets the Default User policy and one or more global group-based policies. Thus, you must consider all the possible combinations of selected, grayed out, and blank policy items. If a user is a member of multiple policy groups, you need to know how each policy group is applied, and what the effective policy is when you apply all groups by priority. You must consider how policy items' states will affect users who move between policy groups. You need to ensure that you undo policy items correctly as a user moves from a more restrictive to a less restrictive policy group. To further complicate matters, some policy items can undo each other.

For example, if you use the ZAK policy templates, the Shell\Restrictions\HideDrives in My Computer policy, which is part of the common.adm template, conflicts with the ZAK Policies\Windows NT\Drives\Restrictions\Show Only Selected Drives policy, which is available in the zakwinnt.adm template. You have to gray the restriction in common.adm and use only the zakwinnt.adm restriction to hide drives. Otherwise, the common.adm policy item will automatically change when you save the policy.

Another challenge in implementing system policies in your enterprise is the limited tools available for managing the policies. Microsoft provides only SPE and a few policy templates. SPE is the only meaningful method for viewing the contents of a policy file. Most large enterprises have strict change and version control practices to prevent one change from crippling the computing environment. If you make frequent changes to an enterprise policy file that affects hundreds or thousands of workstations, the only version-control tools available are the policy file's date and timestamp. Moreover, the SPE has no reporting tools for you to list current policy settings or analyze the effective policies for user and group combinations. NT 5.0 needs to provide more tools for organizations to use policies effectively.

Despite the caveats, you can use system policies to deliver meaningful desktop control in your NT 4.0 environment. But you'll want to keep the number of global policy groups to a minimum (three to five), avoid using user- or machine-specific policies, avoid placing users in many overlapping global policy groups, and limit users to one global policy group at a time if possible.

You apply a policy file at the domain level. Therefore, try to keep the policies that you implement relevant to domain-wide changes, and avoid making many small, application-specific Registry changes within the policy. For example, if you want your users to use the same background bitmap on their desktops, use the policy file for such a change. But don't use policies to map drives or printer connections that might change frequently or that are specific to a user's location.

Avoid making frequent changes to your policy file. If you have to update your policy file frequently, record the date and timestamp of each version, and keep the last version available in case you need

Windows NT System Policies

Darren Mar-Elia

(Reprinted from WindowsItPro Magazine)

to revert to it. Remember that changing policy files for your entire domain is as easy as copying a file to your domain's replication directory.

Keep a chart that lists your current policies by group and current state. This chart helps you see how NT applies different policies as users move between global policy groups or become members of multiple global policy groups.

Troubleshooting and Tweaking

After you decide to implement system policies, you need to learn how to troubleshoot problems.

System policies don't log information to NT's event logs when a user logs on, but you can log a policy file's activity. First, you need a copy of a checked-build or debug version of the NT userenv.dll system file. You can get this file from Microsoft Support or if you subscribe to the Microsoft Developer Network (MSDN). Make a backup copy of userenv.dll, and copy the debug version of this file to the %systemroot%\system32 folder on the system where you want to log the policy file's application.

You need to modify the Registry to enable logging. In the Registry key HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsNT\ CurrentVersion\Winlogon, create a new value entry, User-EnvDebugLevel (type REG_DWORD), with hexadecimal value 10002. Shut down and restart your system. At next logon, your system will write the text file userenv.log to the C drive root. This file shows user profile and system policy processing. For information about reading the log file, see the Microsoft Support Online article Q154120 (<http://support.microsoft.com/support/kb/articles/q154/1/20.asp>).

The default policy filename is ntconfig.pol, and the default location for the file is the Netlogon share on any domain controller in your authentication domain. However, if you need to implement multiple policy files in a domain, you can change the policy filename and location for each workstation. NT's common.adm template file has a policy setting for making these changes. From Default Machine, go to the Network\System Policies Update\Remote Update field. This policy adjusts the values in the Registry key HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Update. The UpdateMode value controls whether the workstation looks in the Netlogon share (Automatic Path: Registry value of 0*1) or a path that you specify (Manual Path: Registry value of 0*2). If you specify a value of 0, the workstation ignores the policy files you have in place. If you specify Manual Path, the workstation uses the NetworkPath value. You can use NetworkPath to redirect your standard policy file to a directory other than Netlogon (e.g., c:\localpolicy\ntconfig.pol). You can also specify a different policy filename in the Netlogon share: Use the NT 4.0 environment variable %logonserver% in the path you specify. For example, to point a workstation to an ntconfig2.pol policy file, specify %logonserver%\ntconfig2.pol as the value of NetworkPath. Using alternative paths to policy files is useful for testing new policies before you put them into practice and affect all your users.

NT 4.0 policies have some unusual behavior. You must install Service Pack 3 (SP3) on your workstations for policies to function correctly. Without SP3, your system ignores global policy groups and adheres only to the Default User policy. This behavior also occurs if you use a version earlier than 4.11 of Novell's intraNetWare Client for Windows NT. I've had problems applying policies against built-in NT groups such as Domain Admins. When you create a policy to use against Domain Admins, the policy works only the first time a user in the group logs on to a workstation in the domain.

Subsequent logons don't use the policy. This problem is most likely a bug in NT's system policies, although Microsoft hasn't documented the problem on its Support Online Web site.

Windows NT System Policies

Darren Mar-Elia

(Reprinted from WindowsItPro Magazine)

System Policies in NT 5.0

NT 4.0's system policies are useful, and you can expect even better policies in NT 5.0. Policies will be more granular, and you will be able to apply them at the domain, site, or organizational unit (OU) level in Active Directory (AD). Policies will control more than desktop lockdown. NT 5.0's group policies will be a key element in the application management feature of the operating system (OS), where you distribute icons to users' desktops based on their location in the directory and the user groups they belong to. To prepare yourself for NT 5.0's system policies, you can implement policies in NT 4.0 today and learn about their powerful features and functions.