Global Information Assurance Certification Paper

Michael Ross
GCNT Securing Windows version 2.1b
Windows 2000 Domain Controller Security templates
23 July 2001

Windows 2000 takes security to a whole new level. Unlike previous version upgrades (3.1 – 3.5 – 4.0), WIN2K was designed with security integral to the operating system. At the heart of the WIN2K security you will find the Active Directory and Global Groups.

To make life easier on the administrator, Group Policy allows for the import of security templates. Although the mythical Panic Button, which sets every security setting to maximum, did not materialize, the security templates apply security at two levels, not including basic settings. WIN2K comes with the following pre-configured templates; basicdc, basicsv, basicwk, compatws, dcsecurity, hisecdc, hisecws, notssid, ocfilesss, ocfilesw, securedc, securews, and setupsecurity. These available templates automatically configure settings for; Account Policies, Local Policies, Event Log, Restricted Groups, System Services, Registry, File System, and Public Key Policies. Importing one of these templates makes setting the initial security configuration easy. An administrator could modify the local computer policies; export the changes to a new template (.inf) file. The new template could then be imported into other domains or distributed throughout the remote business offices as a corporate standard.

What settings do these templates make? Are these settings appropriate for your organization? Locally? Or across the corporate? In this paper I intend to explain what changes the securedc and hisecdc templates make. I will point out some of the good and bad points of those changes and in some cases recommend some possible tweeks.

It is recommended that you import the basicdc template into a clean install of WIN2K before importing these templates. If you must upgrade your previous operating system the basicdc template must be imported first to return the security configuration to default. The templates can be imported on their own; the hisecdc template includes all settings contained in the securedc template without needing to import the securedc template first. To import a security template into Group Policy go to Computer Configuration\Windows Settings\Security Settings; right click on security settings and choose Import from the context menu.

Account Policies – Password Policy

MaximumPasswordAge: The range for this setting is 1-999 days. The age could also be set to 0, which sets the password to never expire. Passwords need to be changed on a regular basis. This limits the time an unauthorized person has to attempt to crack the password, and in the event the password is cracked, it limits the time the unauthorized person has to do damage with the password. Both the securedc and the hisecdc template set the maximum age to 42 days. This may be appropriate in many sensitive or high security areas where there is a realistic threat of a dedicated hacker. Users in the high security areas are more likely to receive frequent security training and will be more careful about protecting their passwords. However, this may be too

restrictive for the lower security areas using the securedc template. In these places users will receive less security training and be less protective about their password. In lower security areas, when the password changes too often, users will be more likely to write down their passwords. I would recommend setting the maximum password age to 90 days in the lower security areas.

MinimumPasswordAge: When users are forced to change their password, many will immediately change it back to the one they like. By setting the minimum age a user would have to wait before changing their password again. Minimum age can be set for 0-999 days, with 0 allowing immediate changing. Both templates set the minimum age to 2 days. Two days is appropriate for both security levels. Setting the minimum age too much longer could cause problems if a user suspects his password has been compromised. The user would have to contact the administrator before the password could be changed. The next setting, password history prevents the die-hard user from waiting the two days and changing back to the one he likes.

PasswordHistorySize: Maintaining a history of the last passwords used prevents a user from easily returning to a preferred password. When coupled with the previous setting a user would have to wait two days between each password, then run through the stored history before returning to the preferred password. Both templates set this to a maximum of 24 passwords. Coupled with minimum age, a user would have to wait 48 days before returning to the preferred password. This should discourage any user from returning to their preferred password. The history could also be set to 0 which prevents storing any passwords.

MinimumPasswordLength: Common theory is that the longer the password is, the longer it would take to crack the password. The length setting will allow a password from 1 to 14 characters. Setting this value to 0 allows the password to be blank. When a password is hashed using just LM authentication, NT splits the 14-character password into two seven-character chunks and hashes each one separately, then combines the two chunks into the LM hash. The password cracking tools, such as @Stake's L0pht Crack, take these password hashes, split them in half, then attack each part separately. If you are using a password that is 8-13 characters in length the second half is 1-6 characters, buffered to 7 characters with nulls. The second half being shorter is easier to crack and can improve the chance of guessing the first half. NTLM, on the other hand, hashes the entire 14-character password in one chunk. WIN2K includes NTLM version 2 that improved password security even further by incorporating 128-bit encryption.

Even though this setting only enforces up to 14 characters, WIN2K has increased the password text box to allow passwords of up to 56 characters. Both security templates set minimum length to 8 characters. This is acceptable for common users providing you do not need the backward (Win 9x) compatibility of LM authentication. If you must use LM authentication then passwords should be set to either 7 or 14, not in between. Because their accounts are more sensitive that the average user account, administrators should choose passwords over 10 characters in length.

Passwordcomplexity: This is either enabled, or disabled. Coupled with password length, the more complex a password is the harder it is to crack. LM authentication recognizes letters, numbers, and special characters as three separate character sets when composing passwords.

NTLM adds a fourth character set by differentiating between lower and upper case letters. The security templates enforce a three character set password requirement. User must enter a password that consists of at least three of the character sets. Each character set makes it more difficult for cracking programs, like L0pht Crack, to crack the password. One additional character set is available, but not normally advertised, is the extended ASCII Character set. Administrators should add at least one ASCII character to their password since the ASCII characters must be added to L0pht's user-defined list.

Account Policies – Lockout Policy

LockoutBadCount: When an unauthorized person attempts to gain access to the computer the system needs to be capable of protecting the account. By setting a limit to how many times a user can incorrectly logon, the system can prevent someone from repeatedly trying different passwords on an account until he finds the correct password. The range for this setting is 1-999 tries, with 0 indicating the account cannot be locked out. Both templates set the allowable limit to five tries. After five tries everyone is locked-out of the account. In the high security environments this threshold should be lowered to three.

ResetLockoutCount: It is common for users to forget their password, especially after changing their password. For those who think they know their password, but are not sure, they can wait until the lockout bad count gets reset, and then try again. The reset time can be set from 1-99,999 minutes (approx 70 days), 0 is not an option here. The templates set the reset time to 30 minutes. After 30 minutes the user can try again to enter his password. Allowing the user this chance, may reduce the number of helpdesk calls for lockouts. Setting a lockout-reset time does not necessarily weaken security of the network. If an unauthorized user has to wait 30 minutes after every 2-4 incorrect attempts, he will quickly give up trying to guess the password.

LockoutDuration: Once the user (or intruder) exceeds the lockout bad count the account becomes locked out, preventing anyone from accessing the account. This lockout time can be adjusted from 1 minute to 1,666 hours. It can also be set to 0 for an indefinite lock. The securedc template sets the lockout duration to 30 minutes and the hisecdc template sets the lockout for indefinite. Once the user locks himself out with the duration set for indefinite, the user must contact an administrator to be reset. Administrators in both security environments need to watch their audit logs for frequent lockouts. These frequent lockouts could be early signs of someone trying to gain access to the system.

Account Policies – Kerberos Policy

The templates do not change any of the default policies which are normally sufficient for secure authentication. Since any changes must be changed at the group level, not local machine the changes cannot be exported. The default Kerberos settings are:

Enforce user logon restrictions
Setting: Enabled
The logon restrictions require the Key Distribution Center (KDC) validate every request for a

session ticket by reviewing the user's rights policy on the server the user is requesting access. Verifying every access does increase network traffic and will increase the time it takes to get access to a resource.

Maximum lifetime for service ticket
Setting: 60 minutes (1 hr)
Maximum time a user can present a Service Ticket (ST) and gain access to a resource. When a user needs a network resource he presents his TGT to the domain controller and is issues a ST which can be used for that resource.

Maximum lifetime of a user ticket
Setting: 10 hours
Maximum time the user's Ticket Granting Ticket (TGT) is valid. When a user authenticates to the domain he is issued a TGT. The TGT will be used to request a ST each time he needs to access a network resource.

Maximum lifetime for user ticket renewal
Setting: 7 days
A TGT can only be renewed for seven days. After seven days the user must re-authenticate and be issued a new TGT.

Maximum tolerance for computer clock synchronization
Setting: 5 minutes
Ticket requests include a time stamp to prevent a session request being stolen and presented at a later time. The time stamp must be within 5 minutes of the KDC or the request will not be granted.

Local Policies – Audit Policy

WIN2K maintains six types of event logs, the first three were previously used in Windows NT, and the second three are new to Windows 2000:

System – Contains messages from the base operating system components

Application – Where applications report their event messages

Security – Contains auditing information

Directory Service – Records information regarding the NT Directory Service, problems connecting to the Global catalog and any issues regarding Active Directory in your network

DNS Service – Records events related to running Directory Name Service in your Active Directory

File Replication Service – Records any notable events that took place while the Domain Controller attempted to update other Domain Controllers.

The Templates configure 9 categories of events recorded in the Security Audit log. These categories include Account Log-on Events, Account Management, Directory Service Access, Log-on Events, Object Access, Policy Change, Privilege Use, Process Tracking, and System Events.

	Securedc	Hisecdc
Account Logon Event	Failed	Successful/Failed
Logon Events	Failed	Successful/Failed
Account Management	Successful/Failed	Successful/Failed
Policy Change	Successful/Failed	Successful/Failed
Process Tracking	Not Audited	Successful/Failed
Privilege Use	Failed	Successful/Failed
Directory Service Access	Failed	Successful/Failed
System Events	Not Audited	Successful/Failed
Object Access	Not Audited	Successful/Failed

Some points to consider when modifying these settings:

Audit Logon Event and *Logon Events*: Important for detecting attempted break-ins and monitoring for suspicious user activity – after-hours or weekend use when organizational users are not normally at work. Knowing when a user logged on and off is key evidence when investigating unauthorized system events.

Account Management: Will identify when a user's privileges are elevated. The administrator should be the only person to elevate privileges, and users should only have minimum privileges necessary to do their work. Any time privileges get elevated without knowledge of the administrator could indicate a compromised account.

Policy Changes: Identify changes to the security policy that could weaken the policy initially set by the administrator. These could be indications of intruders weakening system defense to make a future return easier.

Process Tracking: Tracks process activation and termination. Useful in a development environment or when tracking virus behavior, but will create massive amounts of data in a production environment. The securedc template does not activate this item but the hisecdc template does. If using the hisecdc template administrators should consider deactivating auditing.

Use of Privilege: Monitors use of administrator's privileges, such as changing system time, adding workstations to the domain, creating pagefile, load and unload device drivers, manage auditing and security logs, etc… This setting can create large quantities of audit data, but in the high security environment administrators will want to track this activity.

Directory Service Access: Tracks access to the directory service. This is not essential for member servers, but should be activated on domain controllers in the high security environments.

System Events: Logs shutdowns and restarts on the local workstation.

Object Access: Enabling object access does not start recording events, After enabling object

access administrators must select the objects to be audited. With WIN2K an object could be anything; an individual file, a directory, a printer, a registry key, or internal operating system data structure. Auditing can be configured to monitor one specific action, on one specific file, for one specific user. Setting auditing to this granularity, for every object, would create massive log files, making reviewing virtually impossible. Easiest way to set-up object auditing is through Windows Explorer. Choose the object you wish to audit, right click, select Properties from the context menu, select the Security tab, Click on the Advanced button, select Auditing tab, then add.

Local Policies – Security Rights Assignment

The securedc and hisecdc templates do not set User Rights beyond what is set by the default however, when an administrator configures a local template and exports it as a standard template for use in other corporate domains, User Rights are also exported. The system administrator must determine which users will need what rights, according to company policy. Rights which can be configured are:

Access this computer from the network
Allows users to access the server over the network. For the Domain Controller this right should be removed from the administrator accounts. Removing the right prevents an intruder from accessing the DC using a stolen administrator account. Administrators are forced to logon locally where, if physical security is adequate, intruders cannot gain access.

Act as part of the Operating System
Allows a process to act as part of the trusted computing base. A modified service, uploaded in place of a system level service would allow a hacker to execute additional code at the operating system level. The only account that should have this right is the LocalSystem account.

Add workstation to domain
Users with this right, normally administrators, can add workstations and servers to the domain. This could allow an intruder with a stolen administrator account to add a domain controller and obtain a copy of the SAM database. This right should be limited to the administrators who are responsible for adding new workstations and servers to the domain.

Back-up files and directories
This privilege overrides all NTFS permissions and allows users with this right to read all files on the system. This right is required for a user to perform system back-ups. The user with this right does not need to be a member of the administrator group; a separate Backup Operator group should be created for this right.

Bypass traverse checking
This right allows a user to gain access to files and folders regardless of the permissions of the parent folder. This right should be given to administrators, server operators and backup operators.

Change the system time
Kerberos relies on accurate time. If the time is changed Kerberos authentication will be disabled. Additionally, accurate audits require a time standard across the network. Only administrators should have the right to change the system time.

Create a pagefile
Allows a user to add or change the pagefile. Normally only administrators should be permitted this right.

Create a token object
Allows the creation of a security access token. Processes that require this privilege should use the LocalSystem account instead of being given this right. No user should have this right.

Create permanent shared objects
This right allows the creation of special permanent objects, such as the floppy. Only the administrator should have this right.

Debug programs
Allows users to attach a debugger to any process. Also permits a user to modify programs. An intruder with a stolen account that has this privilege could insert and run malicious code. No one should have this privilege.

Deny access to this computer from the network
Forces administrators to logon locally to servers. Domain Controllers should enforce this right; member servers can allow administrator access from the network.

Deny logon as a batch job
Determines which accounts are prevented from being able to logon as a batch job. This setting supercedes the Logon as a batch job right.

Logon as a batch job
Allows a user to be logged on by a batch-queue facility. When a user submits a job by means of the task scheduler, the job is launched as a batch user rather than an interactive user. Only the LocalSystem account has this privilege.

Deny logon as a service
Determines which accounts are prevented from being able to logon as a service. This setting supercedes the Logon as a service job right.

Logon as a service
Determines which service accounts can register a process as a service. Accounts with this privilege can logon with full control of the system. Virus scanners require this right. No user accounts have this privilege.

Deny logon locally
Determines which users are prevented from logging on at the computer. By default, there are no accounts denied this ability.

Logon locally
Determines which users can logon to the computer. Servers should be restricted to administrators, server operators, and backup operators. Normal Users should not be given this right since there are hacker programs than can elevate users' permissions if run from the console.

Enable Computer and User accounts to be trusted for delegation
With this right a user can set the Trusted for Delegation setting on an object however, the user must have write access to the account control flags on the object. Misuse of this privilege could make the network vulnerable to sophisticated attacks using Trojan horse programs that impersonate incoming clients using their credentials to gain access to network resources. This right should be limited to administrators.

Force shutdown from a remote system
Users with this privilege can shutdown computers from across the network. This right should not be delegated on a domain controller; shutdowns should be done locally by an administrator to prevent regular users from creating a denial of service for the DC.

Generate security audits
Required for processes to be able to generate entries in the security log. Only the LocalSystem account has this right by default.

Increase quotas
Required to call the *CreateProcessAsUser* API function, which creates a new process running in the security context of another user. Only administrators should have this right.

Increase scheduling priority
Allows a user to change the base priority of a process. Setting a priority too high can consume system resources creating a denial of service on the computer. Only administrators should be given this privilege.

Load and unload device drivers
Determines which users can dynamically load and unload device drivers. A user with this privilege could load a modified driver, containing a Trojan horse, to the system. Device drivers should only be loaded by administrators.

Lock pages in memory
Allows pages to be locked in memory, pages locked in memory can not be pages to the pagefile on disk. Locking pages can create a denial of service attack. No one should have this privilege.

Manage auditing and security log
Allows viewing and clearing the audit logs. An intruder who steals an account with this privilege can clear the security log to erase his tracks. Only administrators should be permitted access to the audit logs.

Modify firmware environment values
This right is required to change the contents of a computer's NVRAM. Someone with this right could modify a variable such that it points to malicious programs. This right should be limited to administrators only.

Profile single process
With this right a user can access the performance counters for a specific non-system process. Since logging consumes resources, only administrators should have this right.

Profile system performance
With this right a user can access the performance counters for system processes. Since logging consumes resources, only administrators should have this right.

Replace a process level token
This token allows a user to replace the security token of a process with a different token, possibly allowing the process to run with a higher security level token. This right should only be assigned to LocalSystem account.

Restore files and directories
This privilege allows a user to set any valid or group SID as the owner of an object. This is required to restore files and folders previously backed-up. With this privilege and the backup privilege, a user could backup a malicious program and restore it over critical system files. This right should only be given to the user responsible for system restores.

Shutdown the system
Allows the logged-on user to shutdown the system. With this privilege a user could shutdown the system in the middle of critical jobs. Only administrators should be allowed to shutdown the DC.

Take ownership of files or other objects
With this right a user could take ownership of any file or object on the system, he could then modify the permissions to give himself full access. Administrators require this permission for system maintenance.

Local Policies – Security Options

The security templates also set several registry keys. Some of these registry keys do not exist in the default configuration and would require the administrator to create these keys individually. Using these templates reduces the chances that an administrator could make a mistake with a key and require reloading the operating system. As with all other registry modifications,

administrators should make a current back up of the registry prior to loading these templates, just-in-case.

Policy: *Audit access to global system objects*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Control\Lsa*
Value Name: *AuditBaseObjects*
Type: *REG_DWORD*
Value: Both templates set this to 0, do not audit the base objects (i.e. known dlls, data structure, and device names). Setting this to 1 tells the Local System Authority (LSA) to create base objects with the default system audit control list. Enabling this auditing would quickly fill the event log and, if configured for it, will crash the server (see CrashOnAuditFail).

Policy: Shutdown system immediately if unable to log security audits
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\lsa*
Value Name: *CrashOnAuditFail*
Type: *REG_DWORD*
Value: Both templates set this value to 0 – Feature is off, system will not halt even when it cannot record events in the Security Log. When coupled with the event log setting "Overwrite Events as Needed" when the log fills, it starts writing over previous events. A hacker could take advantage of this as a way to cover his tracks by flooding the log with a common event. If the Value is set to 1 – Feature on, the system will halt when it cannot record an event in the Security Log. This prevents further damage to the system and the hacker cannot wipe his tracks, but it also creates a denial of service for the server – if the hacker cannot do what he planned to do he could flood the log and crash your server. Before using either of these options the administrator needs to have an enforced policy of reviewing and backing up the event log on a regular basis. If the Value has been set to 2 then the feature is on and has been triggered. The system has halted and only members of the administrator group will be able to log on to clear the log and reset the value to 1.

Policy: *Audit use of backup and restore privilege*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Control\Lsa*
Value Name: *FullPrivilegeAudit*
Type: *REG_BINARY*
Value: Both templates set this to 0, which prevents auditing backup and restore events. If set to 1, enable auditing, it could create a massive event log as it records thousands of events, one for every folder and every file backed-up. This should be enabled only in high security environments. Users with the privilege to do backups have read access to every file on the system and should be audited. Administrators should be cautioned that enabling this will flood the Security Log and may need to back-up and clear the log after a system backup.

Policy: *Lan Manager Authentication level*
Settings: *Send LM & NTLM responses*
Send LM & NTLM – use NTLMv2 session security if negotiated
Send NTLM response only
Send NTLMv2 response only
Send NTLMv2 response only\refuse LM
Send NTLMv2 response only\refuse LM and NTLM
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Control\Lsa*
Value Name: *LmCompatabilityLevel*
Type: *REG_DWORD*
Value (securedc): 2, The server will initiate NTLM authentication, but will accept any, to include LM. Allowing LM authentication decreases server security because if the LM password is sniffed on the network it can easily be cracked with L0pht Crack.
Value (hisecdc) 5, This is most secure because it will not accept any other authentication except NTLMv2 authentication.
Possible registry values for this key are 0 – 5. Zero is the least secure, up through Value 5, which is the most secure and recommended for the high security environments.

Policy: *Additional restrictions for anonymous connections*
Settings: *None, rely on default permissions*
Do not allow enumeration of SAM accounts and shares
No access without explicit anonymous permissions
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Control\Lsa*
Value Name: *RestrictAnonymous*
Type: *REG_DWORD*
Value: Possible values are 0, 1, and 2. Value 0 indicates the restriction is not being enforced; anonymous users will not be restricted. Value 1 indicates the restriction is enabled; anonymous users will be granted limited access. Value 2 indicates anonymous users will have no access without explicit permissions.

The securedc template is set to 1 to allow restricted access, whereas the hisecdc template is set to 2 to deny anonymous access. Restrict anonymous should only be set to value 2 in environments running native mode WIN2K.

Policy: *Allow server operators to schedule tasks*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Control\Lsa*
Value Name: *SubmitControl*
Type: *REG_DWORD*
Value: Value 0 disables Server Operators and restricts task scheduling to Administrators and Power Users. To enable a Server Operator to schedule jobs the value needs to be set to 1. This key defines who can schedule processes. If a hacker knew what processes were scheduled

then he could replace a scheduled process by uploading a Trojan horse. When the scheduled time arrives, the uploaded process will be launched.

Policy: *Prevent Users from installing printer drivers*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Control\Print\Providers\LanMan Print Services\Servers*
Value Name: *AddPrinterDrivers*
Type: *REG_DWORD*
Value: Possible values are 0 and 1. Value 0 disables the prevention and allows all users to add new printer drivers. Value 1 enables the restriction that prevents users from adding drivers, but allows administrators and power users the privilege.

Both templates set this key to value 1. This key does not prevent a user from mapping to an available printer; it prevents him from adding a new driver. Adding new drivers runs in Kernel Mode. Should an operator replace a valid driver with a modified driver, the modified code could run at Kernal level.

Policy: *Clear virtual memory pagefile when system shuts down*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Control\SessionManager\MemoryManagement*
Value Name: *ClearPageFileAtShutdown*
Type: *REG_DWORD*
Value: Possible values are 0 and 1. Value 0 indicates the key has not been enabled, value 1 indicated the key has been enabled and will fill the inactive pages with zeros.

The securedc template does not enable this key, but the hisecdc does. Filling the inactive pages prevents the pages from being read by another process. The pagefile contains virtual memory when not in use and could contain usernames and passwords. If a computer were started in an alternate operating systems those passwords could be discovered in a sector-by-sector read. The system cannot fill the active pages because they may be in use by the system or another process.

Policy: *Strengthen default permissions of Global system objects (e.g. symbolic links)*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Control\Session Manager*
Value Name: *ProtectionMode*
Type: *REG_DWORD*
Value: Possible values are 0 and 1. Value 0 indicates the key is not enabled, 1 indicated the value has been enabled.

Both templates enable this key. Enabling this key prevents someone with programming knowledge to redefine system-wide resource attributes. By redefining certain attributes a user can deny access to other users. Additionally a knowledgeable user could load one of their own

dlls into memory, using the same name as a system dll, then change the entry point in the KnownDll list to point to their copy. When the dll is invoked by a privileged process, it can grant the user administrative rights.

Policy: *Digitally sign server communication (when possible)*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Services\LanManServer\Parameters*
Value Name: *EnableSecuritySignature*
Type: *REG_DWORD*
Value: Possible values are 0 and 1. Value 0 disables security signatures while value 1 enables security signatures. This key must be used in conjunction with the following key. Both templates set this value to 1 to enable signing SMBs.

Policy: *Digitally sign server communication (always)*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Services\LanManServer\Parameters*
Value Name: *RequireSecuritySignature*
Type: REG_DWORD
Value: Possible values are 0 and 1. The securedc template sets the key to 0 to disable the security signature. The hisecdc template sets the value to 1, which requires security signatures.

These two keys can enable security signatures for Server Messaging Blocks (SMB). Security signatures at each end are a countermeasure to man-in-the-middle attacks. If security signatures are required at each end then the server will only respond to clients with message signing. The securedc template does not require security signature to be compatible with down-level file systems (FAT), but will accept security signatures from NTFS clients. Administrators must ensure that if security signatures are required, that the key to enable security signatures must also be enabled. Setting Require to 1, while Enable is set to 0, prevents all access to server SMB shares.

Policy: *Automatically logoff users when logon time expires*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Services\LanManServer\Parameters*
Value Name: *EnableForcedLogOff*
Type: *REG_DWORD*
Value: Possible values are 0 and 1; both templates set this key to 1. When set to 1, if the User Policy is set to restrict the time and hours a user can logon, then when the logon time expires the user is forced off. If company policy dictates logon hours then this key needs to be enabled.

Policy: *Amount of idle time required before disconnecting session*
Settings: *0 – 99,999 minutes*

Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Services\LanManServer\Parameters*
Value Name: *AutoDisconnect*
Type: *REG_DWORD*
Value: Possible values from 0x0 through to 0xFFFFFFFF. 0x0 indicates the inactive connections can never be disconnected automatically. 0x1 through 0xFFFFFFFF (4294967295) are how many seconds the inactive connection can stay open before being closed. A setting of 0x1 would disconnect the connection almost immediately (1 second), where FFFFFFFF would keep the connection open for approx. 136 years. If the administrator were to configure this option at the command line, and choose to disable the AutoDisconnect, he would set the value to –1. This sets AutoDisconnect to the upper value in the registry, which is approx. 8,171 years, essentially turning off the AutoDisconnect.

Both templates set this key to 15 minutes. If the connection does not transfer any NetBIOS session data (i.e. file copying, network resource access, or e-mail) it is considered inactive and will be closed.

Process: *Prevent system maintenance of computer account password*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Services\Netlogon\Parameters*
Value Name: *DisablePasswordChange*
Type: *REG_DWORD*
Value: Possible values are 0 and 1. Value 0 disables the key and allows the system to automatically change the computer account password; value 1 enabled the key and disables the ability to automatically change the password.

Both templates set this key to 0, allowing the system to change the password regularly. Changing the password regularly reduces system vulnerability.

The following four keys work together.
Policy: *Secure Channel: Digitally sign secure channel data (whenever possible)*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Services\Netlogon\Parameters*
Value Name: *SignSecureChannel*
Type: *REG_DWORD*
Value: Possible values are 0 and 1. This key determines if outgoing secure channel traffic is signed. Value 0 disables the signing requirement and outgoing traffic need not be signed; value 1 enables signing of outgoing traffic. This entry is used only when the value of RequireSignOrSeal is 0. Otherwise, the system requires that traffic is signed, and it does not consult this entry. This key is superceded when the SealSecureChannel value is set to 1. Both templates set this value to 1, require sign.

Policy: *Secure Channel: Digitally encrypt secure channel data (whenever possible)*
Settings: *Enabled or disabled*

Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Services\Netlogon\Parameters*
Value Name: *SealSecureChannel*
Type: *REG_DWORD*
Value: Possible values are 0 and 1. This key determines if outgoing traffic must be encrypted (sealed). Value 0 disables outgoing traffic encryption; value 1 indicates outgoing traffic must be encrypted. When RequireSignOrSeal is set to 1, SealSecureChannel determines whether encryption is required. When RequireSignOrSeal is set to 0, SealSecureChannel specifies the system's preferences when negotiating with the domain controller on the other side of the channel. When set to 1 this key takes precedence over SignSecureChannel. Both templates set this key to 1 to require channel encryption.

Policy: *Secure Channel: Digitally encrypt or sign secure channel data (always)*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\CurrentControlSet\Services\Netlogon\Parameters*
Value Name: *RequireSignOrSeal*
Type: *REG_DWORD*
Value: Possible values are 0 and 1. When set to 0 channel traffic need not be signed or sealed. System preferences when negotiating with other domain controllers are determined by the values of SignSecureChannel and SealSecureChannel. The securedc template disables this key with value 0, and the hisecdc template enables the key with value 1.

Policy: *Secure Channel: Require strong (Windows 2000 or later) session key*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\currentControlSet\Services\Netlogon\Parameters*
Value Name: *RequireStrongKey*
Type: *REG_DWORD*
Value: Determines whether the system requires that all secure channel keys be computed using a strong key. Possible values are 0 and 1. The securedc template sets this to 0, disabling the requirement for the trusted domain controller be able to compute a strong key. The hisecdc template sets the value to 1; requires the trusted domain controller be able to compute a strong key. It the domain controller on the other side of the channel does not support strong key encryption; this system refuses to establish a channel. This should only be set to 1 when all of the trusted domains are able to compute strong keys.

Policy: *Disable CTRL+ALT+DEL requirement for logon*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows\CurrentVersion\Policies\System*
Value Name: *DisableCAD*
Type: *REG_DWORD*
Value: Determines whether users must press the CTL+ALT+DEL security attention sequence to log on to Windows 2000. Both templates disable the key by setting the value to 0,

which requires the user to press CTL+ALT+DEL to logon to the system. The other value, 1 enables the disable and suppresses the "Press Ctl+Alt+Del to begin" message.

Suppressing the Ctl+Alt+Del sequence can compromise system security. Only Windows responds to this sequence and guarantees that passwords entered after the sequence are sent only to Windows. Eliminating the security attention sequence can allow malicious programs to request and receive your Windows password.

Policy: *Do not display last user name in logon screen*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows\CurrentVersion\Policies\System*
Value Name: *DontDisplayLastUserName*
Type: *REG_DWORD*
Value: Enabling this value, set to1, prevents the last user's name from being displayed in the log-in box. This is not much of a deterrent to hackers because most organizations have a naming convention that would not be too hard to guess. However, by removing the previous user's name from the log-in box forces the next user to enter their name and not assume that they were the last person to log-on, this can cut down on the number of calls to your helpdesk. Leaving this value set at default, 0 displays the name of the last person to successfully log-in. The securedc template leaves the default 0 for ease of use, but the hisecdc template sets the value to 1 to obscure the name in the event the physical security of the server is compromised.

Policy: *Message title for users attempting to logon*
Settings: *User defined text block*
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows\CurrentVersion\Policies\System*
Value Name: *LegalNoticeCaption*
Type: *REG_SZ*
Value: Provides a heading for the warning banner displayed when a user presses CTL+ALT+DEL during logon. Both templates leave the value to this key blank. The administrator needs to add a title; most appropriate would be "Warning". Maximum size for the caption is approx 65 characters.

Policy: *Message text for users attempting to logon*
Setting: *User defined text block for "Consent to Monitor" banner*
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows\CurrentVersion\Policies\System*
Value Name: *LegalNoticeText*
Type: *REG_SZ*
Value: Provides the capability for an administrator to create a message that will be displayed when a user attempts to logon. Both templates leave the value for this key blank. When determining what message to display, administrators should contact the corporate legal office. The message displayed should reflect the corporate policy on authorized system use, and be supportable in a court of law if the policy is violated. When this key is configured the user must acknowledge the message before logon.

Administrators should be cautious in the words they choose for the banner. In previous court cases, intruders have been able to escape prosecution because the banner started with Welcome.

Policy: *Allow system to be shut down without having to log on*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *System\Microsoft\Windows\CurrentVersion\Policies\System*
Value Name: *ShutdownWithoutLogon*
Type: *REG_DWORD*
Value: Determines if the Shutdown button is available in the Logon to Windows dialog box. With the value set to 0, the button is present, but dimmed and does not operate. When enabled a user could shutdown the operating system without logging on to the server. Both templates set this value to 0 preventing unauthorized people from shutting down the DC.

Policy: *Restrict CD-ROM access to locally logged-on user only*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows NT\CurrentVersion\Winlogon*
Value Name: *AllocateCDRoms*
Type: *REG_SZ*
Value: This key determines if domain administrators can access data on the disk in the CD-ROM drive from across the network. When set to 0 (default) administrators can gain access to the CD-ROM data. When enabled, access to the data on the CD-ROM is restricted to only the user logged on locally. Both templates set the value to 1 to protect the users' data. Administrators should be cautioned that once the user logs off, the drive is available across the network and any data on the disk becomes vulnerable.

Policy: *Allowed to eject removable NTFS media*
Settings: *Administrators*
Administrators and Power Users
Administrators and Interactive Users
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows NT\CurrentVersion\Winlogon*
Value Name: *AllocateDASD*
Type: *REG_SZ*
Value: Determines which users can format and eject removable hard disks. Possible values are 0, 1, and 2. Value 0 limits this capability to administrators. Value 1 expands this to include power users too. And value 2 allows the local current user the privileges to format and eject the removable hard drive. Both templates set the value to 0 so only administrators can remove the controller's hard drive.

Policy: *Restrict floppy access to locally logged-on user only*
Settings: *Enabled or disabled*
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows NT\CurrentVersion\Winlogon*
Value Name: *AllocateFloppies*
Type: *REG_SZ*
Value: Like CDRoms, by default floppies are shared as an administrator share across the

network. The default value 0 allows administrators access to floppies from across the network, whereas the value 1 limits access to the locally logged-on user. The templates set the value to 1 to protect the user's data. Administrators should be cautioned that once the user logs off, the drive is shared again and data on floppies in the drive becomes vulnerable.

Policy: *Number of previous logons to cache*
Settings: *Range from 0-50*
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows NT\CurrentVersion\Winlogon*
Value Name: *CachedLogonsCount*
Type: *REG_SZ*
Value: Determines how many user accounts will be cached on the local computer. The cached data includes a user's permissions and authorities and will be used in the event the domain controller is unavailable. This maintains security of the system by preventing a user from logging on to a workstation that has been removed from the network and gaining privileges not assigned to the user. The range for this key is 0 to 50 accounts. Administrators should be careful about how many accounts are cached; more cached accounts require more disk space. The templates set this value to a moderate 10.

Policy: *Prompt user to change password before expiration*
Settings: *Range from 0 to 999 days*
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\WindowsNT\CurrentVersion\Winlogon*
Value Name: *PasswordExpiryWarning*
Type: *REG_DWORD*
Value: Range for this key is 0x0 through 0xFFFFFFFF.

Both templates set the warning period to 14 days. In most cases this will ensure each user gets notified that their password is about to expire, even if they are about to go on vacation.

Policy: *Smart Card removal behavior*
Settings: *No action*
Lock workstation
Force logoff
Hive: *HKEY_LOCAL_MACHINE*
Key: *Software\Microsoft\Windows NT\CurrentVersion\Winlogon*
Value Name: *ScRemoveOption*
Type: *REG_SZ*
Value: This key is set to 0, not enabled, by default. There are two options that can be enabled for this key; value 1 locks the computer allowing the user to walk away with the smart card and still maintain a secure session, value 2 logs off the user when the smart card is removed. Both templates set the value to 2 to log off the computer.

Local Policies – Security Options (no registry settings)

Policy: *Recovery Console: Allow automatic administrative logon*
Settings: *Enabled and disabled*
Allows the system to be booted into the recovery console without needing to use the administrator password. This creates a significant security risk in that if the machine becomes physically compromised.
Policy: *Recovery Console: Allow floppy copy and access to all drives and folders*
Settings: *Enabled and disabled*
Enabling this feature allows unrestricted access to the server when booted to the recovery console. Disable the feature to protect your system, you will still have access to the root directory and to parts of the WINNT folder when using recovery console. Disabling will still allow copying files from a floppy to the system, but not from the system to the floppy.

Policy: *ForceLogoffWhenHourExpires (local policy)*:
Settings: Enabled or disabled.
Any organization where users are restricted, by time of day, from accessing their accounts should enable the forced logoff. If a user is only expected to work from 8:00am to 5:00pm and the account is set to allow access from 6:00am to 7:00pm then at 7:00 the user will be logged off the workstation. This prevents users from using company resources after hours, and possibly stealing company data or performing other unauthorized actions on the network.

Event Log – Log Settings

Security Log

RestrictGuestAccess: The Event Log will be an administrator's primary tool when investigating a possible break-in. The logs, if configured correctly, will provide a trail of where an intruder went, and what he did. Because of the amount of information the logs will provide to an administrator, guests should be restricted from having access to the logs. Allowing guest access is also allowing null user access, which is how an intruder would normally be seen. If the log permitted read access (delete restricted to administrators) then the intruder could watch the log to see what trail he was leaving. He could also determine if the system was set to halt if the log filled (more on this later). Both templates enable this setting and restrict the guest, null, user.

MaximumLogSize: The more objects you choose to audit, the larger the audit log needs to be. If the log size is set too small then it will fill quickly. The basic setting for this entry is 512K. This is appropriate for the System and Application logs, which is why they are not addressed beyond default in these templates, but will not hold enough data for the Security Log. The securedc template expands the size of the Security Log to 5120K, and the hisecdc template expands this further to 10240K. Depending on how the registry setting is configured (see *CrashOnAuditFail*) you will either lose valuable information about the intruder's actions, or if configured for it the server will crash.

AuditLogRetentionPeriod: By default the audit log is retained for seven days. After which it can

be overwritten by new events. Neither template extends the log period for the System or Application logs, but the Security log is extended to indefinite. Setting it to indefinite prevents the log from being overwritten by new events; the log must be cleared manually. Configuring the log to be overwritten by newer events would allow an intruder to flood the log with arbitrary errors, covering his tracks. Preventing the log from being overwritten would prevent identifying an intruders actions should the log be filled previously by normal events, or the intruder could flood with arbitrary errors before the actual attack. In either case the administrator would have to have an enforced policy of backing up and clearing the logs. Keeping the logs cleared would make it more difficult for the intruder to cover his tracks, and maintaining the backups would give a history of events and may be useful later when looking for early signs of a subtle attack.

System Log, Application Log

RestrictGuestAccess: As with the Security Log, the System and Application logs need to be restricted from access by guest, or null, users.

This is the only setting for the Security and System logs configured by the securedc and hisecdc templates. However the administrator can make changes to the other settings and export the settings as part of the corporate template.

The configuration described in this paper, whether they are the default, securedc, hisecurdc, or some of the settings I recommended, may not be appropriate in all situations. Before applying any of the settings administrators need to compare the settings to the corporate policy. Users' requirements, administrators' needs, and results of an in-depth risk assessment will impact each setting. Whenever possible, registry settings should be changed by selecting one of the Group Policy options. This prevents typos, or incorrect settings which could require system reload to correct.

Resources

Fossen, Jason. Windows 2000: Active Directory and Group Policy (Track 5.5). Baltimore: SANS Institute, May 2001

Boswell, Bill. Windows 2000: How it Works (Track 5.4). Baltimore: SANS Institute, May 2001

Norberg, Stefan. Securing Windows NT/2000 Servers for the Internet. Sebastopol: O'Reilly & Associates, Inc, 2001

Schmidt, Jeff. Microsoft Windows 2000 Security Handbook. Indianapolis: QueCorp, 2000

Hipson, Peter D. Mastering Windows 2000 Registry, Alameda: Sybex, Inc., 2000

Minasi, Mark; Anderson, Christa; Smith, Brian M.; Toombs, Doug. Mastering Windows 2000 Server, Alameda, 1999

Consensus document. Securing Windows 2000 Step-by-step (Preliminary Edition) v1.0. SANS Institute, 2001

Consensus document. Windows NT Security Step-by-step v3.03, SANS Institute, 2001

Microsoft Corp. "Microsoft 2000 Server, Resource Kit: Supplement 1", http://www.microsoft.com/WINDOWS2000/techinfo/reskit/en/Regentry/, July 2001

Microsoft Corp. MSDN Library, Multiple articles, http://www.msdn.microsoft.com/library/, July 2001

Microsoft Corp. Knowledge Base, Multiple articles, http://support.microsoft.com/directory/, July 2001