Comprehensive Review of Windows 2000 Security Policy Templates and Security Configuration Tool

March 6, 2001

David B. Koconis
SD453373

Introduction

Windows 2000 provides a laundry list of new features, including ones that are advertised to greatly enhance the security of individual server systems and the network communications of the enterprise[1]. These security-related features include Active Directory service, support for the Kerberos version 5 authentication protocol, authentication using public key certificates, Encrypting File System (EFS) for protection of local data, and support for secure communication across public networks using Internet Protocol Security (IPSec). The presence of these features enables the security-conscious system administrator to implement his prepared security policy in its entirety, thereby constructing a highly secure enterprise network. In a Windows 2000 environment, the first and most important step in accomplishing this is an appropriately configured security policy template.

Scope and Intent

This document begins with a brief discussion of what security policy templates are; including where they are located, how to view and customize them, and what security settings are accessible from templates. The reader will become familiar with the structure of a security template.

The document then presents the Microsoft Management Console Security Configuration and Analysis snap-in, a graphical user interface (GUI) tool for working with security templates. The user-friendly way to apply the security policies defined in a security policy template is by using this snap-in. This section also demonstrates how the snap-in can be used to audit the current security state of a system by comparing it with a security policy template database.

The next section contains a detailed discussion of the more than twenty pre-configured security policy templates found in an out-of-the box Windows 2000 installation. The document describes how the design of many of the templates is tailored to achieve a specific security stance for a specific type of system. In addition, special purpose templates are discussed that narrowly configure only certain areas of the system. Lastly, a pre-configured template for a highly secure Internet Information Server system is reviewed

The last section gives a detailed look at the engine behind the scenes of the GUI tool, a command-line program secedit. Using the command-line tool, a system administrator can perform all the actions that are possible with the GUI tool, and more. Some examples are discussed that demonstrate how secedit can be used.

The Security Configuration and Analysis snap-in and command-line secedit tool can only apply security policy templates on one workstation or server at a time. What is not presented in this document is how to propagate the security policies throughout the enterprise. This propagation is done using Group Policy Objects (GPO) and for a detailed review of how it is done, refer to the Microsoft Step-by-Step documentation on this topic[2].

Given the power and flexibility, as well as, the obvious security implications of the Windows 2000 Security Templates and Configuration Tools, a system administrator who is conscious of network security (as all system administrators should be) would be remiss to not understand fully and take advantage of these built-in tools. The intent of this document is to assist a Windows 2000 system administrator in attaining a higher level of understanding of these fundamental tools.

Security Policy Templates

What is a security policy template and for what can it be used? The following description is taken directly from the documentation provided at the Microsoft Windows 2000 web site[3]:

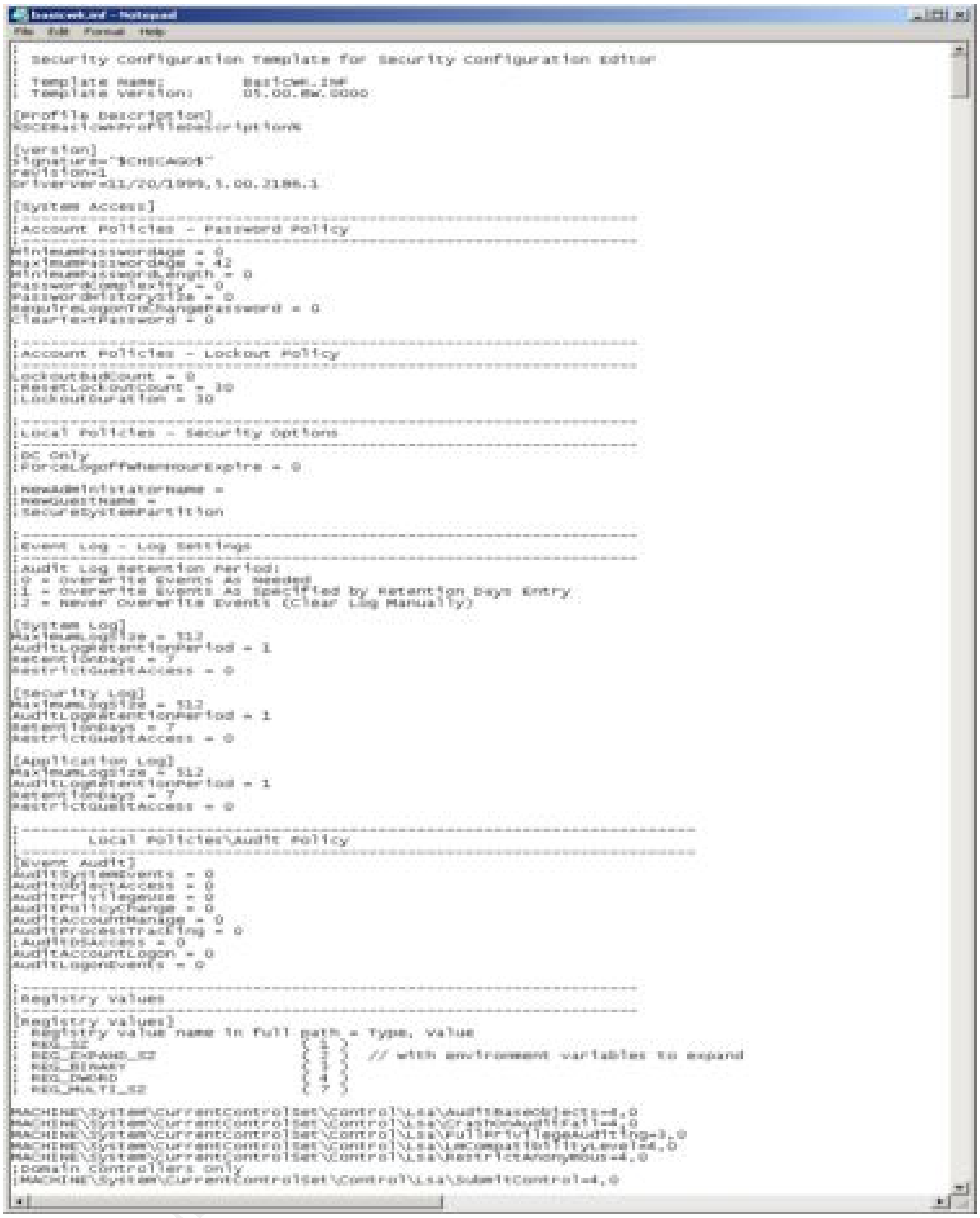
> Windows 2000 provides a centralized method of defining security with the Security Template … It is a single point of entry where the full range of system security can be viewed, adjusted, and applied to a local computer …

It is clear from the quote that Microsoft intended a security template to be an integral part of any implementation of any security policy. In fact, during the Windows 2000 Professional or Server installation procedure, a security policy template is used to configure the security settings of a system, including enforcing password and account lockout policies, configuring auditing, enforcing appropriate permissions on certain Registry items, setting up correct access control lists (ACL) for relevant areas of the filesystem, and enabling or disabling services for the system.

As can be seen in Figure 1, a security template is nothing more than a text file. In the file, a semicolon (;) at the beginning of a line denotes a comment. The sections of the file corresponding to the security areas being configured are headed by text in brackets (*e.g.* [System Log]). These headings are very important, as the option will be ignored if it does not appear in a section with the appropriate header. Finally, the lines describing the version of the file must be included, or the template will be corrupted. Keeping all of these restrictions in mind, an example of a minimal security policy template that could be successfully applied is the following:

```
[version]
signature="$CHICAGO$"
[System Access]
MinimumPasswordLength = 8
```

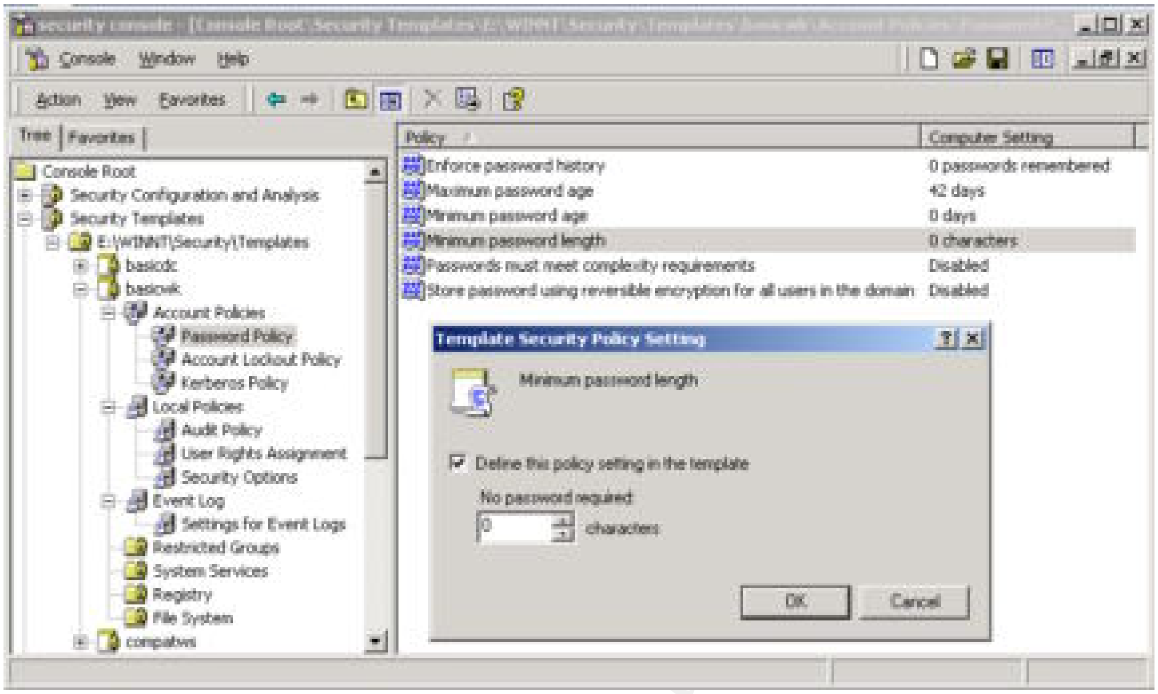
If applied (details on how this can be done are discussed later), this template would only enforce that all passwords must have at least eight (8) characters. All other template settings would be undefined.



```
;
; Security Configuration Template for Security Configuration Editor
;
; Template Name:        BasicWk.INF
; Template Version:     05.00.BW.0000

[Profile Description]
%SCEBasicWkProfileDescription%

[version]
signature="$CHICAGO$"
revision=1
DriverVer=11/20/1999,5.00.2186.1

[System Access]
;----------------------------------------------------------------
;Account Policies - Password Policy
;----------------------------------------------------------------
MinimumPasswordAge = 0
MaximumPasswordAge = 42
MinimumPasswordLength = 0
PasswordComplexity = 0
PasswordHistorySize = 0
RequireLogonToChangePassword = 0
ClearTextPassword = 0

;----------------------------------------------------------------
;Account Policies - Lockout Policy
;----------------------------------------------------------------
LockoutBadCount = 0
;ResetLockoutCount = 30
;LockoutDuration = 30

;----------------------------------------------------------------
;Local Policies - Security Options
;----------------------------------------------------------------
;DC Only
;ForceLogoffWhenHourExpire = 0

;NewAdministratorName =
;NewGuestName =
;SecureSystemPartition

;----------------------------------------------------------------
;Event Log - Log Settings
;----------------------------------------------------------------
;Audit Log Retention Period:
;0 = Overwrite Events As Needed
;1 = Overwrite Events As Specified by Retention Days Entry
;2 = Never Overwrite Events (Clear Log Manually)

[System Log]
MaximumLogSize = 512
AuditLogRetentionPeriod = 1
RetentionDays = 7
RestrictGuestAccess = 0

[Security Log]
MaximumLogSize = 512
AuditLogRetentionPeriod = 1
RetentionDays = 7
RestrictGuestAccess = 0

[Application Log]
MaximumLogSize = 512
AuditLogRetentionPeriod = 1
RetentionDays = 7
RestrictGuestAccess = 0

;--------------------------------------------------------------------
;       Local Policies\Audit Policy
;--------------------------------------------------------------------
[Event Audit]
AuditSystemEvents = 0
AuditObjectAccess = 0
AuditPrivilegeUse = 0
AuditPolicyChange = 0
AuditAccountManage = 0
AuditProcessTracking = 0
;AuditDSAccess = 0
AuditAccountLogon = 0
AuditLogonEvents = 0

;----------------------------------------------------------------
;Registry Values
;----------------------------------------------------------------
[Registry Values]
;  Registry value name in full path = Type, Value
;  REG_SZ                         ( 1 )
;  REG_EXPAND_SZ                  ( 2 )  // with environment variables to expand
;  REG_BINARY                     ( 3 )
;  REG_DWORD                      ( 4 )
;  REG_MULTI_SZ                   ( 7 )

MACHINE\System\CurrentControlSet\Control\Lsa\AuditBaseObjects=4,0
MACHINE\System\CurrentControlSet\Control\Lsa\CrashOnAuditFail=4,0
MACHINE\System\CurrentControlSet\Control\Lsa\FullPrivilegeAuditing=3,0
MACHINE\System\CurrentControlSet\Control\Lsa\LmCompatibilityLevel=4,0
MACHINE\System\CurrentControlSet\Control\Lsa\RestrictAnonymous=4,0
;Domain Controllers Only
;MACHINE\System\CurrentControlSet\Control\Lsa\SubmitControl=4,0
```

Figure 1 – Example of a Security Policy Template

Working with text files can be very powerful, especially when combined with command line tools (as will be discussed later), but there is a more user-friendly way to view and edit security policy templates. Figure 2 shows how the template shown in Figure 1 can be viewed using the Microsoft Management Console (MMC) Security Templates snap-in.



Policy	Computer Setting
Enforce password history	0 passwords remembered
Maximum password age	42 days
Minimum password age	0 days
Minimum password length	0 characters
Passwords must meet complexity requirements	Disabled
Store password using reversible encryption for all users in the domain	Disabled

Minimum password length

☑ Define this policy setting in the template

No password required

0 characters

Figure 2 –Security Templates (MMC Snap-In) View of a Security Policy Template

As can be seen in the figure, the snap-in provides a GUI interface to edit each policy item in the template. To bring up the dialogue box that is overlaid on the figure, select a policy item in the right-hand frame (in this case Minimum password length), click with the right mouse button, and select "Security…" from the resulting pop-up menu. Note that when the setting in the dialogue box is changed, the new value is not saved to the template unless the check box is selected. Also, to save the value to disk, the template must be saved. In order to save a template, highlight the template name, click with the right mouse button, and select "Save" from the resulting pop-up menu.

Previously, a very minimal, customized security policy template was presented that could be easily created using a text editor. However, if a more populated, customized template were needed, creating it from scratch using a text editor would be undesirable. A better way to create one would be to locate the existing, pre-configured template that is closest to the desired one, copy it to a new name, and make the necessary modifications using the snap-in tool. The method to copy a template using the MMC tool is the same as saving one, with the exception that "Save As…" should be selected from the pop-up menu instead of "Save".

Figure 2 also shows that each security policy template is divided into a standard, default set of categories: Account Policies, Local Policies, Event Log, Restricted Groups, System Services, Registry, and File System. Table 1 includes a brief description of each category and a sample entry.

Category	Items in the Category	Sample Entry

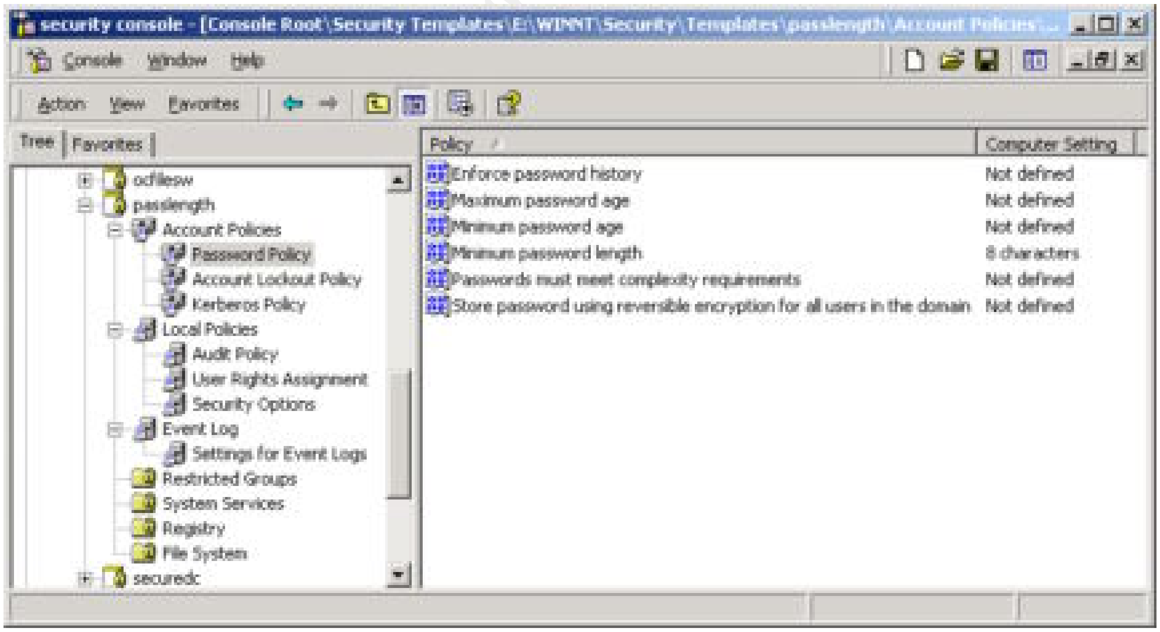
Account Policies	Password policy (history, age, length, complexity and encryption type).
	Lockout Policy (duration, threshold, and counter reset).
	Kerberos Policy (logon restrictions, ticket lifetimes, clock synchronization tolerance).
		Minimum PasswordAge = 2
		LockoutDuration = 30
		MaxTicketAge= 10

Local Policies	Audit Policy (log success and/or failure for certain events).
	User Rights Assignment (restrict what groups can perform what actions).
	Security Options (security related Registry options).
		AuditPrivilegeUse = 2
		SeTakeOwnershipPrivilege = Administrators
		MACHINE\System\CurrentControlSet\Control\Lsa\RestrictAnonymous=4,2

Event Log	Configurations for event logs (Maximum size, guest access, retention policy)	RestrictGuestAccess = 1
Restricted Groups	Local Group Membership Administration	Users__Members = AuthenticatedUsers, INTERACTIVE
System Services	Security and Start-up Mode for local Services	Fax Service - Disabled
Registry	Any user-defined local Registry key	MACHINE\SOFTWARE\Classes\helpfile
File System	Any user-defined local file or directory	%SystemRoot%\explorer.exe

Table 1 – Description of Categories Included in Every Security Policy Template

The complete set of categories is displayed in the GUI tool when any selected template is selected, regardless of whether the template defines any of the values in that category or not. Figure 3 shows the minimal security template presented earlier that only sets the password length to 8.



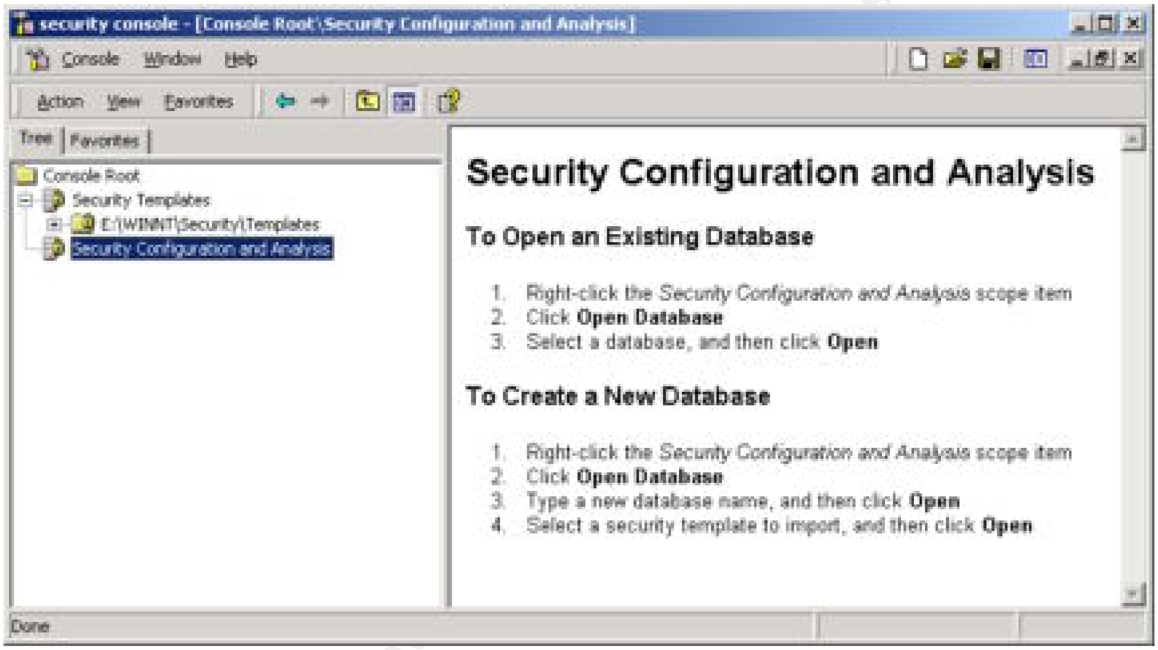
Policy	Computer Setting
Enforce password history	Not defined
Maximum password age	Not defined
Minimum password age	Not defined
Minimum password length	8 characters
Passwords must meet complexity requirements	Not defined
Store password using reversible encryption for all users in the domain	Not defined

Figure 3 – A Minimal Security Policy Template Restricting Password Length Only

As shown in the figure, all settings in the Password Policy section of the template are displayed, even though only one is set. Policy items that are not specifically defined in the template are displayed as “Not defined”.

Security Configuration and Analysis Tool

The definition of an appropriate security policy template is only the beginning. This section discusses the MMC Security Configuration and Analysis Tool, the GUI that can be used to audit the existing security configuration on a system or configure the system to conform to a selected security template or templates. Figure 4 gives a snapshot of the tool once it has been added to the MMC Console.


security console - [Console Root\Security Configuration and Analysis]
Console
Window
Help
Action
View
Favorites
Tree
Favorites
Console Root
Security Templates
E:\WINNT\Security\Templates
Security Configuration and Analysis
Done

Security Configuration and Analysis

To Open an Existing Database

1. Right-click the *Security Configuration and Analysis* scope item
2. Click **Open Database**
3. Select a database, and then click **Open**

To Create a New Database

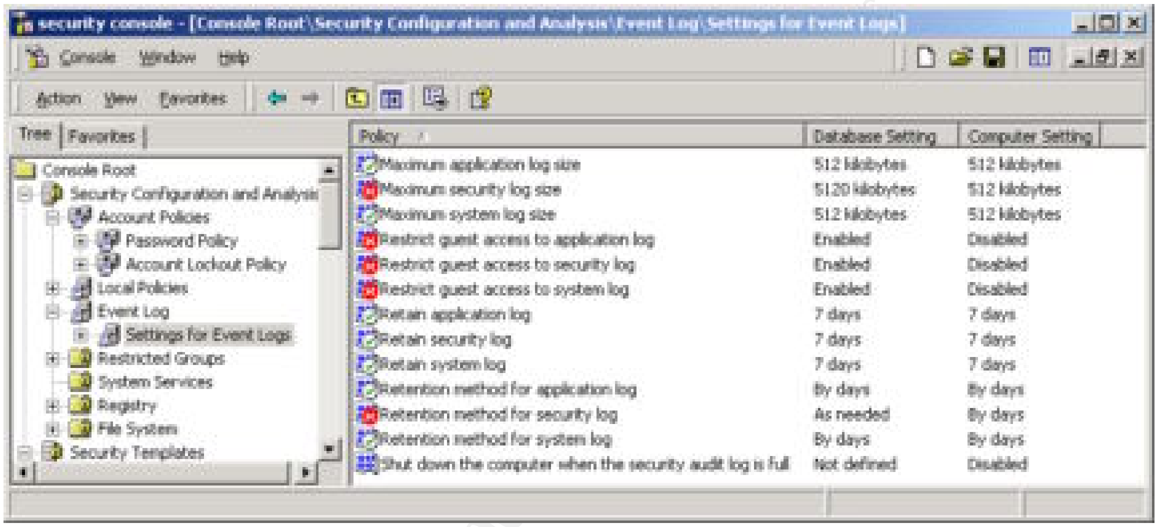
1. Right-click the *Security Configuration and Analysis* scope item
2. Click **Open Database**
3. Type a new database name, and then click **Open**
4. Select a security template to import, and then click **Open**

Figure 4 - Security Configuration and Analysis Tool Splash Page

The step-by-step procedure for using the Security Configuration and Analysis Tool to perform an analysis on a system or to configure a system will not be repeated here, since the Microsoft Step-by-Step guide covering the use of the GUI tool is very good and quite thorough[4]. However, a few notes about the tool are included here for clarity.

The tool does not work directly from a security policy template file, but instead uses a database in order to analyze or configure the security settings on a system. Instructions for opening an existing database or creating a new one are given in the right-hand frame of the snap-in when it is opened for the first time (see the Figure 4). Use of a database increases the capability and flexibility of the tool because it enables layering of multiple templates one over another. This simulates the method of incremental application of security policy templates. As will be discussed in the next section, most of the pre-configured Windows 2000 templates are designed to be used in this manner. For example, the appropriate database for a secure or highly secure configuration of a workstation can be created by first importing the basic workstation template (*basicwk*), and then importing the secure (*securews*) or highly secure (*hisecws*) template, respectively.

Once a database has been constructed, it can be used to either analyze the current security state of the system or configure the security state of the system. Analysis is simply an audit that compares the security settings active on the system to the ones defined in the database. To perform an analysis, right click on the Security Configuration and Analysis item, select "Analyze Computer Now …" from the pop-up menu, specify or select the name of the log file path in the dialogue box, and then click OK. Figure 5 shows a sample result window from an analysis.



Policy	Database Setting	Computer Setting
Maximum application log size	512 kilobytes	512 kilobytes
Maximum security log size	5120 kilobytes	512 kilobytes
Maximum system log size	512 kilobytes	512 kilobytes
Restrict guest access to application log	Enabled	Disabled
Restrict guest access to security log	Enabled	Disabled
Restrict guest access to system log	Enabled	Disabled
Retain application log	7 days	7 days
Retain security log	7 days	7 days
Retain system log	7 days	7 days
Retention method for application log	By days	By days
Retention method for security log	As needed	By days
Retention method for system log	By days	By days
Shut down the computer when the security audit log is full	Not defined	Disabled

Figure 5 – Output from a Security Configuration Analysis

As shown in the figure, a quick glance at the results reveals the settings that are in compliance with the database (a white circle with a green check mark) and the ones that are in violation of the database setting (a red circle with a white X). The ones that are not defined in either the database or the current computer setting are unmarked. In addition to the GUI display, a plain text log file containing the complete results is also created.

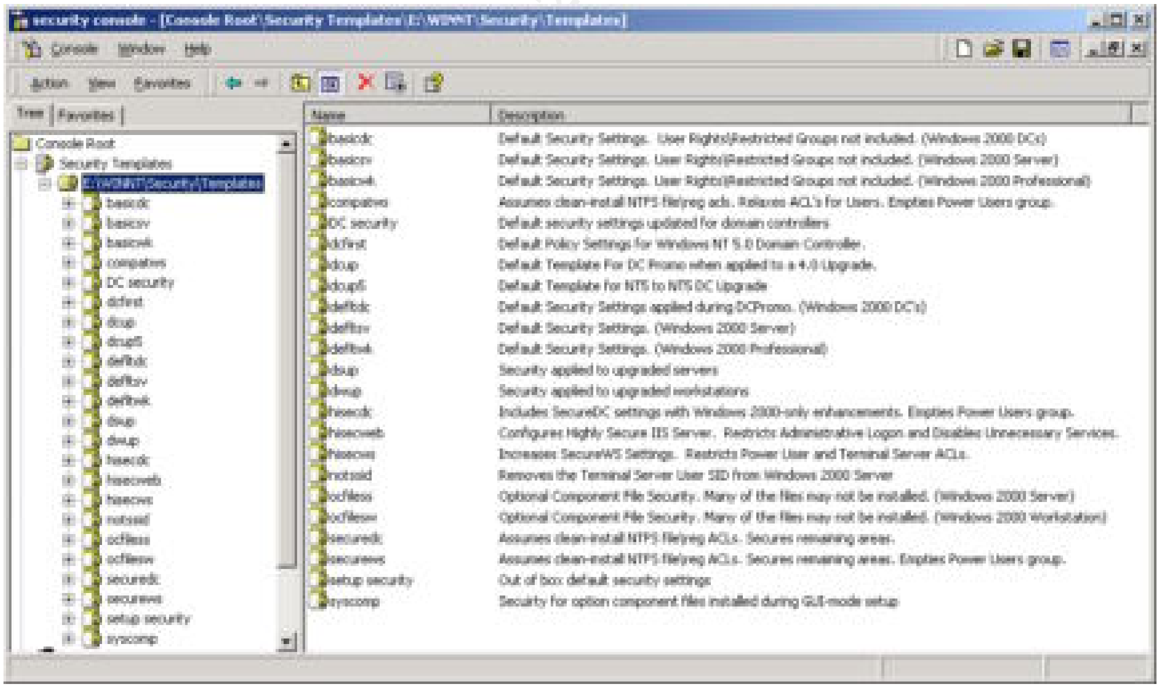
When ready, the snap-in can be used to configure the system, and the security settings from the current database will be applied to the system. Simply select "Configure Computer Now …" from the pop-up menu, specify or select the name of the log file path in the dialogue box, and then click OK.

Finally, there is a command line tool available (secedit.exe) that can perform all of the functions that the Security Configuration and Analysis Tool can and more. A discussion of the many options available using secedit, as well as some examples, is presented in the section following the discussion of pre-configured templates.

Pre-Configured Security Templates

This section presents a detailed review of all the pre-configured security policy templates available for Windows 2000 Professional, Server and Advanced Server. The default installation of Windows 2000 will automatically install several pre-configured security policy templates into the *%SystemRoot%\Security\Templates* directory. These are the templates that will appear in the right-hand frame of the Security Template MMC by default, and unless otherwise specified, all templates discussed in this section can be found there. In addition to these, several other templates that are used to configure the security of the system during the installation of the operating system and the promotion of servers to domain controllers are stored in the *%SystemRoot%\inf* directory. There are two ways to get these templates to appear in the Security Templates snap-in. Either copy them to the *%SystemRoot%\Security\Templates* directory or add a new search path to the snap-in that points to the *%SystemRoot%\inf*. The former option is better, since the snap-in assumes that any file with a **.inf* suffix is a security policy template and it will attempt to load it. The *%SystemRoot%\inf* directory is full of **.inf* files, only a handful of which are security policy templates. If the latter option is chosen, the right hand frame of the snap-in will become cluttered and a nuisance to sift through.

Finally, a specially designed template for configuring a highly secure Internet Information Server (IIS) system that can be obtained from the security area of the Microsoft Web Site is also discussed. All in all, 23 templates have been discovered and are shown in Figure 6 along with a one-line description for each template. A summary of the naming convention for the templates appears in Table 2, following the figure.


security console - [Console Root\Security Templates\E:\WINNT\Security\Templates]
Console
Window
Help
Action
View
Favorites
Tree
Favorites
Console Root
Security Templates
E:\WINNT\Security\Templates

Name	Description
basicdc	Default Security Settings. User Rights\Restricted Groups not included. (Windows 2000 DCs)
basicsv	Default Security Settings. User Rights\Restricted Groups not included. (Windows 2000 Server)
basicwk	Default Security Settings. User Rights\Restricted Groups not included. (Windows 2000 Professional)
compatws	Assumes clean-install NTFS file\reg acls. Relaxes ACL's for Users. Empties Power Users group.
DC security	Default security settings updated for domain controllers
dcfirst	Default Policy Settings for Windows NT 5.0 Domain Controller.
dcup	Default Template For DC Promo when applied to a 4.0 Upgrade.
dcup5	Default Template for NT5 to NT5 DC Upgrade
defltdc	Default Security Settings applied during DCPromo. (Windows 2000 DC's)
defltsv	Default Security Settings. (Windows 2000 Server)
defltwk	Default Security Settings. (Windows 2000 Professional)
dsup	Security applied to upgraded servers
dwup	Security applied to upgraded workstations
hisecdc	Includes SecureDC settings with Windows 2000-only enhancements. Empties Power Users group.
hisecweb	Configures Highly Secure IIS Server. Restricts Administrative Logon and Disables Unnecessary Services.
hisecws	Increases SecureWS Settings. Restricts Power User and Terminal Server ACLs.
notssid	Removes the Terminal Server User SID from Windows 2000 Server
ocfiless	Optional Component File Security. Many of the files may not be installed. (Windows 2000 Server)
ocfilesw	Optional Component File Security. Many of the files may not be installed. (Windows 2000 Workstation)
securedc	Assumes clean-install NTFS file\reg ACLs. Secures remaining areas.
securews	Assumes clean-install NTFS file\reg ACLs. Secures remaining areas. Empties Power Users group.
setup security	Out of box default security settings
syscomp	Security for option component files installed during GUI-mode setup

Figure 6 – Pre-Configured Security Policy Templates Provided for Windows 2000

Prefix	**Meaning**
basic	Basic (lowest) level of security
secure	Secure (mid) level of security
hisec	Highly Secure (highest) level of security
compat	Compatible (to Windows NT)
ocfiles	Optional Component Files
syscomp	System Component
deflt	Default
dw	Default workstation
ds	Default server
Suffixes	
wk or w	Workstation
sv or s	Server
dc	Domain controller
web	Web Server
up	Upgrade
first	First domain controller in a tree

Table 2 – Naming Convention for Pre-Configured Security Templates

Note that in the Security Templates snap-in, filename suffixes are not shown. All of the pre-configured security policy template filenames end in **.inf*. Whereas this is not absolutely necessary for security policy templates (the command line tool will work with any filename), the snap-in will not show templates with any other filename suffix. Throughout this section, the templates will be referred to only using their base names. Also, some of the templates shown in Figure 6 (namely *dcfirst*, *dcup*, *dcup5*, and *notssid*) are only present on a Windows 2000 Server or Advanced Server, and one is only present after the system has been promoted to a domain controller (*DC security*).

Before proceeding with a discussion of each of the security policy templates, a very important concept must be mentioned. Many of the security policy templates are incremental[5]. An incremental template does not contain the complete set of policies needed to obtain a desired security stance, but instead, assumes some pre-existing security posture as a starting point. A good example of this is the *basicdc* template that is intended to configure a domain controller with a basic level of security. It does not define any password policies whatsoever, and if applied by itself, would result in a security configuration with no requirements for password history, aging, length or complexity. The assumption is that the system must have been a server prior to becoming a domain controller, and therefore, password policies must already exist. Therefore, by design, the template does not overwrite them. **In the following discussion, all templates are incremental unless otherwise noted.**

Templates Relating to the Standard Windows 2000 Security Levels
basicdc, *basicsv*, *basicdc*, *securews*, *securedc*, *hisecws*, and *hisecdc*

This group of these templates is intended to enforce three levels security policy (basic, secure and highly secure) on workstations, servers or domain controllers. The basic templates are discussed first. As the descriptions of these templates indicate, none of the policy items in the User Rights Assignment sub-category or Restricted Groups category is defined by any of these templates. Basic level of security enforces the Windows 2000 default security settings. The policy settings in the categories of Account Policies, Local Policies and Event Log, are covered in great detail in a SANS reading room document[6], so only a brief summary is presented here.

- Account Policies
    - No password history, minimum age or length, or complexity is required
    - Maximum password age set to 45 days
    - Account lockout policies are not set
- Local Policies
    - No auditing of any kind is configured
    - No additional restrictions are placed on anonymous connections
    - System can be shut down without logging on (workstation only)
    - Only Administrators can eject removable NTFS media
    - Local users will be automatically logged off after 15 minutes idle time.
    - The virtual memory page is not cleared on system shutdown
    - Network communications are digitally signed when possible
    - Secure channel communications are digitally signed or encrypted when possible
    - Ctrl-Alt-Del requirement is disabled for logon (server only)
    - LM and NTLM responses are sent for LAN Manager Authentication
    - Users are prevented from installing printer drivers (server only)
    - Users are prompted to change their password 14 days before expiration.
    - Automatic administrative logon and floppy copy access is disabled for Recovery Console
    - Unencrypted passwords are not sent to third-party SMB servers
    - System does not shut down if unable to log security audits
    - No action is taken if the smart card used to logon is removed
    - Default permissions on global system objects are strengthened (*e.g.* symbolic links)
- Event Log (application, security and system logs)
    - All log sizes are limited to 512 kB.
    - Guests may access all logs.
    - All logs are overwritten every 7 days, full or not.
    - System will not automatically shut down when security log is full.

In the other categories of the security template, basic security means enforcement of the default access permissions for four groups: Administrators, Power Users, Terminal Server Users, and Users (see Table 3)[7].

Group	Brief Description	Capabilities
Administrators	All-powerful	Unrestricted access to any registry or system object. Any right they do not have by default, they can grant to themselves.
Power Users*	Some Privileges	Install and remove local applications that do not install System Services Customize system resources (e.g. Time, Shares, Printers, etc.) Create local users and groups and modify local groups they have created.
Terminal Server Users* (Server only)	Some Privileges	Granted access to certain files, directories, and registry keys that normal users may not access
Users	Very limited privileges	Man run any application installed by Administrator, Power User or themselves. May not access registry settings, operating system files, program files, or other users' data.

*Provided for backward compatibility to Windows NT

Table 3 – Windows 2000 basic security configuration for native operation

This basic security stance for native Windows 2000 operation is an example of the well-known principle of least privilege. It is enforced by configuring the System Services, Registry, and File System categories, and is designed to prevent Users on a workstation from compromising the integrity of the operating system while allowing them to perform the actions necessary to complete their tasks. Note that this change in the default security stance for the Users group means that software applications must be written that are compatible with this policy[8]. Many legacy applications will fail to run in the normal Users context (as they may try to write to the now protected system area). The Power Users group is provided as one option to achieve backwards compatibility. Users who need to run legacy applications can be placed in this group, and they will have appropriate permissions to run these applications. However, they will also be granted other privileges beyond what is necessary and sufficient (*e.g.* the ability to create local user accounts), violating the least privilege principle. This is an unacceptable security risk and a safer option is available using the *capatws* security template (discussed later). A similar problem with server based legacy applications arises on a Windows 2000 server system. This is automatically addressed in the default basic level security stance for a server with the Terminal Server Users group. This group is granted additional access to certain files, directories and registry keys that normal users are not granted access to. Again, this may be an unacceptable security risk, and a pre-configured security template (*notssid*) is provided to remove these privileges (discussed later).

The templates designed to increase the security of the system configuration to the secure level provide increased security settings to those areas of the operating system not covered by permissions. They do not make any modifications to the System Services, File System or Registry categories. A complete table listing the options applied by the secure template is given in [6]. The following is a summary of the items that are modified:

- All members of the Power Users Group are removed (by *securews* only, not by *securedc*)
- Stronger password and account lockout policies are implemented
  - Password history, minimum age and length, and complexity are enforced
  - Account lockout is enabled for 30 minutes after 5 invalid logons and lockout counter resets after 30 minutes.
- Some auditing is configured
  - Success and failure of 1) account logon events and management and 2) policy changes (Failure only for domain controllers)
  - Failure of privilege use.
  - Failure of directory services access for domain controllers
- Event Log settings are modified
  - Allow 10 times the space for the security log and prevent automatic overwriting after 7 days
  - Restrict guest access to system, security and application logs
- Anonymous users are restricted from enumerating SAM accounts and shares
- Digitally signed server communications are used when possible
- Requirement for Ctrl+Alt+Del for console logon is disabled
- LM authentication requests are refused
- Users are prevented from installing printer drivers,
- The workstation is locked when the smart card used for logon is removed
- Installation policy for unsigned drivers and non-drivers is defined
  - Server and Workstation – Warn but allow installation of unsigned drivers, silently succeed to install unsigned non-drivers
  - Domain Controllers – Do not allow unsigned drivers; warn but allow installation of unsigned non-drivers.

The highly secure security stance primarily enforces requirements on network traffic and protocols. It is designed for Windows 2000 computers operating in a native Windows 2000 environment. All network communications are digitally signed or encrypted and these systems will not be able to communicate with Windows NT, 98 or 95 hosts. Note all of the changes listed above for the secure template are also configured in the highly secure template, **with the exception that the Power Users group is not emptied**. Again, a complete table listing the options applied by the highly secure template is given in [6]. The following is a summary of the items that are modified:

- Locked out accounts must be manually unlocked by an Administrator
- Additional auditing is configured
  - Success and failure of logon events, privilege use, object access, and system events.
  - Success and failure of directory service access for domain controllers
- Space allotted for the security log is 20 times basic level (twice the secure level)
- Anonymous connections are not granted access without explicit permissions.
- The virtual memory page is cleared on shutdown
- Only NTLMv2 requests are accepted. LM & NTLM are refused.
- Installation of unsigned drivers is prevented.

- Installation of unsigned non-drivers succeeds silently
- Client and Server communications must be digitally signed.
- Secure channel communications must be signed or encrypted.
- Privileges granted to the Terminal Server Users group by the basic template are revoked

Care should be taken when using these templates to increase the security posture of a system, as it can be path dependent. As an example, consider the path that results in a secure domain controller beginning with a server at a basic level of security (meaning there may be some members in the Power Users group). If the server is promoted to a domain controller before the *securedc* template is applied, the resulting security stance does not empty the Power Users group. However, if the *securews* template is applied before promotion to a domain controller, the Power Users group will be emptied. Both paths should result in a domain controller with a secure stance, but clearly the latter is more secure.

The author in [6] presents a long table of recommended settings for each item in the Account Policies, Local Policies, Event Log, and User Rights. To briefly summarize the recommendations are as follows:

- Password and account lockout policies of the highly secure stance are recommended (with minor modifications)
    - Maximum password age 45 days instead of 42
    - Minimum password age 1 day instead of 2
    - Account lockout threshold 3 invalid logins instead of 5
    - Reset lockout counter after 15 minutes instead of 30
- Success and failure auditing of everything but process tracking.
- Limit all User Rights to Administrators, Backup Operators, and Authenticated Users
- Security Options from highly secure stance plus
    - Restrict floppy drive and CD-ROM access to locally logged-on users
    - Force Logoff on smart card removal

As discussed earlier, the highly secure stance requires a level of network communication only supported by Windows 2000; therefore, the recommendations presented assume a native Windows 2000 environment. If that is not the case, a security stance close to the recommended one that allows non-Windows 2000 clients to communicate can be maintained by changing the following items in the Security Options sub-category:

- Disable digitally sign server communication (always)
- Disable digitally sign client communication (always)
- Set LAN Manager Authentication level to Send NTLM response only.

A final word on the shortcomings on this group of templates is that secure and highly secure templates do not disable any of the system services. Surely there is no good reason why a highly secure domain controller should be running a Fax Service or DHCP client. Later in this document, the security template designed for a highly secure web server reviews some of the other possibly unnecessary services that system administrators should consider disabling.

Templates Relating to Backwards Compatibility
compatws and *notssid*

As mentioned in the previous section, the *compatws* template is designed to relax the ACLs for the Users group so that software applications not conforming to [8] will run correctly. The template grants members of the Users group permission to modify and write to certain areas of the file system (*%Program Files%, %SystemRoot%\Downloaded Program Files,* and *%System Root%\temp*) and certain registry keys (*MACHINE\Software*) commonly accessed by legacy applications. It only has settings for items in the Registry and File System categories with one exception. It is assumed that anyone applying this template does not want users to be Power Users, so it includes a setting in the Restricted Groups category that removes all the members of the Power Users group.

Application of the *notssid* template will remove the additional privileges granted to the Terminal Server User group by the default, basic security level template on a Windows 2000 server. The *notssid* template only has settings in two categories: Registry (MACHINE\Software and sub keys) and File System (*%Program Files%* and subdirectories). The settings simply remove the Terminal Server User group from the ACL for these areas. Note that the description of the template ("Removes the Terminal Server User SID from Windows 2000 Server") is a bit misleading. After application of the template, the Terminal Server User group still exists on the server and can be used for administration[9]. Also, application of this template does not result in the removal of members from the Power Users group.

Templates Relating to Optional Component Files
syscomp, *ocfilesw*, and *ocfiless*

This group is intended to properly configure ACLs on files and directories that are not automatically included in a default installation of Windows 2000, but may be selected for installation during GUI-setup as optional components (*e.g.* Windows Media Player, OLE database, Outlook Express, *etc.*). The only category in which these templates have settings is File System. All files that will be installed by these applications in the *%ProgramFiles%*, *%CommonProgramFiles%*, *%SystemRoot%* and *%SystemDirectory%* directories are configured to inherit the permissions on their respective parent folders. The first template in this category (*syscomp*) is applied during GUI-mode setup and can be found in the *%SystemRoot%\inf* directory. The file found there differs depending on whether the system is a Professional or Server installation because some optional components are only available for server installations. The last two templates are located in the *%SystemRoot%\Security\Templates* directory. They are intended for workstation and server installations, respectively, and they include the files covered by *syscomp* as well as files in the *%SystemRoot%*\spool\drivers directory (which are omitted from *syscomp*).

Installation or Upgrade Templates

Clean-Installation: *defltwk*, *defltsv*, and *defltdc*
Upgrade: *dwup*, *dsup*, *dcup*, and *dcup5*

These templates are located in the *%SystemRoot%\inf* directory and are applied during the Windows 2000 installation procedure to configure all the default security settings. The templates applied for a clean installation of a workstation or server (*defltwk* and *defltsv*) are **not incremental**. The general philosophy evident in the upgrade group of templates is the assumption that the system being upgraded already has adequate security policies in place that should not be overwritten. Therefore, the main difference between the clean-installation templates and upgrade templates is in the categories of Account Policies, Local Policies, and Event Log. The settings in the clean installation templates match those in the templates intended to provide a basic level of security (discussed in an earlier section). However, all settings in these categories of the upgrade templates are left undefined so previously configured values on the Windows NT system being upgraded will not be overwritten. The exception is that any Windows 2000 specific item in these categories that did not exist for NT (*e.g.* “Smart card removal behavior”) is assigned the setting of the corresponding item in the basic level of security template (e.g. “No Action”).

The security configuration defined in the categories of System Services, Registry and File System are the almost identical between the clean installation and upgrade templates for a given system type (*e.g.* workstation). The result is that the default Windows 2000 permissions for file system, registry and service objects are configured consistently regardless of whether the system was upgraded or cleanly installed[7]. The exception is that in the upgrade templates, the registry key that stores configuration data for classes of hardware devices (HKLM\SYSTEM\CurrentControlSet\Control\Class) is configured to ignore any changes to sub-keys in this area that might be attempted during the installation. This is intended to insure that hardware devices connected to the system continue to be recognized. In the clean installation templates, this registry key is left undefined.

The settings in the remaining category of Restricted Groups differ in that the clean-installation templates restrict the members of the Users group to interactive users (*i.e.* those logged in at the console) and authenticated users. The upgrade templates do not configure this restriction.

One of the two domain controller upgrade templates is applied when upgrading Windows NT servers. The difference between the two is that one is intended for systems that were already domain controllers prior to upgrading (*dcup5*) and the other is intended for systems that were not (*dcup*). Both templates configure Registry and File System categories, however, *dcup* also includes configuration for the User Rights Assignment sub-category that match the *defltdc* template.

An important thing to realize is that, if Windows 2000 is to be installed from a distribution share, modifications to the “out-of-the-box” security configuration are possible simply by editing the templates discussed in this section, and replacing the original ones in the share with the edited ones. A Microsoft Knowledge Base article presents a step-by-step discussion of using this technique to prevent a Windows 2000 upgrade from modifying custom security for categories not already left undefined[10]. However, the same technique could be employed so that all cleanly installed workstations are configured with an appropriately customized template instead of *defltwk*.

Templates Documenting Installation Setup
setup security and *DC security*

These two templates are essentially logs of the security configuration that was applied during installation. The *setup security* template is a copy of the security settings at the time of installation of the workstation or server. For clean installations, it is essentially identical to the *defltwk or defltsv*, template, respectively, with the exceptions that it is stored in UNICODE format (which approximately doubles the file size) and it has the actual values for variables that appear in the template instead of the variables. For example, the path *c:\winnt* appears instead of the variable *%SystemRoot%* or the security identifier *S-1-5-32-544* appears instead of the variable *%SceInfAdmins%*. The DC security template is a copy of the security settings applied by *dcpromo* during promotion of a server to a domain controller. For clean installations, it is essentially the same as *defltdc*. It is also stored in UNICODE format and actual values of variables for the current system appear in the template, instead of the variables.

Highly Secure Web Server Template
hisecweb

The *hisecweb* security policy template does not come with the Windows installation, but can be obtained from Microsoft. There is script-driven Internet Server Security Configuration Tool, IIS Lock, that can be obtained by visiting http://www.microsoft.com/security/ and searching for IISLock.exe. The tool comes with a very restrictive version of *hisecweb* that is based on the following assumptions

- The machine is a not a Domain Controller
- The machine is a standalone server (*i.e.* not joined to a domain)
- The machine is a dedicated web-server and physically protected
- The machine has the Windows 2000 clean-install defaults (*i.e.* no modifications have been made to ACLs, User Rights etc.)
- No one is allowed to log on locally to the machine accept an administrator
- Administrators are not allowed to log on over network (they have to go to the Web server to administer it)
- Admin\Guest accounts are not renamed via this template

The template contains no settings in the categories of Registry and File System. The settings for items in the other categories match those of a highly secure server (*hisecws*) with the following exceptions:

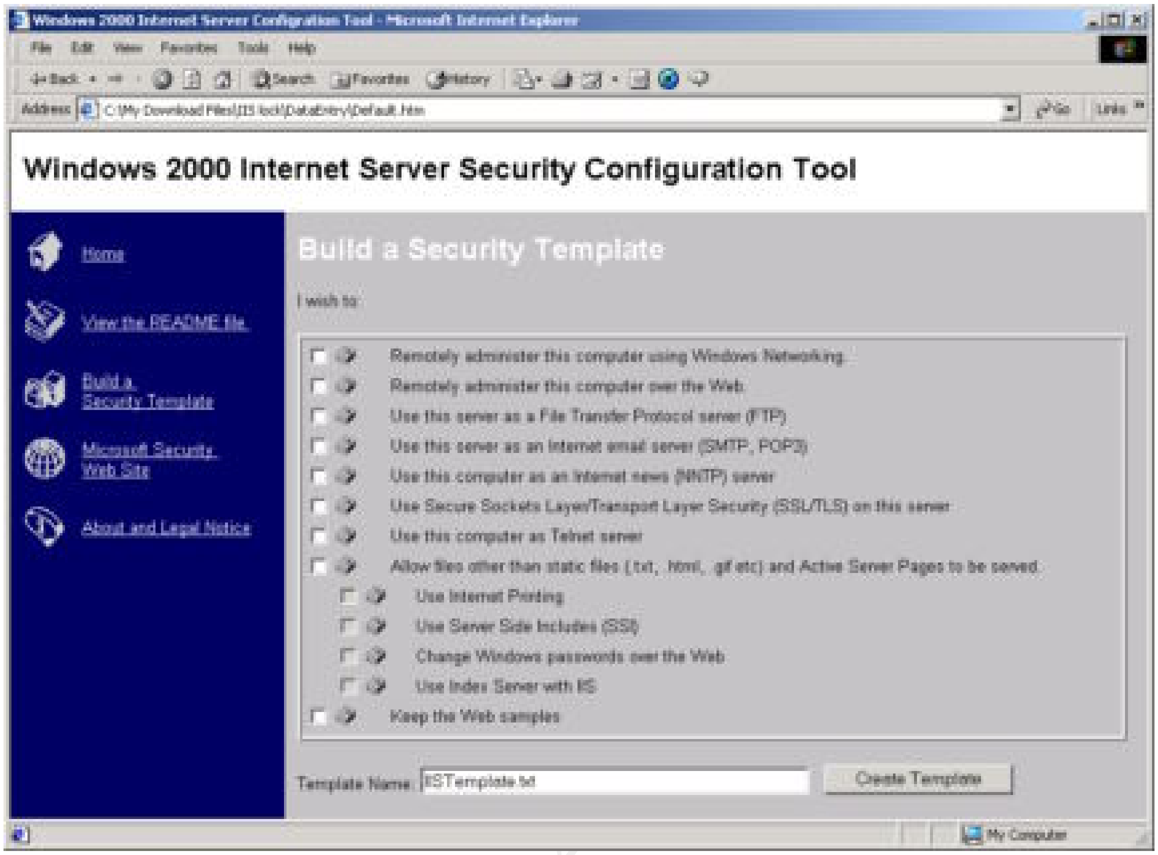
- Audit Policy - Only Failures of object access are audited.
- Security Options
  - System can not be shut down without logging on
  - Message text for users attempting to logon is defined
  - Floppy disk and CD-ROM access is restricted to logged-on users only

In addition, the template adds restrictions in the following categories that are left undefined by the highly secure template:

- Restricted Groups – All members of the Power Users Group are removed.
- User Rights Assignment – Only authenticated users can access the computer from the network
- System Services – The following unnecessary system services are disabled
  - Alerter
  - ClipBook
  - Computer Browser
  - DHCP client
  - Fax Service
  - Internet Connection Sharing
  - Irmon
  - Messenger
  - NetMeeting Remote Desktop
  - Print Spooler
  - Remote Access Auto Connection Manager
  - Remote Access Connection Manager
  - Remote Registry Service
  - Task Scheduler
  - Telephony
  - TermService

The list of system services disabled by this template is perhaps it's the most useful aspect. A well know tenet in computer security is that if a service is not necessary for a system to perform its duties and is not being used, turn it off. The other pre-configured, highly secure templates should also follow this policy.

Note that if a web server does not conform to all of the assumptions, IIS Lock provides a GUI interface to modify the *hisecweb* template. Scripts provided with the distribution use answers to a set of questions on an HTML page with check boxes to create a customized version of the template (see Figure 7).


Windows 2000 Internet Server Configuration Tool - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address C:\My Download Files\IIS lock\DataEntry\Default.htm
Go Links
Home
View the README file
Build a Security Template
Microsoft Security Web Site
About and Legal Notice
Create Template
My Computer

Windows 2000 Internet Server Security Configuration Tool

Build a Security Template

I wish to:

- □ Remotely administer this computer using Windows Networking.
- □ Remotely administer this computer over the Web.
- □ Use this server as a File Transfer Protocol server (FTP)
- □ Use this server as an Internet email server (SMTP, POP3)
- □ Use this computer as an Internet news (NNTP) server
- □ Use Secure Sockets Layer/Transport Layer Security (SSL/TLS) on this server
- □ Use this computer as Telnet server
- □ Allow files other than static files (.txt, .html, .gif etc) and Active Server Pages to be served.
  - □ Use Internet Printing
  - □ Use Server Side Includes (SSI)
  - □ Change Windows passwords over the Web
  - □ Use Index Server with IIS
- □ Keep the Web samples

Template Name: IISTemplate.txt

Figure 7 – IIS Lock Interface to Generate Customized Security Policy Template

Clearly some of the options presented by the HTML page cannot be configured solely through the application of a security policy template (*e.g.* setting up FTP, SMTP or NNTP servers). However, other options directly correspond to items in the System Services category (*e.g.* enabling remote administration).

Miscellaneous Templates

dcfirst and *dedica**

The *dcfirst* template is located in the *%SystemRoot%\inf* directory, and according to the comments included within the template, it is applied via Group Policy during *Winlogon* for the first domain controller in a tree. It contains a very minimal number of settings. It is the only template that includes any settings in the Kerberos Policy sub-category, although the settings that are defined simply re-apply the default Kerberos settings[11]. Aside from these, the following settings are explicitly defined:

- Basic security level Password Policy (except that password history set to 1 not 0)
- Account lockout is disabled
- Automatic logging off of users when logon time expires is disabled

The usefulness of this template is questionable. A search of the Microsoft support web site, Windows 2000 related news groups, and the Internet as a whole did not reveal any additional documentation on exactly what the purpose is intended to be. Given the absence of additional information, the minimal increase in security posture provided by this template (enabling minimal password history) is not worth the decrease in security posture resulting from disabling account lockout (which enables the possibility for brute force password attacks) and automatic log off.

The second template in this section (*dedica**) is mentioned in the Windows 2000 Server Documentation. According to the documentation application of the template has the following effect[12]:

> Local user security on domain controllers running Windows 2000 is not ideally secure by default. This enables an administrator to run existing server-based applications on domain controllers (not recommended) in a backwards-compatible fashion. If you do not run server based-applications on domain controllers (recommended), the default file system and registry permissions for the local users group can be defined in the same ideal fashion as that defined by default for Windows 2000 workstations and stand-alone servers. By implementing a dedicated security template these ideal security settings for local users on Windows 2000 domain controllers are applied.

While this sounds quite useful, the claims could not be confirmed, because a search of Windows 2000 Professional and Server installations, the Microsoft support web site, Windows 2000 related news groups, and the Internet as a whole did successfully not locate the template.

Working with Security Templates from the Command Line

As with most of the Windows 2000 GUI tools, the functionality provided by the Security Configuration and Analysis MMC snap-in is also available with a command line tool. Online documentation for secedit is available by entering the command with no arguments at the command prompt. Also, [4] contains a very good summary of how to use secedit, so only an overview will be presented here.

Purpose	**Valid Options**
Analyze security settings	**secedit /analyze** [**/DB** *filename*] [**/CFG** *filename*] [**/log** *logpath*] [**/verbose**] [**/quiet**]
Configure security settings by applying a stored database.	**secedit /configure** [**/DB** *filename*] [**/CFG** *filename*] [**/overwrite**][**/areas** *area1 area2...*] [**/log** *logpath*] [**/verbose**] [**/quiet**]
Re-apply a Group Policy object to refresh security settings	**secedit /refreshpolicy** {**machine_policy** | **user_policy**}[**/enforce**]
Export database security settings to a template	**secedit /export** [**/mergedPolicy**] [**/DB** *filename*] [**/CFG** *filename*] [**/areas** *area1 area 2...*] [**/log** *logPath*] [**/verbose**] [**/quiet**]
Validate a security template	**secedit /validate** *filename*

Table 4 – Valid Options for the Command Line Tool secedit

Table 4 shows a brief summary of the valid options. The security state of the system can be analyzed or configured using a database constructed by overlaying one or more security templates and the settings in the database can be exported to a template, just as with the MMC snap-in. However, additional functionality is provided by *secedit* that is not available using the GUI tool. First, the syntax of a security policy template can be validated before it is imported into a database. Second, the verbosity of the log file that is generated can be controlled; either increased to give detailed progress (**/verbose**) or suppressed altogether to give no output (**/quiet**). Third, a group policy propagation event, that occurs by default when a machine boots and every 60-90 minutes thereafter, can be forced. Finally, and probably most useful, *secedit* includes an option (**/areas**) that can be used with the **/configure** and **/export** flags to apply or export only certain categories in the database instead of the whole template. A summary of valid areas is given in Table 5.

Area Name	Description
SECURITYPOLICY	Local policy and domain policy for the system, including account policies, audit policies, and so on.
GROUP_MGMT	Restricted group settings for any groups specified in the security template
USER_RIGHTS	User logon rights and granting of privileges
REGKEYS	Security on local registry keys
FILESTORE	Security on local file storage
SERVICES	Security for all defined system services

Table 5 – Summary of Valid Security Template Areas Allowed with secedit

The following examples provide a better understanding of how to use the command line tool. First, the following command will validate the syntax of the example minimal security policy template discussed earlier that configures only the password length:

```
C:\>secedit /validate passlength.inf
Template C:\passlength.inf is validated
```

As discussed earlier, the *hisecweb* security policy template includes settings to disable unnecessary services on a web server. It is likely that the same services should be disabled on a secure mail server, which must be connected to the Internet to perform its function, and, therefore, needs to be protected. To configure the System Services settings of the *hisecweb* template on a mail server system, first copy the template to the system. Then issue the following at the command prompt:

```
C:\>secedit /configure /db”c:\temp\hisecweb.sdb”
/cfg”c:\temp\hisecweb.inf” /log”c:\temp\confserv.log” /verbose /areas
services
```

This command assumes the *hisecweb* security template was copied to *c:\temp*. It will import the template into a database and configure the system security settings in the System Services category only. Verbose output will be sent to a log file named *confserv.log*.

Note that care should be taken to insure that application of settings in the database does not result in unintended consequences on a system when using either the secedit or the Security Configuration and Analysis Tool. Strictly speaking, there is no “Undo” functionality to return the system to the state it was in prior to configuring it. It would be necessary to construct a template that undoes each of the changes, one-by-one, setting-by-setting. Therefore, testing security policy templates on a test system prior to using them on a production system is highly recommended.

Concluding Remarks

The first, and probably most essential, step in securing any computer network is a thorough, well-written security policy. With a Windows 2000 system or network, the next step is translating that policy into an appropriate security policy template. In order to facilitate this translation, this document has presented an in-depth review of what a security policy template is, how to work with one in the context of a single system, what pre-configured templates are actually used during installation of the operating system, and what tools are available to review and customize them for specific requirements. Armed with the comprehensive knowledge provided here, a system administrator has taken a big step towards achieving the goal of a secure enterprise network.

List of References

1. Microsoft Windows 2000 Technical Library, "Windows 2000 Security Technical Overview", (http://www.microsoft.com/WINDOWS2000/library/howitworks/security/sectech.asp)
2. Microsoft TechNet Article, "Step-by-Step Guide to Configuring Enterprise Security Policies", (http://www.microsoft.com/TechNet/win2000/entsec.asp)
3. Microsoft Windows 2000 Server Documentation – "Security Templates Overview", (http://www.microsoft.com/WINDOWS2000/en/advanced/help/sag_SCEwhatis.htm)
4. Microsoft Windows 2000 Technical Library, "Step-by-Step Guide to Using the Security Configuration Toolset", (http://www.microsoft.com/windows2000/library/planning/security/secdefs.asp)
5. Microsoft Knowledge Base Article Q234926 – "Windows Security Templates Are Incremental"
6. Security Configuration Tool and Template Settings, Usefulness and Shortcomings of the Pre-Configured Security Policy Templates that are included with Windows 2000, Robert Huie, December 2000. (http://www.sans.org/infosecFAQ/win/settings.htm)
7. Microsoft Windows 2000 Server Documentation – "Default Access Control Settings in Windows 2000"
8. Windows 2000 Application Specification, (http://msdn.microsoft.com/certification/default.asp)
9. Microsoft Knowledge Base Article Q238965 – "Removing Additional Permissions Granted to Terminal Services Users"
10. Microsoft Knowledge Base Article Q260242 – "How to Prevent Windows 2000 Upgrade from Modifying Custom Security"
11. Microsoft Knowledge Base Article Q231849 – "Description of Kerberos Policy Settings in Windows 2000"
12. Microsoft Windows 2000 Server Documentation – "Predefined Security Policy Templates" (http://www.microsoft.com/WINDOWS2000/en/server/help/sag_SCEdefaultpols.htm)