Global Information Assurance Certification Paper

GIAC Certified Windows Security Administrator (GCWN)
Practical Assignment
V3.1 (revised April 8, 2002)

Securing Windows 2000 with Security Templates

Prepared by:
Todd Ladner

Table of Contents

1.0 Introduction

The whole idea behind the Windows operating systems is to make the act of using a computer simple. After all, when trying to market a relatively new product to the general population, you want the learning curve to be as short as possible. A steep learning curve equals non-acceptance unless this new product is absolutely necessary. When personal computers where first introduced, most people were of the belief that owning a personal computer was not that critical. Based on the popularity of the Windows operating systems and the size of Bill Gates' fortune, it is obvious that the Microsoft corporation has convinced the general population that personal computers are necessary, and that their software is the easiest to use.

Once computers were more widely accepted, Microsoft's challenge was to incorporate new features and functionality into their operating systems, while continuing to make them as simple as possible. Microsoft has not been successful at this simplification in all areas, but in the area of security, and in particular the deployment of security settings, Microsoft has made great strides with the introduction of their security templates.

This document will give you an explanation of what a security template is, including a discussion of how to view and modify a security template within the Microsoft Management Console (MMC) Security Template snap-in. A brief discussion of the different sections in the template will be included. The document will then walk through the process of setting up a Microsoft Windows 2000 server running Internet Information Server (IIS) 5.0. The focus of the server setup will be on the use of a security template to establish the initial security settings, with each relevant security setting explained. Following the explanation of the applied security settings, there will be an explanation of how tests can be performed on the server to insure that the security is performing as expected. The functionality of the server will also be addressed, as it is important to make sure that your security settings have not affected the server in any adverse way. Finally, an opinion of the effectiveness of the security template will be given with suggestions on how the template could be improved.

1.1 What is a Security Template?

The security templates provided by Microsoft serve as starting points for system security and can be customized to fit individual needs. The default installation of Windows 2000 includes several security templates that provide basic, secure, and highly secure settings for workstations, servers, and domain controllers. These templates are located in the WINNT\Security\Templates folder. There are several other templates installed, but they are not discussed in this document. Other organizations such as the National Security Agency and the Defense Information Systems Agency have provided additional security templates. With

all of the canned templates available, a system administrator should be able to find a good starting point for whatever type system being created. It is even possible to build custom security templates from scratch, but this is not the preferred method.

The security templates themselves are nothing more that text files. An example of one can be seen in Figure 1.

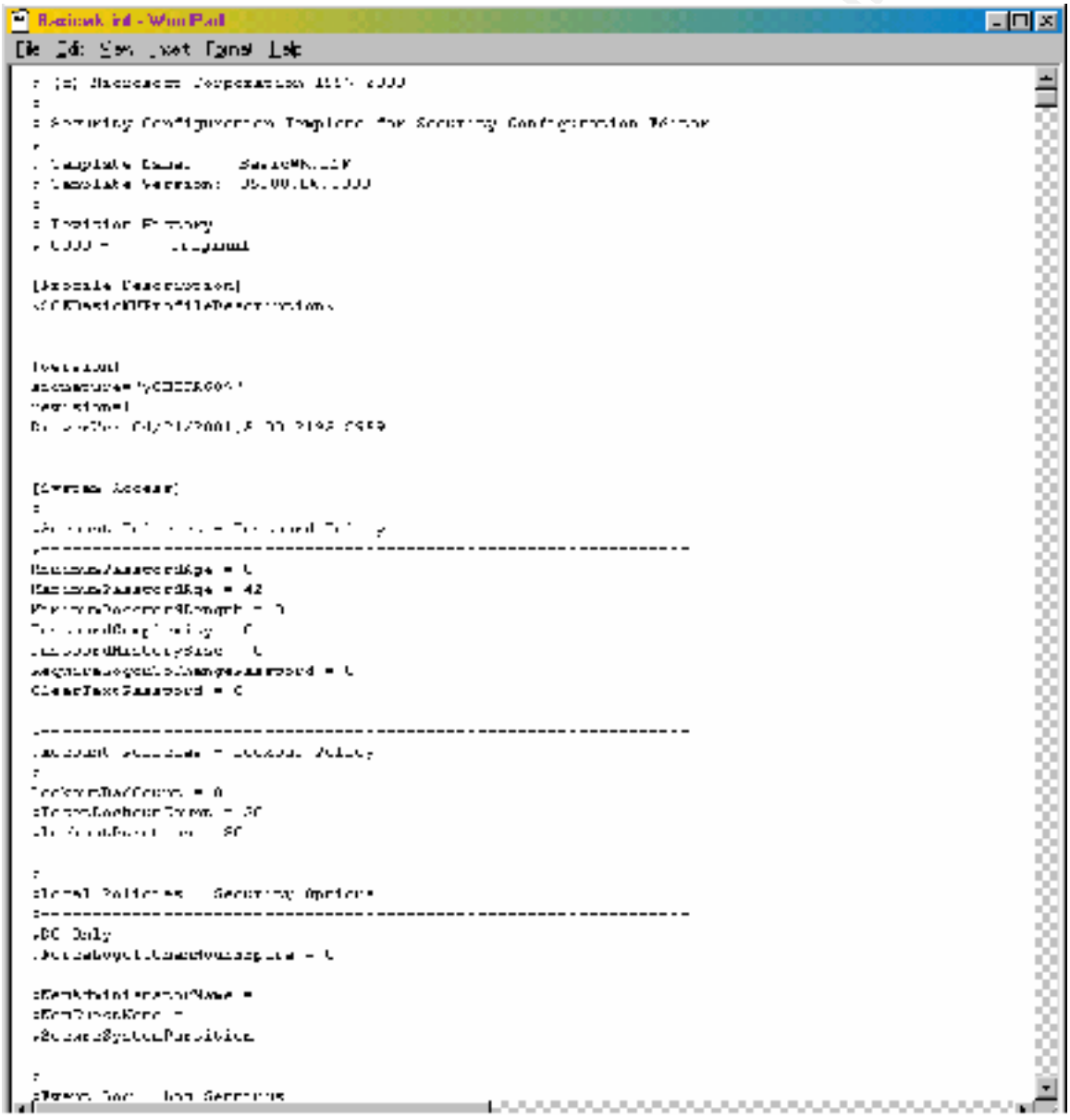

Figure 1- Security Template Example

Fortunately, the creation or modification of a security template does not have to be performed with a text editor. Microsoft has provided a Security Template snap-in that can be utilized by the MMC. The snap-in provides the user with a graphical interface that can be used to view the individual policy settings and allows the user to modify the individual policy settings. The MMC with the Security Templates snap-in can be seen in Figure 2. In order to load the snap-in into the MMC, start the MMC by clicking "Start" on the taskbar and then click "Run" from the menu. Type "mmc" in the "Run" window to open a blank MMC window. Now click on "Console" in the top menu and then click on the "Add/Remove Snap-in…" option. In the "Add/Remove Snap-in" window click on the "Add…" button. In the resulting "Add Standalone Snap-in" window, scroll down until the "Security Templates" snap-in is found, highlight it, and click on the "Add" button. The "Security Templates" snap-in should appear in the "Add/Remove Snap-in" window. Now click on the "Close" button in the "Add Standalone Snap-in" window, and then click on the "OK" button in the "Add/Remove Snap-in" window. The MMC should now contain the "Security Templates" snap-in. Drilling down into the Security Templates folder will allow all of the templates located in the WINNT\Security\Templates folder to be seen. Any new templates not part of the standard load or obtained from sources other than Microsoft should be placed in this folder. It is important to keep all other files with a ".ini" extension out of this folder because they too will show up as a security template.

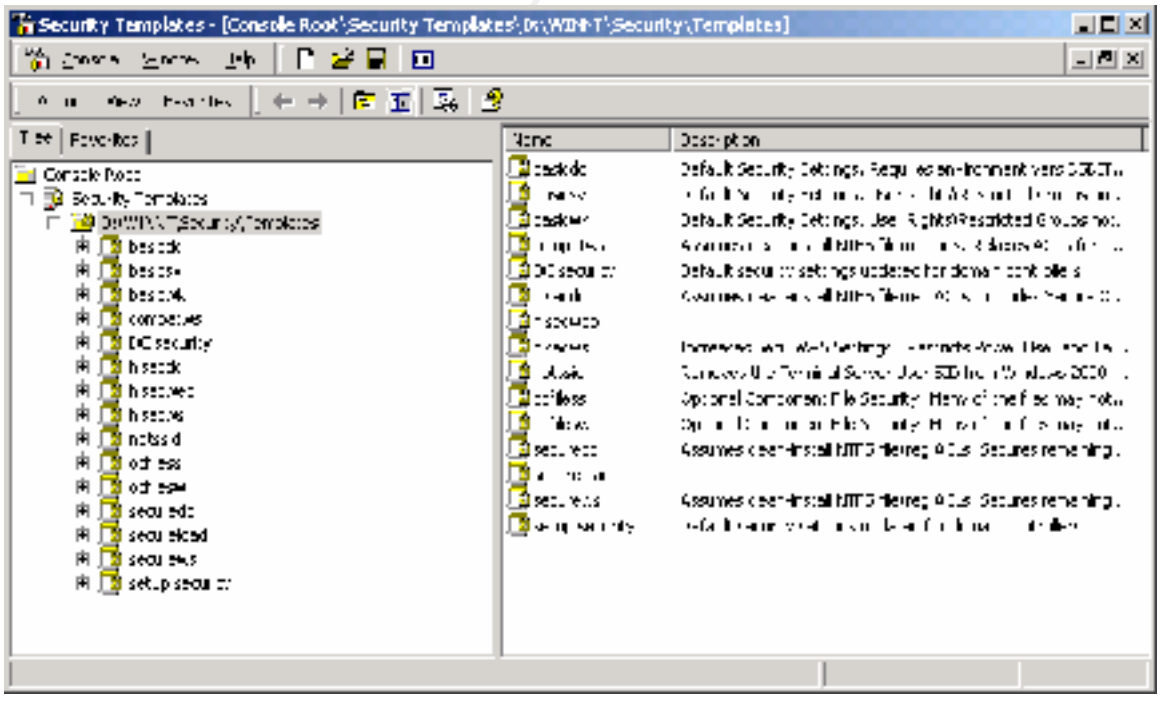

Figure 2 – MMC with Security Templates Snap-in

A security template consists of a default set of policies and categories. These are Account Policies, Local Policies, Event Log, Restricted Groups, System Services, Registry, and File System. Account Policies is broken down further into Password Policy, Account Lockout Policy, and Kerberos Policy. Local Policies is broken down into Audit Policy, User Rights Assignment, and Security Options. Event Log contains setting for the three event logs in Windows 2000, Application, Security, and System. Restricted Groups administers local group membership. System Services contains security and start up modes for local services. Registry has security settings for local registry keys. File System can set permissions on local files and directories.

As mentioned above, the Security Template snap-in allows the user to select an existing security template, modify the chosen template, and save this customized template. This is accomplished by simply opening the security template that has been chosen as a starting point, modifying the settings to the desired values and then using the "Save As" menu item to create a new customized security template. There is another method to create customized templates using the Security Configuration and Analysis snap-in, but that topic is beyond the scope of this paper. The Security Configuration and Analysis snap-in is also used to apply security templates, and will be discussed later. Detailed instructions on working with security templates can be found in Microsoft Knowledge Base Article Q313434. [1]

2.0 Computer System

When a security configuration is being developed for a new system, there are many factors to consider. The primary factor is the role that the system will play. Is the system going to be a domain controller? Is the system going to serve web pages on the Internet or just on an organizational intranet? Will the system contain sensitive data that could be damaging to the organization? These are the types of questions that must be answered. Unless it is known how the system will be used, the security options cannot be tailored to fit specific needs. Another consideration is the software that is, or will be, installed on the system. The operating system is the first item to look at. Even within the bounds of a Microsoft operating system, it is imperative to know if the machine is running Windows 2000, Windows NT, or any other flavor of the windows operating system. The role of the system will, in most cases, determine what application software will be loaded, and the security settings that are chosen must allow the software to function correctly. A final consideration is the system hardware. While this is not as important as system role or system software, items such as smart card readers, removable hard drives, and wireless network devices can have a huge impact on your security configuration.

2.1 System Hardware

The hardware used in the creation of this paper is not what you would expect. My job evolution has taken me away from many system administration duties and my access to hardware resources is not what it used to be. The purpose of the system does reflect an actual web server under my supervision that is currently running Windows NT 4.0. This exercise will serve me well when it is decided that the current system will be upgraded.

The actual hardware used is a Compaq Armada1750 laptop with a 10GB Toshiba hard drive partitioned into two 5GB drives. The system is loaded with 192MB RAM. The system is equipped with a CD-ROM drive and a 3.5 inch floppy drive. A LinkSys EtherFast 10/100 PCMCIA card is installed for network access. A Compaq 56VL Global Internal Modem is also installed for dial-up access.

2.2 System Software

The Compaq laptop is loaded with Microsoft Windows 2000 Server with Service Pack 2 and is configured as a stand-alone server. Microsoft IIS 5.0 is also loaded. Norton AntiVirus Corporate Edition version 7.6 has been installed and File System Realtime Protection is enabled. Microsoft Office 2000 Professional Edition is installed as well.

2.3 System Role

The system will perform as an IIS/Win2K web server and will serve pages only to users located on the organization's intranet. No access from the Internet will be allowed, but everyone located within the boundaries of the firewall will have access. The hosted web sites will contain information relevant to various departments that are located on site. The server will not host any mission critical or sensitive information. No one other than an administrator will be allowed to log on to this system, and all system administration functions will be performed from the console. There will be no remote administration allowed. The server, once put into production, will be located in an access controlled environment. Some of these restrictions may seem a little too vigorous for a web server that is restricted to serving pages on the intranet, but it is a well known fact that threats from within are often times more destructive than threats from the outside.

3.0 Checklist/Security Template Selection

Both the IIS 5.0 Baseline Security Checklist[2] provided by Microsoft and the *hisecweb* security template will be used to secure the system. The checklist has

some recommendations that cannot be accomplished by using the template alone, and these recommendations are vital to a complete security posture. Had the web server been accessible from the Internet, the Secure Internet Information Services 5 Checklist[3] also provided by Microsoft would have been used. This checklist covers all of the features in the Baseline Security Checklist, and includes other steps to further tighten the security of a web server. The *hisecweb* security template can be downloaded from a link on the Secure Internet Information Services 5 Checklist web page. The template chosen may seem to be too restrictive for an internal web server, but it was decided to err on the side of caution. It would be preferable to have to go back and loosen some of the restrictions than to have a security related event occur that alerts the administrator to the fact that the machine was not locked down tight enough.

4.0 Checklist/Security Template Security Settings

4.1 Checklist

The IIS 5.0 Baseline Security Checklist contains recommendations and best practices to help achieve a baseline level of security on IIS 5.0 servers. It offers 6 steps to achieve this security foundation.

The first step is to set appropriate access control lists (ACL) on the virtual directories. The checklist offers a recommendation as to how a new directory structure can be created to make the task of setting ACLs on different file types easier. It suggests creating a different directory for each file type and then setting the ACL on the directory. Files of each type should then be placed in the appropriate directory and allowed to inherit the ACL from the directory level.

The second step is to set the appropriate ACLs on the IIS log file. The permissions should be "Full Control" for "Administrators" and "System", while the "Everyone" group is assigned "Read", "Write", and "Change".

The third step is to enable logging using the W3C Extended Logging format. A procedure is given to turn on the logging feature. Enabling logging goes hand in hand with the second step. It is vitally important that logging of events occur in relation to the web activity, but it is also important that these logs be protected. What good is a log if a user with malicious intent is able to erase the evidence that they leave behind?

The fourth step is to disable or remove all sample applications. The locations of these samples are:
C:\inetpub\iissamples
C:\winnt\help\iishelp
C:\program files\common files\system\msadc.

The fifth step is to remove the IISADMPWD virtual directory, which allows you to reset Windows NT and Windows 2000 passwords. This directory should only be present if you have upgraded from IIS 4.0 to IIS 5.0.

The sixth and final step is to remove unused script mappings. The mappings tell the system which .dll file to utilize when a call is made to a particular filename extension. If the system is not using a particular filename extension, then there is no need for the mapping associated with it. A procedure to remove these mappings is given in the document.

4.2 Security Template

The *hisecweb* security template is based on the following assumptions:

- The machine is not a Domain Controller
- The machine is a standalone server
- The machine is a dedicated web-server and physically protected
- The machine has the Windows 2000 clean-install defaults and no modifications have been made to ACLs, User Rights, etc.
- No one is allowed to log on locally to the machine except an administrator
- Administrators are not allowed to log on over the network.

The following sections contain descriptions of the settings established by the *hisecweb* security template. Many of these descriptions are derived from the MSDN Windows 2000 Security Settings web pages.[4]

4.2.1 Account Policies

The Password policy settings are:

Enforce password history – 24 passwords remembered
This setting will prevent a user from reusing a password until they have used 24 unique passwords. Users, especially administrators, should not be reusing passwords anyway, so this setting is fine.

Maximum password age – 42 days
This forces the user to change their password after 42 days. This is fine for this environment.

Minimum password age – 2 days
This policy will not allow a user to change their password until 2 days have passed. This prevents the user from cycling through passwords so that they can use their old, familiar password again.

Minimum password length – 8 characters
This requires a user account to be protected by a password consisting of at least 8 characters. When setting minimum password length, the administrator is looking for a balance between a length long enough to provide adequate protection and short enough that the user will not write it down. For this particular environment, the setting of 8 is adequate.

Passwords must meet complexity requirements – enabled
This is probably the most important password policy to be enabled. It requires a password to meet the minimum requirements described below:

- Does not contain all or part of the user's account name
- Is at least 6 characters in length
- Contains characters from three of the following four categories
  - English upper case characters (A..Z)
  - English lower case characters (a..z)
  - Base 10 digits (0..9)
  - Nonalphanumeric (For example, !,$,#,%)

The complexity requirements are enforced upon password creation or change.

With the increase in computing power and the ability to crack passwords, password complexity is vitally important. All passwords can be cracked eventually, but the idea is to make it so difficult that by the time the password is cracked, the user has changed their password, or the theft of the password has been discovered.

Store password using reversible encryption for all users in the domain – disabled
Storing passwords using reversible encryption is essentially the same as storing clear-text passwords, and this is not desirable unless absolutely necessary.

The Account Lockout Policy settings are:

Account lockout duration – 0
This determines the number of minutes a locked out account will remain locked out before becoming unlocked automatically. The setting of zero will prevent the account from being unlocked automatically and force an administrator to manually unlock the account.

Account lockout threshold – 5 invalid logon attempts
This determines the number of failed logon attempts that will cause a user account to be locked out.

Reset account lockout counter after – 30 minutes
This determines the number of minutes that must elapse after a failed logon attempt before the failed logon attempt counter is reset to zero.

The Kerberos Policy settings apply to a domain and cannot be configured on individual machines. The policies are all set to "Not defined" and will have no impact.

4.2.2 Local Policies

The Audit Policy settings are:

Audit account logon events – Success, Failure
Determines whether to audit each instance of a user logging on or logging off of another computer where this computer was used to validate the account.

Audit account management – Success, Failure
Determines whether to audit account management events on a computer. This setting gives visibility into such actions as user account creation, and password changes. All activity involving user accounts is of interest, so it is important to audit both successes and failures.

Audit directory service access – Not defined
Since the system is not using Active Directory, this setting has no meaning for this environment.

Audit logon events – Success, Failure
This setting allows the logging of all instances of a user successfully, or unsuccessfully attempting to logon to this computer.

Audit object access – Failure
This setting will log all unsuccessful user attempts to access objects which have their own system access control list specified. Attempting to log successful events would cause the log files to fill up quickly.

Audit policy change – Success, Failure
This setting allows auditing of every incidence of a change to user rights assignment policies, audit policies, and trust policies.

Audit privilege use – Success, Failure
This setting will audit all instances of a user exercising a user right. Auditing both successful and unsuccessful privilege uses is going to cause the audit logs to fill up rapidly.

Audit process tracking – Not defined
If active, this would audit detailed tracking information for events such as program activation, process exit, handle duplication, and indirect object access.

Audit system events – Success, Failure

This will place an entry in the logs when a user restarts or shuts down the computer, or if an event that affects system security or log settings has occurred. Any event that affects system security is important regardless of whether it was successful or not.

The User Rights Assignments are not defined in the *hisecweb* security template with the exception of the "Access this computer from the network" policy. This policy is set to Authenticated Users.

Many of the Security Options settings are not defined for various reasons, and these options will not be discussed. The ones that are defined and their settings are:

Additional restrictions for anonymous connections – No access without explicit anonymous permissions
This option allows the administrator to place additional restrictions on anonymous connections to the computer. This setting will require that "Anonymous" users be given explicit access permissions to any required resources.

Allow system to be shut down without having to log on – Disabled
This setting will remove the Shut Down command from the Windows logon screen, thus forcing a user to log on in order to shut down the system.

Allowed to eject removable NTFS media – Administrators
Only administrators can eject removable NTFS media from the computer.

Audit use of Backup and Restore privilege – Enabled
The "Audit privilege use" option under the Audit Policy does not affect backup and restore rights. Enabling this option will allow backup and restore actions to be audited.

Automatically log off users when logon time expires (local) – Enabled
This setting will cause client sessions with the Server Message Block (SMB) server to be forcibly disconnected when the client's logon hours expire.

Clear virtual memory pagefile when system shuts down – Enabled
The system pagefile is used to swap pages of inactive memory to disk. This setting will cause the pagefile to be cleared when the system is shut down.

Digitally sign client communication (when possible) – Enabled
This setting will force SMB clients to perform SMB packet signing when possible.

Digitally sign server communication (when possible) – Enabled
This setting will force SMB servers to perform SMB packet signing when possible.

Disable CTRL+ALT+DEL requirement for logon - Disabled
When this policy is enabled, the user is not required to press CTRL+ALT+DEL in order to logon. This can open the user up to attacks that try to intercept passwords.

Do not display last user name in logon screen – Enabled
This erases the user name from the logon screen that is enabled by pressing CTRL+ALT+DEL.

LAN Manager Authentication Level – Send LM & NTLM – use NTLMv2 session security if negotiated
This option affects the level of authentication protocol used by clients and the level of session security negotiated. This setting allows clients to use LM and NTLM authentication and use NTLMv2 session security if supported by the server.

Message text for users attempting to log on – This is a private computer system…
This is where a text message displayed just before logon is specified. Many organizations, including mine, require that a warning message be displayed to users logging on to the system. This is mainly for legal purposes.

Message title for users attempting to log on – A T T E N T I O N !
This is simply the title that is displayed when the window containing the logon warning message appears.

Prevent system maintenance of computer account password – Disabled
Default Windows 2000 security causes computer account passwords to be changed automatically every seven days. Enabling this policy prevents the automatic password change.

Prevent users from installing printer drivers – Enabled
This setting prevents members of the Users group from installing printer drivers on the computer.

Restrict CD-ROM access to locally logged-on user only – Enabled
This allows only the user logged on interactively to access the CD-ROM drive. However, if no one is logged on interactively, the CD-ROM is shareable over the network.

Restrict floppy access to locally logged-on user only – Enabled
This option is exactly the same as the one above for CD-ROMs, except this one applies to the local floppy drive.

Secure channel: Digitally encrypt secure channel data (when possible) – Enabled

This option will cause all outgoing secure channel traffic to be encrypted if possible.

Secure channel: Digitally sign secure channel data (when possible) – Enabled
This option will cause all outgoing secure channel traffic to be signed if possible. If the above "Digitally encrypt secure channel data (when possible)" option is enabled, it will cause this option to automatically be enabled.

Send unencrypted password to connect to third-party SMB servers – Disabled
This option will prevent the SMB redirector from sending clear-text passwords to non-Microsoft SMB servers that do not support password encryption during authentication.

Strengthen default permissions of global system objects (e.g. Symbolic links) – Enabled
This setting strengthens the default discretionary access control list for objects on the list of shared system resources.

Unsigned driver installation behavior – Do not allow installation
This option prevents the Windows 2000 device installer from installing any device driver that has not been certified by the Windows Hardware Quality Lab.

4.2.3 Event Log

The "settings for event logs section" of a security template controls many of the properties of the three event logs: application, security, and system. The *hisecweb* security template deals mainly with the security log. All settings for the other two logs are set to "Not defined" except for the "Restrict guest access to application log" and the "Restrict guest access to system log" entries. These are set to "Enabled". Below are the settings for the options referring to the security log.

Maximum security log size – 10240 kilobytes
This determines the maximum size of the security event log. The two factors that determine the size of the log file are, how fast the logs grow and how often are the logs archived. The longer the period between log archives, the larger the log file should be. With hard drives being so large these days, there is no reason to cause yourself problems by making your security log too small.

Restrict guest access to system log – Enabled
This setting prevents guests from accessing the security event log.

Retain security log – Not defined
This setting determines how many days an event should be kept in the security log. It is only valid if the retention method is set to "by days".

Retention method for security log – As needed
This setting will allow the security log to overwrite older events when the log is full. The other options are to allow only events that are a certain number of days old to be overwritten, or to not allow any events to be overwritten and force the log file to be cleared manually.

Shut down the computer when the security audit log is full – Not defined
Microsoft recommends using the "Shut down system immediately if unable to log security audits" setting in the Security options section of the template. Since that particular setting was also "Not defined", there will be no action taken if the system is unable to log a security audit event.

4.2.4 Restricted Groups

The Restricted Groups section of a security template allows an administrator to define which users should and should not be members of a particular group. It also allows the administrator to determine if the restricted group is a member of any other groups. The only restricted group in the *hisecweb* security template is the Power users group. The settings do not allow any users to be a member of this group, and the Power users group will not be a member of any other group.

4.2.5 System Services

The System Services section allows an administrator to set the start-up mode of all system services and the access permissions for those services. The *hisecweb* security template has many services not defined, but the ones that are addressed are listed below.

IIS Admin Service	Automatic
IPSEC Policy Agent	Automatic
World Wide Web Publishing Service	Automatic
Alerter	Disabled
ClipBook	Disabled
Computer Browser	Disabled
DHCP Client	Disabled
Fax Service	Disabled
Infrared Monitor	Disabled
Internet Connection Sharing	Disabled
Messenger	Disabled
NetMeeting Remote Desktop Sharing	Disabled
Print Spooler	Disabled
Remote Access Auto Connection Manager	Disabled
Remote Access Connection Manager	Disabled

Remote Registry Service	Disabled
Task Scheduler	Disabled
Telephony	Disabled
Terminal Services	Disabled

4.2.6 Registry

The Registry section allows an administrator to define access permissions and audit settings for registry keys. There are no registry settings in the *hisecweb* security template.

4.2.7 File System

The File System section allows an administrator to define access permissions and audit settings for file system objects. There are no file system settings in the *hisecweb* security template.

5.0 Applying the Template

The role and configuration of the web server is going to make the application and maintenance of the *hisecweb* security template a relatively easy task. In order to apply the template to the system the Security Configuration and Analysis snap-in is needed. Microsoft provides detailed instructions on how to load the Security Configuration and Analysis snap-in into the MMC and apply a security template to a computer in the Microsoft Knowledge Base Article Q309689.[5] Like many other functions in the Windows operating system, applying the security template can also be accomplished with a command line utility. The secedit.exe command can be used to both apply a template and perform a comparison of a system's existing security settings to a security template.

The maintenance of the security imposed by the security template is going to be a procedure outlined by the security plan. The security plan for the system will dictate how often the security posture of the system should be evaluated, and at that time the Security Configuration and Analysis snap-in can be utilized to make sure no changes have occurred to the system security. The security plan for this particular system calls for a minor security review every six months and a major security audit conducted by the security team will occur once a year. During these review and audit periods, any desired changes to the security template can be applied to the system. Changes to security settings at times other than the review and audit periods must go before a configuration control board for emergency approval. Of course, down time for the system should be scheduled if any changes will be made so that those changes can be tested. It would be

much better if any changes could first be implemented on a test machine, but current resources do not allow that luxury.

6.0 Testing the Template

Now that the security template has been applied, verification is needed to make sure that it is working as expected. In order to do this, tests will be run on several of the security functions.

6.1 Account Policies

The first series of tests to be run will be on several account policies. The first test will be of the password complexity requirements and password length requirement under the Password Policy, and the second test will be of the account lockout threshold under the Account Lockout Policy. The reset account lockout counter setting will also be tested.

Test:
In order to test the functionality of the password complexity requirements, a test account was created with the username of "testuser". The initial password for the "testuser" account met all of the complexity requirements. The administrator then tried to reset the password of "testuser" using the following entries:

- test!123 – failed because it contained part of the account name.
- abergy1234 – failed because it did not meet 3 of the 4 category requirements, no capitalization or special characters.
- Abergykkkk – also failed because it did not meet 3 of the 4 category requirements, no numerics or special characters.
- A!1f5t – failed because it did not meet the length requirement of 8.
- Trst!1234 – meets all requirements and was successful.

Result:
The complexity requirement and the password length requirement are working as expected.

Test:
In order to test the account lockout threshold, an attempt was made to logon to the system using the "testuser" account and an incorrect password. 5 incorrect passwords were entered, and the account was locked out of the system. The administrator logged in and unlocked the "testuser" account. Another attempt to logon to the system using the "testuser" account was made and 4 incorrect passwords were entered. Following the fourth failed logon attempt the correct password was used and the "testuser" account was successfully logged onto the system. The "testuser" account was then logged off of the system. Several more logon attempts were made using incorrect passwords to ensure that the invalid

logon attempt count for the “testuser” account had been reset after a successful logon. 5 invalid logon attempts were again required to lock the “testuser” account out of the system.

Result:
The fact that 5 invalid attempts were needed to log the user out of the system after the successful logon proves that the account lockout threshold of 5 invalid logon attempts is working correctly.

Test:
To test the reset account lockout counter setting, 4 attempts were made to log on the system using the “testuser” account and an incorrect password. After 40 minutes another attempt was made to log on to the system using the “testuser” account and an incorrect password. If the reset account lockout counter policy had not been working correctly, the “testuser” account would have been locked out. 5 logon attempts were made before the “testuser” account was locked out of the system.

Result:
The “Reset account lockout counter after” setting of 30 minutes is working correctly.

6.2 Local Policies

To check the implementation of the local policies, two of the audit policies will be tested; “Audit account logon events” and “Audit account management”. Several of the security options including “Do not display last user name in logon screen”, “Message text for users attempting to log on”, and “Message title for users attempting to log on” will also be checked.

Test:
Testing the audit policies will be an easy task since the testing of the account policies should have created numerous entries in the security log. In order to verify the “Audit account logon events” setting of “Success, Failure” is working correctly, the logs were searched for entries created by previous logon attempts by the account “testuser”. As expected, there were numerous failure audit entries. There were actually two entries in the log reporting each failed logon attempt. The first was created by the “Audit account logon events” setting, and was event id 681. The second was created by the “Audit logon event” setting and was event id 529. In addition to the failure entries, there were also successful logon entries. Like the failure entries, both the “Audit account logon events” setting and the “Audit logon event” setting caused an entry indicating a successful logon attempt. The event ids were 680 and 528, respectively.

Result:

The log entries of both successful and failed logon attempts are being correctly added to the security log.

Test:
Another audit policy to check is the "Audit account management" setting. If there is any type of change to the status of a user account, especially involving the administrator group, the system administrator would want to know about it. As before, our activities in testing the account policies should have created a log entry that we can key on. It was found that the unlocking of the "testuser" account did create a log entry with event id 642 that specifies that the "Administrator" account unlocked the "testuser" account. To further test the Account management setting the administrator added the "testuser" account to the "Administrator" group. This action was successfully reflected in the security event log with event id 636. Following this test, the "testuser" account was removed from the "Administrator" group.

Result:
The account management actions are being correctly added to the security event log.

Test:
In order to test several of the Security options, one will simply have to log out of the system and log back on. If the "Do not display last user name in logon screen" setting is working correctly, the username field on the logon screen will be blank. Also, if the "Message text for users attempting to log on" and "Message title for users attempting to log on" settings are working correctly, the default title and message text will appear after pressing CTRL+ALT+DEL and before the logon screen appears.

Result:
The dialogue box with the default title and text appeared correctly. The logon screen also appeared with a blank username field indicating that these settings are working as planned. While the message text and title may not seem important to some, some organizations have deemed it necessary and stress its implementation on each new system.

6.3 Event Log

One other group of settings that needs to be tested is the Event Log settings for the security event log. It is especially important to make sure that the security log size has been changed to the 10240 kilobytes specified in the *hisecweb* security template. This setting and the retention method setting can be determined by viewing the properties of the security event log. The restriction on guest access to all of the logs will also be tested.

Test:
In order to check the maximum security log size setting and the retention method set for the security log, one simply needs to view the properties dialogue box. This can be accomplished by starting the Event Viewer and right clicking on the Security Log in the left pane. The Properties entry can then be chosen from the menu.

Result:
Both the maximum security log size setting and the retention method are correctly configured.

Test:
While the "Guest" account is disabled by default, there are two built-in accounts that are members of the guest group. These are the IUSR_machinename and IWAM_machinename accounts that are used for anonymous access to Internet Information Services. To perform the test of the restrictions on guest access to the logs, the password of the IUSR_machinename account is manually changed, and then that account is used to log in to the console. An attempt is then made to start the Event Viewer program. As expected, access the event logs was denied and the user was presented with the dialogue box in Figure 3. A similar message was received regarding access to the Application Log.

Figure 3 – System Log Access Denied

What is interesting to note, is the message regarding the Security Log was different. It can be seen in Figure 4. The reason for this is the Security Log is also protected by a setting in the "User Rights Assignments" under "Local Policies". The "Manage auditing and security log" setting is not defined in the *hisecweb* security template, but the default setting is Administrators only.

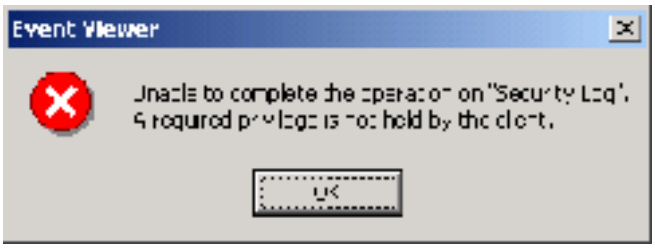

Figure 4 – Security Log Access Denied

Result:
The System, Security, and Application event logs are protected from access by guests, and the Security log has another layer of protection with the "Manage auditing and security log" user right.

6.4 System Services

The final test, and one that is quite important, is a check to see if the system services portion of the template was applied correctly. The disabling of unneeded services is considered a key part of not only the *hisecweb* security template, but of any good security plan.

Test:
To test the system services, a manual review of the Services console will be sufficient. Sorting the Services console and the security template settings by startup type makes it easy to do a comparison. The three services set to automatically start, IIS Admin Service, IPSEC Policy Agent, and World Wide Web Publishing Service, all have the automatic startup type, and have all successfully started. The sixteen services that have been disabled by the security template are indeed disabled in the Services console.

Result:
The services addressed by the *hisecweb* security template all have the appropriate startup type, and those that were configured to automatically start have done so successfully.
It should be noted that there are many services that are not defined by the *hisecweb* security template, and these should be reviewed in the Services console to determine if their default setting is appropriate.

7.0 Testing the System

Now that there is confidence that the security settings from the *hisecweb* security template are correctly applied, further testing is required to make sure that the system still functions as expected.

Since the main purpose of the computer is to act as a web server, the administrator needs to make sure that the Internet Information Service is working properly and the web pages are still accessible. The settings of the World Wide Web Publishing Service were tested in the "Testing the Template" section of this document, but since this service is key to the system role and goes hand in hand with the ability to view a web page, it will be tested again.

Test:
One function of security templates is to allow the administrator to define the start-up mode of the system services and their access permissions. One would expect that a security template designed to secure a web server would automatically start the World Wide Web Publishing Service. In order to test this again, the settings were checked in the "Services" snap-in, and it was verified that the startup type was "Automatic". The machine was then rebooted to verify that the service did indeed start automatically.

After establishing that the World Wide Web Publishing Service is started on system boot, confirmation is needed that the web pages are accessible over the network. To run this test, the system was connected to the network, and then an attempt was made to access the default web page from a user workstation. Using Internet Explorer 6.0, the user was able to view the web page as can be seen in Figure 5. Another user attempted to view the web page from another workstation, and that attempt was successful also.

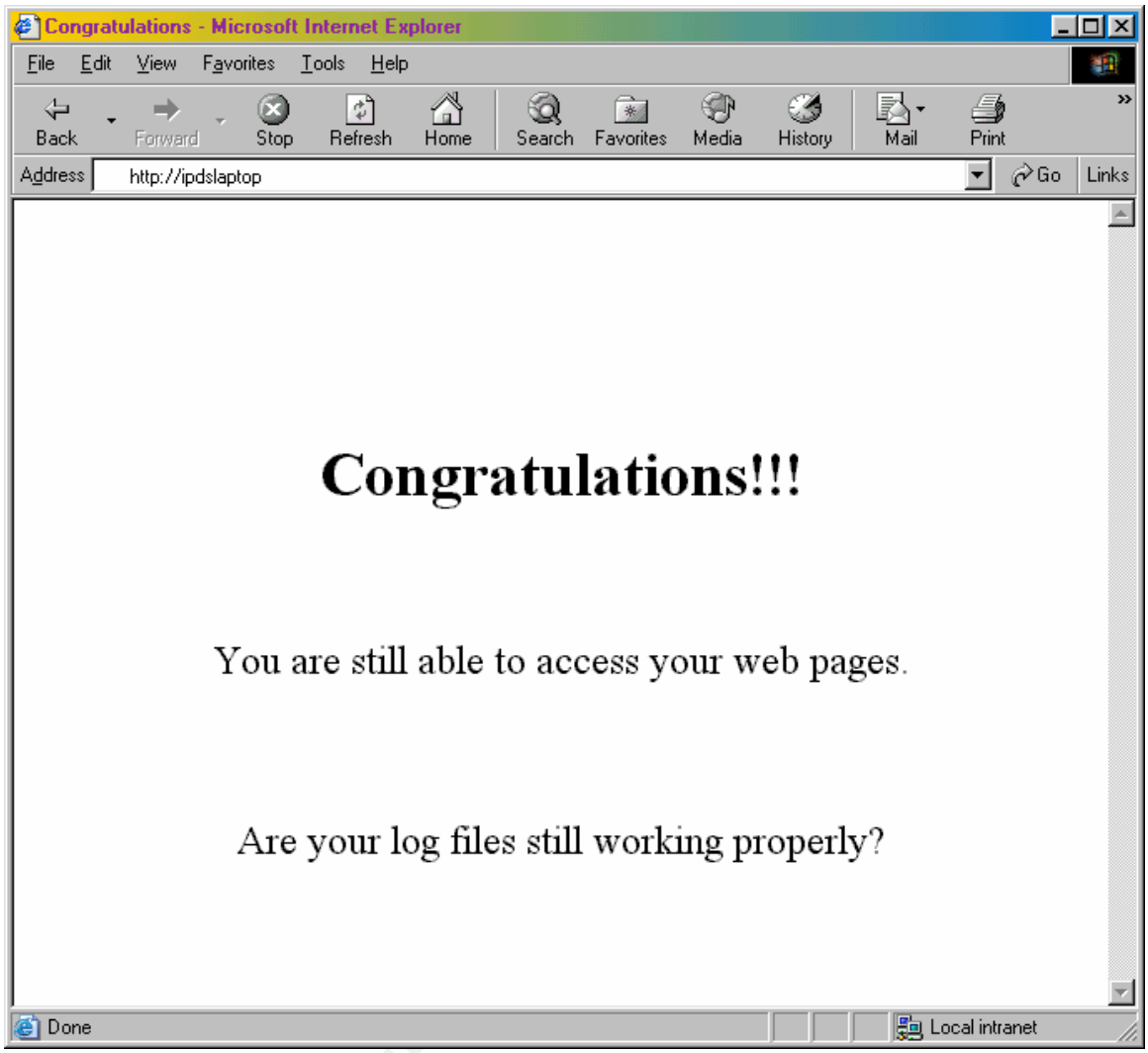

Figure 5 – Default Web Page

Result:
The World Wide Web Publishing Service is started automatically when the computer is booted up, and the web pages being served are accessible from the network.

The second function in need of testing is the generation of log files by the web server. The log files are important not only for the sake of security, but are also helpful in troubleshooting various problems that can occur with the web site. The log file settings can be configured within the properties of the web site, which can be accessed by right clicking on the web site name within the Internet Information Services snap-in. As suggested in the IIS 5.0 Baseline Security Checklist, logging using the W3C Extended Logging format has been enabled.

Test:

The actions taken in testing the accessibility of the web pages should have created entries in the web server log files. These logs are located in the WINNT\system32\LogFiles\W3SVC1 directory and a new log file is created each day. Since the previous tests have been the only access to the web server today, the logs should be short, but should contain information concerning the two web page accesses. The log did contain entries for the two network accesses conducted in the first test, in addition to an entry caused by access from the local machine. The logs can be seen in Figure 6. Note: The IP addresses have been modified for security reasons.

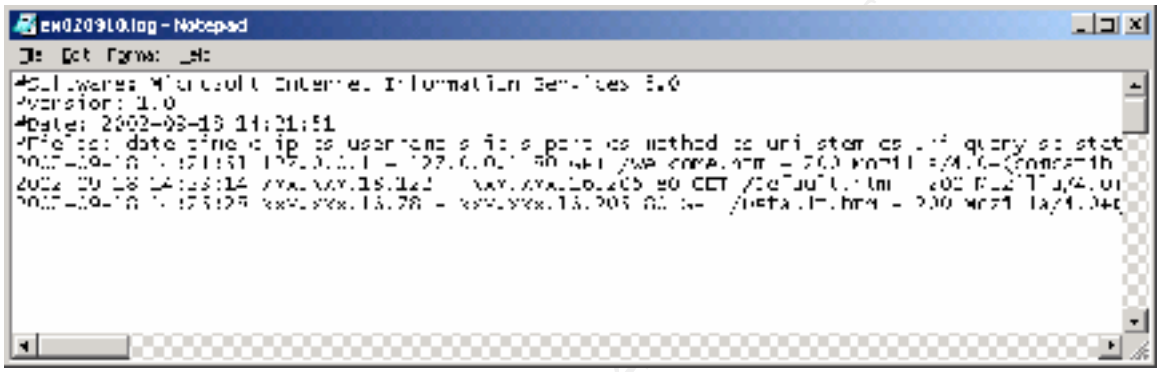

Figure 6 – Web Server Logs

Result:
The application of the *hisecweb* security template has had no effect on the creation of web server log entries.

Since the role of this system is strictly to serve web pages, there are not a lot of other applications installed. One application that is installed and should be installed on all systems is the anti-virus software. As stated earlier, this machine is running Norton Antivirus Corporate Edition, with File System Realtime Protection enabled. It would be a major problem if the application of the security template somehow affected the functionality of the anti-virus software.

Test:
Testing to check the functionality of the anti-virus software can be a tricky endeavor. After all, one does not want to inadvertently infect the system if the software is not working correctly. It was decided to take several CD-ROMs that have been obtained from security courses and browse them with Windows Explorer. If the virus protection software is working correctly, it should produce an alert warning of the presence of a virus.

The first CD-ROM browsed was from the SANS Institute. It is expected that this CD-ROM will have a hacker tool on it that would be considered a virus by the software. It wasn't long before the warning in Figure 7 was received.

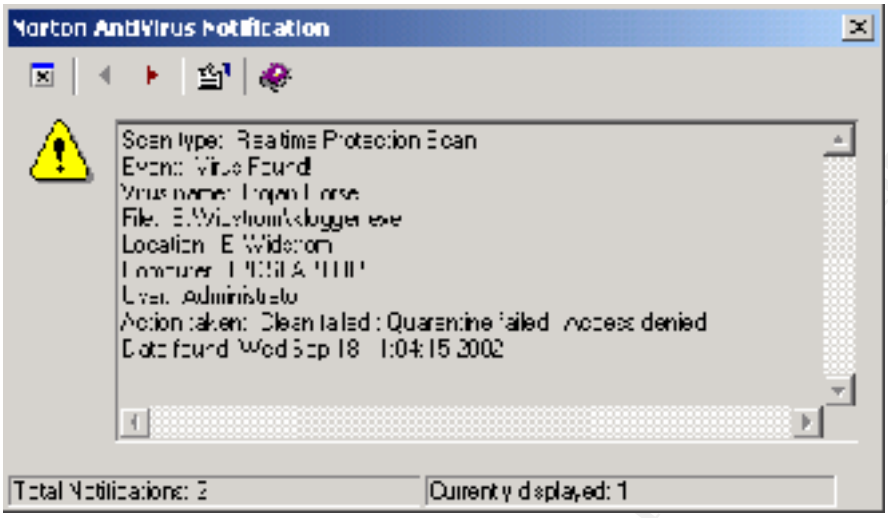

Figure 7 – Virus Warning from SANS CD-ROM

Another CD-ROM that had been obtained from a recent class on hacking was then browsed. It was certain that this CD-ROM contained virus software and the following warning in Figure 8 was displayed immediately.

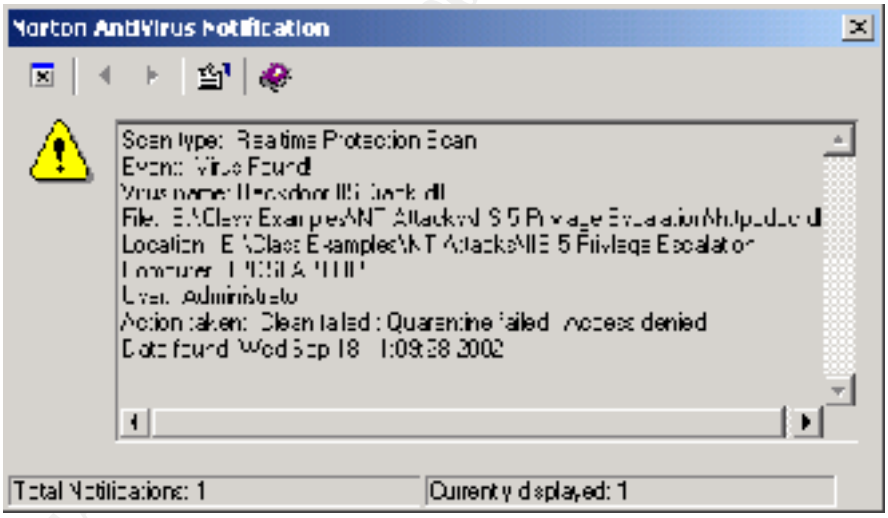

Figure 8 – Virus Warning from Hacking CD-ROM

Result:
The anti-virus software successfully identified virus files on the system. The action taken by the anti-virus software failed due to the fact that the virus files were located on a read-only media.

8.0 Template Evaluation

Many of the settings in the *hisecweb* security template were appropriate for the particular implementation of the test system, but a few were not. To begin with the default password complexity requirements are good, but they need to go one step further. The password should be forced to contain at least one character from all four of the listed categories rather that three. As stated before, strong passwords are key to the security of a system and their complexity prevents them from being easily cracked. The password complexity requirements are derived from the passfilt.dll file. Password complexity requirements can be changed by acquiring or creating a new passfilt.dll file that fits the particular needs of the system.

Referring to the "Account lockout duration" setting, the local environment does not require that an administrator be required to unlock an account. That value will be reset to sixty minutes. Since administrators are only allowed to logon to the console, and an administrator may not be available to logon to the console for quite a while, it would be desirable if the user could be able to try again after one hour. Frequent review of the logon events in the security log file will indicate if there is any malicious intent, and the appropriate action can be taken.

Within the "User Rights Assignments" the only setting defined is the "Access this computer form the network" policy. Another setting that should be defined is the "Deny access to this computer from the network" policy. This should be set to "Administrators". It is an assumption of the security template that administrators are not allowed to log on over the network, and this setting can reinforce that assumption.

Under the "Security Options", the "Do not display last user name in logon screen" option will be disabled. There are good arguments to support each setting for this policy, but since physical access to the server is controlled, there is little chance of someone stealing a username from this system. The ability to see whom the last person was to logon to the system outweighs the risk of username theft in this case. Also under "Security Options" the "Prompt user to change password before expiration" policy will be enabled. This is mainly for convenience sake and will be set to 10 days.

One other change that will be made to the "Security Options" is prompted by changes that will be made to the "Event Log" settings. The "Retention method for security log" will be changed from "as needed' to "by days" and the "Retain security log" policy will be set to 15 days. The retention method setting is being changed because the loss of any of the security logs is not acceptable. Log entries over 15 days old will be allowed to be overwritten because the security plan states that an archive of the security logs will be made every 7 days. This means that there will be at least 2 copies of the log entries before they are potentially erased. With the security event log configured in this manner, the "Shut down system immediately if unable to log security audits" policy will be enabled. Since the information on this system is not sensitive or mission critical,

it is desirable to have the system unavailable rather than allow someone to conceal their attempts to compromise the system and use it to attack other systems on the network. These changes will make it important to monitor the growth rate of the security log to ensure that the combination of settings that have been chosen are appropriate and do not cause the system to shut down needlessly.

The other main deficiency noted in the *hisecweb* security template is the lack of entries relating to the security of the registry and file system objects. The default security of a Windows 2000 server, in regard to the registry and the file system, is not adequate for an IIS 5.0 web server, but the web server security template offered by Microsoft does not address any of these issues. It is suggested that an administrator should investigate other security templates that place restrictions on the registry and file system and incorporate those restrictions into the *hisecweb* template. A good place to start is the *Web_Secure.inf* template created by SystemExperts Corporation.[6]

Overall, I am pleased with the performance of Microsoft's security templates and the *hisecweb* template in particular. It must be stressed though, that the use of the template alone will result in a system configuration that is wide open to vulnerabilities. I chose to highlight the IIS 5.0 Baseline Security Checklist along with the *hisecweb* security template, but there are many other tools that should be utilized to establish and maintain a good security posture. Not discussed in this document was the IIS Lockdown Tool[7] that can perform some of the actions outlined in the IIS 5.0 Baseline Security Checklist. This tool can also be used to apply security templates. Also not discussed was the Microsoft Baseline Security Analyzer (MBSA)[8] that utilizes the HFNetChk[9] tool to test for hotfixes and security packs. The MBSA will also perform some of the actions outlined in the IIS 5.0 Baseline Security Checklist and will even inform the user if the IIS Lockdown tool has been run on the target computer. A system administrator should use all of these tools when initially setting up a system, and should continue to use them to maintain the security of the computer.

The discussion of the various Microsoft tools needed to secure a computer brings up one important question. Why can't a single tool be created that combines the functionality of all of the current tools? The current tools have overlapping functionality, but each one also offers something unique. If all of the functionality of all of the tools were combined into one, it would make the life of an administrator much simpler.

9.0 References

[1] Microsoft Knowledge Base Article Q313434 – "HOW TO: Define Security Templates in the Security Templates Snap-in in Windows 2000"

[2] Microsoft Technet "IIS 5.0 Baseline Security Checklist"
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/chklist/iis5cl.asp

[3] Microsoft Technet Article "Secure Internet Information Services 5 Checklist"
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/chklist/iis5cl.asp

[4] MSDN: Windows 2000 Security Settings Explained
http://msdn.microsoft.com/library/default.asp?url=/library/en-us/gp/615.asp

[5] Microsoft Knowledge Base Article Q309689 – "HOW TO: Apply Predefined Security Templates in Windows 2000"

[6] SystemExperts Corporation – Web_Secure.inf Security Template
http://www.systemexperts.com/win2k/HardenWin2K.html

[7] Microsoft Technet "IIS Lockdown Tool"
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/tools/locktool.asp

[8] Microsoft Technet "Microsoft Baseline Security Analyzer"
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/Tools/MBSAhome.asp

[9] Microsoft Technet "HFNetChk"
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/tools/hfnetchk.asp