



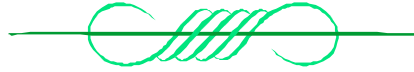
Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Securing Windows and PowerShell Automation (Security 505)"
at <http://www.giac.org/registration/gcwn>



Securing a Web Development Workstation with the NSA Security Template

**GIAC Certified Windows Security Administrator (GCWN)
Practical Assignment v3.1**

Option 2 – Securing Windows 2000 With Security Templates

Author: Bill Sterns, GSEC
Date: 3/12/2003



ABSTRACT

This paper details how to secure a web development workstation using the Windows 2000 Professional Workstation template developed by the National Security Agency. It describes the current state of a development workstation, describes the important settings in the template, and recommends changes to the template to meet the security needs of the computer. It also shows how to use the template to update the security settings of the computer and how to test the machine to insure that the machine made it through the application of the template without any functionality loss.

© SANS Institute 2003, Author retains full rights

CONVENTIONS USED IN THIS DOCUMENT

- **Bold** text indicates clickable items in a window.
- *Italic* text indicates text to be typed or names of files.
- **Bold red** text indicates modified template values.

© SANS Institute 2003, Author retains full rights.

TABLE OF CONTENTS

1	INTRODUCTION	1
2	ROLE.....	1
3	SYSTEM CONFIGURATION.....	1
3.1	HARDWARE CONFIGURATION	1
3.2	SOFTWARE CONFIGURATION	2
4	SECURITY REQUIREMENTS.....	3
5	TEMPLATE SELECTION	4
6	SECURITY SETTINGS.....	5
6.1	THE SECURITY CONFIGURATION AND ANALYSIS TOOL	5
6.2	ANALYZING THE WORKSTATION	6
6.3	REVIEWING THE RESULTS AND MODIFYING THE TEMPLATE	8
6.3.1	Password Policy	8
6.3.2	Account Lockout Policy	11
6.3.3	Kerberos Policy	12
6.3.4	Audit Policy.....	12
6.3.5	User Rights Assignment.....	15
6.3.6	Security Options	17
6.3.7	Settings for Event Logs	20
6.3.8	Restricted Groups	22
6.3.9	System Services.....	22
6.3.10	Registry and File System Settings.....	24
7	APPLYING THE TEMPLATE.....	24
8	TESTING THE COMPUTER.....	27
8.1	TEST 1: PASSWORD AND ACCOUNT LOCKOUT POLICIES	27
8.2	TEST 2: EVENT LOGGING	29
8.3	TEST 3: TIME SYNCHRONIZATION.....	34
8.4	TEST 4: LOGGING ON AS A NORMAL USER.....	34
8.5	TEST 5: DEVELOPER FUNCTIONALITY	38
9	CONCLUSION.....	41
10	REFERENCES.....	42

1 INTRODUCTION

As new updates to the ever-popular Windows operating system continue to be released, corporations with large networks of servers and workstations find it increasingly difficult to keep their individual machines up-to-date. Newer versions of Windows such as 2000 and XP promise previously unheard-of levels of stability and security, but these benefits come at the price of greatly increased complexity. Small corporations with small networks might be able to effectively handle configuring and updating every computer individually. On the other hand, large corporations who are faced with upgrading hundreds or even thousands of computers from one version of Windows to the next would find individually configuring every machine to require an extremely huge amount of manpower. For these large organizations, this process would take so much time that a new version of Windows would likely be released before the last machine was finished. Luckily, with the advent of Windows 2000, a new tool is available which can break this vicious cycle and rescue IT workers from doing nothing but upgrading and configuring Windows from now until eternity. This tool is called Security Configuration and Analysis. Through the use of predefined security templates, this tool can be used to completely configure the security settings of a computer by following a few easy steps, with no manual changes required once the templates have been saved. This paper will step through how to use this tool to check the state of a computer, apply a security template, and verify that nothing broke as a result of the previous two steps.

2 ROLE

The computer I have chosen to configure is a Windows 2000 Professional workstation. It is one of many new PCs that have been purchased by my organization to replace older machines running NT 4. The particular workstation I will be securing is a web development box. The web applications that will be developed on this computer consist of Java Servlet/JSP applications written in Java, ASP applications written in VBScript, and regular HTML pages that make use of client-side JavaScript.

3 SYSTEM CONFIGURATION

3.1 HARDWARE CONFIGURATION

The specs of the workstation are as follows:

Compaq Evo Workstation

CPU: Intel Pentium 4, 1.7 GHz

Memory: 256 MB

Hard drive: 40 GB

Partition 1: 20GB NTFS (C:)
Partition 2: 20GB NTFS (D:)
Video RAM: 32 MB
CD-ROM: 52x

3.2 SOFTWARE CONFIGURATION

Although this system will be acting as a server to some degree due to the web servers that will be installed on it, the primary role of this machine will be a development workstation. The web servers running on this machine will only be used for development activities and will not be used in a production environment. Windows 2000 Server was briefly considered as the operating system of this machine due to the fact that its version of IIS supports having multiple web applications simultaneously listening on different ports, but 2000 Professional was eventually chosen because of the additional overhead of 2000 Server, the fact that 2000 Professional is the company standard for personal workstations, and the fact that the majority of our web development is done with Java and Apache Tomcat rather than VBScript and IIS.

Service Pack 3 has been installed via Windows Update as well as the following critical security updates:

- Q323172:** Flaw in Certificate Enrollment Control Could Allow Deletion of Digital Certificates
- Q323255:** Unchecked buffer in HTML Help can lead to Code Execution
- Q324096:** Buffer Overrun in SmartHTML Interpreter Could Allow Code Execution
- Q324380:** Cryptographic Flaw in RDP Protocol can Lead to Information Disclosure
- Q324929:** Cumulative Patch for Internet Explorer 6 Service Pack 1
- Q326830:** Unchecked Buffer in Network Share Provider Can Lead to Denial of Service
- Q326886:** Flaw in Network Connection Manager Can Cause Rights Elevation
- Q328310:** Flaw in Windows WM_TIMER Message Handling Could Enable Privilege Elevation
- Q329115:** Certificate Validation Flaw Could Enable Identity Spoofing
- Q329170:** Flaw in SMB Signing Could Enable Group Policy to be Modified
- Q329414:** Buffer Overrun in Microsoft Data Access Components Could Lead to Code Execution
- Q329834:** Unchecked Buffer in PPTP Implementation Could Enable Denial of Service Attacks
- 810030:** Flaw in Microsoft VM JDBC Classes Could Allow Code Execution
- 810649:** Hyperlinks Open in Internet Explorer Instead of in the Default Browser
- 810833:** Unchecked Buffer in Locator Service Could Lead to Code Execution

MS02-008: XMLHTTP Control Can Allow Access to Local Files

The following additional software has been installed:

- **Utilities**
 - Acrobat Reader 5.0
 - Meetingmaker Client 7.1
 - Microsoft Office 2000
 - WinZip 8.1
- **Anti-virus software**
 - Norton AntiVirus Corporate Edition 7.61.934
- **Web browsers**
 - IE 6 SP1
 - Netscape Communicator 4.78
 - Netscape 7.01
 - Opera 6.05
- **Web servers**
 - Apache Tomcat 4.0.4
 - IIS 5.0
- **Development tools**
 - Java 2 SDK 1.4.1
 - JCreator 2.5 LE
- **Log consolidation**
 - NTSyslog 1.13
- **Other software**
 - WinDump 3.5 (Windows port of TCPDump)

4 SECURITY REQUIREMENTS

The primary functional requirement of a development machine is that it be fully capable of supporting the various kinds of development activities that occur on it. In this particular case, the users of this machine must be able to use the development tools to create web applications, compile their code, and test out the applications on a variety of platforms.

A very important operational/security issue discovered while researching this paper is described in the Microsoft Knowledge Base article entitled "Web Site Operator Capabilities and Limitations". Basically, in order for a developer to be able to administer IIS, their user account must be a member of the local Administrators group. This requirement goes against the principle of least privilege for these users, but it is unfortunately necessary for developers to be able to do their jobs.

Additional security requirements for this workstation include:

- There are no NT/2000 domains in the particular environment of this workstation, so domain access is not required.
- The workstation must be able to connect to the local NT workgroup.
- Certain security events on all workstations and servers are sent to a central syslog server for consolidation. Each type of syslog event is saved to a different log file on the server, and these logs are consolidated using an internally developed log consolidation tool. To accomplish this, this workstation will need to be able to connect to udp/514 on a Unix box running syslogd. This will be implemented via the free *NTSyslog* utility.
- The workstation must be able to connect to an external NTP server for time synchronization.
- All access to this workstation must be logged.
- Any malicious activity on this workstation should be logged if possible. This is especially important considering that developers will be in the local Administrators group and therefore capable of doing much more damage than the average user. Refer to sections 6.3.4 and 8.2 for more information about this.
- Strong password requirements defined by our organization must be enforced.
- The workstation will not be able to connect to the Internet, so any services that rely on an Internet connection should be disabled.
- Any unnecessary services should be disabled.
- The workstation should remain as up-to-date as possible with service packs and critical security updates.

This machine will be on a secured network inaccessible from the general Internet, so physical and software security is not as critical of an issue as it would be for a machine that anyone on the Internet can potentially connect to. However, organizational and best-practice security policies should be followed at all times.

5 TEMPLATE SELECTION

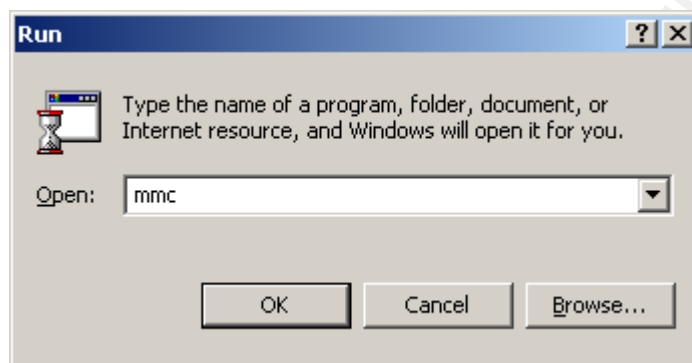
To secure the development machine, I have chosen `w2k_workstation.inf`, the Windows 2000 Professional Workstation template developed by the National Security Agency. This template was chosen partly because of its close approximation to the security needs of our organization. In particular, our organization is required to adhere to strict password policies. The NSA template met our needs in this regard very well. It was also chosen partly due to the fact that a template blessed by a United States government organization carries more weight than one that is not blessed by such an organization. This helps to reduce any internal political tension caused by the changes described in this paper.

6 SECURITY SETTINGS

6.1 THE SECURITY CONFIGURATION AND ANALYSIS TOOL

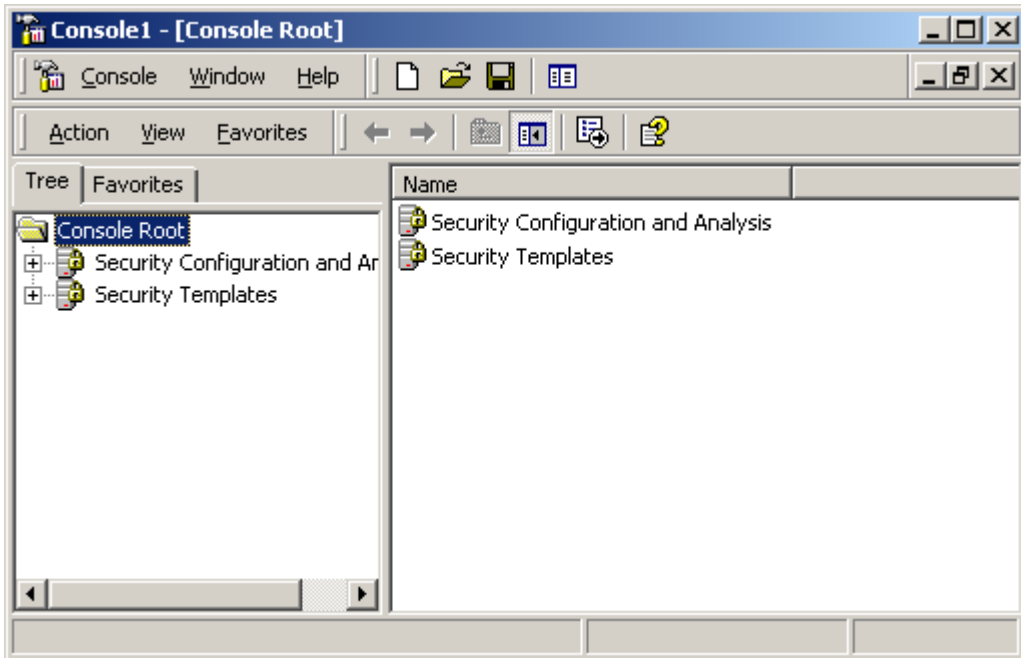
To load up the Security Configuration and Analysis tool, the following steps were followed:

1. I first logged into the workstation as a local Administrator.
2. I then loaded up the Microsoft Management Console.
 - a. I clicked **Start | Run**.
 - b. I typed “*mmc*” and clicked OK.

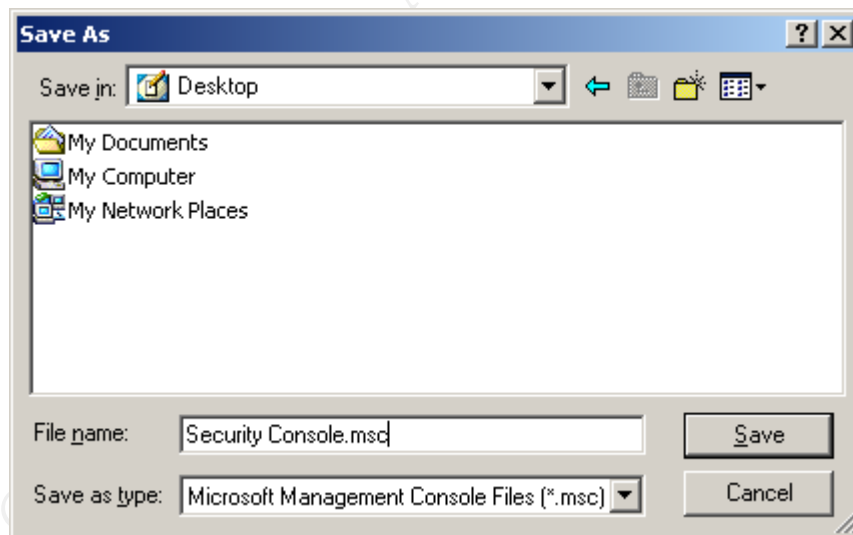


- c. An empty MMC window then appeared.
3. I then added the template-related snap-ins to the MMC console.
 - a. I clicked **Console | Add Remove Snap-in...**
 - b. In the Standalone tab, I clicked the **Add** button.
 - c. I clicked “**Security Configuration and Analysis**” to highlight it and clicked **Add**.
 - d. I clicked “**Security Templates**” to highlight it and clicked **Add**.
 - e. I clicked **Close**.
 - f. I clicked **OK** to close the snap-in manager.
4. The MMC console now looked like the following image:





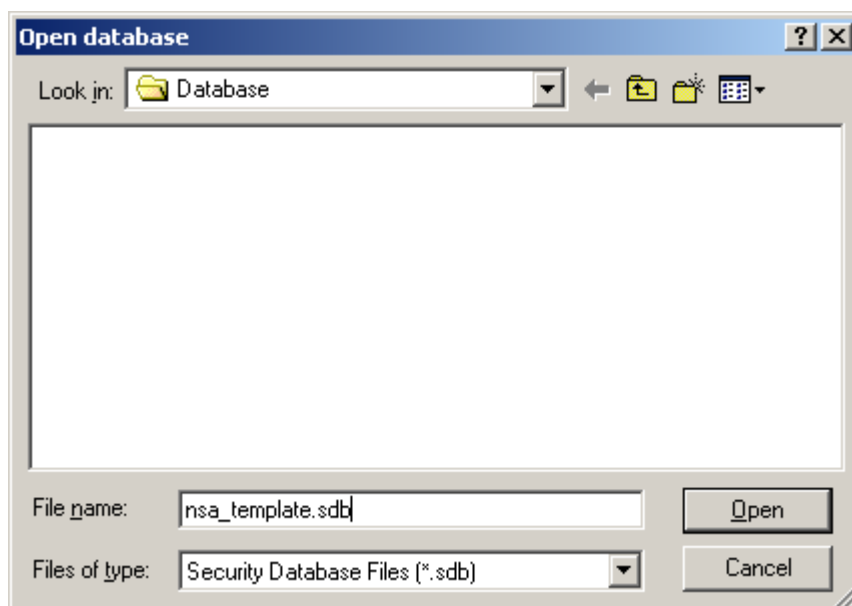
5. Finally, I saved the MMC console to the desktop for later use...
 - a. I clicked **Console | Save As**.
 - b. In the **"Save In"** box, I chose **Desktop**.
 - c. I then picked a name for the console and clicked **Save**.



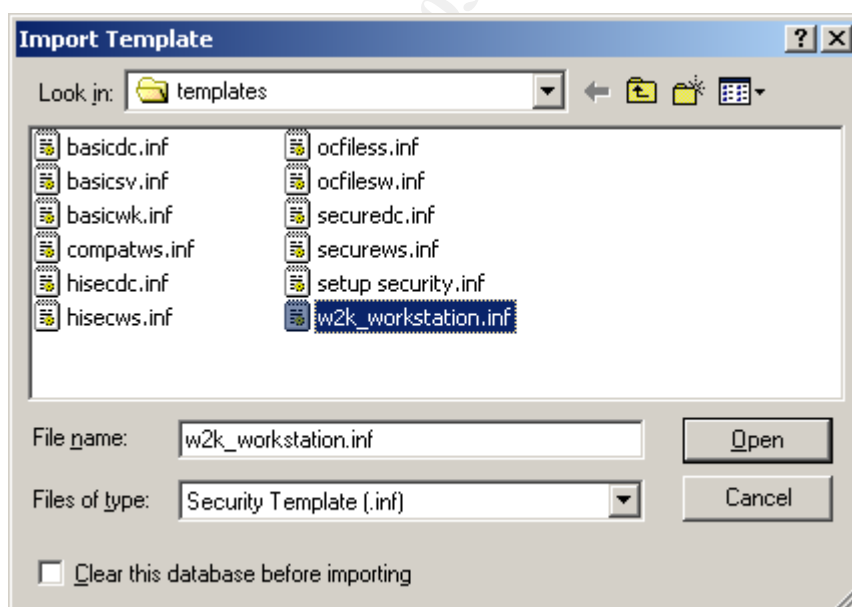
6.2 ANALYZING THE WORKSTATION

To compare the security settings on the workstation to the settings defined in the template, the following steps were followed:

1. In the MMC console, I right-clicked on “**Security Configuration and Analysis**” in the left-hand pane and selected “**Open Database**”
2. I then typed a name for a new database file and clicked the **Open** button

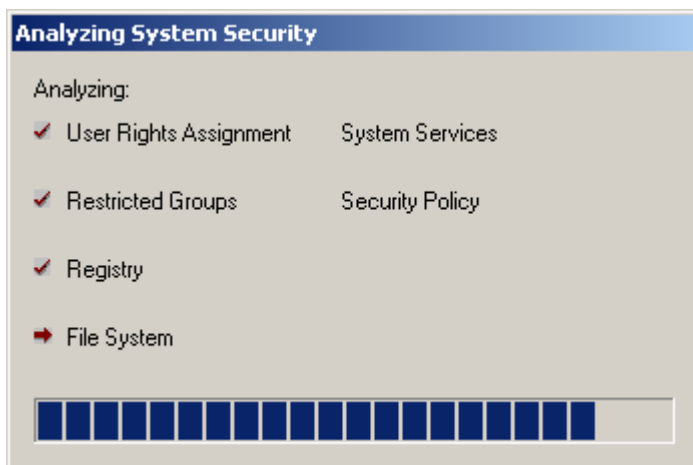


3. The Import Template window appeared. I then selected the desired template to check the computer's configuration against. In this case, the *w2k_workstation.inf* file was the one I wanted.







4. I then clicked **Open** to load the template into the SCA tool
5. To start analyzing the computer, I right-clicked on “**Security Configuration and Analysis**” in the left-hand pane and selected “**Analyze Computer Now**”.

6. I was prompted to choose a location for the error log that would be generated. The default path is usually ok, and is what I selected.
7. The security configuration of the computer was then compared to the settings in the chosen template.



8. When it finished analyzing the computer, I was returned to the MMC console. Note that “**Security Configuration and Analysis**” could now be expanded to view the results.
9. I saw one of the following icons next to each security setting. I used these icons to determine the status of each security setting on my computer.

	The security setting on the computer matches the value defined in the template.
	The security setting on the computer is not defined in the template.
	The security setting on the computer does not match the value defined in the template.
	The security setting for the given template entry is not available. For example, you will see this if the template entry references a registry key that does not exist on the computer.

6.3 REVIEWING THE RESULTS AND MODIFYING THE TEMPLATE







After analyzing the machine, the next step was to review the security settings that will be changed by the template and make sure that they meet organizational security policies.

6.3.1 Password Policy

The password policy determines how strong passwords must be and how often they need to be changed. Having a very lenient password policy would make it much easier for an attacker to compromise the passwords on the system. On the other hand, an extremely strict password policy can potentially cause major

annoyance and anger for the users that have to abide by it. Therefore, a middle ground must be found that keeps the users happy and the system reasonably secure.

The password policy template settings, as compared to the current configuration of the machine, are as follows:

Policy ▲	Database Setting	Computer Setting
 Enforce password history	24 passwords reme...	0 passwords remembered
 Maximum password age	90 days	42 days
 Minimum password age	1 days	0 days
 Minimum password length	12 characters	0 characters
 Passwords must meet complexity requirements	Enabled	Disabled
 Store password using reversible encryption for all users in the domain	Disabled	Disabled

Enforce password history

Old value: 24 passwords remembered

New value: **5 passwords remembered**

This setting determines how many previously used passwords will be remembered by Windows. Remembered passwords cannot be re-used. It is used to keep the user from using the same passwords over and over again. This value can range from 0-24. The template value of 24 seemed quite strict, and seemed to be overkill for my organization's needs. The value of 5 remembered passwords was chosen as a good balance between security and usability. It also matches the password history settings in our existing security policy.

Maximum password age

Value: 90 days

This setting determines how long a password can be used before it must be changed. It is used to limit the amount of time an attacker has to break a compromised encrypted password file before the cracked passwords become useless. This value can range from 0-999 days. The template value of 90 days seemed to be a reasonable value. This value also fits into the range of 30-90 days recommended by Microsoft as a best practice.

Minimum password age

Old value: 1 day

New value: **3 days**

This setting determines how long a password must be used before it can be changed. It prevents users from changing their password repeatedly in rapid succession to bypass the "Enforce password history" setting. This value can range from 0-998 days. The template setting of 1 day seemed to be too lenient.

Changing this to 3 days would make Windows much more resistant to this kind of behavior. As with the password history setting, the new value matches our existing security policy.

Minimum password length

Value: 12 characters

This setting determines how many characters a password must be in order for it to be accepted. Long passwords take longer to crack, so this setting, in combination with the “maximum password age” setting, will help prevent the passwords on the system from being compromised. This value can range from 0-14 characters. Because we will be using NTLMv1 authentication (as described in section 6.3.6), an attacker who has managed to access the network could potentially use L0phtCrack to sniff the LanManager password hashes of the user accounts on the computer. Since a password in a LanManager hash is broken into two 7-byte strings, allowing a password 7 characters or less would make the password length obvious since the second half of the password would be all null characters. In order to make both halves of the password difficult to crack, I have decided to stick with the template value of 12 characters, even though it might cause a bit of dissent among our users that dislike long passwords.

Passwords must meet complexity requirements

Value: Enabled

This setting activates additional checks to make sure the passwords chosen are strong. If this setting is enabled, then all passwords must be at least six characters long, must not contain the login or full name, and must contain three of the following four features:

- At least one uppercase character (A-Z)
- At least one lowercase character (a-z)
- At least one digit (0-9)
- At least one non-alphanumeric character (!, #, %, etc)

My organization requires that all passwords be complex, so this setting was left enabled. There are additional checks that I would like to be in there, but which Microsoft does not provide, such as a check for a minimum number of unique characters and the ability to attempt to crack the password with Crack or John the Ripper before accepting it. These can be added later via development of a custom password filter. A custom password filter, however, is beyond the scope of this paper.

Store password using reversible encryption for all users in the domain




Value: Disabled

This setting, if enabled, stores passwords using a two-way encryption algorithm. If this setting were enabled, the level of security would be similar to storing the plaintext passwords themselves. This policy is required by applications that use protocols that require the user's password for authentication. Challenge-Handshake Authentication Protocol authentication and Digest authentication in IIS are two such protocols that require this setting to be enabled. Since password security is very important to my organization, and since this workstation does not need to support these protocols, this setting can be left disabled.

6.3.2 Account Lockout Policy

The account lockout policy limits the number and frequency of invalid logon attempts for accounts on the system. As with the password policy settings, a middle ground should be found between security and user happiness.

The account lockout policy template settings, as compared to the current configuration of the machine, are as follows:

Policy ▲	Database Setting	Computer Setting
 Account lockout duration	15 minutes	15 minutes
 Account lockout threshold	3 invalid logon attempts	0 invalid logon attempts
 Reset account lockout counter after	15 minutes	15 minutes

Account lockout duration

Old value: 15 minutes
New value: **30 minutes**

This setting determines how long a locked account will stay locked before automatically becoming unlocked. Windows only uses this setting if the "Account lockout threshold" setting is used. This value can range from 0-99,999 minutes. Setting this value to 0 has a special meaning; if this value is set to 0, the account will never automatically become unlocked. In this case, if an account becomes locked, an administrator will have to manually unlock it. Having a low (but non-zero) value for this setting makes it easier for attackers to make multiple guesses of a user's account password, since there would be less time that would need to elapse between groups of guess attempts. Changing this to 30 minutes makes the machine more resistant to this type of attack. This value matches our existing security policy.

Account lockout threshold

Value: 3 invalid logon attempts

This setting determines how many invalid logon attempts can be made before an account is automatically locked out. It is used in combination with the "Account

lockout duration” setting to prevent attackers from attempting to access an account by making multiple guesses of a password. This value can range from 0-999 failed logon attempts. If this value is set to 0, the account will never be automatically locked out. The template value of 3 invalid logon attempts seems reasonable and matches our existing security policy.

Reset account lockout counter after

Old value: 15 minutes

New value: **30 minutes**

This setting determines how much time must elapse before the counter of failed logon attempts is reset to 0. Windows only uses this setting if the “Account lockout threshold” setting is used. This value can range from 1-99,999 minutes. Since it did not make much sense to set this value any lower than the account lockout duration value, this value was also changed to 30 minutes. If it were set to a lower value than the account lockout duration value, it would partially negate the advantages of having the account automatically lock out. If an attacker knew what this setting was, the attacker would be able to guess more passwords in a shorter period of time by making just enough guesses to avoid having the account lock out, wait for the lockout counter to reset, make more guesses, wait again, and so on.










6.3.3 Kerberos Policy

The servers that this workstation would potentially be connecting to are all Windows NT machines, none of which use Kerberos for authentication purposes. Also, this particular workstation will not be connecting to a domain, but only an NT Workgroup. Therefore, Kerberos policies do not apply to this workstation.

6.3.4 Audit Policy

These settings determine the types of events that will be stored in the Security event log. These settings can provide a good audit trail if the machine is attacked, but can also significantly slow down the performance of the machine if too many events are being logged.

The audit policy template settings, as compared to the current configuration of the machine, are as follows:

Policy ▲	Database Setting	Computer Setting
 Audit account logon events	Success, Failure	No auditing
 Audit account management	Success, Failure	No auditing
 Audit directory service access	No auditing	No auditing
 Audit logon events	Success, Failure	No auditing
 Audit object access	Failure	No auditing
 Audit policy change	Success, Failure	No auditing
 Audit privilege use	Failure	No auditing
 Audit process tracking	No auditing	No auditing
 Audit system events	Success, Failure	No auditing

The template values for these settings seem to be an excellent compromise between providing a good audit trail and limiting the performance hit on the system. However, some of these are not needed by this particular workstation. All of these settings are important to the security of the system, so I will briefly discuss each of them.

Audit account logon events

Old value: Audit Success and Failure

New value: **No auditing**

This setting determines whether to audit situations in which a user logs into another machine, but this machine actually does the authentication. If this machine were a domain controller, an entry would be made every time a user logs on or off a machine that is validated against it. On a regular workstation, this setting is identical to the “Audit logon events” setting. Since this machine is not a domain controller, and since logon event auditing will be enabled, auditing account logon events is redundant and unnecessary.

Audit account management

Value: Audit Success and Failure

This setting determines whether to audit account management activities such as creating new accounts, changing passwords, creating groups, etc. If this type of auditing were not enabled, an attacker would be able to change any account-related information on the machine without any trace of their activities. Microsoft recommends against enabling account management failure auditing due to the possibility that an attacker could cause a denial-of-service attack by filling up the log with denial messages. However, due to the relatively small size of the network that this computer will reside on, and due to the fact that this computer will not be accessible via the Internet, the benefits of turning on failure auditing seem to outweigh the possible problems that might arise.

Audit directory service access

Value: No auditing

This setting determines whether to audit situations in which a user accesses an Active Directory object that has a system access control list specified for it. Since this computer is not a domain controller running Active Directory, this type of auditing event will never happen. Therefore, the template value will suffice.

Audit logon events

Value: Audit Success and Failure

This setting determines whether to record every time a user logs on or off the computer. Microsoft recommends only enabling success auditing to prevent against denial-of-service attacks, as described in the “Audit account management” description above. However, for the same reasons described above, I have decided to enable failure auditing as well.

Audit object access

Value: Audit Failure

This setting determines whether to audit accesses of objects such as files, folders, registry keys, etc. Every object to be monitored must have auditing enabled for it by modifying its system access control list. If success auditing were enabled for this audit type and a large number of files had auditing enabled, then the security log might fill up very quickly, especially if a large number of users were simultaneously accessing the computer. Therefore, it makes sense to stick with the template value and only enable failure auditing.

Audit policy change

Value: Audit Success and Failure

This setting determines whether to audit changes to the Local Security Authority security policy configuration on the computer. Any changes to the security policy configuration on the computer would potentially leave the computer more vulnerable to attackers, so all pertinent information should be gathered. This is especially important since all developers will have the ability to change the security policy. Therefore, the template value for this setting is fine.

Audit privilege use

Old value: Audit Failure

New value: **Audit Success and Failure**

This setting determines whether to record every time a user exercises a special privilege, such as the ability to back up files or debug programs. Since developers are members of the local Administrators group, they have privileges on the system far beyond that of the average user. As a result, it will not be possible to prevent developers from abusing their extra privileges. However, turning on all auditing of privilege uses, in combination with the use of NTSyslog to store all logs on a separate Unix box and our custom log consolidation tool, will allow system administrators to be immediately notified if such abuses occur. Therefore, all successful and unsuccessful privilege uses should be audited.

Audit process tracking

Value: No auditing

This setting determines whether to record detailed information about the processes running on the computer. Enabling process tracking would significantly degrade the performance of the computer, as well as fill up the security log very quickly. Because of this, and because any security-related information gleaned from process tracking would be marginal at best, process tracking would be best left disabled. Therefore, the template value for this setting is fine.

Audit system events

Value: Audit Success and Failure

This setting determines whether to audit events such as the system starting up or shutting down, or events that might affect the system security or the security log. Enabling auditing for system events would log activities that might be indicative of an attack, such as an attacker wiping out the security log to cover their trail. Therefore, the template value for this setting is fine.

6.3.5 User Rights Assignment

These settings determine which users can perform certain actions on the system. Developers that use this computer will be placed into the Administrators group since they will need to be able to administer IIS, as described in the Security Requirements section. Being in this group will also give them the added benefit of being able to run legacy Windows applications, which is a benefit since our working environment has been completely NT-based for a number of years.

The user rights assignment template settings, as compared to the current configuration of the machine, are as follows:

Policy ▲	Database Setting	Computer Setting
Access this computer from the network	Users,Administrators	Backup Operators,Power Us...
Act as part of the operating system		
Add workstations to domain		
Back up files and directories	Administrators	Backup Operators,Administra...
Bypass traverse checking	Users	Backup Operators,Power Us...
Change the system time	Administrators	Power Users,Administrators
Create a pagefile	Administrators	Administrators
Create a token object		
Create permanent shared objects		
Debug programs		Administrators
Deny access to this computer from the network		
Deny logon as a batch job		
Deny logon as a service		
Deny logon locally		
Enable computer and user accounts to be trusted for delegation		
Force shutdown from a remote system	Administrators	Administrators
Generate security audits		
Increase quotas	Administrators	Administrators
Increase scheduling priority	Administrators	Administrators
Load and unload device drivers	Administrators	Administrators
Lock pages in memory		
Log on as a batch job		MDESS10\IWAM_MDESS10,...
Log on as a service		
Log on locally	Users,Administrators	Backup Operators,Power Us...
Manage auditing and security log	Administrators	Administrators
Modify firmware environment values	Administrators	Administrators
Profile single process	Administrators	Power Users,Administrators
Profile system performance	Administrators	Administrators
Remove computer from docking station	Users,Administrators	Power Users,Users,Administr...
Replace a process level token		
Restore files and directories	Administrators	Backup Operators,Administra...
Shut down the system	Users,Administrators	Backup Operators,Power Us...
Synchronize directory service data		
Take ownership of files or other objects	Administrators	Administrators

These settings are acceptable for this workstation. Several of these are particularly important privileges that developers need to have in order to do their job. These privileges are described below.

Access this computer from the network

Value: Users, Administrators

This setting determines which users can access this computer remotely via the network. Developers may need to do testing on other computers running older browsers; in order to do this, they would need to be able to access the code running on their own workstation.

Change the system time

Value: Administrators

This setting determines which users can change the time and date on the system clock. Time synchronization between the workstation and external LDAP and/or database servers is very important for web applications to function correctly. If the external NTP server becomes unavailable for some reason, developers would need a way to perform this time synchronization manually.

Profile single process

Value: Administrators

This setting determines which users can use performance-monitoring tools to analyze the performance of non-system processes running on the machine. It is often beneficial for a developer to be able to closely analyze a running process associated with an application being developed. It allows problems such as memory leakage, infinite processing loops, and excessive system resource usage to be diagnosed in a much easier fashion. Therefore, it would benefit developers to have this privilege.

Profile system performance

Value: Administrators

This setting determines which users can use performance-monitoring tools to analyze the performance of system processes running on the machine. Certain system processes, such as inetinfo.exe (the IIS webserver process) would potentially need to be analyzed for performance problems as described in the previous section. Therefore, it would benefit developers to have this privilege.

Shut down the system

Value: Users, Administrators

This setting determines which users can shut down the computer. Developers should be able to shut down their own machine, so they should have this privilege.

6.3.6 Security Options

These settings are used to enable or disable additional security settings for the computer, including driver installation, media access, warning banners, and many others.

The security options template settings, as compared to the current configuration of the machine, are as follows:

Policy ▲	Database Setting	Computer Setting
Additional restrictions for anonymous connections	No access without explic...	None. Rely on default permissions
Allow server operators to schedule tasks (domain controllers only)	Not defined	Not defined
Allow system to be shut down without having to log on	Disabled	Enabled
Allowed to eject removable NTFS media	Administrators	Administrators
Amount of idle time required before disconnecting session	30 minutes	15 minutes
Audit the access of global system objects	Enabled	Disabled
Audit use of Backup and Restore privilege	Enabled	Disabled
Automatically log off users when logon time expires (local)	Enabled	Enabled
Clear virtual memory pagefile when system shuts down	Enabled	Disabled
Digitally sign client communication (always)	Disabled	Disabled
Digitally sign client communication (when possible)	Enabled	Enabled
Digitally sign server communication (always)	Disabled	Disabled
Digitally sign server communication (when possible)	Enabled	Disabled
Disable CTRL+ALT+DEL requirement for logon	Disabled	Not Available
Do not display last user name in logon screen	Enabled	Disabled
LAN Manager Authentication Level	Send NTLMv2 response ...	Send LM & NTLM responses
Message text for users attempting to log on	Not defined	
Message title for users attempting to log on	Not defined	
Number of previous logons to cache (in case domain controller is not available)	0 logons	10 logons
Prevent system maintenance of computer account password	Disabled	Disabled
Prevent users from installing printer drivers	Enabled	Disabled
Prompt user to change password before expiration	14 days	14 days
Recovery Console: Allow automatic administrative logon	Disabled	Disabled
Recovery Console: Allow floppy copy and access to all drives and all folders	Disabled	Disabled
Rename administrator account	Not defined	Administrator
Rename guest account	Not defined	Guest
Restrict CD-ROM access to locally logged-on user only	Enabled	Disabled
Restrict floppy access to locally logged-on user only	Enabled	Disabled
Secure channel: Digitally encrypt or sign secure channel data (always)	Disabled	Disabled
Secure channel: Digitally encrypt secure channel data (when possible)	Enabled	Enabled
Secure channel: Digitally sign secure channel data (when possible)	Enabled	Enabled
Secure channel: Require strong (Windows 2000 or later) session key	Disabled	Disabled
Send unencrypted password to connect to third-party SMB servers	Disabled	Disabled
Shut down system immediately if unable to log security audits	Enabled	Disabled
Smart card removal behavior	Lock Workstation	No Action
Strengthen default permissions of global system objects (e.g. Symbolic Links)	Enabled	Enabled
Unsigned driver installation behavior	Warn but allow installation	Warn but allow installation
Unsigned non-driver installation behavior	Warn but allow installation	Silently succeed

Several of these settings are notable to the security of the computer, and these are discussed below.

Do not display last user name in logon screen

Value: Enabled

This setting determines whether the username of the last user to log on to the computer will be displayed. This value can be set to Enabled or Disabled. If this feature is enabled and a malicious user gains access to this machine, the malicious user will immediately have the username of an existing account. Since it is more secure to not willingly give away account information, I have decided to stick with the template value and leave this setting enabled.

LAN Manager Authentication Level

Old value: Send NTLMv2 response only\refuse LM & NTLM
New value: **Send NTLM response only**

This setting determines the level of authentication used for network logons. Since the majority of the servers this machine will be connecting to are running versions of Windows NT that may or may not have NTLMv2 support enabled, the only way to guarantee that all logons will succeed is to only use NTLM authentication for network logons.

Message text for users attempting to log on

Old value: Not defined
New value: **Machine-specific warning message**

This setting specifies a message that will be shown to all users immediately before logging on the computer. My organization has a standard warning message that must be displayed on all computers. This warning message will be entered into our official template.

Message title for users attempting to log on

Old value: Not defined
New value: **Machine-specific warning title**

This setting specifies a title to appear in the window containing the message described in the previous section. The security policy of my organization specifies the text of this title. This text will be entered into our official template.

Prompt user to change password before expiration

Old value: 14 days
New value: **15 days**

This setting determines how many days in advance a user will be warned that his/her password is about to expire. The value of 15 days is defined by our security policy, so I have changed the template value accordingly.

Rename administrator account

Old value: Not defined
New value: **<org name>AdminAcct**

This setting specifies a new name for the built-in local Administrator account. Changing the name of this account provides additional security and will defeat any automated hacking scripts that have the Administrator logon hard-coded into

them. I wanted to keep the new name fairly standard, so I have chosen the name listed above. <org name> will be replaced by the name of my organization.

Rename guest account

Value: Not defined

This setting specifies a new name for the built-in local Guest account. Changing the name of this account might provide some additional security, but the confusion of having a changed Guest account name often outweighs any benefits that it would provide. Therefore, I have left the name of the Guest account organization alone. The best practices involving the Guest include disabling the account and setting its password to a long random string, so this is the approach I will take.

Shut down system immediately if unable to log security audits

Old value: Enabled

New value: **Disabled**

This setting determines whether to shut down the computer if the security log cannot be written to, usually because it has become too full. This value can be set to Enabled or Disabled. All relevant security logs will be automatically sent to a syslog server, and because of this, the maximum log size limits are set to fairly small values. As a result, this should never be an issue. If it ever becomes an issue, however, the developer should be able to figure out why the hard drive space is so limited. Having a personal workstation shut down for this reason seems to be an unreasonable inconvenience.

Smart card removal behavior

Old value: Lock Workstation














New value: **No Action**

This setting determines the action the computer will take when a smart card is removed. This value can be set to No Action, Lock Workstation, or Force Logoff. Since our organization does not use smart cards, the value of this setting does not really matter. Therefore, I have changed it back to the standard Windows default value.

6.3.7 Settings for Event Logs

These settings are used to define how large event logs can get, how often they are rolled over, and who can access them. Event logs are often the only way an administrator can know that a security problem has occurred on a machine, so these values should be set with care.

The event log template settings, as compared to the current configuration of the machine, are as follows:

Policy ▲	Database Setting	Computer Setting
 Maximum application log size	4194240 kilobytes	512 kilobytes
 Maximum security log size	4194240 kilobytes	512 kilobytes
 Maximum system log size	4194240 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	Manually	By days
 Retention method for security log	Manually	By days
 Retention method for system log	Manually	By days
 Shut down the computer when the security audit log is full	Enabled	Disabled

Maximum application log size

Maximum security log size

Maximum system log size

Old values: 4194240 kb

New values: **64000 kb**

These settings specify the maximum sizes that the various log files can grow to. Since all relevant logs will be sent to an external syslog server, it does not really matter how many log entries are stored on the computer itself. The value of 64000 kb was chosen for three reasons; it is large enough to allow the developer to view logs from the recent past, small enough to not take up a significant amount of hard drive space, and it is a multiple of 64k (a requirement for Windows log file sizes).

Retention method for application log

Retention method for security log

Retention method for system log

Old values: Manually

New values: **By days**

These settings specify the rollover policy for event logs. The template value of “Manually” would cause the logs to never be rolled over; any rollover would have to be done by hand. Since an external syslog server is being used, it is safe to let the logs be rolled over automatically.

Shut down the computer when the security audit log is full

Old value: Enabled
New value: **Disabled**

This setting determines whether to shut down the computer if the security audit log cannot be written to because it is full. This setting is very similar to the “Shut down system immediately if unable to log security audits” setting described earlier, and I have disabled this setting for the same reason.

6.3.8 Restricted Groups

This section allows an administrator to specify which members should be in each of the local predefined security groups. This section is machine-specific, so the template does not define anything for most of the groups. The Power Users group, however, is set in the template to have no members. This makes sense, since one of the things the template does is to strip the Power Users group of most abilities on the system. Since the Power Users group will not be used on this computer, this is acceptable. I have therefore left this section unchanged.

6.3.9 System Services

The following chart describes which system services should be explicitly enabled or disabled on the machine. The NSA template does not set any of these; the individual administrator must adjust these settings appropriately.

These settings can each be set to one of the following values:

Not defined – The current computer setting for this service will not be changed when the template is applied.

Automatic – The service will be enabled and will be started automatically when Windows starts up.

Manual – The service will be enabled, but it must be started manually.

Disabled – The service is disabled and cannot be started.

The services I have set to Disabled are generally services that are known to not need to be running for a developer to perform his/her job. A notable disabled service is the Automatic Updates service, which is disabled because this machine will not have a connection to the Internet. The two Smart Card-related services are also worth mentioning; since our organization does not use smart cards, these services are also not needed.

The services I have set to Automatic, on the other hand, are known to be necessary for a developer to perform his/her job. Norton Antivirus uses the DefWatch and Norton AntiVirus Client services; the DefWatch service periodically checks and warns the user if the virus definition files are out of date and the Norton AntiVirus Client service actively scans files for viruses. The

DHCP Client service is needed for this machine to get an IP address. The Event Log service is needed in order to generate the audit logs that my organization requires. The IIS Admin Service and the World Wide Web Publishing Service are used by IIS to serve up web pages. The Ntssyslog service is needed to automatically send all audit logs to an external syslog server. The Windows Time service is needed to synchronize the system time with an external NTP server. Finally, the Workstation service is needed in order this machine to function as a Windows 2000 workstation.

Name	Startup setting
Alerter	Not defined
Application Management	Not defined
Automatic Updates	Disabled
Background Intelligent Transfer Service	Not defined
ClipBook	Not defined
COM+ Event System	Not defined
Computer Browser	Not defined
DefWatch	Automatic
DHCP Client	Automatic
Distributed Link Tracking Client	Not defined
Distributed Transaction Coordinator	Disabled
DNS Client	Not defined
Event Log	Automatic
Fax Service	Disabled
FTP Publishing Service	Disabled
IIS Admin Service	Automatic
Indexing Service	Not defined
Internet Connection Sharing	Disabled
IPSEC Policy Agent	Not defined
Logical Disk Manager	Not defined
Logical Disk Manager Administrative Service	Not defined
Messenger	Disabled
Net Logon	Not defined
NetMeeting Remote Desktop Sharing	Not defined
Network Connections	Not defined
Network DDE	Not defined
Network DDE DSDM	Not defined
Norton AntiVirus Client	Automatic
NT LM Security Support Provider	Not defined
NTssyslog	Automatic
Performance Logs and Alerts	Not defined
Plug and Play	Not defined
Print Spooler	Not defined
Protected Storage	Not defined

QoS RSVP	Not defined
Remote Access Auto Connection Manager	Not defined
Remote Access Connection Manager	Not defined
Remote Procedure Call (RPC)	Not defined
Remote Procedure Call (RPC) Locator	Not defined
Remote Registry Service	Disabled
Removable Storage	Not defined
Routing and Remote Access	Disabled
RunAs Service	Disabled
Security Accounts Manager	Not defined
Server	Not defined
Simple Mail Transport Protocol (SMTP)	Disabled
Smart Card	Disabled
Smart Card Helper	Disabled
System Event Notification	Not defined
Task Scheduler	Disabled
TCP/IP NetBIOS Helper Service	Not defined
Telephony	Not defined
Telnet	Disabled
Uninterruptible Power Supply	Not defined
Utility Manager	Not defined
Windows Installer	Not defined
Windows Management Instrumentation	Not defined
Windows Management Instrumentation Driver Extensions	Not defined
Windows Time	Automatic
Workstation	Automatic
World Wide Web Publishing Service	Automatic

6.3.10 Registry and File System Settings

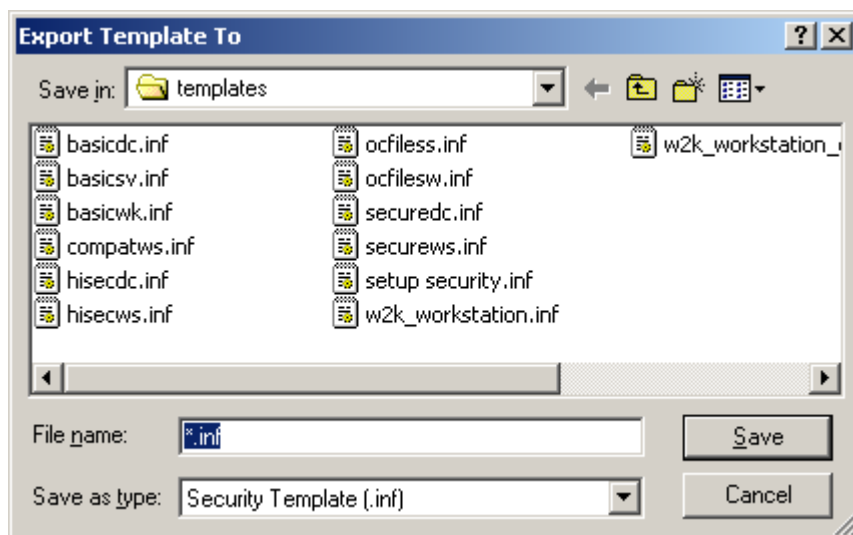
The NSA template specifies permission changes to many critical Registry keys and system files. The number of keys and files affected is extensive and to list them all would be overkill. Suffice to say, the default values set in the template are the result of countless of hours of research. Since my organization does not have any particular security policies concerning any registry keys or files, and since the values in the template seem to be reasonable, I have decided to leave the values in these sections untouched.

7 APPLYING THE TEMPLATE

Before applying the template on the computer, it is a good idea to save off the modified template as a new template file for later use on other developer

workstations. This new template can be stored in a secure location and updated as the needs of the organization change. To do this, I performed the following steps:

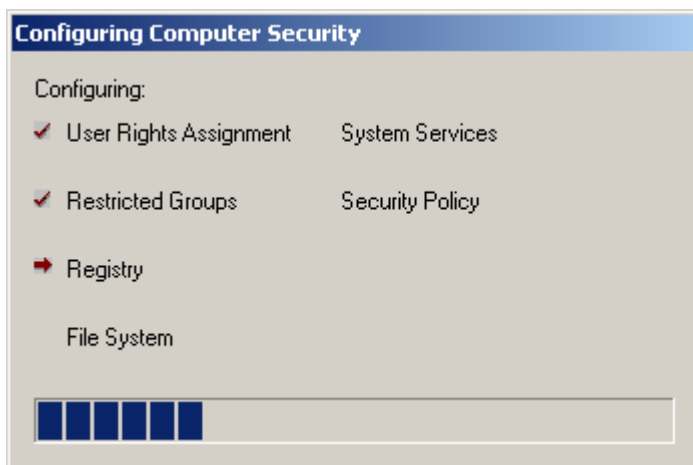
1. In the MMC console that we've been using to modify the NSA template, I right-clicked on **"Security Configuration and Analysis"** and then clicked on **"Export Template..."**



2. I typed a name for the template, i.e. *"w2k_workstation_modified.inf"* and clicked **Save**.
3. The updated template file could now be imported into additional computers using the steps described in section 6.2.

The template could now be applied to the computer. It could be applied either by using the secdit.exe command-line tool or via the MMC console. The main benefit of using secdit.exe is that individual sections of the template can be applied independently of each another. Since the entire template needs to be applied to this machine, and since the MMC console is much more user-friendly, I chose to use the MMC console to apply the template. To do this, I performed the following steps:

1. In the MMC console, I right-clicked on **"Security Configuration and Analysis"** and then clicked on **"Configure Computer Now"**.
2. I was prompted to choose a location for the error log that would be generated. I chose the default path.
3. The security settings defined in the template were now applied to the computer. There is no undo feature for this, so before I did this, I made sure that I was ready.



4. When the settings had all been applied, the plus symbol next to "Security Configuration and Analysis" disappeared.
5. Next, I re-analyzed the computer to make sure all security settings now agreed with those in the template. To do this, I right-clicked on "**Security Configuration and Analysis**" and then clicked on "**Analyze Computer Now**".
6. I selected the default path for the error log.
7. I then verified that every setting now had a green check mark next to it.
For example, this computer now had the following under Audit Policy; the rest of the sections also had all green check marks.

Policy	Database Setting	Computer Setting
<input checked="" type="checkbox"/> Audit account logon events	No auditing	No auditing
<input checked="" type="checkbox"/> Audit account management	Success, Failure	Success, Failure
<input checked="" type="checkbox"/> Audit directory service access	No auditing	No auditing
<input checked="" type="checkbox"/> Audit logon events	Success, Failure	Success, Failure
<input checked="" type="checkbox"/> Audit object access	Failure	Failure
<input checked="" type="checkbox"/> Audit policy change	Success, Failure	Success, Failure
<input checked="" type="checkbox"/> Audit privilege use	Success, Failure	Success, Failure
<input checked="" type="checkbox"/> Audit process tracking	No auditing	No auditing
<input checked="" type="checkbox"/> Audit system events	Success, Failure	Success, Failure

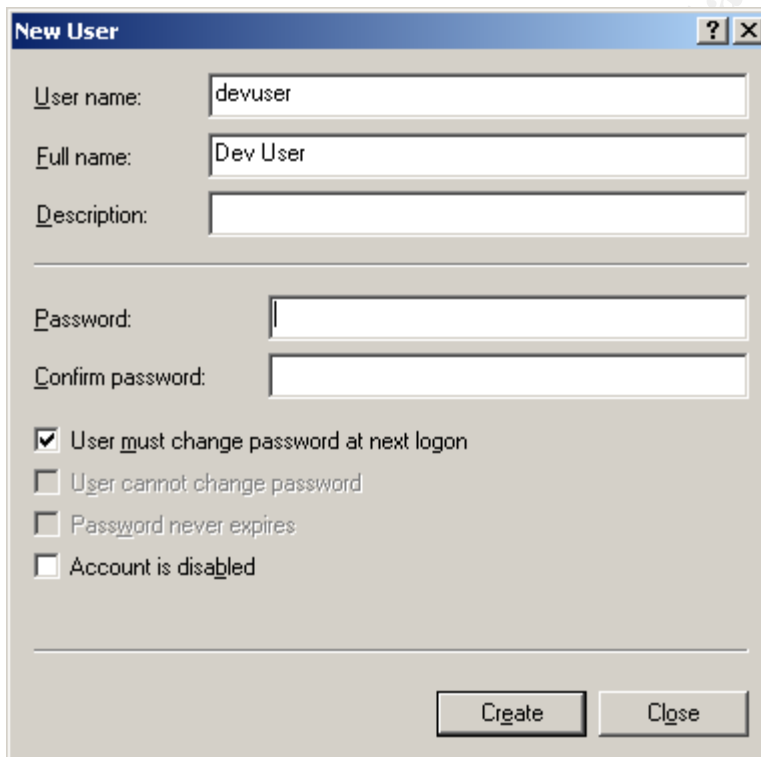
8. Finally, I rebooted the computer. This ensured that any security settings that might require Windows to restart were put into effect. Windows did not warn me that I had to reboot, but I did just to be on the safe side.

8 TESTING THE COMPUTER

The testing of the functionality of the machine was performed using three user accounts: The local administrator account, a development user account called devuser (Dev User) that is a member of the Users and Administrators groups, and a regular user account called juser (Joe User) that is only a member of the Users group. Actual IP addresses have been changed to protect the identities of the machines on our network.

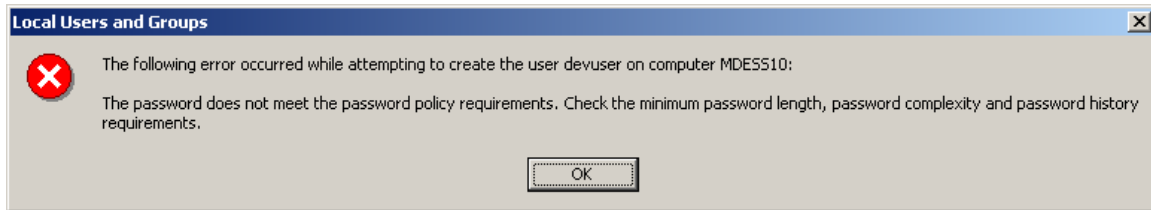
8.1 TEST 1: PASSWORD AND ACCOUNT LOCKOUT POLICIES

First, I tested the password and account lockout policies on the computer. To do this, I first logged in as the local administrator and attempted to create the devuser account.



The screenshot shows the 'New User' dialog box. The 'User name' field is filled with 'devuser' and the 'Full name' field is filled with 'Dev User'. The 'Description' field is empty. The 'Password' and 'Confirm password' fields are empty. The 'User must change password at next logon' checkbox is checked. The other three checkboxes are unchecked. The 'Create' button is highlighted.

I left the password fields blank and clicked **Create**. I then got the following error message:



I then tried the passwords “*abcdef*” (too few characters), “*abcdefg1*” (contains only two of the strong password requirements), “*Abcdefg*” (also contains only two of the strong password requirements), and then finally “*Abcdefg1*” (meets password policies). The first three passwords generated the same error, but the final password was accepted and the account was created. I then added devuser to the local Administrators group and created the user account as well with the same strong initial password.

I then tried logging on as devuser. Upon logging in with the initial password, I was then prompted to change my password to something else. I tried entering “*abcdef*” as the password, and I got the following error message:

Your password must be at least 8 characters; cannot repeat any of your previous 5 passwords; must contain capitals, numerals, or punctuation; and cannot contain your account or full name. Please type a different password. Type a password which meets these requirements in both text boxes.

I then typed a strong password, and it was accepted and I was allowed to log in.

Next, I tried to change my password again. I pressed **Ctrl-Alt-Del**, clicked on **Change Password**, and typed in a new strong password. I got the following error message:

The password on this account cannot be changed at this time.

This verified that the minimum password age policy was in effect. I then logged off, logged back on as the local Administrator, and temporarily set this policy value to 0 through Local Policy in order to test the password history policy. After doing this, I logged off, logged back on as devuser, and tried to change my password again. This time, I used the initial password that the account had been created with. I once again got the following error message:

Your password must be at least 8 characters; cannot repeat any of your previous 5 passwords; must contain capitals, numerals, or punctuation; and cannot contain your account or full name. Please type a different

password. Type a password which meets these requirements in both text boxes.

I then chose another different strong password. This time, the password change was accepted. This verified that the password history policy was in effect. I then put the minimum password age policy back to 2 days like it was before. Next, I logged off and tried logging on as devuser again. This time, I deliberately typed the password incorrectly, and I got the following error message:

The system could not log you on. Make sure your User name and domain are correct, then type your password again. Letters in passwords must be typed using the correct case. Make sure that Caps Lock is not accidentally on.

I then typed the password incorrectly three more times. On the fourth attempt, I got the following error message:

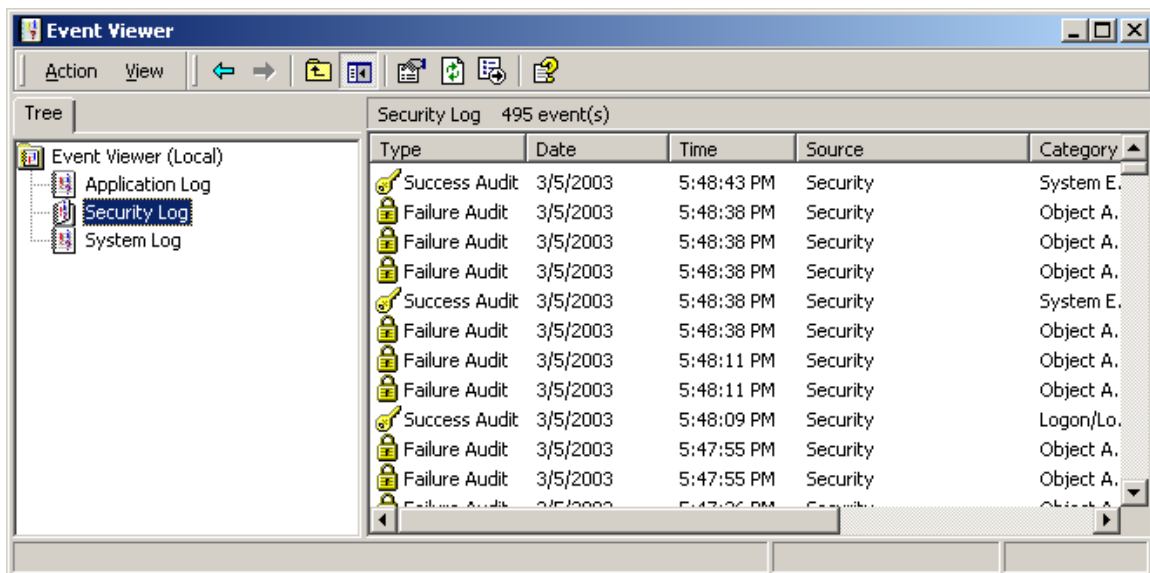
Unable to log you on because your account has been locked out, please contact your administrator.

I tried logging on with the correct password, and I got the same error message. This verified that the account lockout policy was in effect.

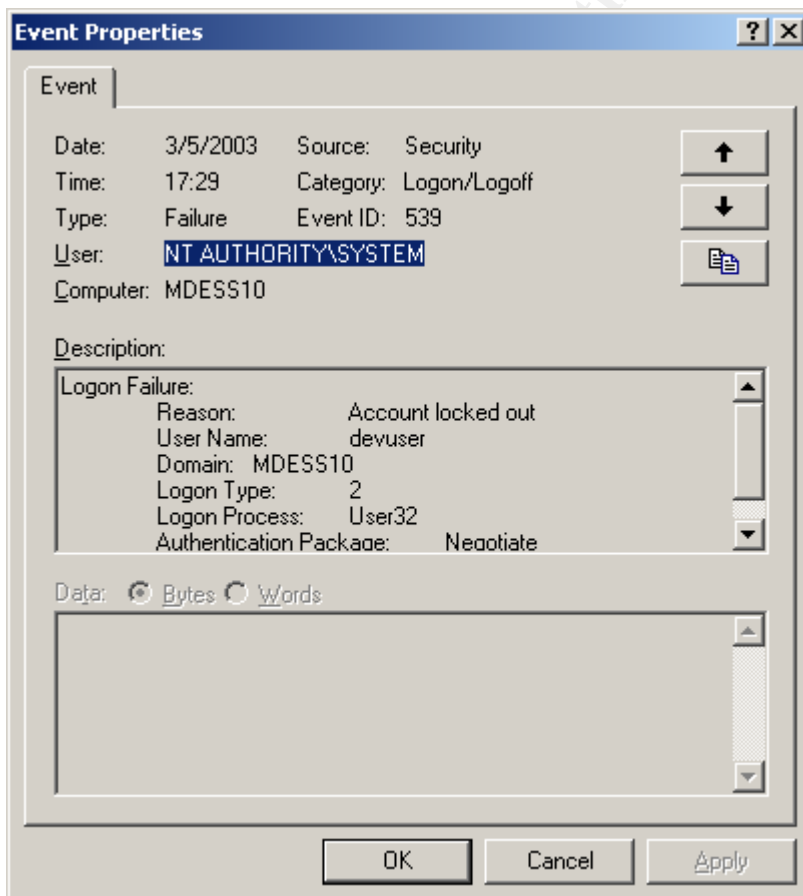
I waited a little over 30 minutes and then tried logging in again with the correct password. This time, it let me in with no errors. This verified that the account lockout duration policy was in effect.

8.2 TEST 2: EVENT LOGGING

After the previous tests, I logged off and logged back on as the local administrator. I then loaded up the Event Viewer and looked at the Security Log. I could see that there were many entries in it:

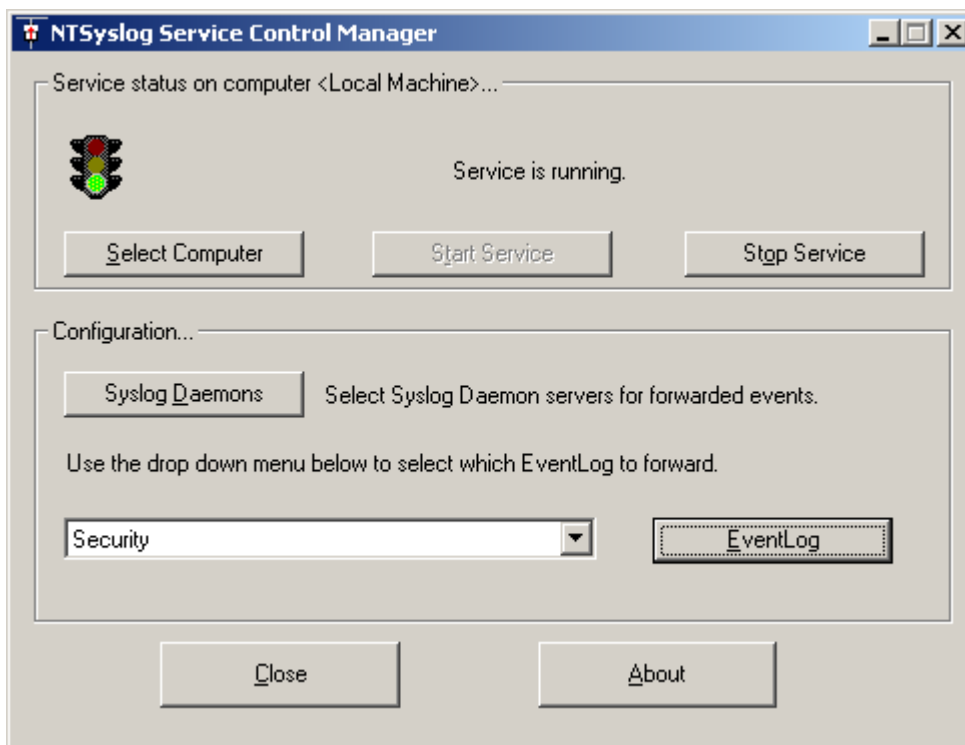


This verified that audit log entries were being stored. I then took a look at a couple of the most recent entries, and I found the entries for when devuser was locked out and for when I successfully logged in after being locked out. Here is one of those events:

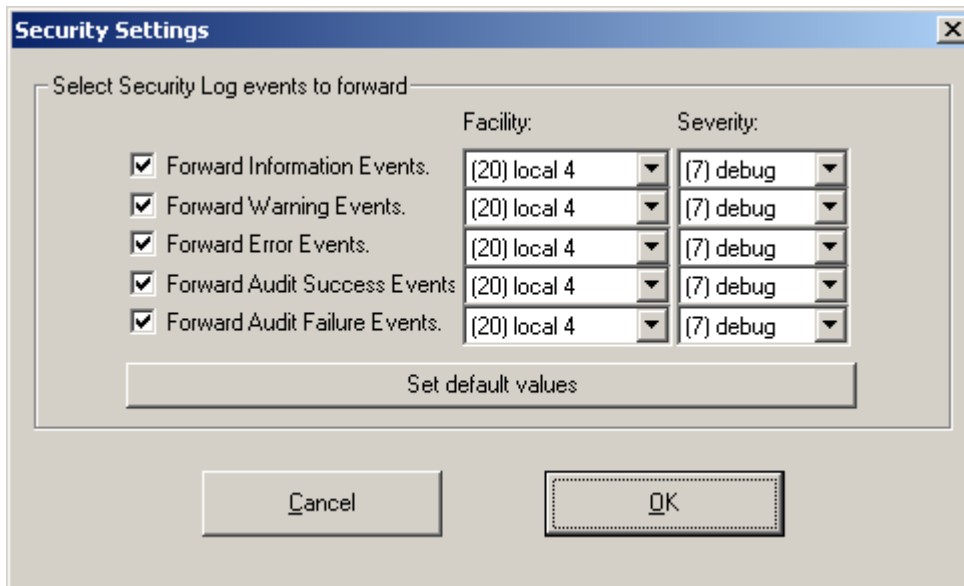


This verified that at least logon-related events are being recorded. The other various audits were assumed to be working as well.

Next, I tested the ability of the computer to send audit logs to the syslog server. First, I needed to configure ntsyslog. I did this by running the *NTSyslogCtrl.exe* utility that came with it:



First, I set the location of the syslog server by clicking on the “**Syslog Daemons**” button and filling out the entries in the window that popped up. I then configured the Security log settings by choosing “**Security**” from the drop-down select box and clicking the **EventLog** button. For each type of event, a syslog facility and severity value must be chosen. I chose a syslog facility that was not currently being used by our syslog server (local 4) and a severity value of debug:



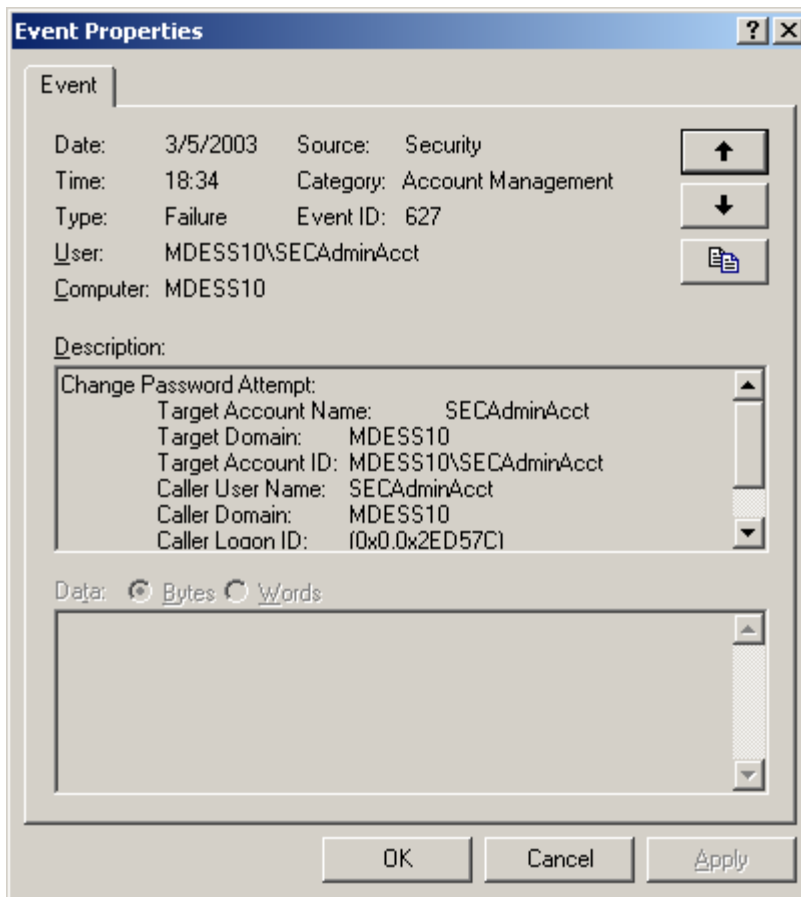
In addition to forwarding all Security events, I also enabled forwarding of Application Information Events to the same facility/severity. This will allow the syslog server to store startup/shutdown events for the NTSyslog service itself. This will make it more difficult for malicious users to cover their tracks, as described in the “important note” below. I then clicked **OK**, and then stopped and started the NTSyslog service to make the changes effective.

IMPORTANT NOTE: It is important to note that the syslog server logs are only accessible by root. This will ensure that any logs of attempted abuses of authority by malicious developers are kept in an unreachable location (assuming the developers don't have root on the Unix box). The malicious user could wipe out the logs on the Windows machine, but they would not be able to completely cover their tracks. In addition, if a malicious user tries to hide their activities by stopping the NTSyslog service on the machine, NTSyslog will log the fact that the service has been shut down. This will allow administrators monitoring the log consolidation tool to be notified that something might be amiss.

I then stopped and started syslogd to put the configuration changes into effect. Next, I started WinDump on the Windows machine with the following command in order to view any outgoing packets on their way to the Unix box:

```
windump -i 2 dst host aaa.bbb.228.26
```

I was now ready to run the test. To do this, I needed to generate an auditable security event. I did this by attempting to change the local administrator's password to something weak. This generated the following message in the event log:



I also saw the following message generated by WinDump:

```
windump: listening on \Device\Packet_{D019AD77-6EB6-4DD0-808E-80134677D617}
18:37:53.024098 mdess10.1542 > aaa.bbb.228.26.514:
udp 317
```

This verified that a packet was sent to port udp/514 of the syslog server. Finally, I made a telnet connection to the syslog server and looked in the proper log file. The following entry was at the end of the file:

```
Mar  5 18:37:52 [aaa.bbb.228.10.6.6] security[failure]
627 MDESS10\SECAdminAcct  Change Password Attempt:
Target Account Name:SECAdminAcct  Target Domain:MDESS1
0  Target Account ID: %{S-1-5-21-1078081533-507921405-
1801674531-1016}  Caller User Name:SECAdminAcct
Caller Domain:MDESS10  Caller Logon ID:(0x0,0x2ED57C)
Privileges:-
```

This verified that the syslog server successfully captured the audit log from the Windows machine.

8.3 TEST 3: TIME SYNCHRONIZATION

The next test I ran was to prove that NTP time synchronization was working. To do this, I first opened up a command prompt and ran the following command:

```
net time /setsntp:aaa.bbb.228.20
```

This command sets the address of the NTP server that the Windows Time service should connect to in order to set the system time. Next, I loaded up WinDump with the following command:

```
windump -i 2 host aaa.bbb.228.20
```

I then set the system time to 10 minutes in the past (note that NTP will refuse to update the time if the client time is more than around 20 minutes off, apparently for security reasons). I then stopped and started the Windows Time service. When the service started up, it connected to the NTP server to get the current date and time. The following messages appeared in the WinDump window:

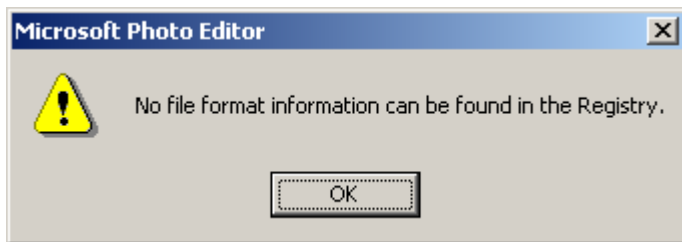
```
13:11:21.253285 mdess10.3581 > aaa.bbb.228.20.123:
[len=48] v2 client strat 0 poll 11 prec 0

13:11:21.253912 aaa.bbb.228.20.123 > mdess10.3581:
[len=48] v2 server strat 5 poll 11 prec 0 (DF)
```

This shows the computer connecting to port udp/123 to request the current time, and the NTP server responding with the current time. I then looked at the system time, and I saw that it had been corrected to match the time on the NTP server. This verifies that time synchronization is functional.

8.4 TEST 4: LOGGING ON AS A NORMAL USER

I previously verified that a developer can log onto the machine, but I have not yet tested juser, the regular user account. Do this, I simply logged off and logged back on as juser using the initial password that I had specified. I was then prompted to change my password. I picked a strong password and was allowed into the system. I tested some basic activities such as bringing up Internet Explorer, bringing up Netscape Messenger (for email), bringing up Microsoft Office components, and navigating through the file system. Most of the applications worked fine. However, when I brought up Microsoft Photo Editor, one of the components of the Microsoft Office suite, I got the following error message:



When I tried loading up and saving an image, the application crashed. Looking on the Internet for information about this error, I found Microsoft Knowledge Base article Q260151, which indicated that the error occurs when an account does not have proper access to the following registry key:

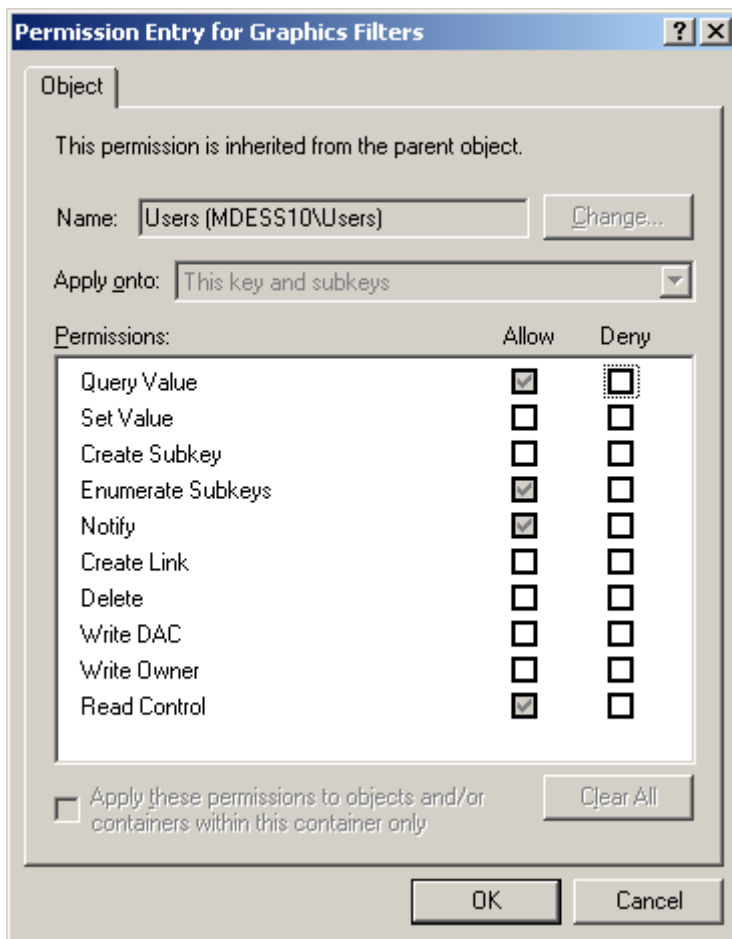
```
HKEY_LOCAL_MACHINE\Software\Microsoft\Shared  
Tools\Graphics Filters
```

According to the article, the registry key needed the following permissions for the Users group:

```
Query value  
Set value  
Create subkey  
Enumerate subkeys  
Notify  
Create Link  
Write DAC  
Write Owner  
Read control
```

However, when I looked at the security settings for this registry key, this is what I found:

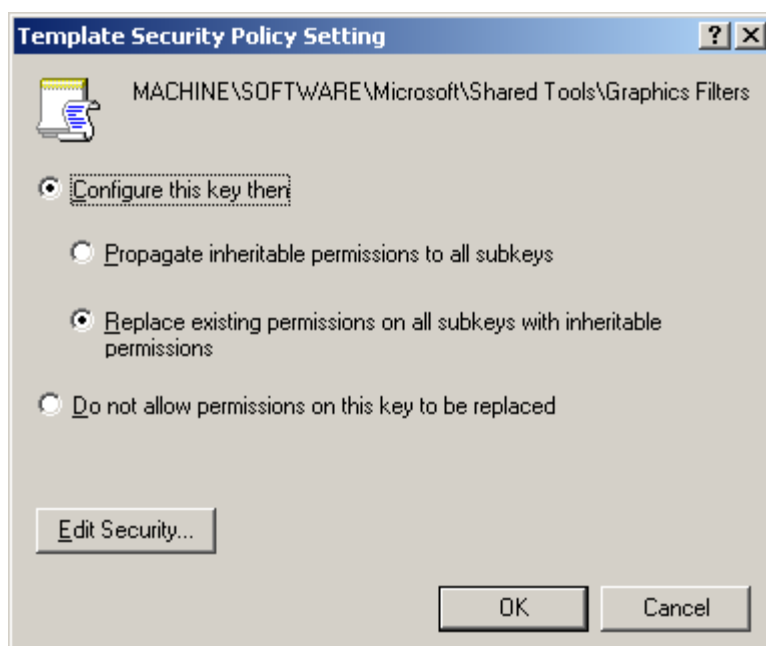
© SANS Institute 2003, Author retains full rights.



This registry key had inherited its permissions from the `HKEY_LOCAL_MACHINE\Software` key, which is defined in the NSA template to have reduced permissions for the Users group. So, I needed to fix the permissions for this registry key in order for the program to work. To implement the change, I added the necessary permissions to the registry key, then logged off and logged on as user. When I loaded up Microsoft Photo Editor again and tried saving a file, everything worked normally.

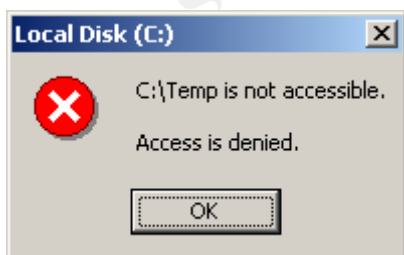
I then performed the following steps to modify the template that I had saved earlier with the Security Templates MMC snap-in to reflect this new change:

1. Expanded the `wk2_workstation_modified` template.
2. Went to the Registry section of the template.
3. In the right-hand pane, right-clicked in the area below the last registry key and clicked **"Add Key..."**
4. Browsed through the registry tree until the proper key was found.
5. Clicked **OK**.
6. Left the default security settings in place and clicked **OK**.
7. Selected **"Replace existing permissions on all subkeys with inheritable permissions"**.



8. Clicked **OK**.
9. Right-clicked on the template name in the left-hand pane and clicked **"Save"** to save the template.
10. Loaded up the modified template in a text editor.
11. I then copied the permission section from an existing registry entry with permissions similar to those of the key being changed. In this case, I copied the permissions from the *"machine\software"* line since this was the parent of the key I needed to set the permissions for.
12. Closed and reopened the Security Template MMC and went back into the security properties for the new key.
13. Added additional permissions as needed.
14. Resaved the security template.

In addition, when I tried to view the *C:\Temp* folder as juser, I got the following error message:

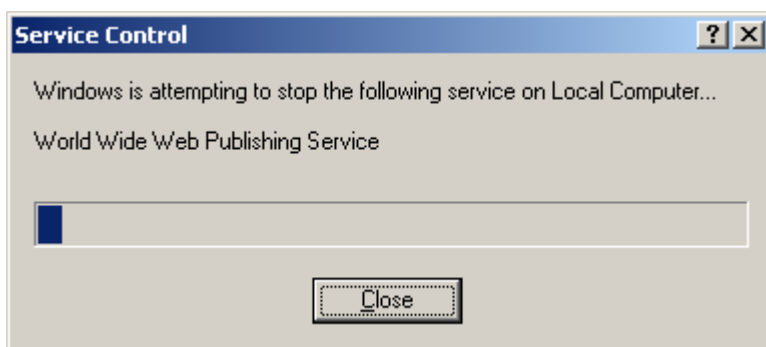


This is expected, since the template sets the permissions for the C:\Temp folder to allow the Users group to create files in this folder, but not to view the folder's contents.

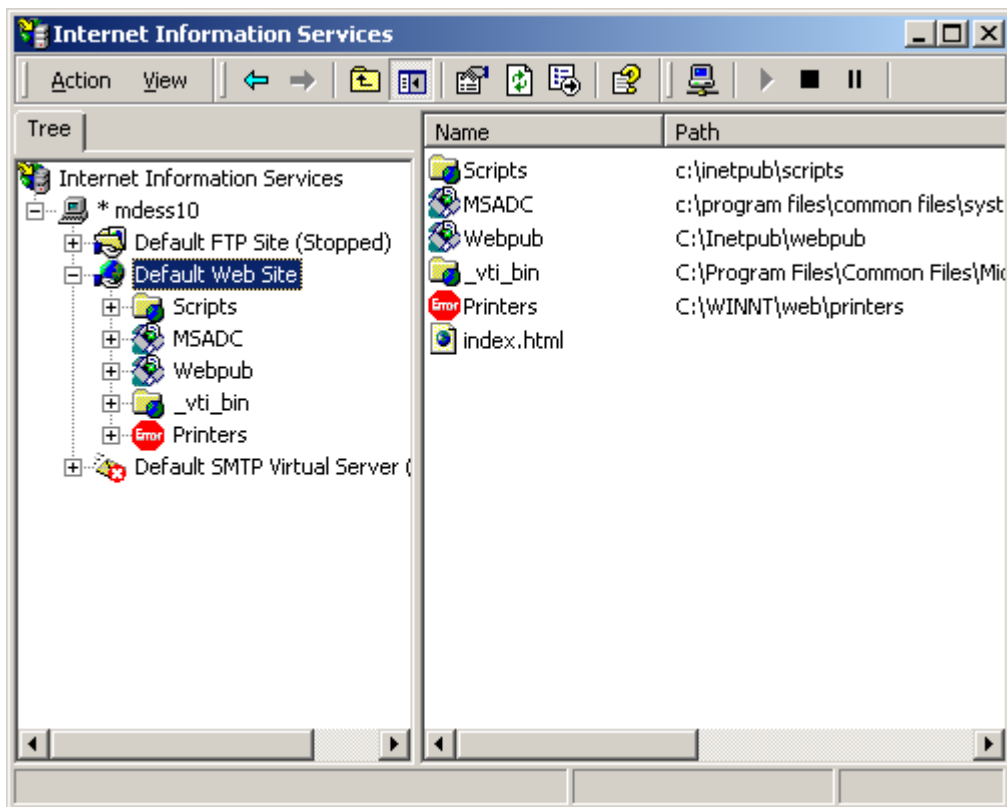
8.5 TEST 5: DEVELOPER FUNCTIONALITY

The final test I ran was to prove that developers could perform their job on this machine. I first logged off and logged back on as devuser. I then tried stopping and starting IIS and Tomcat.

To stop and start IIS, I went into the IIS MMC console tool and stopped the World Wide Web Publishing Service:



I then started it back up. This verified that devuser could cycle the IIS service. I then went into the IIS MMC console:

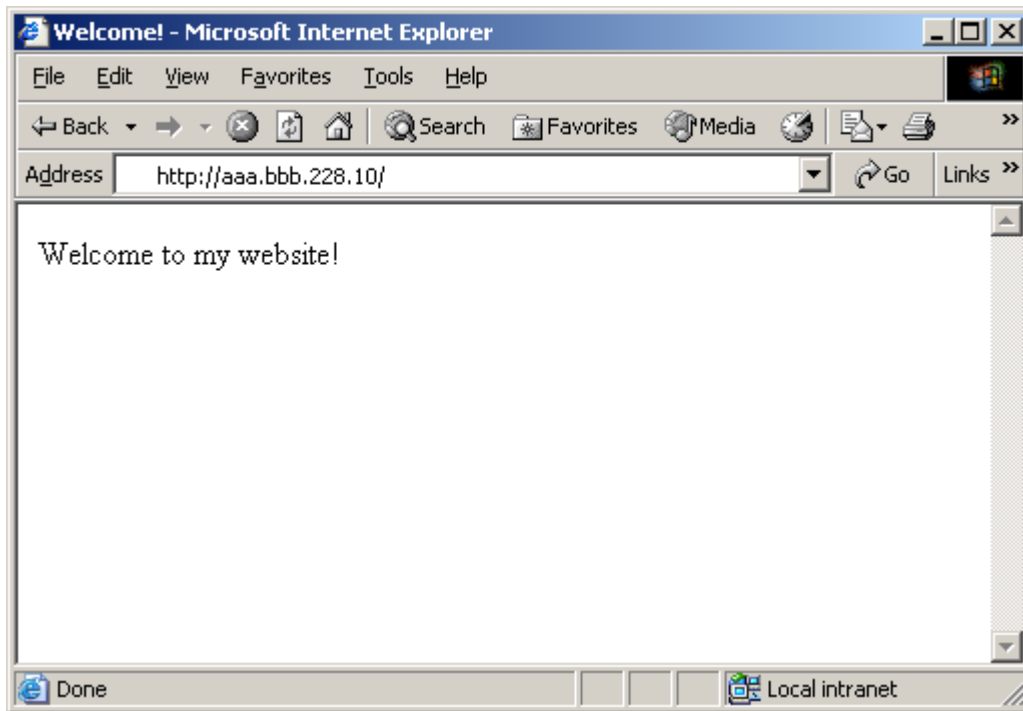


Note that the FTP and SMTP servers are not running. I then tried stopping and starting the website. This worked with no problems. Next, I went to another computer and went to the url:

`http://aaa.bbb.228.10/`

I then saw the following page come up, which is a replacement I had made earlier of the default IIS start page:

© SANS Institute 2003

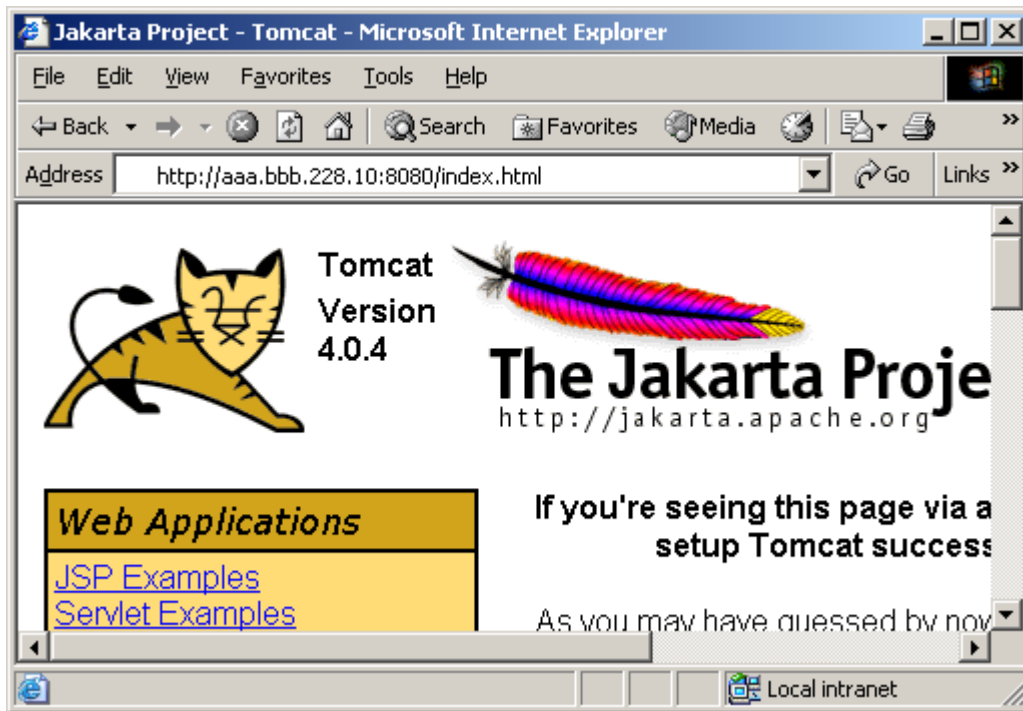


This verified that IIS could serve up web pages and that another computer could connect to IIS running on this computer.

Next, I tested Apache Tomcat. I ran the startup.bat script that came with it:

```
C:\WINNT\System32\cmd.exe
Using CATALINA_BASE:  ..
Using CATALINA_HOME:  ..
Using CATALINA_TMPDIR:  ..\temp
Using JAVA_HOME:  D:\jdk1.3.0_02\
Using CLASSPATH:  D:\jdk1.3.0_02\lib\tools.jar;..\bin\bootstrap.jar
Starting service Tomcat-Standalone
Apache Tomcat/4.0.4
Starting service Tomcat-Apache
Apache Tomcat/4.0.4
```

I then went to another computer and loaded up the following page:



This verified that Tomcat could start up successfully and bind to port 8080 to serve up web pages.

9 CONCLUSION

Overall, the NSA Windows 2000 Professional Workstation template met the needs of my organization very well. The password and account lockout policies were the only settings that significantly differed from what this workstation needed in terms of security. Some of these settings seemed a bit too strong while others seemed a bit too weak. That said, however, the default security settings are fairly secure in their own right, even if they did not exactly sync up with what our organization considers to be secure.

The only major problem I ran across with this template is the problem with Microsoft Photo Editor discussed in section 8.4. The template gave all registry entries under `HKEY_LOCAL_MACHINE\Software` the same security permissions. However, this completely broke Microsoft Photo Editor, since it assumes greater permissions on one of the keys beneath this one. Adding an additional template entry to relax the permissions on this particular subkey would allow this problem to be corrected without adversely affecting the permissions of any of the other keys defined in the template.

Numerous individuals have undoubtedly tested the security of this template very thoroughly, so additional testing would not likely uncover any gaping security holes. However, as demonstrated by the problem I found with Microsoft Photo

Editor, this template would greatly benefit from the testing of the functionality of COTS applications on machines where the template has been applied. Small tweaks to a few additional registry keys might be all that is needed to insure that most computers are completely problem-free after the application of the template, thereby saving the valuable time and lowering the stress level of IT administrators everywhere.

10 REFERENCES

1. Fossen, Jason. DNS and Group Policy. The SANS Institute, version 1.0, Aug 2002.
2. Microsoft Corporation. "Security setting descriptions". URL: <http://www.microsoft.com/technet/treeview/default.asp?url=/technet/proddocs/chnol/windowsserver2003/proddocs/server/615.asp> (27 Feb 2003)
3. Microsoft Corporation. "Passwords - Best practices". URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/proddocs/chnol/windowsserver2003/proddocs/server/windows_password_protect.asp (27 Feb 2003)
4. Microsoft Corporation. "Auditing Security Events - Best practices". URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/proddocs/chnol/windowsserver2003/proddocs/server/sag_SEconceptsImpAudBP.asp (27 Feb 2003)
5. SaberNet.net. "NTSyslog". URL: <http://ntsyslog.sourceforge.net/> (27 Feb 2003)
6. WinDump: tcpdump for Windows. URL: <http://windump.polito.it/> (27 Feb 2003)
7. National Security Agency. "Windows 2000 Security Recommendation Guides". <http://nsa2.www.conxion.com/win2k/download.htm> (27 Feb 2003)
8. Microsoft Corporation. "Web Site Operator Capabilities and Limitations". <http://support.microsoft.com/default.aspx?scid=KB;EN-US;298969> (4 Mar 2003)
9. Microsoft Corporation. "'No File Format Information Can Be Found in the Registry' Error Message When You Start Photo Editor". <http://support.microsoft.com/default.aspx?scid=KB;en-us;q260151> (6 Mar 2003)