Global Information Assurance Certification Paper

Automated Auditing in a Windows 2000 Environment

(Option 1 – Windows 2000 Security 2.1.1)

Steve Elky
Monday, August 13, 2001

Table of Contents

Introduction

Auditing is often overlooked as a vital tool to prevent system intrusion and compromise. Just as specific security measures are suspect without a security policy providing the underlying structure and control, a security policy itself it suspect without a methodology to audit that policy for compliance.

The audit policy is actually considered part of the security policy. It can be defined as the set of tests that ensure that the security rules laid out in the security policy are actually being put in place. Moreover, the audit policy should contain the consequences of not adhering to the security policy.

It is impossible to enforce a security policy without managerial controls, operational controls and technical controls. **Table 1** summarizes these controls. These controls can end up being implemented using a variety of methodologies, including written policy, training and technical tools.

Management Controls	Risk Management and Assessment
	Review of Security Controls
	Rules of Behavior
	Planning for Security in the Life Cycle
Operational Controls	Personnel security
	Physical and Environmental Protection
	Interception of data
	Contingency Planning
	Hardware and Systems Software Maintenance Controls
	Documentation
	Security Awareness and Training
	Incident Response Capability
Technical Controls	Identification
	Authentication
	Authorization
	Logical Access Controls
	Public Access Controls
	Audit Trails

Table 1 - Security Controls

The audit policy, being a subset of the security policy, follows the same guidelines. A policy can be written, but to actually use it, the controls must be implemented in some form. A control does not need to be computer-based in nature in order to fulfill this requirement.

The implementation of a control to audit social engineering may be to log suspicious activity in some type of logbook. While this is not an automated procedure, this information could be vital to detect and stop social engineering. Moreover, when checking the effectiveness of the anti-social engineering controls, management may choose to order a penetration test using social engineering. The logbook would act as the audit to ensure that the Help Desk personnel acted responsibly during the test.

Therefore, the purposes of an audit policy are to:
- Provide a record of activities to assist in possible prosecution
- Monitor the compliance with the security policy

Environment

Windows 2000 was designed to be backward compatible. In the Microsoft world that means connectivity and ease of use are emphasized and security is de-emphasized. Out of the box, the Windows 2000 operating system has a number of security holes that need to be closed up for safe operations.

Additionally, as new security features and hotfixes are added to combat newly discovered vulnerabilities, these too need to be put into place. Overall, keeping desktop and server operating systems configured securely takes a lot of time and effort as well as expertise. While most security groups can outline the proper technical controls to secure systems, it is up to the operations group to put these controls into place.

Wherever possible, it is best to enforce security policy using technical controls, such as those built into modern operating systems, such as identification (who you are), authentication (proof that this is really you) and authorization (what you are allowed to do.) However, in many large organizations, those responsible for operations make operating system security a priority.

Operations personnel tend to be driven reactively by the needs of the end users. As a result, these people may lack the time to learn about and configure the systems in a secure manner. This is compounded by the territoriality of computer personnel in general and the egos that inevitably come into play. This is made worse by the fact that most internal desktops, domain controllers and file servers are overseen by more junior personnel. The prevailing "wisdom" is that internal machines are protected by the firewall.

Differences in corporate culture and turf wars can prevent those responsible for system security from directly implementing anything stronger than a written security policy. Though upper management will enforce this security policy, there often needs to be proof that the systems are not in compliance before management will move on it. Additionally, after being repeated embarrassed by audit reports showing non-compliance, even the most egocentric certification holders will eventually toe the line.

Auditing systems by hand is extremely time consuming. It may prove impossible to get access to the systems because of territoriality. Lastly, it provides a personal reason to dislike the individual doing the auditing. In such situations, an automated auditing tool can sometimes provide the impetus for compliance. While the operations staff may not like seeing the audit failure reports coming from their managers, they won't be able to vent their spleen on the hapless security administrators!

The environment in which this auditing system is intended to be implemented in has a number of autonomous business groups with their own internal IT departments. Each of these business groups has their own Windows 2000 domain or domains. They internal groups have complete control over their domains.

The organization has implemented a single Active Directory, run by a central group. This central group runs the Active Directory and is responsible for

- Managing the Forest Root
- Managing the Active Directory Schema
- Active Directory replication topology and operations
- Management of Active Directory objects related to replication (i.e., site, site-link, subnet)
- Authorization of DHCP servers
- Creation of new domains
- Security policy for the Windows 2000 infrastructure

The central group falls under the same management structure as the group providing Internet connectivity.

Security Policy Subset

The following is the subset of the security policy that is audited by the auditing subsystem. There are several possible ways to implement this policy. Implementation of the policy is outside the scope of this discussion.

This policy was based on recommendations from the National Security Agency (NSA). Changes were made to these recommendations wherever it was deemed that the NSA guidelines would impact systems operations or where additional security was indicated.

Rather than echoing a clear and concise document, read the original paper, Guide to Securing Microsoft Windows 2000 Group Policy: Security Configuration Tool Set, available at: http://nsa2.www.conxion.com/win2k/download.htm
Changes from the NSA recommendations are discussed below.

- A single cached logon is allowed on workstations only. This allows users to work locally without a local account if no domain controller can be contacted. Note that this is not foolproof, but it should reduce the user frustration level. If an administrative user must log on to a workstation, he or she should use a user level account and use the RUNAS command to perform administrative functions. If an

administrative account is ever used at a local machine, the administrator must log out and log back in as a regular user (and then log out again) before leaving the machine in order to clear the cache of the administrative logon.

- Users are allowed to install printer drivers on workstations only. This is an accepted risk to allow ease of use.
- Workstations will not be set to crash if the security log is full. This is an accepted risk to allow ease of use.
- Workstation application, security and system logs are set to 2 GB and are overwritten if necessary after seven days. This is an accepted risk to allow ease of use.
- On domain controllers the following groups are restricted, i.e. can have no members:
  - Account Operators
  - Backup Operators
  - Guests
  - Pre-Windows 2000 Compatible Access
  - Print Operators
  - Server Operators
  - This was done because with the advent of Active Directory, delegation of authority is far more granular than the far-reaching rights of these built-in groups.
  - **Table 2** lists the services restricted. Restricted services are set to:
    - Startup of the service is Disabled
    - DACL (permissions):
      - Authenticated Users – Allow Read
      - Everyone – Deny Change Permissions, Deny Take Ownership, Deny Start, Deny Stop, Deny Pause
    - SACL (Audit):
      - Everyone – Failed Start, Failed Change Permissions and Failed Take Ownership
  - **Table 3** explains the reasoning behind the service restrictions.

Service	**Workstation**	**Server**	**Domain Controller**
DNS Server	Restricted	Restricted	
File Replication Service	Restricted	Restricted	
FTP Publishing Service	Restricted	Restricted	Restricted
IIS Admin Service	Restricted	Restricted	Restricted
Internet Connection Sharing	Restricted	Restricted	Restricted
Intersite Messaging	Restricted	Restricted	
Kerberos Key Distribution Center	Restricted	Restricted	
License Logging Service	Restricted	Restricted	
Remote Access Auto Connection Manager	Restricted	Restricted	Restricted
Remote Access Connection Manager	Restricted	Restricted	Restricted
Routing and Remote Access	Restricted	Restricted	Restricted
Simple Mail Transport Protocol (SMTP)	Restricted	Restricted	Restricted
TCP/IP Print Server	Restricted		
Telnet	Restricted	Restricted	Restricted
World Wide Web Publishing Service	Restricted	Restricted	Restricted

Table 2 – Restricted Services

Service	Reason for Restriction
DNS Server	DNS zones are AD-integrated in this environment, thus this service only can run properly on domain controllers.
File Replication Service	This service is used for the SYSVOL, which is found only on domain controllers.
FTP Publishing Service	Uses unencrypted passwords.
IIS Admin Service	The IIS services have a long history of security problems.
Internet Connection Sharing	Could create an inadvertent backdoor into the network by installing a modem to dial out to an ISP.
Intersite Messaging	Only used on domain controllers.
Kerberos Key Distribution Center	Only used on domain controllers.
License Logging Service	Licenses will be controlled in this environment from the domain controllers.
Remote Access Auto Connection Manager	Could create an inadvertent backdoor into the network by installing a modem to dial out to an ISP.
Remote Access Connection Manager	Could create an inadvertent backdoor into the network by installing a modem to dial out to an ISP.
Routing and Remote Access	Could create an inadvertent backdoor into the network by installing a modem to dial out to an ISP.
Simple Mail Transport Protocol (SMTP)	If incorrectly configured, could be used to bounce spam.
TCP/IP Print Server	Workstations are not allowed to act as print servers.
Telnet	Uses unencrypted passwords.
World Wide Web Publishing Service	The IIS services have a long history of security problems.

Table 3 – Rationale for Service Restrictions

In order to allow the auditing system to operate properly and securely, several additional requirements were added to the security policy.

- The Windows 2000 Task Scheduler will be configured to startup automatically as the Local System account.
- Each machine must set the %MACHINEROLE% environmental variable to indicate whether the machine is a:
  - workstation
  - server
  - dc (i.e., domain controller)
- Each machine must have file sharing enabled, though users must be blocked from being able to share files or change the ACLs on file shares. These file shares will only be used for administrative purposes.

Auditing System Architecture

The auditing system is made up of four components. The first component, the Audit Setup subsystem, performs the initial setup of the Local Audit subsystem on each machine. The Audit Setup subsystem runs as a Site Group Policy startup script. The second component, the Local Audit subsystem, runs on the local machine and generates the raw audit data. The third component, the Audit Collection subsystem, runs on the audit server and collects the raw audit data from all of the audited machines. The final component, the Audit Report subsystem, runs on the audit server and generates Summary Reports and machine-specific Exception Reports. **Figure 1** shows the auditing system architecture.

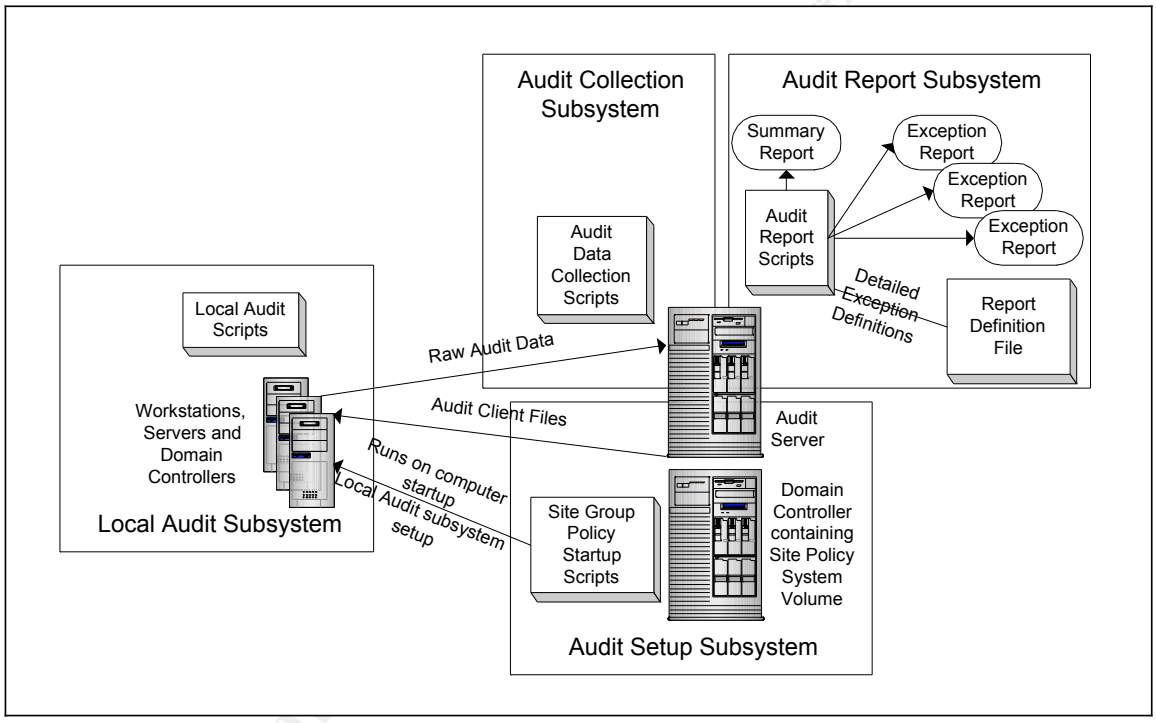

Figure 1 - Auditing System Architecture

Tools Utilized in the Audit System

Four tools were used to create this system. Batch files, PERL and the Microsoft Security Configuration and Analysis command line tool (**`SECEDIT.EXE`**) and the **`SUBINACL.EXE`** tool. The PERL engine used was ActivePerl, PERL version 5.005_03. ActivePerl and the **`SUBINACL.EXE`** tool are included on the Windows 2000 Resource Kit CD. No complex PERL modules were used, so any PERL engine should work.

Environmentally specific values such as file paths and environmental variables were passed to the PERL scripts as command line variables. Similarly, the PERL scripts communicated their results to the batch files by creating text files, for which the batch files could check existence. The PERL scripts also wrote batch files, which could be in turn executed by the batch file that executed the PERL script. No operating system specific commands were used with the PERL scripts.

The Microsoft Security Configuration and Analysis command line tool, **`SECEDIT.EXE`**, performs the bulk of the data collection. This tool is part of the Windows 2000 default installation. It is present on all workstations and servers.

Using the SECEDIT Command Line Tool

The SECEDIT command line tool is a powerful security analysis and configuration tool that is part of the default installation of Windows 2000. The SECEDIT tool is the command line version of the Microsoft Security Configuration and Analysis MMC snap-in. The Audit System only makes use of the analysis functionality of the tool. The SECEDIT tool has several requirements/limitations:

- The tool can not be run against a remote machine.
- The tool requires an analysis (or configuration) template
- The tool requires a database with the analysis (or configuration) template loaded into the database.

These requirements present some difficulties in using the tool in a distributed environment. The Audit System solves these problems as described in the following sections.

In order to create the template and the database, the Security Configuration and Analysis MMC snap-in is used. The base templates were obtained from NSA, who in turn based theirs on the templates Microsoft ships with Windows 2000. The NSA templates are available at: http://nsa2.www.conxion.com/win2k/download.htm

Once the proper template and database files are created, it is easy to use the SECEDIT tool. The format of the SECEDIT tool in analysis mode is:

```
SECEDIT /ANALYZE /DB databse.sdb  /CFG template.inf" /LOG outputfile.txt /VERBOSE
```

Where **`database.sdb`** is the database file, **`template.inf`** is the template file and outputfile.txt is the output of the tool. Keep in mind that this tool must be executed on the local workstation.

Data Formats

This system deals with files in several different formats. The preferred format is unformatted ANSI text. The **`SECEDIT.EXE`** tool outputs text data in Unicode. The Local

Audit Script converts the input data from Unicode into ANSI text before it is used elsewhere. The report files are HTML-encoded ANSI text. This was done to provide a more effective report.

The Report Definition file, which provides user-friendly messages for the Exception Reports, is an ANSI text file with two fields. The first field is the raw exception report that will match the raw input data. The first field must only contain lowercase. The second field contains the user friendly report data formatted in HTML. The two fields are separated using the pattern "tab" "pipe" "tab". The "pipe" character, |, is also known as the "bar" character. This was done for readability in the Report Definition file. **Table 4** summarizes the data files and their formats.

Data files are named on the local system using the NetBIOS name, which is represented by the %COMPUTERNAME% environmental variable. In order to prevent name collisions, the audit data files, machine-tracking files and audit exception reports on the Audit Server are named using Fully Qualified Domain Names (FQDN), i.e., DNS names.

Data File Name	Purpose	Format	Subsystem
`y-with-crlf.txt`	Provides affirmative input to commands that prompt for Y/N with a carriage return/linefeed	ANSI text	Audit Setup
`results.txt`	Output from the AT command showing the current configuration of the Windows Task Scheduler	ANSI text	Audit Setup
`found.txt`	A text file used to indicate that the Audit Collection subsystem is correctly configured in the Windows Task Scheduler	ANSI text	Audit Setup
`fqdn.raw`	The output from the `IPCONFIG /ALL` command. Used to determine the FQDN of the system.	ANSI text	Local Audit
`%COMPUTERNAME%.raw`	Output from the `SECEDIT` utility	Unicode text	Local Audit
`yyyy-mm-dd-`*`FQDN`*`.raw`	Audit raw data file	ANSI text	Local Audit
`%COMPUTERNAME%.log`	Machine-tracking update file. Indicates that the machine is configured correctly and when the local audit was last run	ANSI text	Local Audit
`audit_dc.inf`	Audit template used by `SECEDIT` to audit domain controllers	ANSI text	Local Audit
`audit_server.inf`	Audit template used by `SECEDIT` to audit servers	ANSI text	Local Audit
`audit_workstation.inf`	Audit template used by `SECEDIT` to audit workstations	ANSI text	Local Audit
`audit_.inf`	Copy of domain controller audit template used by `SECEDIT` to audit machines without %MACHINEROLE% set	ANSI text	Local Audit
`COLLECT-ERRORS.LOG`	Collection process error report	ANSI	Audit Collection
`FQDN``.log`	Machine-tracking files on the audit server	ANSI	Audit Collection

`reportdefinintions.txt`	Report Definition file	Two ANSI text fields separated by “tab” “pipe” “tab”	Audit Report
`Summary Report-yyyy-mm-dd.html`	Summary Report	ANSI HTML	Audit Report
`yyyy-mm-dd-`*`FQDN`*`-Exception Report.html`	Exception Report	ANSI HTML	Audit Report

Table 4 - Data Files

Bandwidth Considerations

Since this system is initially rolled out using a script that is run per Active Directory Site, the scripts could be easily tailored to use a local server to store the Audit Setup and Audit Collection scripts and support files. However, this should not be necessary in most circumstances. The system has been designed to minimize the impact of file copying across slow network links.

The auditing system only updates client scripts and support files if necessary. Therefore, it is unlikely that there will be a major impact on end users except for the very first time the machine starts up after the auditing system is rolled out or updated. Additionally, the files most likely to change, the audit template files, are checked and updated during the Audit Collection process. This process occurs in the background at 5:00 AM and should not impact end users.

The Audit Setup Subsystem

A site policy startup script (**`site.cmd`**) initiates the Audit Setup Subsystem upon machine start up. This script is the only thing being controlled by the Site Policy. Site Policies are controlled by the central group and are not used for anything else, so as not to violate the autonomy of the business units.

The site policy startup script first creates the directory to house the Audit Collection subsystem. Next it checks for the existence of the files necessary for the Audit Setup subsystem. If these files are not present, the script will copy them from the Audit Server. At this point, the script will set the attributes and/or Access Control Lists (ACLs) on the directory and all the related files to keep the end users from interfering with the Audit Collection process. The directories created will have the Read Only, System and Hidden attributes set. All the files in the directories will have the rights shown in **Table 5.**

On the Audit Server, the Audit Collection subsystem is run under an account with membership in the AuditCollect group in the domain housing the Audit Server. This allows the Audit Collection subsystem to centrally collect the raw audit files in a controlled manner.

Access Control Entry	Reason
SYSTEM: Full control	The local system needs to generate the audit data
Audit_Server_Domain\AuditCollect: Full Control	The Audit Collection subsystem uses these right to collect the data
Everyone: Deny all	Do not allow users to access the scripts, engines or data.

Table 5 - Access Control Entries on Local Audit Files

Since the Local System account will be running both the Group Policy engine and the Windows 2000 Task Scheduler, these ACLs should not interfere. The Windows 2000 Task Scheduler service always runs as the Local System account. The new Scheduled Tasks GUI interface allows certain tasks to run under other accounts, but this cannot be configured via the AT command. The Windows 2000 Task Scheduler service on the local machine runs the Local Audit subsystem as the Local System account.

The PERL engine is not available to the end users, only the Local System and it is not on the %PATH%. This was purposely done to prevent any viruses or worms that may infect the machine under the user context from using the scripting engine to cause further damage. In a production system, the executable, **`PERL.EXE`**, would also be renamed, but it was left with its default name to make the scripts more understandable. This will not interfere with normal PERL use on the machine. If the machine were being used to run PERL scripts an instance of the PERL scripting engine would have already been elsewhere on the machine and thus will still be available.

Next, the script cleans up any temporary files from any previous iteration of the Audit Setup subsystem. After this task has completed, the Audit Setup subsystem checks to see if the Windows Task Scheduler is configured to run the Local Audit subsystem every night at 4:00 AM. If it is not, the Audit Setup subsystem configures it. The command to run the Local Audit subsystem is:

```
AT 4:00 /EVERY:M,T,W,Th,F,Sa,Su "C:\AUDIT\HIDDEN\LOCALAUDIT.CMD"
```

Since file servers and domain controllers are only restarted occasionally, the Audit Collection subsystem is executed by the Windows 2000 Task Scheduler. Therefore, after ensuring that the proper support files and script files are in place, the Audit Setup subsystem checks the configuration of the Windows 2000 Task Scheduler. If the Windows 2000 Task Scheduler is not correctly the Audit Setup subsystem configures it correctly.

The Audit Setup subsystem also creates a hidden share, **`HIDDEN$ = C:\HIDDEN\AUDIT,`** to allow the Audit Collection subsystem to collect the data. Having a central system collect the data in this manner requires that file sharing be enabled on the workstations. The use of file shares on workstations solely for administrative use is defined in the

security policy. Once the share has been created, a resource kit utility, **`subinacl.exe`**, is used to set the ACL on the share so that only the AuditCollect group has any access.

Next, the Audit Collection subsystem creates an update for the machine-tracking file. After completing this task, the Audit Collection subsystem uses the **`checktracking.pl`** script to check for a machine-tracking file for the machine on the Audit Server. If there is not one present, it is created. On the audit server, the machine tracking files have a format of ***`FQDN`*`.log`**, where *FQDN* is the Fully Qualified Domain Name of the computer. This ensures there will be no name collisions within the Active Directory Forest, since each machine in an Active Directory Forest shares a single DNS namespace and thus must have a unique FQDN. **Table 6** summarizes the files used by the Audit Setup subsystem.

File Name	Purpose	Language/ Format
`site.cmd`	Check for proper audit files and configure the Windows 2000 Task Scheduler to run the audit collection script.	Batch language
`y-with-crlf.txt`	Provides affirmative input to commands that prompt for Y/N with a carriage return/linefeed	ANSI text
`attest.pl`	Check the Windows 2000 Task Scheduler configuration	PERL
`recordfqdn.pl`	Generates the DNS name of the machine	PERL
`checktracking.pl`	Creates the machine-tracking file on the Audit server if none exists	PERL
`perl.exe`	PERL scripting engine	Executable
`perlcrt.dll`	PERL scripting engine	Executable
`perlcore.dll`	PERL scripting engine	Executable
`subinacl.exe`	Resource kit utility to set ACLs on shares	Executable
`results.txt`	Output from the AT command showing the current configuration of the Windows Task Scheduler	ANSI text
`found.txt`	A text file used to indicate that the Audit Collection subsystem is correctly configured in the Windows Task Scheduler	ANSI text

Table 6 -Files Used by the Audit Setup Subsystem

The Local Audit Subsystem

Each night at 4:00 AM, the Windows Task Scheduler initiates the Local Audit subsystem by executing **`localaudit.cmd`**. This script first cleans up any temporary files from the previous running of the script.

Next the script creates an update for the machine tracking-file. The full machine-tracking file is stored on the audit server and is used to detect if machines are not running the audit scripts regularly and to ensure that the machines have been set up so that the proper audit template is used in auditing. The script writes the NetBIOS name of the computer (%COMPUTERNAME%), the date, the time and the machine role (%MACHINEROLE%) to the machine-tracking update file on the local machine.

As mandated by the security policy, each machine must contain the %MACHINEROLE% environment variable set to declare the role of the system in the Active Directory. **Table 7** shows the possible settings. The %MACHINEROLE% variable dictates the audit template and the corresponding Security Configuration and Analysis database the machine will be audited against. If the variable is not set, the system is audited against the most restrictive audit policy (the domain controller audit policy.) Additionally, the machine-tracking file will indicate that the environmental variable is not set and the date. This data is not case sensitive.

%MACHINEROLE% Value	Usage
workstation	Windows 2000 Professional workstations
server	Windows 2000 member servers
dc	Windows 2000 domain controllers

Table 7 - %MACHINEROLE% Values

After the machine-tracking file update has been created, the Fully Qualified Domain Name (FQDN) of the computer will be determined and the computer name field in the machine-tracking update file will be replaced with the computer's FQDN. The FQDN is determined by using the output from the **`IPCONFIG /ALL`** command. The **`IPCONFIG /ALL`** command is run and redirected into a text file. The **`recordfqdn.pl`** script reads this text file and uses this information to revise the machine-tracking update file.

At this point the Microsoft Security Configuration and Analysis tool (SECEDIT) will be run in command line mode, writing verbose output to a specified log file. The audit template corresponding to the %MACHINEROLE% will be used. If the %MACHINEROLE% variable is not set, the machine will be audited against the strictest audit template, in this case, the domain controller audit template.

Once the Microsoft Security Configuration and Analysis tool has completed, a PERL script will run. This script (**`localaudit.pl`**) will convert the data from Unicode to ANSI and add the FQDN and the %SYSTEMROOT% and %SYSTEMDRIVE% variables to the beginning of the data file. The script will then write out the pre-processed audit data to a filename with the format ***`yyyy-mm-dd-FQDN`*****`.raw`**, where:

- yyyy is the year.

- mm is the month
- dd is the day of the month
- FQDN is the fully qualified domain name of the machine (i.e., ws.realm.org)

This format will be preserved when the data is later copied to the Audit Server. The format prevents name collisions between computers in the enterprise, since the FQDN is a unique computer identifier within an Active Directory forest. The FQDN is also well known and understood by humans. The format also allows audit files from machines to sort together by date if being viewed in Windows Explorer. This can also prove useful. **Table 8** summarizes the files used by the Local Audit subsystem.

File Name	Purpose	Language/ Format
`localaudit.cmd`	Create machine-tracking file Collects local audit data through the use of `SECEDIT.EXE` Passes environmental variables to `localaudit.pl` Copies raw data file to the audit server	Batch language
`localaudit.pl`	Converts data from Unicode to ANSI Adds environmental variables to data file Creates correct file name	PERL
`recordfqdn.pl`	Determines the FQDN of the machine from the output of the `IPCONFIG /ALL` command and revises the machine-tracking update file with this information.	PERL
`perl.exe`	PERL scripting engine	Executable
`perlcrt.dll`	PERL scripting engine	Executable
`perlcore.dll`	PERL scripting engine	Executable
`secedit.exe`	Compares the local system against an audit template and creates a report file	Executable
`%COMPUTERNAME%.raw`	Audit data output from `secedit.exe`	Unicode text
`fqdn.raw`	Output from the `IPCONFIG /ALL` command used to determine the FQDN of the machine	ANSI Text
`yyyy-mm-dd-`*`FQDN`*`.raw`	Audit data output from `localaudit.pl`	ANSI text
`%COMPUTERNAME%.log`	Machine-tracking update file. Indicates that the machine is configured correctly and when the audit script was last run	ANSI text
`audit_%MACHINE ROLE%.inf`	Audit template used by `secedit.exe` to audit machines	ANSI text

Table 8 – Files Used by the Local Audit Subsystem

The Audit Collection Subsystem

Each day at 5:00 AM, the Windows Task Scheduler runs the Audit Collection subsystem on the Audit Server. The **collect.cmd** script controls the Audit Collection subsystem. The Windows 2000 Task Scheduler runs the Audit Collection subsystem as a user that is a member of the AuditCollect global security group in the domain where the Audit Server is housed. If this is not the case, the Audit Collection subsystem will not be able to collect any data files.

The first task of the **collect.cmd** script is to clean up the dynamically created data collection script from the previous iteration of the Audit Collect process. After this has been done, the **collect.pl** script is executed.

The **collect.pl** script builds a batch file, **getdata.cmd**, which will actually collect all the audit data. This methodology was used to avoid use the system() function in PERL and to keep operating system command out of the PERL scripts wherever possible. The script steps through all the files in the machine-tracking file directory. There is one file in the machine-tracking directory for each machine that has been set up by the Audit Setup subsystem. These files are used to build the machine list, which in turn determines which systems for which the **getdata.cmd** script will collect data and update the machine-tracking files.

The **getdata.cmd** script is really a series of commands repeated for each machine with a corresponding entry in the machine-tracking directory. The first commands check the files used by the Local Audit subsystem and if they are not present, replace them and set the proper ACLs on them.

Next, as previously stated, the **getdata.cmd** script collects the local audit data from each machine with a corresponding entry in the machine-tracking directory. After this is done, a check is made to ensure that the audit file was copied and the machine-tracking file for that machine is updated from the local machine-tracking update file. Both the local audit data and the local machine-tracking update file are created by the Local Audit subsystem.

After the **getdata.cmd** script is created, the **collect.pl** script returns control to **collect.cmd**. This script then executes the **getdata.cmd** script. Any errors not internally redirected are placed into a single log file, **COLLECT-ERRORS.LOG** in the **AUDITLOGS** share of the audit server. **Table 9** summarizes the files used by the Audit Collection subsystem.

File Name	Purpose	Language/ Format
collect.cmd	Clean up previous iterations Run the script that builds the data collection script Run the data collection script	Batch language
collect.pl	Build the data collection script	PERL
getdata.cmd	The data collection script	Batch language
perl.exe	PERL scripting engine	Executable
perlcrt.dll	PERL scripting engine	Executable

`Perlcore.dll`	PERL scripting engine	Executable
`COLLECT-ERRORS.LOG`	Collection process error report	ANSI text
`yyyy-mm-dd-`*`FQDN`*`.raw`	Audit data output from `localaudit.pl`	ANSI text
`%COMPUTERNAME%.log`	Machine-tracking update file. Indicates that the machine is configured correctly and when the audit script was last run	ANSI text
`yyyy-mm-dd-`*`FQDN`*`.raw`	Machine-specific audit file on the audit server	ANSI text
`FQDN``.log`	Machine-tracking files on the audit server	ANSI text

Table 9 - Files Used by the Audit Collection Subsystem

The Audit Report Subsystem

The Audit Report subsystem runs on the audit server, once per day. It is controlled by the Windows 2000 Task Scheduler and runs at 8:00 AM daily. The Audit Report subsystem uses three sources of data. The first source is the machine-tracking files. This information is used to:

- Discover which machines should have run an audit
- Discover if these machines did and if not, when the last valid audit was completed
- Discover if these machines had the %MACHINEROLE% variable set to a valid value at the time the Local Audit process ran on them
- Determine which machines to examine for detailed exceptions to the audit policy

The second source of data is the actual machine-specific audit data. This data is generated by the Local Audit subsystem and moved to the Audit Server by the Audit Collection subsystem. This information is used to determine if a particular machine has any exceptions to the audit policy

The final source of information is the Report Definition file. This file is maintained on the Audit Server. It contains all the audit exceptions that can be reported by the SECEDIT utility and friendly error messages for these audit exceptions. This file is developed from the audit templates (**`audit_*.inf`**) generated by the Microsoft Security Configuration and Analysis tool.

Each line in an audit template file corresponds to the unique part of the error message in the verbose output from the SECEDIT tool. For example, the audit template file contains the following line:

```
MaximumPasswordAge = 90
```

The output from the SECEDIT utility contains the following corresponding line:

```
Mismatch        - MaximumPasswordAge.
```

The Report Definitions file contains the common error definition and a friendly error message. Here is the corresponding line from the Report Definitions file:

```
maximumpasswordage    |    <li><b>Maximum password age</b> is incorrectly set<p>
```

The first field in the "unfriendly" or raw error message. The second field is the friendly error message. Note that the error message looks better when viewed in a web browser, as the audit reports are design to be viewed. The two fields are separated by a tab-pip-tab sequence for readability when working on the Report Definition file.

The Audit Report subsystem produces at least one output, the Summary Report. There may also be one or more Exception Reports. Exception Reports are per machine and are only generated if:

- The local audit ran successfully that day and the data was successfully moved to the audit server
- The machine was not in compliance with all the items in the audit template

The Summary Report shows:

- For machines that have successfully completed the audit process:
  - Whether or not the %MACHINEROLE% was configured properly at the time of the Local Audit
  - Whether or not the machine had any exceptions to the audit policy. If this is the case, a hyperlink to the Exception Report for that machine will be provided.
- For machines that have been set up by the Site Group Policy setup script:
  - The last time the audit was successfully completed
  - Whether or not the %MACHINEROLE% was configured properly at the time of that audit

The Summary Report also shows how long it took to execute the Summary Report and any associated Exception Reports. **Figure 2** shows an example of a Summary Report.

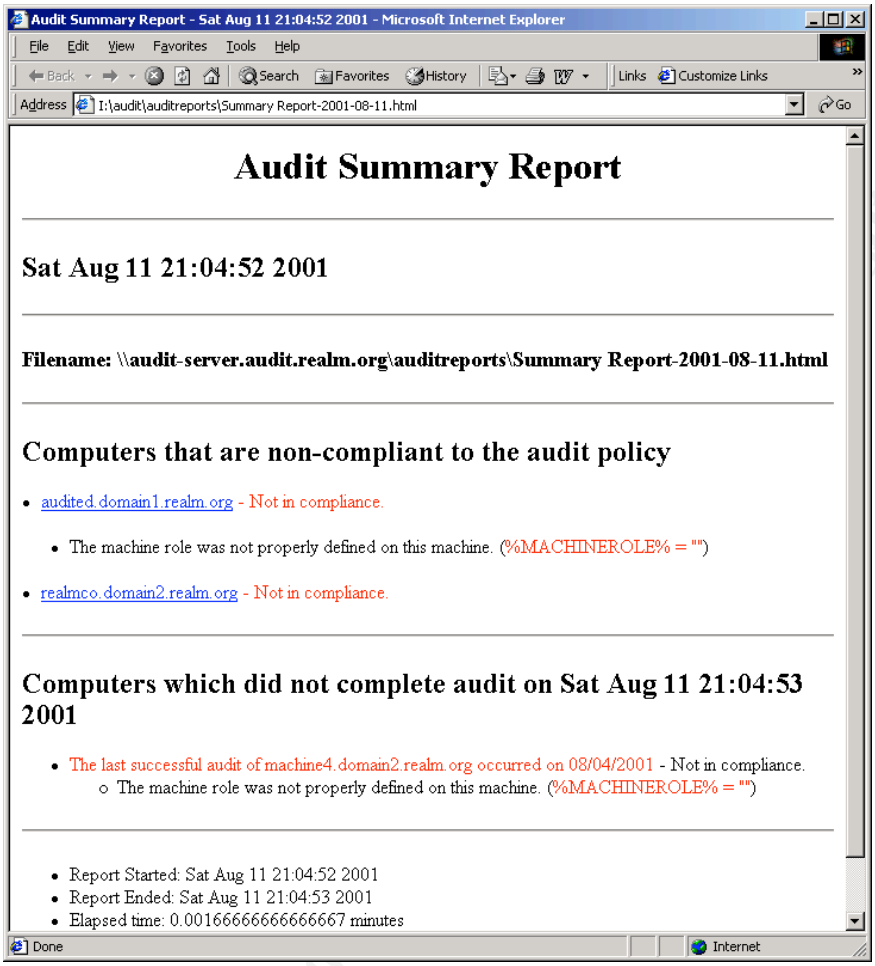

Figure 2 - Summary Report

An Exception Report contains friendly error messages for each exception generated by the SECEDIT tool. These audit exceptions are grouped into subsections that correspond to the subsections in the Microsoft Security Configuration and Analysis Tool:

- Account Policies
- Audit Policy
- User Rights Assignment
- Security Options (Registry Settings)
- Event Log Settings
- Restricted Groups
- System Services
- Registry Permissions
- File System Permissions

Figure 3 and **Figure 4** show an Exception Report. **Table 10** summarizes the files used in the Audit Report Subsystem.

File Name	Purpose	Language/ Format

auditreport.cmd	Clean up previous iterations Run the script that builds the data collection script Run the data collection script	Batch language
auditreport.pl	Build the data collection script	PERL
perl.exe	PERL scripting engine	Executable
perlcrt.dll	PERL scripting engine	Executable
Perlcore.dll	PERL scripting engine	Executable
yyyy-mm-dd-*FQDN*.log	Machine-specific audit file on the audit server	ANSI text
FQDN.log	Machine-tracking files on the audit server	ANSI text
Summary Report-yyyy-mm-dd.html	Summary Report	ANSI HTML
yyyy-mm-dd-*FQDN*-Exception Report.html	Exception Report	ANSI HTML

Table 10 - Files Used by the Audit Report Subsystem

Figure 3 - Exception Report (Part I)

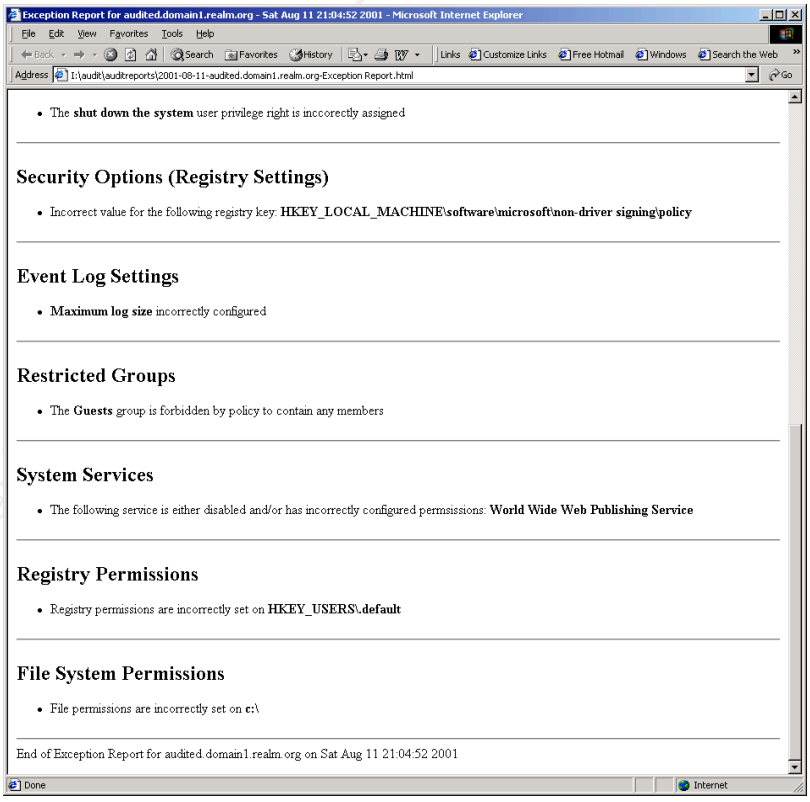

Figure 4 - Exception Report (Part II)

Implementation Details

This system will deploy to the workstations, collect audit data and produce audit reports automatically. However, in order to do this a number of tasks must first be completed. These tasks are grouped into the following categories.

- Create audit templates and corresponding analysis databases
- Configure Audit Server
- Configure Site Group Policy

Create the Template and Database Files for Use with the SECEDIT Tool

1) Download the NSA templates and recommendations from: http://nsa2.www.conxion.com/win2k/download.htm
   a) Make sure to obtain the following:
      i) **`sceregvl2.inf`**
      ii) **`W2K Server.inf`**
      iii) **`W2K Workstation.inf`**
      iv) Guide to Securing Microsoft Windows 2000 Group Policy: Security Configuration Tool Set
2) Read the recommendations: Guide to Securing Microsoft Windows 2000 Group Policy: Security Configuration Tool Set
3) Make backup copies of the NSA templates to make local changes. This retains the original templates untouched for future reference.
   ```
   COPY "W2K Server.inf " audit_server.inf
   COPY "W2K Server.inf " audit_dc.inf
   COPY "W2K Workstation.inf " audit_ workstation.inf
   ```
4) Copy the NSA templates and the audit system templates into the **`%SYSTEMROOT%\security\templates`** directory.
5) Rename the original **`%SYSTEMROOT%\inf\ sceregvl.inf`** file:
   ```
   REN %SYSTEMROOT%\inf\ sceregvl.inf %SYSTEMROOT%\inf\ sceregvl.inf.old
   ```
6) Copy the NSA Security Configuration and Analysis update file to the **`%SYSTEMROOT%\inf`** directory renaming it to the name of the original file.
   ```
   COPY sceregvl2.inf %SYSTEMROOT%\inf\ sceregvl.inf
   ```
7) Refresh the **`SCECLI.DLL`** settings. This adds some new items to the Security Configuration and Analysis interface.
   ```
   REGSVR32 SCECLI.DLL
   ```
8) Set up a custom Microsoft Management Console for working with this system.
   a) Click Start->Run, type mmc and press Enter.
   b) Select **Add/Remove Snap-in** from the **Console** menu.
   c) Click the **Add** button on the Add/Remove Snap-in window.
   d) Select Security Configuration and Analysis from the Add Standalone Snap-in window and press the **Add** button one time.

e) Select **Security Templates** from the Add Standalone Snap-in window and press the **Add** button one time.
f) Press the **Close** button on the Add Standalone Snap-in window.
g) Press the **OK** button on the Add/Remove Snap-in window.
h) The MMC should look similar to **Figure 5**.

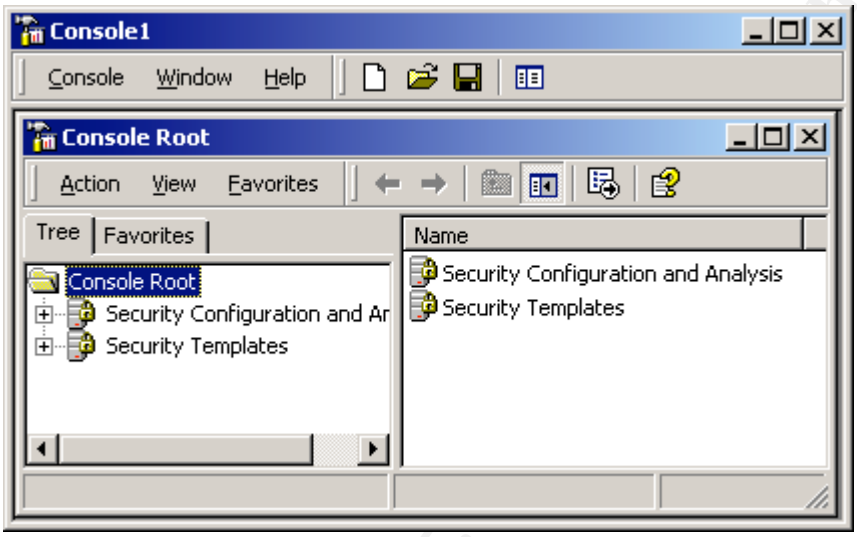

Figure 5

i) Select **Save** from the **Console** menu and save the MMC as Audit Template Configuration. By default this saves it in the current user's Administrative Tools menu. This is a good place.
j) Open each of the **`audit_*`** templates and customize them to suit the security policy. The audit reports will be based off the settings in the templates. **Figure 6** shows the **`audit_workstation`** template being updated.

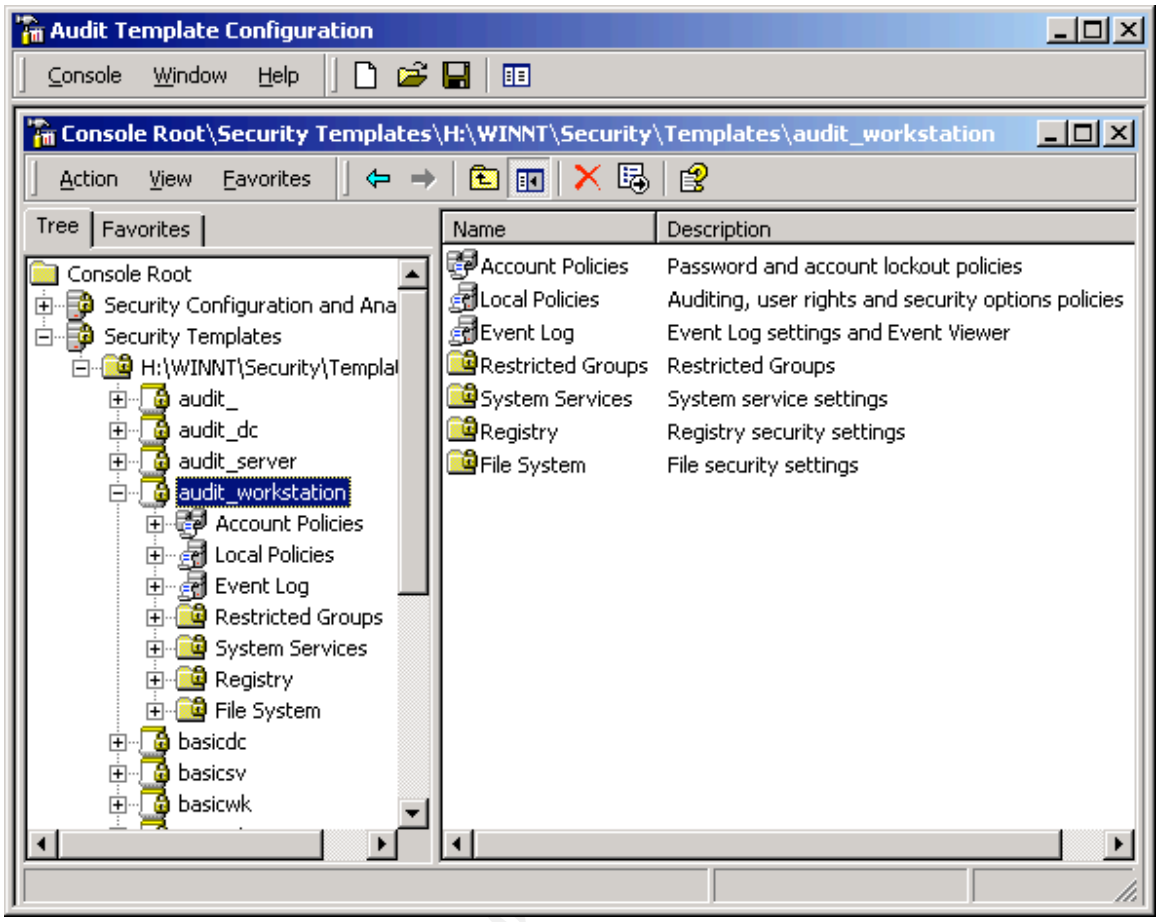

Figure 6

k) Once any changes have been made, highlight the template name in the left-hand window of the Audit Template Configuration MMC and select **Save** from the **Action** menu.
l) Repeat this on any other templates that have been updated.
m) Do not close the Audit Template Configuration MMC.

9) Create the databases.
   a) Click on Security Configuration and Analysis in the left-hand window.
   b) **Figure 7** shows the screen that comes up if no database has been defined. This should be the case since this MMC was just created.

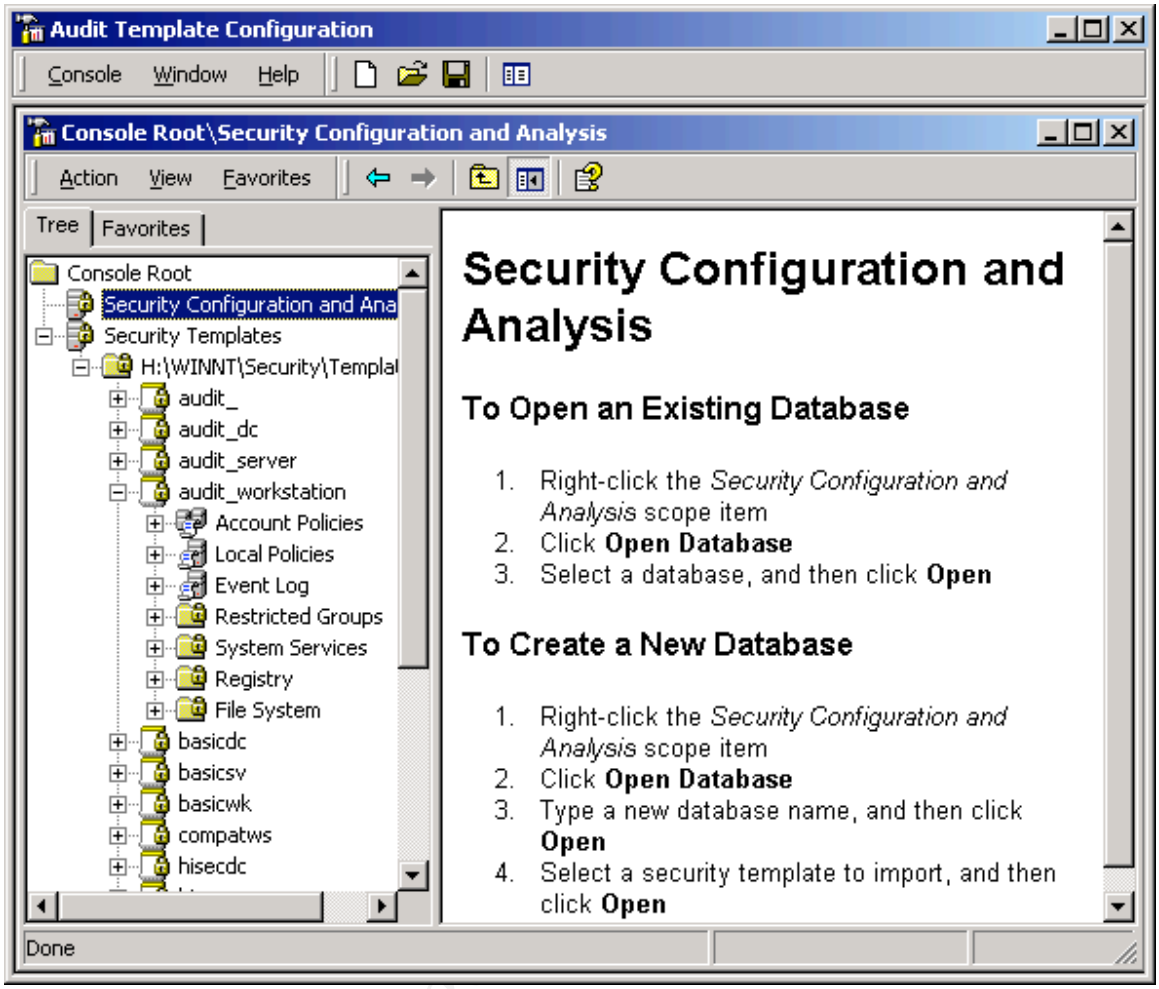

Figure 7

c) Following the directions on the screen:
   i) Right-click on Security Configuration and Analysis in the left-hand window.
   ii) Choose **Open Database**.
   iii) Type in a new database name (audit_dc.sdb, audit_server.sdb or audit_workstation.sdb) and click the **Open** button as shown in **Figure 8**.

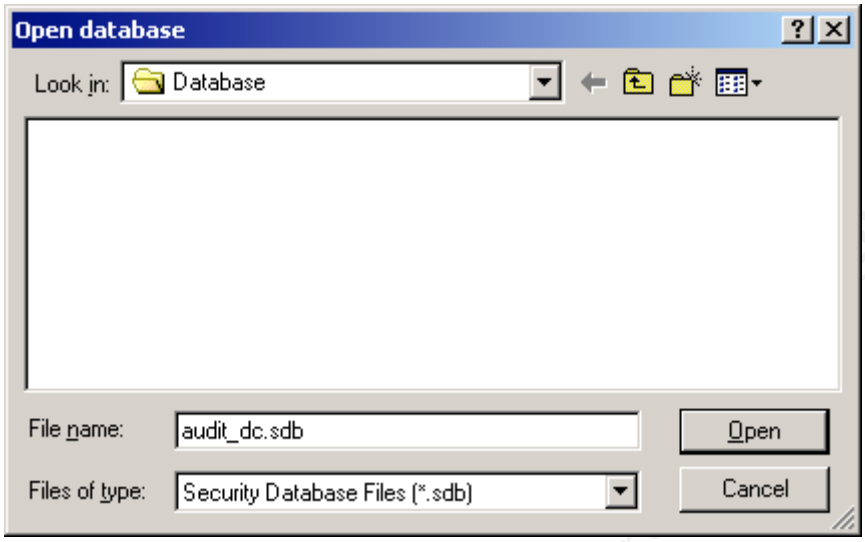

Figure 8

iv) In order to create databases for use with the Audit System, select the security template that corresponds with the database name as shown in **Table 11** when prompted to Import Template. See **Figure 9**.

Database Name	Template Name
Audit_dc.sdb	audit_dc.inf
Audit_server.sdb	audit_server.inf
Audit_ workstation.sdb	audit_ workstation.inf

Table 11 - Database and Template Names for the Audit System

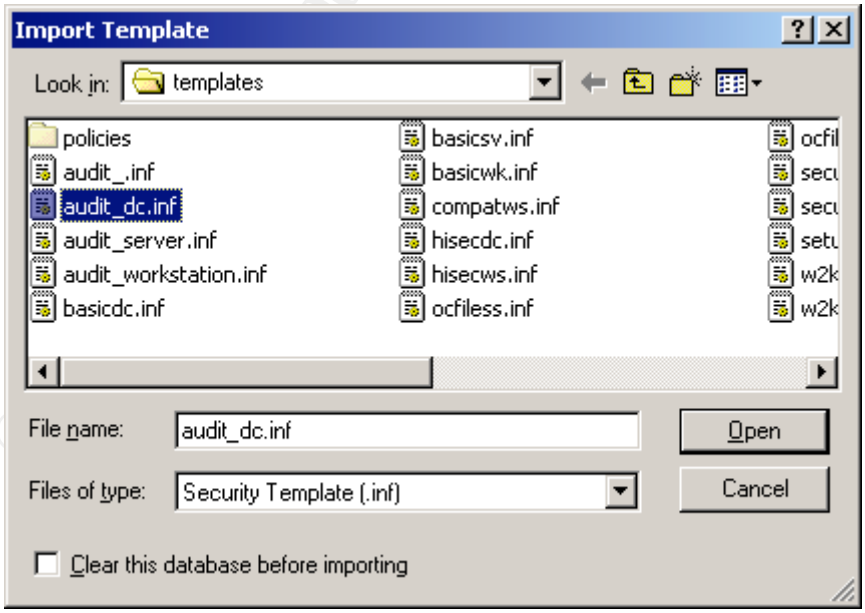

Figure 9

v) Repeat these steps until the three databases listed in **Table 11** have been created.

10) Close the Audit Template Configuration MMC. If a message similar to **Figure 10** comes up, the click **Yes** and recreate the corresponding database. The best way to do this is to delete the database file and then go through the preceding steps to recreate the database.

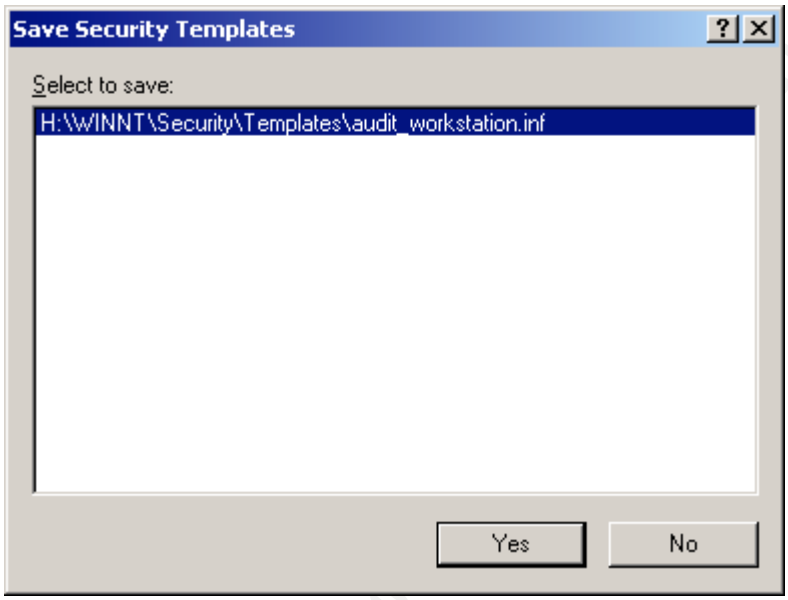

Figure 10

11) The templates end up stored in:

```
%SYSTEMROOT%\security\templates.
```

12) The databases end up stored in:

```
%SYSTEMDRIVE%\Documents and Settings\%USERNAME%\My Documents\Security\Database
```

Copy the most restrictive template and database to the template and database that will be used when there is no %MACHINEROLE% on the system.

```
COPY audit_dc.sdb audit_.sdb
COPY audit_dc.inf audit_.inf
```

Audit Server Setup

The next task is to set up the Audit Server. It involves creating a user and a group under which to run the Audit Collection and Audit Report subsystems as well as the ability to access the audit data files. Next the proper shares must be created and the permissions set. After this, the audit system files must be placed in the proper locations. Lastly, the Windows 2000 Task Scheduler on the Audit Server must be configured to run the Audit Collection and Audit Report subsystems at the proper times as the proper user.

1) Create the user and group for running the Windows 2000 Task Scheduler on the audit server.

a) Click Start->Run->Administrative Tools->Active Directory Users and Computers.
b) Double-click the domain containing the Audit Server from the right-hand window.
c) Click the users folder (or any organizational unit that would hold the proper user for this task.)
d) Create a user named Auditor with a strong password.
    i) Do not allow the user to change the password or allow the password to expire. *Automatic expiration of service accounts usually severely impacts the system. These passwords should still be changed, but this should be done manually and then tested using a Standard Operating Procedure on a regular basis.*
e) Create a global security group called AuditCollect. Make the user Auditor the only member. Members of this group will be able to access the HIDDEN$ shares on the machines and view and/or alter audit data. Members of this group will also be able to alter the audit data, scripts, reports and support files on the Audit Server.
f) Create a global security group called AuditReports. Place any one who should be able to view all the audit reports in this group.

2) Create the file structure and place the files for the Audit System in the proper places.
a) Create the following file structure, where E: is not the system or boot partition.

```
E:\
__ AUDIT
    ___auditlogs
    _       ____machinelist
    ____auditreports
    ____auditscripts
```

b) Place the Audit System files in the proper locations as indicated in **Table 12**.

Location	File
E:\AUDIT\auditscripts	y-with-crlf.txt
	audit_.inf
	audit_dc.inf
	audit_server.inf
	audit_workstation.inf
	localaudit.cmd
	localaudit.pl
	recordfqdn.pl
	attest.pl
	PerlCRT.dll
	perlcore.dll
	perl.exe
	subinacl.exe
	checktracking.pl
	auditreport.cmd
	audit_workstation.sdb
	audit_dc.sdb
	audit_server.sdb
	audit_.sdb
	auditreport.pl
	auditreport.cmd

	reportdefinitions.txt
	collect.pl
	collect.cmd

Table 12 - Audit System File Locations on the Audit Server

c) Remove the ACLs for Everyone on these files and directories. This can be done by simply unchecking the "Allow inheritable permissions from parent to propagate to this object" checkbox on the Security properties page for the `E:\AUDIT` directory.

d) Grant Full Control to the AuditCollect group on the `E:\AUDIT` directory. This will automatically propagate downwards. These rights should also be granted to any other group whose members may be performing maintenance and troubleshooting of the Audit System.

e) Grant Read permissions to the AuditReports group on the `E:\AUDIT\auditreports` directory.

3) Create the shares used for the Audit System.

a) Create shares and assign share permissions as outlined in **Table 13**.

Share Name	Path	Permissions
Auditlogs	`E:\AUDIT\auditlogs`	Audit_server_domain\AuditCollect: Full Control
Auditscripts	`E:\AUDIT\auditscripts`	Everyone: Read Audit_server_domain\AuditCollect: Full Control
Auditreports	`E:\AUDIT\auditreports`	Audit_server_domain\AuditReports: Read Audit_server_domain\AuditCollect: Full Control

Table 13 – Shares on the Audit Server Used by the Audit System

4) Configure the Windows 2000 Task Scheduler to execute the Audit Collection subsystem.

a) Click on Start->Control Panel

b) Double-click the Scheduled Tasks applet as shown in **Figure 11**.

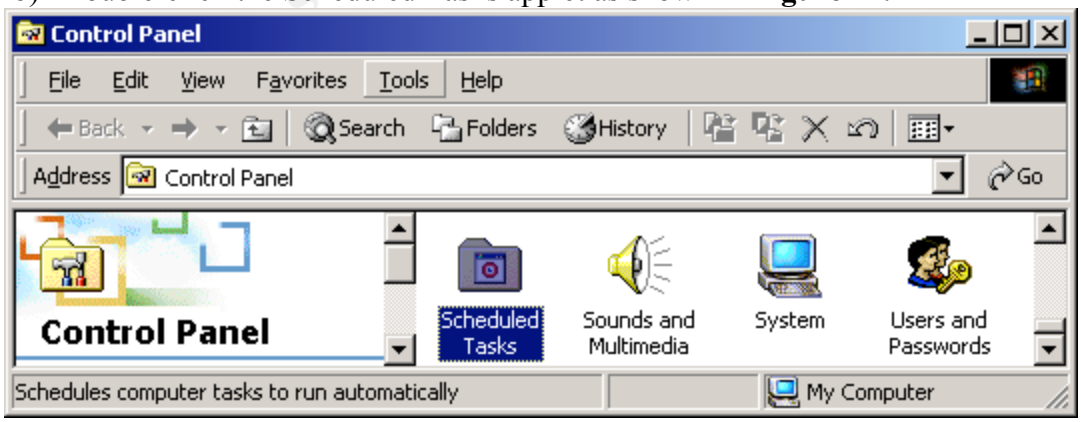

Figure 11

c) Double-click the Scheduled Tasks applet and the Scheduled Tasks applet will start as shown in **Figure 12**.

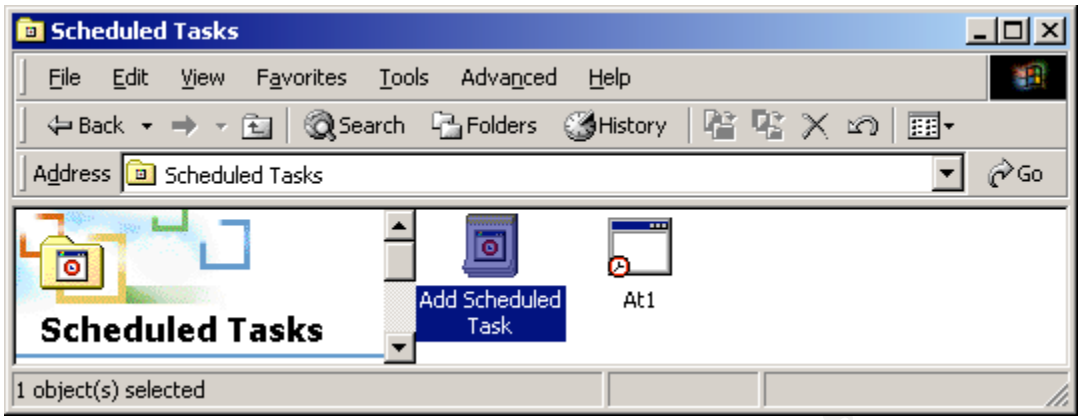

Figure 12

d) Click **Next** as shown in **Figure 13**.

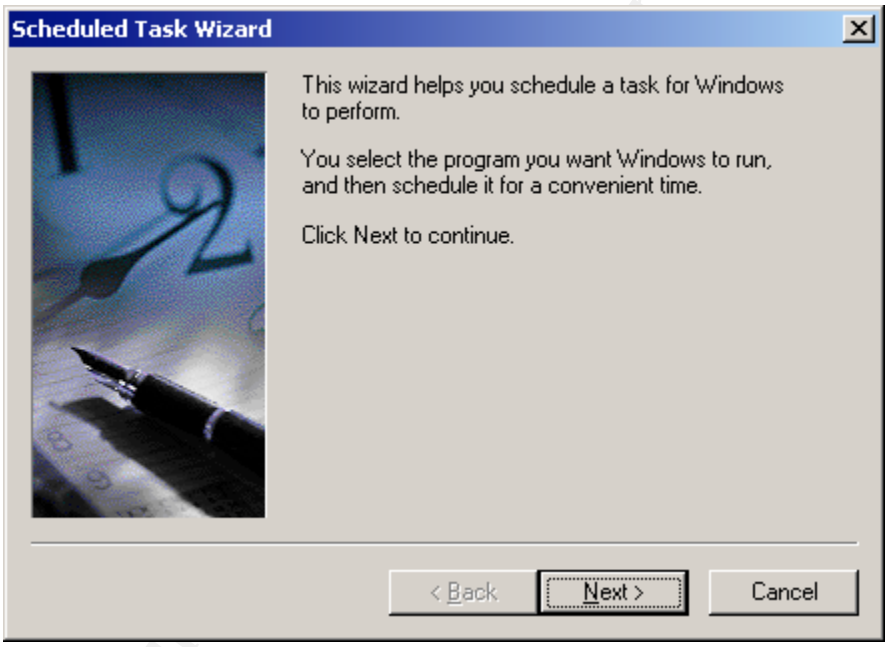

Figure 13

e) Click the **Browse** button as shown in **Figure 14**.

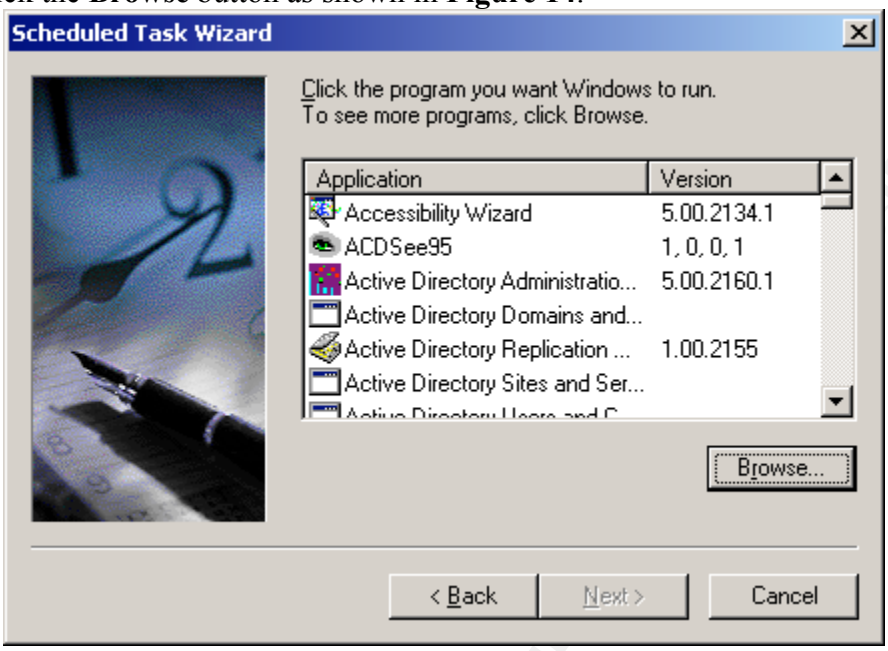

Figure 14

f) Browse to the `E:\AUDIT\auditscripts` directory and choose `collect.cmd` as shown in **Figure 15**.

g) Click **Open**.

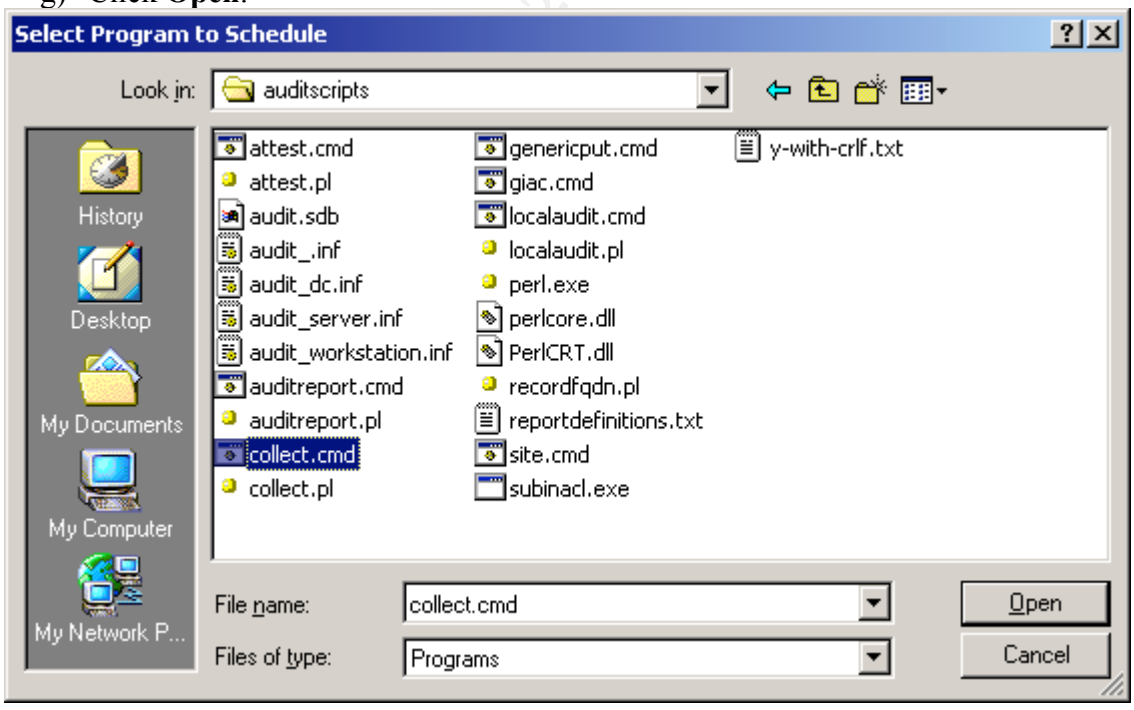

Figure 15

h) Enter **Audit Collection Subsystem** in the task name field and select the **Daily** radio button as shown in **Figure 16**.
i) Click **Next**.

Figure 16

j) Set the Start Time for 5:00 AM and accept the rest of the defaults on this page as shown in **Figure 17**.
k) Click **Next**.

Figure 17

l) Place the Audit_server_domain\Auditor in the user name field and place the correct password in the password and confirmation fields as shown in **Figure 18**.

m) Click **Next**.

Figure 18

n) Click the **Finish** button as shown in **Figure 19**.

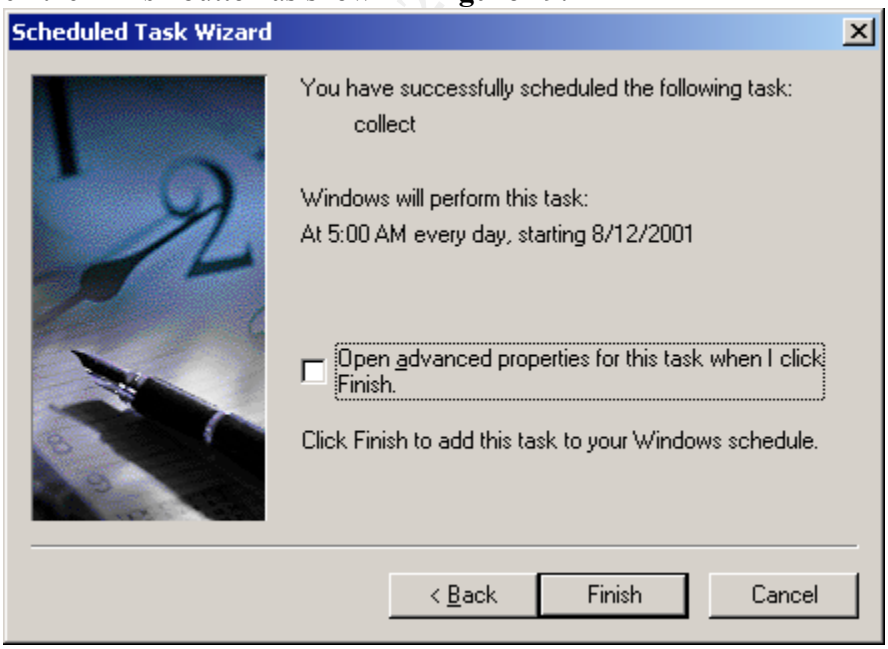

Figure 19

5) The settings for the Audit Collection and Audit Report subsystems are contained in **Table 14**.

Setting	Audit Collection Subsystem	Audit Report Subsystem
Program to Schedule	collect.cmd	auditreport.cmd

Task Name	Audit Collection Subsystem	Audit Report Subsystem
Perform This Task	Daily	Daily
Start Time	5:00 AM	8:00 AM
User Name	Audit_server_domain\Auditor	Audit_server_domain\Auditor

Table 14 – Task Scheduler Configuration Summary for Audit System Tasks on the Audit Server

6) Configure the Windows 2000 Task Scheduler to execute the Audit Report subsystem.
   a) Use the same methodology as discussed above for configuring the Audit Collection subsystem.
   b) The settings for the Audit Collection subsystem are contained in **Table 14**.

Site Group Policy Setup

The Site Group Policy has a single function in the Audit System. It functions as a tool to deliver the Audit System components to the local machine, the Audit Setup subsystem. In order to ensure that the system is properly configured the Audit Setup subsystem runs each time the machine is started, checking the Local Audit system and updating it if necessary.

1) Click Start->Programs->Administrative Tools->AD Sites and Services.
2) Select the Site you want to configure a Group Policy for. In this example, Default-First-Site-Name is used as shown in **Figure 20**.

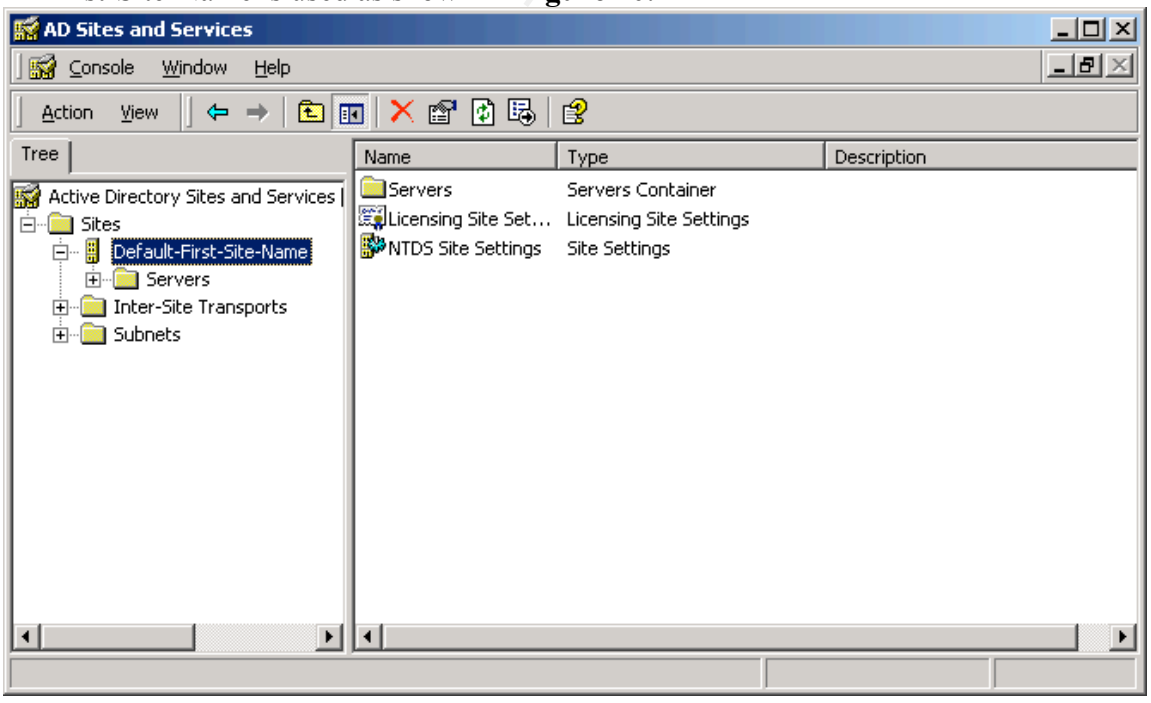

Figure 20

3) Select **Properties** from the **Action** menu and the properties for the Default-First-Site-Name will be displayed as shown in **Figure 21**.

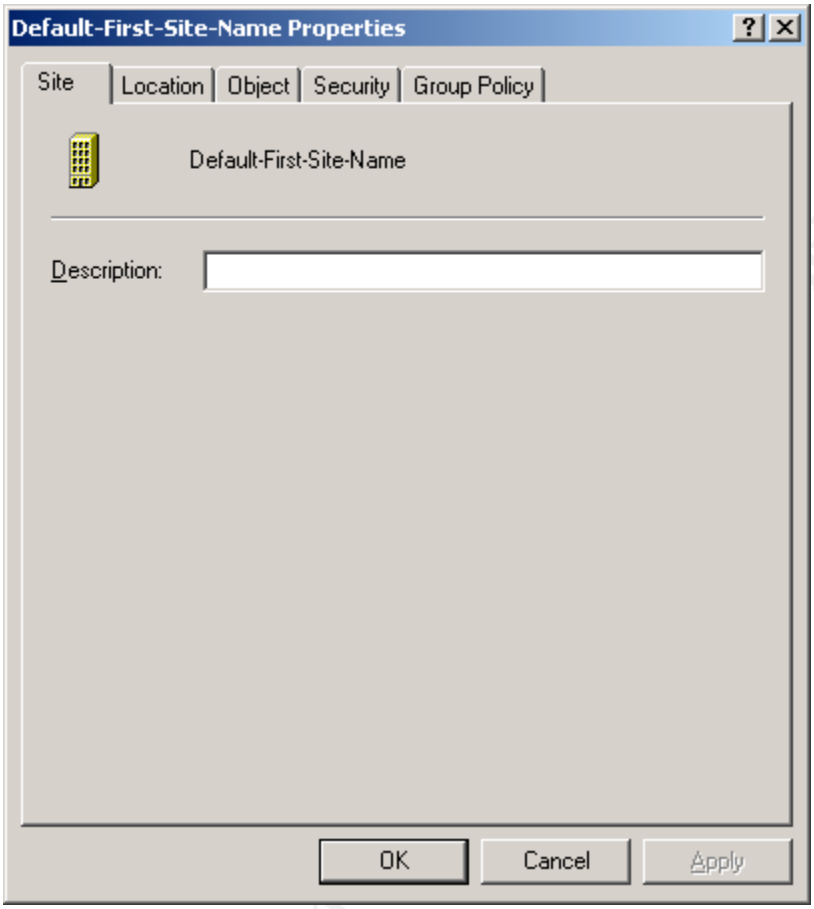

Figure 21

4) Click on the **Group Policy** tab.
5) Click the **New** button and rename New Group Policy Object to Audit Setup as shown in **Figure 22**.

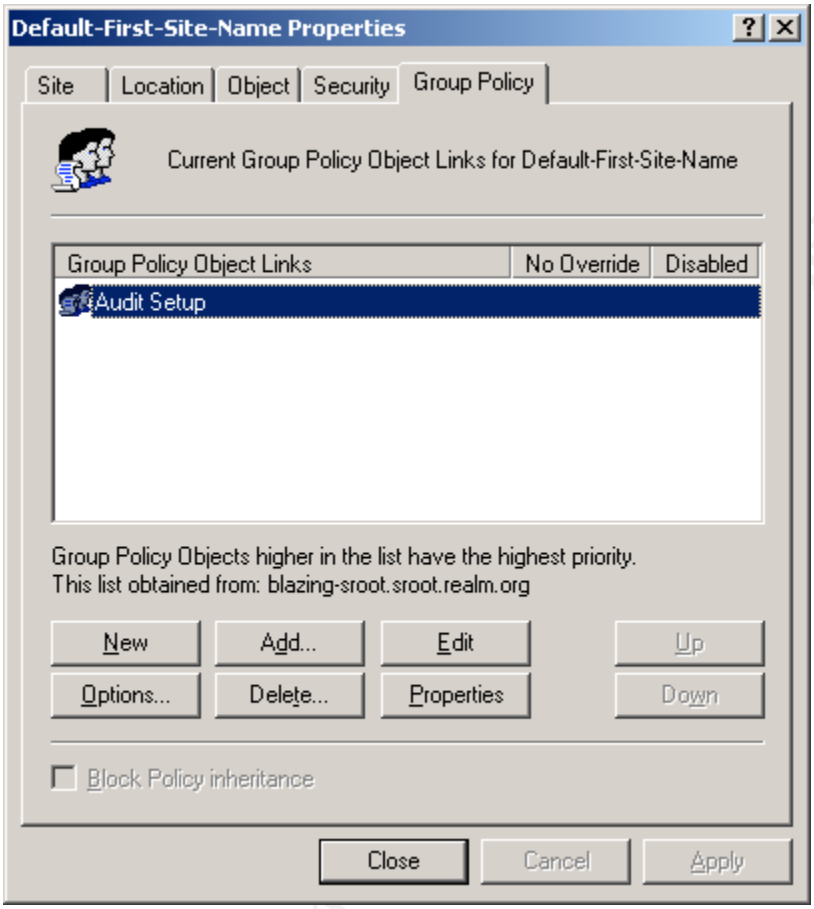

Figure 22

6) While Group Policy Object Link labeled Audit Setup is selected, click the **Properties** button. The Audit Setup Properties will be displayed as shown in **Figure 23**.

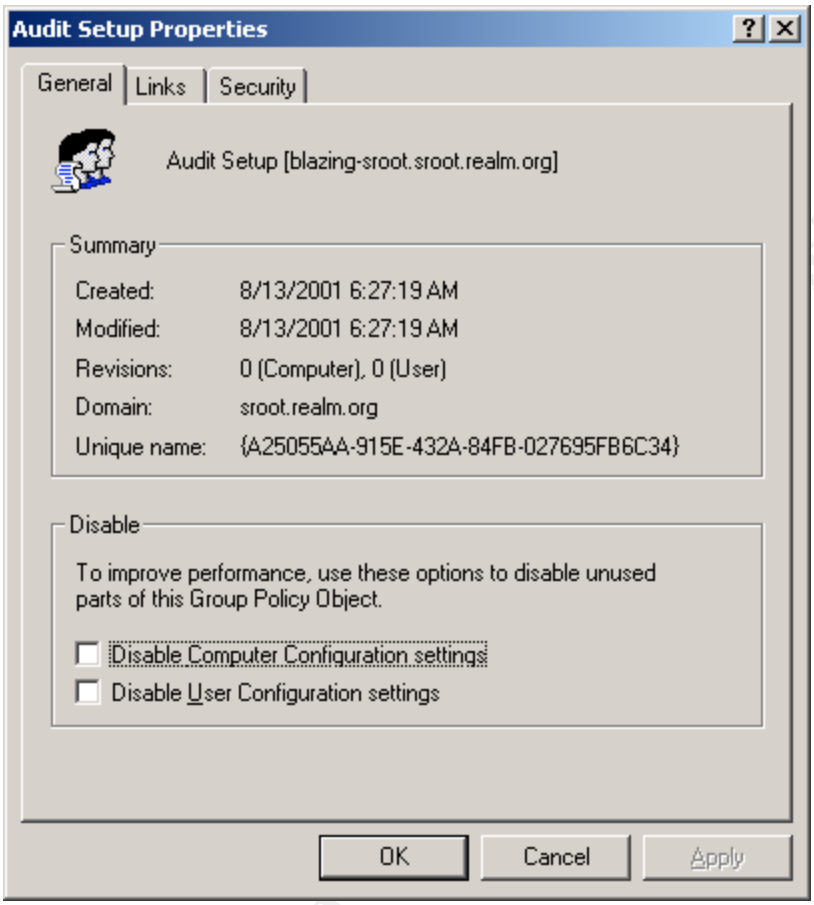

Figure 23

7) Click the "Disable User Configuration settings" checkbox. No user configuration settings are used and this will speed up the processing of the Group Policy object. A confirmation dialog will be presented as shown in **Figure 24**.
8) Click **Yes**.

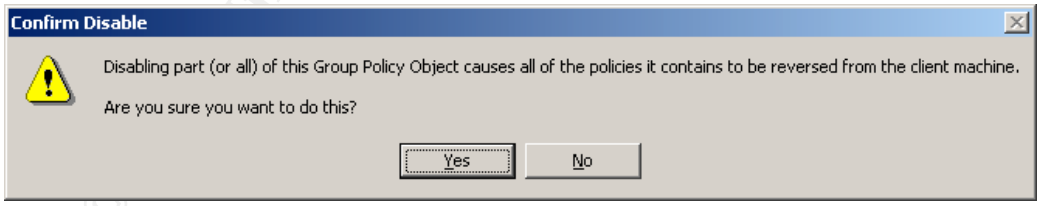

Figure 24

9) Click **OK** to close the Audit Policy Properties window.
10) While Group Policy Object Link named Audit Setup is selected, click the **Edit** button. The Group Policy will be displayed as shown in **Figure 25**.

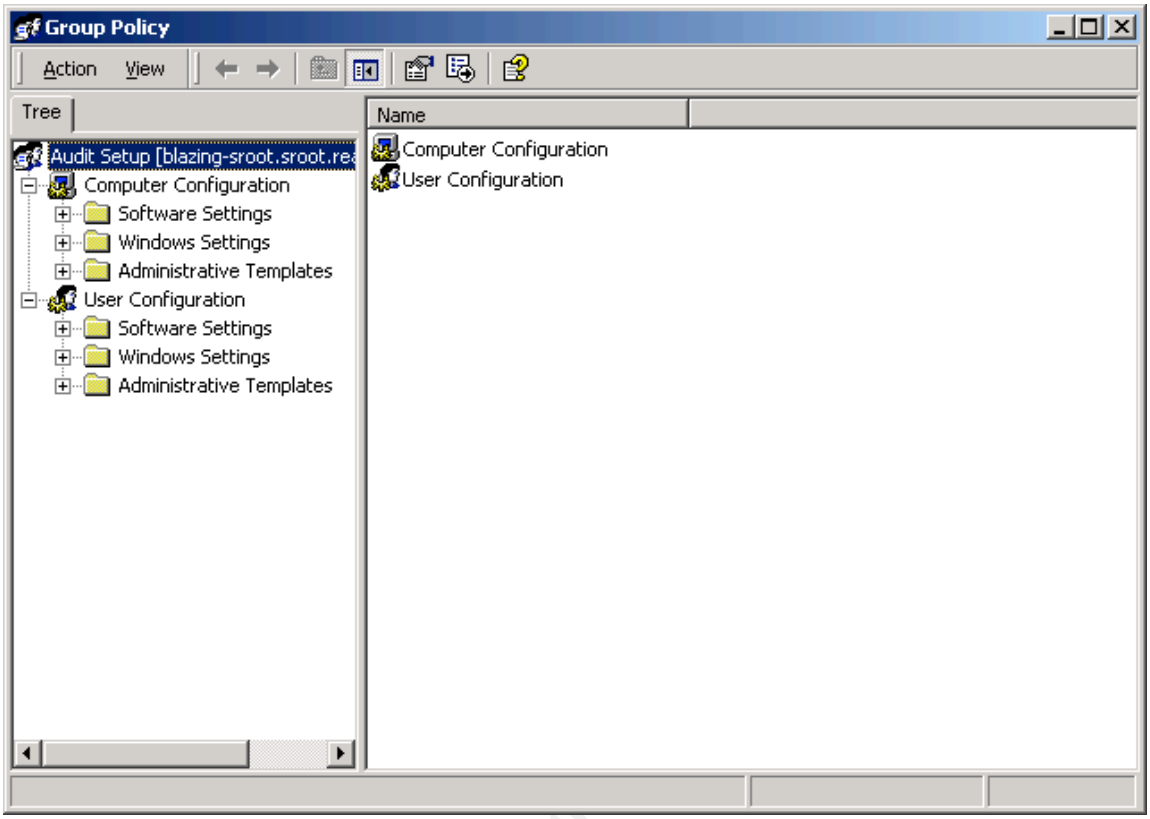

Figure 25

11) In the left-hand window, under Computer Configuration, double-click **Windows Settings**.
12) Next, select Scripts as shown in **Figure 26**.

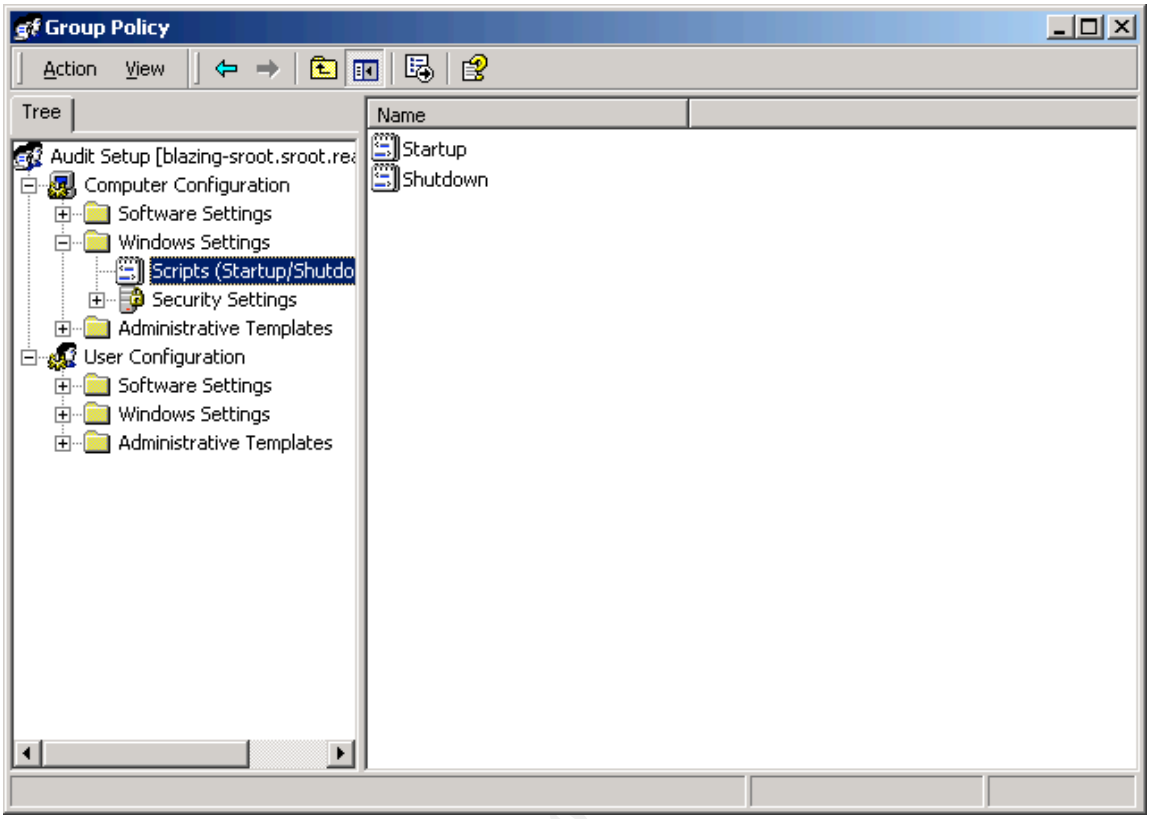

Figure 26

13) In the right-hand window, select Startup, then choose **Properties** from the **Action** menu. The Startup Properties will be displayed as shown in **Figure 27**.

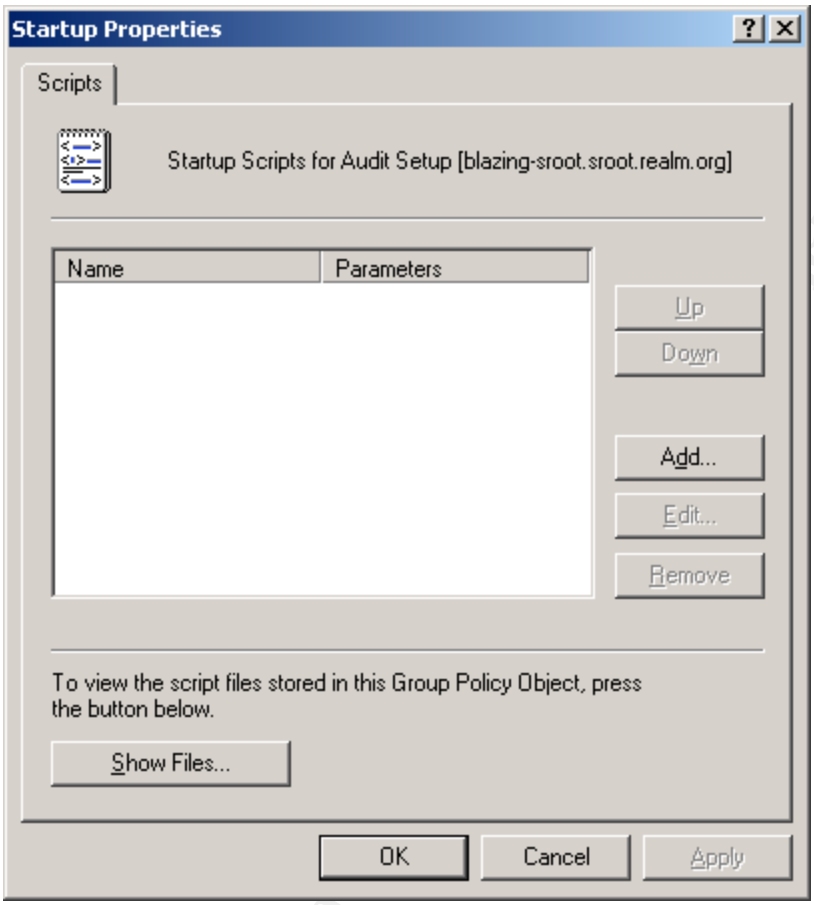

Figure 27

14) Click on the **Show Files** button. This will bring up a Windows Explorer window showing the scripts defined for this Group Policy object. There should not be any files in this directory.
15) Copy the `site.cmd` file into this directory. The easiest way to do this is to open up another instance of Windows Explorer and copy and paste the file into the Group Policy object scripts Window. When this step has been competed, the Group Policy object scripts window should look like **Figure 28**.

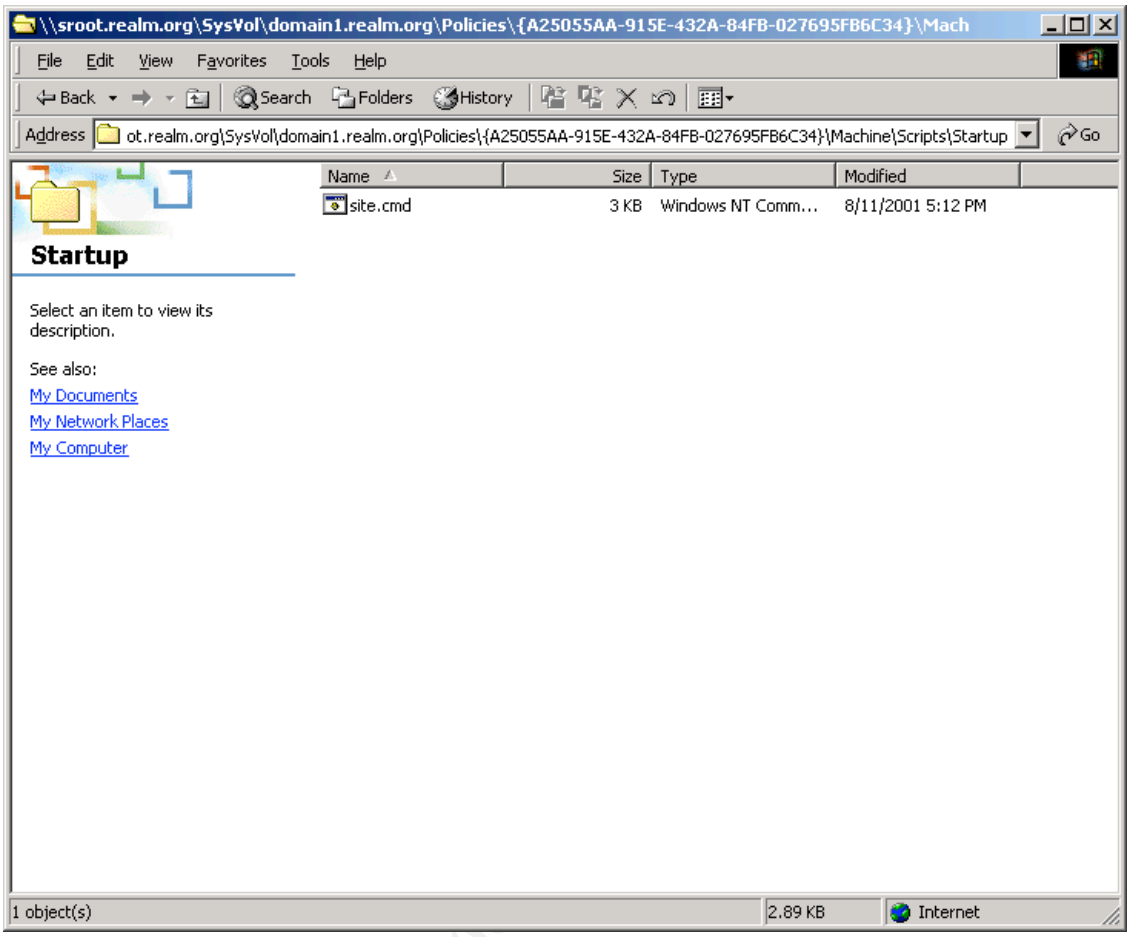

Figure 28

16) Close the Group Policy object scripts window.
17) On the Startup Properties window, click the **Add** button. The Add a Script dialog will be presented as shown in **Figure 29**.

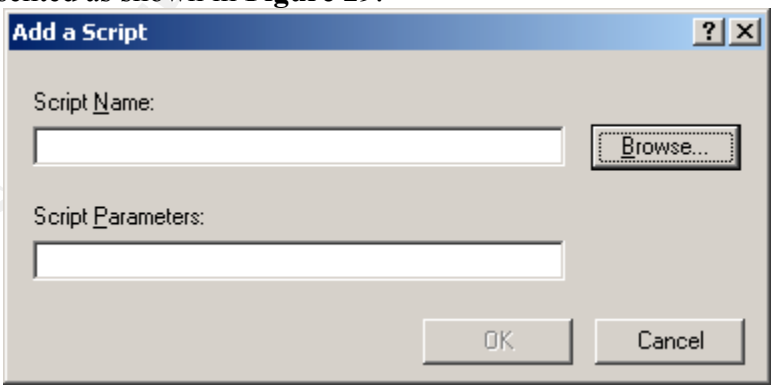

Figure 29

18) Click the **Browse** button and select the `site.cmd` file as shown in **Figure 30**.

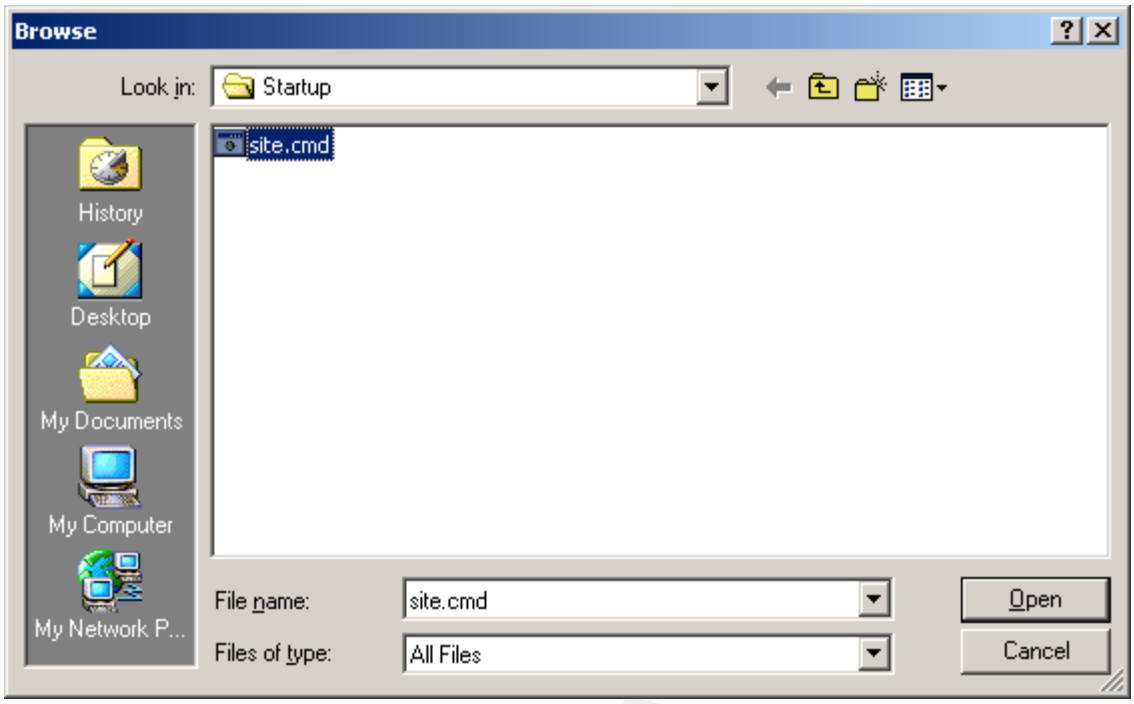

Figure 30

19) Click the **Open** button and the `site.cmd` script will show up in the Add a Script dialog as shown in **Figure 31**.

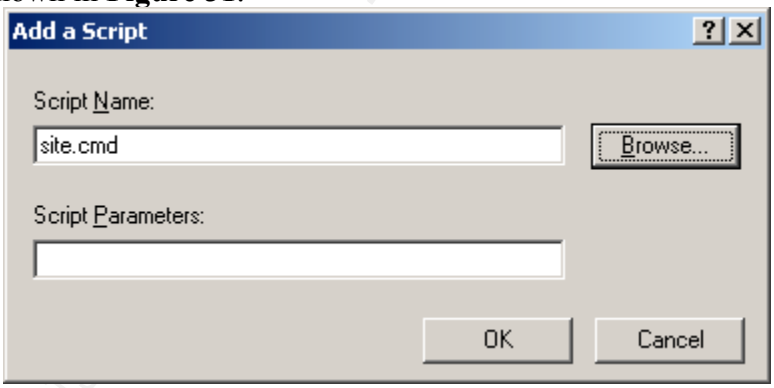

Figure 31

20) Click the **OK** button on the Add a Script dialog. At this point the Startup Properties will show the `site.cmd` script as shown in **Figure 32**.

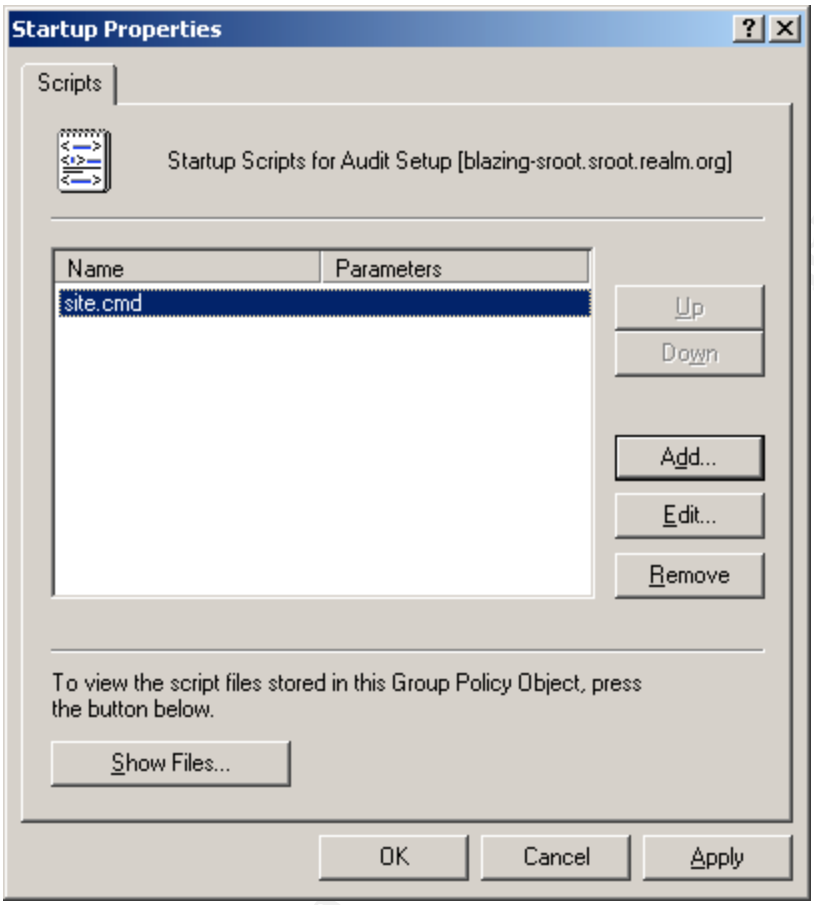

Figure 32

21) Click the **OK** button on the Startup Properties window.
22) Close the Group Policy window.
23) Click the **Close** button on the Default-First-Site-Name Properties window.
24) Close the AD Sites and Services window.

File Locations

Table 15 summarizes all the files and file locations in the Audit Subsystem

`C:\HIDDEN\AUDIT` on the local machine	`y-with-crlf.txt`
	`audit_%MACHINEROLE%.inf`
	`localaudit.cmd`
	`localaudit.pl`
	`recordfqdn.pl`
	`attest.pl`
	`PerlCRT.dll`
	`perlcore.dll`
	`perl.exe`
	`subinacl.exe`
	`checktracking.pl`
	`audit_%MACHINEROLE%.sdb`

	results.txt found.txt %COMPUTERNAME%.raw fqdn.raw yyyy-mm-dd-FQDN.raw *FQDN*.log
E:\AUDIT\auditlogs on the Audit Server	COLLECT-ERRORS.LOG yyyy-mm-dd-FQDN.raw getdata.cmd
E:\AUDIT\auditlogs\machinelist on the Audit Server	*FQDN*.log
E:\AUDIT\auditscripts on the Audit Server	y-with-crlf.txt audit_.inf audit_dc.inf audit_server.inf audit_workstation.inf localaudit.cmd localaudit.pl recordfqdn.pl attest.pl PerlCRT.dll perlcore.dll perl.exe subinacl.exe checktracking.pl auditreport.cmd audit_workstation.sdb audit_dc.sdb audit_server.sdb audit_.sdb auditreport.pl auditreport.cmd reportdefinitions.txt collect.pl collect.cmd
E:\AUDIT\auditreports on the Audit Server	Summary Report-yyyy-mm-dd.html yyyy-mm-dd-FQDN-Exception Report.html
Startup scripts directory of the Audit Setup Site Group Policy Object	site.cmd

Table 15 – File Locations of All Audit System Files

Subsystem Executors

Table 16 summarizes all the subsystems, how they are executed and the credentials that they run under.

Subsystem	Executes on	Credentials
Audit Setup	Local machines	Group policy engine
Local Audit	Local machines	Local System
Audit Collection	Audit Server	Auditor
Audit Report	Audit Server	Auditor

Table 16 – Subsystem Execution Details

Summary

The Auditing System laid out in this paper will provide a fairly complete level of automated auditing against Windows 2000 systems. It can aid administrators in maintaining properly secured operating systems on server and workstation platforms. This system can be a valuable aid to monitor all the systems in an Active Directory Forest for compliance with technical security controls.

Bibliography

- Haney, Julie, M. Guide to Securing Microsoft Windows 2000 Group Policy: Security Configuration Tool Set. National Security Agency. 2001
- Pierce, Clinton. Sams Teach Yourself Perl in 24 Hours. Sams Publishing. 2000.
- Opitz, David. Guide to Windows 2000 Kerberos Settings. National Security Agency. 2001
- McGovern, Owen R., Haney, Julie, M. Guide to Securing Microsoft Windows 2000 File and Disk Resources. National Security Agency. 2001
- Fossen, Jason. Windows 2000: Active Directory and Group Policy. SANS Institute. 2001.
- Lowe-Norris, Alistair G. Windows 2000 Active Directory.O'Reilly & Associates, Inc. 2000.
- Distributed Perl Documentation (5.005_02): http://www.perl.com/CPAN-local/doc/manual/html/index.html

Author's Note: *A good deal of research went into using Visual Basic or Visual Basic Scripting to assist in gathering the audit data, but due to time restrictions this could not be placed in the final paper. These are all excellent books.*

- Esposito, Dino. Windows Script Host Programmer's Reference. Wrox Press Ltd. 1999.
- Eck, Thomas. Windows NT/2000 ADSI Scripting for System Administration. MTP. 2000.
- Perry, Greg., Hettihewa, Sanjaya. Sams Teach Yourself Visual Basic 6 in 24 Hours. Sams Publishing. 1998.

Appendix – Code Listings

The following annoted script files are contained in this appendix. The audit template files and the Report Definitions file are also included at the end of the Appendix. These files are summarized in **Table 17**.

Script File	Purpose	Subsystem(s)	Language
`Site.cmd`	Setup the Local Audit subsystem	Audit Setup	Batch
`Attest.pl`	Check to set if the Local Audit subsystem is configured to run on the local machine	Audit Setup	PERL
`Checktracking.pl`	Check to see if the machine-tracking file is created and if not create it	Audit Setup	PERL
`Recordfqdn.pl`	Determine the DNS name of the computer	Audit Setup Local Audit	PERL
`Localaudit.cmd`	Collect the local audit data	Local Audit	Batch
`Localaudit.pl`	Convert the local audit data to ANSI text and create the proper file name	Local Audit	PERL
`Collect.cmd`	Collect the local audit data	Audit Collection	Batch
`Collect.pl`	Collect the local audit data	Audit Collection	PERL
`Auditreport.cmd`	Create the audit report	Audit Report	Batch
`Auditreport.pl`	Create the audit report	Audit Report	PERL
`Audit_dc.inf`	Audit template for domain controllers	Local Audit	N/A
`Audit_server.inf`	Audit template for servers	Local Audit	N/A
`Audit_workstation.inf`	Audit template for workstations	Local Audit	N/A
`Reportdefinitions.txt`	Report Definitions for the Exception Reports	Audit Report	N/A

Table 17 – Script, Template and Report Definition Files

SITE.CMD

```
@ECHO OFF

REM     SITE.CMD
REM
REM     Version 1.4
REM     Last revised 8/12/1
REM
REM This script runs whenever a machine starts up.
REM This script is run at startup by the site group policy.
REM
REM This script ensures that the proper directory structure and startup
REM scripts and support files are present on the local machine and then
REM checks the Task Scheduler to confirm that the audit script is scheduled
REM to run.
REM Lastly, the script creates the machine-tracking file on the Audit Server
REM if it is not present


REM Create directory for audit scripts and support files
MD C:\HIDDEN\AUDIT > NUL 2> NUL

REM Switch to the local machine
C:
CD\HIDDEN\AUDIT


REM Check startup scripts and support files, copying them from the audit
REM server if necessary
IF NOT EXIST C:\HIDDEN\AUDIT\localaudit.cmd (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\localaudit.cmd " C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\localaudit.pl (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\localaudit.pl " C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\recordfqdn.pl (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\recordfqdn.pl " C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\subinacl.exe (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\subinacl.exe" C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\perl.exe (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\perl.exe" C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\perlcrt.dll (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\perlcrt.dll" C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\perlcore.dll (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\perlcore.dll" C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\attest.pl (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\attest.pl" C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\y-with-crlf.txt (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\y-with-crlf.txt" C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\audit_%MACHINEROLE%.inf (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\audit_%MACHINEROLE%.inf" C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\audit_%MACHINEROLE%.sdb (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\audit_%MACHINEROLE%.sdb" C:\HIDDEN\AUDIT) > NUL 2> NUL
IF NOT EXIST C:\HIDDEN\AUDIT\checktracking.pl (copy /y "\\blazing-
sroot.sroot.realm.org\auditscripts\checktracking.pl" C:\HIDDEN\AUDIT) > NUL 2> NUL


REM Protect the directory and the local audit scripts and support files by
REM marking their attributes and/or setting their ACLs
ATTRIB C:\HIDDEN +r +s +h > NUL 2> NUL
ATTRIB C:\HIDDEN\AUDIT +r +s +h > NUL 2> NUL

REM Y-WITH-CRLF.TXT is simply a text file containing y and a carriage
REM return/linefeed.
REM This technique is used to force the CACLS command to complete without
REM user intervention.
CACLS C:\HIDDEN\AUDIT /G SYSTEM:F SROOT.REALM.ORG\AuditCollect:F /D EVERYONE < y-with-
crlf.txt > NUL 2>NUL
```

```
CACLS C:\HIDDEN\AUDIT\*.* /G SYSTEM:F SROOT.REALM.ORG\AuditCollect:F /D EVERYONE < y-
with-crlf.txt > NUL 2>NUL

REM Delete any old output from the AT command or the ATTEST.PL script
DEL results.txt > NUL 2> NUL
DEL found.txt > NUL 2> NUL

REM Create the input file for the ATTEST.PL script
AT > results.txt 2> NUL

REM Check the output of the AT command for the correct configuation
PERL attest.pl results.txt > NUL 2> NUL

REM If the Task Scheduler was not correctly configured, issue the AT command
REM to configure the Task Scheduler to run LOCALAUDIT.CMD
IF NOT EXIST found.txt  (AT 4:00 /EVERY:M,T,W,Th,F,Sa,Su
"C:\AUDIT\HIDDEN\LOCALAUDIT.CMD") > NUL 2> NUL

REM Create the share for the collection script to use to collect the audit data
REM Set the permissions on the share to allow the user that the Windows Task
REM Scheduler on the audit server runs as to collect the audit data.
REM No one else should have access to this share
NET SHARE HIDDEN$=C:\HIDDEN\AUDIT > NUL 2> NUL
SUBINACL /SHARE HIDDEN$ /GRANT=SROOT.REALM.ORG\AuditCollect=F /REVOKE=EVERYONE > NUL 2>
NUL


REM Create the machine-tracking file on the local machine.
ECHO %COMPUTERNAME% %DATE% %TIME% %MACHINEROLE% > %COMPUTERNAME%.log 2> NUL

REM Create a file containing the output from the IPCONFIG /ALL command.
REM This information is used to update the machine-tracking file with
REM the fully qualified domain name (FQDN) of the machine.
IPCONFIG /ALL > C:\HIDDEN\AUDIT\fqdn.raw 2> NUL

REM Update the machine-tracking update file to replace the NetBIOS computer
REM name with the FQDN
C:\HIDDEN\AUDIT\perl C:\HIDDEN\AUDIT\recordfqdn.pl C:\HIDDEN\AUDIT\%COMPUTERNAME%.log
C:\HIDDEN\AUDIT\fqdn.raw > NUL 2>NUL

REM Check to see if the machine-tracking file exists.  If not, then create it.
C:\HIDDEN\AUDIT\perl C:\HIDDEN\AUDIT\checktracking.pl C:\HIDDEN\AUDIT\%COMPUTERNAME%.log
\\blazing-sroot.sroot.realm.org\auditlogs\machinelist > NUL 2>NUL
```

ATTEST.PL

```
#
#       ATTEST.PL
#
#       Version 1.2
#       Last revised 7/31/1
#
# This script tests to ensure that the Task Scheduler
# (AT.EXE) is configured to run the Audit Collection script,
# LOCALAUDIT.CMD.
#
# The script uses a text file containing the output
# of th AT command as an input.  The name of this
# file is passed to the script via a command line
# variable.
# If the script finds the proper Task Scheduler configuration
# a text file, FOUND.TXT, will be created.  Only the presence
# of this file, not its contents are important.



my $target = $ARGV[0]; # The output from the AT.EXE command
my @raw;               # An array to hold the data during processing
my $rc=0;              # A counter

################################################################################
#                                                                              #
# The || die command is read "or" die.  In PERL logical operators can be used  #
# to link certain commands.  These commands return some type of value when     #
# they are used.  The return value often does not need to be used for          #
# anything.  For instance, the open() function returns a true value if the     #
# file open succeeds and a false value if the file open fails. Since PERL      #
# stops evaluating a logical construct as soon as it is either true or false   #
# without a doubt, the second command will only ever execute if the first one  #
# is false.  This is because as long as the first component of an OR statement #
# is true, the entire statement will be true.  If the first component of the   #
# OR statement is false, the second component will have to be evaluated in     #
# order to determine the outcome of the logical statement.                     #
# In this case, the script exits with an error if it can't open the output     #
# file captured from the AT command.                                           #
#                                                                              #
################################################################################


# Open the input file and read it into @raw for processing
open (SOURCEFILE, $target) || die scalar(localtime) . "\nError opening AT list, $target:
$!\n";
@raw = <SOURCEFILE>;
close SOURCEFILE;

################################################################################
#                                                                              #
# PERL has extremely powerful text matching and handling functions.            #
# the $variable =~ /pattern/ command will find the first occurence of          #
# "pattern" in $variable.  This uses a syntax known as a Regular Expression.   #
# However, sometimes there are difficulties finding patterns containing        #
# special characters.  Most of the special characters have a special meaning   #
# within the match command.  There are multiple ways to get the special        #
# characters recognized as part of the pattern and not as special characters.  #
# Any characters between \Q and \E are treated as part of the pattern.         #
# However, the backslashes in the path in the statement below were not being   #
# treated as part of the pattern.  The solution was to "Escape" each backslash #
# individually.  "Escaping" a special character tells the regular expression   #
# to treat it as part of the pattern and is accomplished by placing a          #
# backslash in front of the special character.                                 #
# Therefore, in order to search for the pattern:                               #
#    Each M T W Th F S Su    4:00 AM       C:\AUDIT\HIDDEN\LOCALAUDIT.CMD      #
```

```
# The first part of the pattern could be grouped with \Q and \E.  However,     #
# the part of the pattern with the backslashes had to have the backslashes     #
# escaped individually.  Additionally, the escaped backslashes had to be       #
# outside of the \Q \E enclosed text.  The proper regular expression ended up  #
# being:                                                                       #
# \QEach M T W Th F S Su    4:00 AM       C:\E\\AUDIT\\HIDDEN\\LOCALAUDIT.CMD  #
#                                                                              #
################################################################################

# Step through the contents of the input array
while (defined(@raw[$rc])) {
       # If the AT.EXE output contains a line indicating the correct
       # configuration, create FOUND.TXT
       if (@raw[$rc] =~ /\QEach M T W Th F S Su    4:00 AM
C:\E\\AUDIT\\HIDDEN\\LOCALAUDIT.CMD/i) {
              open (DESTFILE, ">found.txt") || die scalar(localtime) . "\nError opening
found.txt: $!\n";;
              print DESTFILE ("Found it\n");
              close DESTFILE;
       }
       $rc++;
}
```

CHECKTRACKING.PL

```
#
#       CHECKTRACKING.PL
#
#       Version 1.0
#       Last revised 8/12/1
#
# This script:
#       Check to see the machine-tracking file on the Audit Server exists for
#       this local machine.  If not, create a batch file to create the
#       machine-tracking file with the correct name.
#
#       Output from the IPCONFIG /ALL is used to generate the FQDN
#
#       The file containing the current update to the machine-tracking file
#              as the first command line variable
#       The directory containing the machine-tracking files on the Audit Server
#              as the second command line variable



my @raw;                  # Array containing the raw or source data
my $rc;                   # Counter, generally used for the raw array
my $currentTrackingUpdate; # The latest record for the machine-tracking file
my $machineTracking;       # The machine-tracking directory on the audit server
my @rawTrackingFiles;     # The filenames of all the files in the
                          # machine-tracking directory
my $machineTrackingName;   # The Fully Qualified Domain Name of the machine
my $day;                  # The day of the machine-tracking update
my $date;                 # The date of the machine-tracking update
my $time;                 # The time of the machine-tracking update
my $machineRole;          # The %MACHINEROLE% from the machine-tracking update
my $newTrackingFile;      # The newly created machine-tracking file
my %trackingHash;         # A hash used to store all the names of the machines
                          # in the machine-tracking directory

# Set the tracking update filenames and the directory containing the machine-
# tracking files on the Audit Server
$currentTrackingUpdate = @ARGV[0];
$machineTracking = @ARGV[1];


################################################################################
#                                                                              #
# Open the source data file and read it into the raw array for processing      #
# The || die command is read "or" die.  In PERL logical operators can be used  #
# to link certain commands.  These commands return some type of value when     #
# they are used.  The return value often does not need to be used for          #
# anything.  For instance, the open() function returns a true value if the     #
# file open succeeds and a false value if the file open fails.  Since PERL     #
# stops evaluating a logical construct as soon as it is either true or false   #
# without a doubt, the second command will only ever execute if the first one  #
# is false.  This is because as long as the first component of an OR statement #
# is true, the entire statement will be true.  If the first component of the   #
# OR statement is false, the second component will have to be evaluated in     #
# order to determine the outcome of the logical statement.                     #
# In this case, the script exits with an error if it can't open the input file.#
#                                                                              #
################################################################################

# Retrieve the FQDN from the local machine-tracking file update


# Read in the machine tracking-update file
open (MACHINEFILE, $currentTrackingUpdate) || die scalar(localtime) . " Error opening
machine tracking file, $currentTrackingUpdate: $!\n";
@raw=<MACHINEFILE>;
```

```
close MACHINEFILE;

#################################################################
#                                                               #
# The format of a record (row) in the machine-tracking file is:#
# FQDN Day-of-week Date Time %MACHINEROLE%                      #
#                                                               #
#################################################################

($machineTrackingName, $day, $date, $time, $machineRole) = split (' ', @raw[0]);


# Determine if the machine-tracking file for this machine already exists by
# checking the contents of the machine-tracking directory on the Audit Server

#################################################################################
#                                                                               #
# The list of machine-tracking files without the .log extension is the same as #
# the list of Fully Qualified Domain Names of machines that have run SITE.CMD   #
# via the site group policy startup script.                                     #
#                                                                               #
# In order to quickly determine if the machine-tracking file exists, all the    #
# machine-tracking files names without their .log extensions are placed into a #
# hash.  The name of the present machine is matched against this hash.  If it   #
# does not match, there is no machine-tracking file for the machine and one     #
# will be created.                                                              #
#                                                                               #
#################################################################################


# Open machine-tracking files directory
opendir (SOURCEDIR, $machineTracking) || die scalar(localtime) . " Error opening
directory to get input files, $machineTracking: $!\n";

# Read a list of all the files in the directory into the raw array
@rawTrackingFiles=readdir(SOURCEDIR);
close SOURCEDIR;

# Clear the counter
$rc = 0;
# Step through the file list of machine-tracking files
while (defined(@rawTrackingFiles[$rc])){
        # Filter the . and .. designators from the list
        if (!(@rawTrackingFiles[$rc] eq "." || @rawTrackingFiles[$rc] eq "..")) {
                # Assign the filename without the extension to the hash
                $trackingHash{substr(@rawTrackingFiles[$rc],0,-4)} = "true";
        }
        # Increment the filename counter
        $rc++;
}

# If there is no instance in the tracking hash, there is no machine-tracking
# file
if (!($trackingHash{$machineTrackingName})) {
        # Set the filename for the machine tracking file in the proper format
        $newTrackingFile = $machineTracking . "\\" . $machineTrackingName . ".log";
        # Create the machine-tracking file
        open (MAKETRACK, ">" . $newTrackingFile) || die scalar(localtime) . " Error
creating machine tracking file, $newTrackingFile: $!\n";
        print MAKETRACK "$machineTrackingName $day $date $time $machineRole\n";
        close MAKETRACK;
}
```

RECORDFQDN.PL

```
#
#       RECORDFQDN.PL
#
#       Version 1.0
#       Last revised 8/11/1
#
# This script:
#       Updates the %COMPUTERNAME% in the machine-tracking update file to a
#       Fully Qualified Domain Name (FQDN)
#       Output from the IPCONFIG /ALL is used to generate the FQDN
#
#       The file containing the current update to the machine-tracking file
#               as the first command line variable
#       The file containing the output from the  IPCONFIG /ALL command
#               as the second command line variable



my @raw;                  # Array containing the raw or source data
my $rc;                   # Counter, generally used for the raw array
my $currentTrackingUpdate; # The latest record for the machine-tracking file
my $ipconfigOutput;        # A text file containing the output from the
                          # IPCONFIG /ALL command
my $fqdn;                 # The Fully Qualified Domain Name (FQDN)
my $machineTrackingName;   # The NetBIOS name of the machine
my $day;                  # The day of the machine-tracking update
my $date;                 # The date of the machine-tracking update
my $time;                 # The time of the machine-tracking update
my $machineRole;          # The %MACHINEROLE% from the machine-tracking update

# Set the tracking update and ipconfig output filenames
$currentTrackingUpdate = @ARGV[0];
$ipconfigOutput = @ARGV[1];

################################################################################
#                                                                              #
# Open the source data file and read it into the raw array for processing      #
# The || die command is read "or" die.  In PERL logical operators can be used  #
# to link certain commands.  These commands return some type of value when     #
# they are used.  The return value often does not need to be used for          #
# anything.  For instance, the open() function returns a true value if the     #
# file open succeeds and a false value if the file open fails.  Since PERL     #
# stops evaluating a logical construct as soon as it is either true or false   #
# without a doubt, the second command will only ever execute if the first one  #
# is false.  This is because as long as the first component of an OR statement #
# is true, the entire statement will be true.  If the first component of the   #
# OR statement is false, the second component will have to be evaluated in     #
# order to determine the outcome of the logical statement.                     #
# In this case, the script exits with an error if it can't open the input file.#
#                                                                              #
################################################################################

################################################################################
#                                                                              #
# Read in the host name and the primary suffix name from a text file containing #
# the output from the IPCONFIG /ALL command                                    #
#                                                                              #
################################################################################

open (SOURCEFILE, $ipconfigOutput) || die scalar(localtime) . " Error opening input file:
$ipconfigOutput: $!\n";
@raw=<SOURCEFILE>;
close SOURCEFILE;

# Clear the counter
$rc = 0;
```

```
# Find the host name
while (defined(@raw[$rc])) {
        if (@raw[$rc] =~ /Host Name/) {
                @raw[$rc] =~ /\:/;
                $fqdn = substr($',1,-1);
        }
        $rc++;
}


# Clear the counter
$rc = 0;
# Find the primary DNS suffix
while (defined(@raw[$rc])) {
        if (@raw[$rc] =~ /Primary DNS Suffix/) {
                @raw[$rc] =~ /\:/;
                $fqdn .= "." . substr($',1,-1);
        }
        $rc++;
}


# Update the machine name in the local machine-tracking file with the FQDN


# Read in the machine tracking-update file
open (MACHINEFILE, $currentTrackingUpdate) || die scalar(localtime) . " Error opening
machine tracking file, $currentTrackingUpdate: $!\n";
# Clear the raw array before using it
@raw = "";
@raw=<MACHINEFILE>;
close MACHINEFILE;

################################################################
#                                                              #
# The format of a record (row) in the machine-tracking file is:#
# %COMPUTERNAME% Day-of-week Date Time %MACHINEROLE%           #
#                                                              #
# These fields are delimited by spaces.  The split() function  #
# splits the record into its component fields and assigns each #
# to a PERL variable.  At this point, the fields will be       #
# written back out the the file which will be overwriten with  #
# an updated version containing the FQDN:                      #
# FQDN Day-of-week Date Time %MACHINEROLE%                     #
#                                                              #
################################################################

($machineTrackingName, $day, $date, $time, $machineRole) = split (' ', @raw[0]);

# Overwrite the machine-tracking update file with the FQDN replacing the
# NetBIOS computer name
open (MACHINEFILE, ">" . $currentTrackingUpdate) || die scalar(localtime) . " Error
opening machine tracking file, $currentTrackingUpdate: $!\n";
print MACHINEFILE "$fqdn $day $date $time $machineRole\n";
close MACHINEFILE;
```

LOCALAUDIT.CMD

```
@ECHO OFF

REM     LOCALAUDIT.CMD
REM
REM     Version 1.5
REM     Last revised 8/11/1
REM
REM This script is executed daily at 4:00 by the Task Scheduler.
REM
REM This script:
REM             Creates an update record for the machine-tracking file
REM             Provides the information so that the data files use
REM             the Fully Qualified Domain Name (FQDN)
REM             Performs the local audit


REM Switch to the audit directory
C:
CD\HIDDEN\AUDIT


REM Clean up old data files from previous iteration
DEL "C:\HIDDEN\AUDIT\*.raw" > NUL 2>NUL
DEL "C:\HIDDEN\AUDIT\*.log" > NUL 2>NUL

REM Create an update for the machine-tracking file on the local machine.
REM The collection script will use this information to update the
REM machine-tracking file on the audit server.
REM The local data is overwriten every time the audit is run.
REM The audit server preserves the local data.
ECHO %COMPUTERNAME% %DATE% %TIME% %MACHINEROLE% > %COMPUTERNAME%.log 2> NUL

REM Create a file containing the output from the IPCONFIG /ALL command.
REM This information is used to update the machine-tracking file with
REM the fully qualified domain name (FQDN) of the machine.
IPCONFIG /ALL > C:\HIDDEN\AUDIT\fqdn.raw 2> NUL

REM Update the machine-tracking update file to replace the NetBIOS computer
REM name with the FQDN
C:\HIDDEN\AUDIT\perl C:\HIDDEN\AUDIT\recordfqdn.pl C:\HIDDEN\AUDIT\%COMPUTERNAME%.log
C:\HIDDEN\AUDIT\fqdn.raw > NUL 2>NUL


REM Use the Microsoft Security Configuration and Analysis command line tool
REM to generate the raw audit data
SECEDIT /ANALYZE /DB "C:\HIDDEN\AUDIT\audit_%MACHINEROLE%.sdb" /CFG
"C:\HIDDEN\AUDIT\\audit_%MACHINEROLE%.inf" /LOG "C:\HIDDEN\AUDIT\%COMPUTERNAME%.raw"
/VERBOSE > NUL 2> NUL

REM Note that the audit template is specific for the machine role,
REM %MACHINEROLE%.  If %MACHINEROLE% is not set, the machine will be audited
REM against the domain controller audit template


REM Add the enviromental variables to the data file, convert it to ANSI text
REM and give it the proper name.
C:\HIDDEN\AUDIT\perl C:\HIDDEN\AUDIT\localaudit.pl %COMPUTERNAME% %SYSTEMDRIVE%
%SYSTEMROOT% C:\HIDDEN\AUDIT C:\HIDDEN\AUDIT\fqdn.raw > NUL 2>NUL
```

LOCALAUDIT.PL

```
#
#       LOCALAUDIT.PL
#
#       Version 1.4
#       Last revised 8/11/1
#
# This script:
#       Converts the output of the SECEDIT tool from Unicode to ANSI
#       Adds the Fully Qualified Domain Name (FQDN), %SYSTEMROOT% and
#       %SYSTEMDRIVE% to the beginning of the raw data file.
#       The FQDN is derived from the output of the IPCONFIG /ALL command.
#       Creates the properly datestamped file name
#       Creates a file with the batch file code to set the %DATESTAMP% variable
#
# The script takes as input:
#       The %COMPUTERNAME% as the first command line variable
#       The %SYSTEMDRIVE% as the second command line variable
#       The %SYSTEMROOT% as the third command line variable
#       The current directory as the fourth command line variable
#       The file containing the output from the  IPCONFIG /ALL command
#              as the fifth command line variable


my $dateStamp;         # The datestamp in the proper format
my $machineName;       # The %COMPUTERNAME% variable from the command line
my $systemDrive;       # The %SYSTEMDRIVE% variable from the command line
my $systemRoot;              # The %SYSTEMROOT% variable from the command line
my $sourceFilename;    # The raw data file
my $destFilename;      # The destination data file
my @raw;               # Array containing the raw or source data
my @filtered;          # Array containing the filtered or final
my $rc;                # Counter, generally used for the raw array
my $ic;                # Counter, generally used for inner loops
my $fc;                # Counter, generally used for the filtered array
my $currentDir;              # Current directory
my $stampFilename;     # The name of the file containing the code to set the
                       # %DATESTAMP% environmental variable

# Read the Windows environmental variables into PERL variables
$machineName = @ARGV[0];
$systemDrive = @ARGV[1];
$systemRoot = @ARGV[2];

# Set the current directory
$currentDir = @ARGV[3];

# The source data file will always be the same for each computer.  This file
# is deleted by the calling batch file when the PERL script is done using it
# to prevent the same input file from being used for multiple audit reports.
$sourceFilename = $currentDir ."\\" . $machineName . ".raw";

# Get the datestamp in the proper format
$dateStamp = &dateAsString(time());


# Set the ipconfig output filenames
$ipconfigOutput = @ARGV[4];

###############################################################################
#                                                                             #
# Open the source data file and read it into the raw array for processing     #
# The || die command is read "or" die.  In PERL logical operators can be used #
# to link certain commands.  These commands return some type of value when    #
# they are used.  The return value often does not need to be used for         #
# anything.  For instance, the open() function returns a true value if the    #
# file open succeeds and a false value if the file open fails.  Since PERL    #
```

```
# stops evaluating a logical construct as soon as it is either true or false   #
# without a doubt, the second command will only ever execute if the first one  #
# is false.  This is because as long as the first component of an OR statement #
# is true, the entire statement will be true.  If the first component of the   #
# OR statement is false, the second component will have to be evaluated in     #
# order to determine the outcome of the logical statement.                     #
# In this case, the script exits with an error if it can't open the input file.#
#                                                                              #
################################################################################

################################################################################
#                                                                              #
# Read in the host name and the primary suffix name from a text file containing #
# the output from the IPCONFIG /ALL command                                    #
#                                                                              #
################################################################################

open (SOURCEFILE, $ipconfigOutput) || die scalar(localtime) . " Error opening input file:
$ipconfigOutput: $!\n";
@raw=<SOURCEFILE>;
close SOURCEFILE;

# Clear the counter
$rc = 0;
# Find the host name
while (defined(@raw[$rc])) {
        if (@raw[$rc] =~ /Host Name/) {
                @raw[$rc] =~ /\:/;
                $fqdn = substr($',1,-1);
        }
        $rc++;
}



# Clear the counter
$rc = 0;
# Find the primary DNS suffix
while (defined(@raw[$rc])) {
        if (@raw[$rc] =~ /Primary DNS Suffix/) {
                @raw[$rc] =~ /\:/;
                $fqdn .= "." . substr($',1,-1);
        }
        $rc++;
}



# Create the name of the destination file in the proper format
$destFilename = $currentDir . "\\" . $dateStamp . "-" . $fqdn . ".raw";


# Open and read the unprocessed output from the SECEDIT tool
open (SOURCEFILE, $sourceFilename) || die scalar(localtime) . " Error opening input file:
$sourceFilename: $!\n";
@raw=<SOURCEFILE>;
close SOURCEFILE;

# Place a carriage return (\n) in the last line of the array read by the file as an end
of line marker.
$rc = 0;
# Move to the end of the array
while (defined(@raw[$rc++])) {;}
# Place the end of line marker on the last line containing text
@raw[$rc-2] .= "\n";


# Clear the filtered array counter
$fc = 0;

# Write the file description and the local environmental information to the
```

```
# filtered array. The contents of the filtered array will become the
# destination file
@filtered[$fc++]="Raw data file";
@filtered[$fc++]="$fqdn";
@filtered[$fc++]="$systemDrive";
@filtered[$fc++]="$systemRoot";

################################################################################
#                                                                              #
# Filter extra spaces out of raw report data                                   #
# The raw data is Unicode.  Text represented in Uunicode has 0 for the high    #
# byte, therefore PERL interprets this as a space.  Therefore, every other     #
# character must be filtered out.                                              #
# The methodology used:                                                        #
#       For each line in the array                                             #
#       Start on the second character of a line (the first is always a space)  #
#       Copy every character whose position within the line is odd.            #
#               This is accomplished by using the modulus function.            #
#               The modulus function divides the number and returns the        #
#               remainder.  Any number modulus 2 which returns a remainder     #
#               is odd.                                                        #
#       Use the newline (\n) to indicate the end of a line                     #
#                                                                              #
################################################################################

# Set the raw array counter to zero
$rc = 0;

# Step through each line in the array
while (defined(@raw[$rc])) {
        # Set the character position within the line counter to 1, since the
        # first is always a space
        $ic = 1;
        # Each character is isolated using the substring function and tested
        # against the newline character (\n)
        while ((substr(@raw[$rc],$ic,1) ne "\n")) {
                # % is the modulus operator.  This expression is true when $ic
                # is odd
                if ($ic%2) {
                        # Copy the character to the end of the filtered array
                        # entry for this line
                        @filtered[$fc] = @filtered[$fc] . substr(@raw[$rc],$ic,1);
                        # Increment the character position counter
                        $ic+=1;
                }
                else {
                        # This branch is reached on the characters in even
                        # positions which are the extra space in the Unicode
                        # Format
                        # Increment the character position counter
                        $ic++;
                }
        }
        # Place a newline at the end of the filtered array to show the end
        # of the line
        @filtered[$rc] .= "\n";
        # Increment the raw and filtered array counters
        $rc++;
        $fc++;
}

# Open the audit data destination file
open (DESTFILE, ">" . $destFilename) || die scalar(localtime) . " Error opening
destination file, $destFilename: $!\n";
# Write the filtered array to the destination file
print DESTFILE @filtered;
# Close the destination file
close (DESTFILE);

################################################################################
#                                                                              #
```

```
#                                 Subroutines                                  #
#                                                                              #
################################################################################


sub dateAsString {
        # This function takes input from the time() function or a number that
        # indicates the number of seconds since 1900 and converts it
        # into a date string of format: yyyy-mm-dd.
        # Calling dateasstring() with time() as the input, returns the
        # present date.
        # You can get other dates by manipulating the value returned by time
        # by adding or subtracting the number of seconds in a day or
        # a week and then calling dateasstring().
        # 60s/m * 60m/h * 24h/d * 7d/w = 604800 seconds in a week.
        # 60s/m * 60m/h * 24h/d = 86400 seconds in a day.

        my $passedtime = $_[0];                       # Time passed by the calling program
        my @timearray=localtime($passedtime); # Time stored in a standard array format
        my $date;                             # The return datestamp

        ###########################################################################
        #                                                                         #
        # The standard time array format is structured                            #
        # @timearray[0] = second                                                  #
        # @timearray[1] = minute                                                  #
        # @timearray[2] = hour                                                    #
        # @timearray[3] = day of the month                                        #
        # @timearray[4] = month, 0 = January                                      #
        # @timearray[5] = years since 1900                                        #
        # @timearray[6] = day of week, 0 = Sunday                                 #
        # @timearray[7] = day of year                                             #
        # @timearray[8] = daylight savings time, 0 = no, 1 = yes                  #
        #                                                                         #
        ###########################################################################

        #Add the year to the datestamp
        $date=($timearray[5]+1900);
        $date .= "-";

        #Add the month to the datestamp
        if ((@timearray[4]+1) < 10){
                #Add a leading 0 if the month is a single digit
                $date .= "0" . (@timearray[4]+1) ."-";
        }
        if ((@timearray[4]+1) > 9){
                $date .= (@timearray[4]+1) ."-";
        }

        #Add the day of the month to the datestamp
        if (@timearray[3] < 10){
                #Add a leading 0 if the day is a single digit
                $date .= "0" . ($timearray[3]);
        }
        if (@timearray[3] > 9){
                $date .= (@timearray[3]);
        }
        #Return the datestamp
        return ($date);
}
```

COLLECT.CMD

```
@ECHO OFF

REM     COLLECT.CMD
REM
REM     Version 1.1
REM     Last revised 8/11/1
REM
REM This script runs at 5:00 AM daily
REM This script is run by the Windows Task Scheduler on the audit server.
REM
REM This script collects the audit files from the local machines.
REM First it cleans up the files from the previous iteration.  Next, it
REM uses a PERL script to build a list of machines from the machine-tracking
REM files.  The PERL script then builds a temporary batch containing the
REM commands to actually copy the data to the audit server.
REM
REM
REM

SET AUDITSCRIPTS= \\blazing-sroot.sroot.realm.org\auditscripts
SET AUDITLOGS= \\blazing-sroot.sroot.realm.org\auditlogs


REM Clean up the batch file that actually collects the data from the previous
REM iteration.
DEL /Y %AUDITLOGS%\getdata.cmd  > NUL 2> NUL

REM Utilize the COLLECT.PL PERL script to build the actual data collection
REM script using the computer names from the machine-tracking filenames.
%AUDITSCRIPTS%\PERL.EXE collect.pl %AUDITSCRIPTS% %AUDITLOGS%  >> %AUDITLOGS%\COLLECT-
ERRORS.LOG 2> NUL


REM Execute the batch file to collect the data
REM This batch file relies on DNS name resolution
%AUDITLOGS%\getdata.cmd >> %AUDITLOGS%\COLLECT-ERRORS.LOG 2> NUL
```

COLLECT.PL

```
#
#       COLLECT.PL
#
#       Version 1.1
#       Last revised 8/12/1
#
# This script:
#       Collects the computer names names of all the machine in the machine
#       tracking directory and uses this information to build the GETDATA.CMD
#       file.  The purpose of the GETDATA.CMD file is to collect the data off
#       all of the machines with machine-tracking files in the machine-tracking
#       directory.


my $dateStamp;           # The datestamp in the proper format
my $machineName;         # The NetBIOS name of the local system whose files are
                         # being processed
my @raw;                 # Array containing the raw or source data
my @filtered;            # Array containing the filtered or final data
my $rc;                  # Counter, generally used for the raw array
my $mc;                  # Counter, used to determine which local machine the
                         # data is being collected from at any given time
my $fc;                  # Counter, generally used for the filtered array
my $auditScripts;        # The the directory where the audit scripts and
                         # support files are located
my $auditLogs;           # The the directory where the audit logs are stored
my $tracking;            # The the directory where the machine-tracking files
                         # are stored
my @rawTrackingFiles;    # The filenames of all the files in the
                         # machine-tracking directory
my $trackingFile;        # The name of the mahcine-tracking file currently
                         # being processed
my @unsortedMachineNames; # The filtered machine-tracking filenames, unsorted
my @machineName;         # The filtered machine-tracking filenames, sorted
my $machineTrackingName;  # The Fully Qualified Domain Name of the machine
my $day;                 # The day of the machine-tracking update
my $date;                # The date of the machine-tracking update
my $time;                # The time of the machine-tracking update
my $machineRole;         # The %MACHINEROLE% from the machine-tracking update



# Read the Windows environmental variables for the audit scripts and audit logs
# directories into PERL variables
$auditScripts = @ARGV[0];
$auditLogs = @ARGV[1];



# Process the machine-tracking files

# The machine-tracking files are in the machinelist subdirectory of the $auditLogs
# directory by design
# Set the correct directory for the machine-tracking files
$tracking = $auditLogs . "\\machinelist";


#############################################################################
#                                                                           #
# The || die command is read "or" die.  In PERL logical operators can be used  #
# to link certain commands.  These commands return some type of value when     #
# they are used.  The return value often does not need to be used for anything.#
# For instance, the opendir() function returns a true value if the directory   #
# open succeeds and a false value if the directory open fails.                 #
# Since PERL stops evaluating a logical construct as soon as that contstruct   #
```

```
# is either true or false without a doubt, the second command will only ever   #
# execute if the first one is false.  This is because as long as the           #
# first component of an OR statement is true, the entire statement will be     #
# true.  If the first component of the OR statement is false, the second       #
# component will have to be evaluated in order to determine the outcome of the #
# logical statement.                                                           #
# In this case, the script exits with an error if it can't open the directory  #
# containing the machine-tracking files                                        #
#                                                                              #
################################################################################

# Open machine-tracking files directory
opendir (SOURCEDIR, $tracking) || die scalar(localtime) . " Error opening directory to
get input files, $tracking: $!\n";

# Read a list of all the files in the directory into the raw array
@rawTrackingFiles=readdir(SOURCEDIR);
close SOURCEDIR;


# The list of machine-tracking files without the .log extension is the same
# as the list of Fully Qualified Domain Names of machines that have run
# SITE.CMD via the site group policy startup script.

$rc = 0;
$fc = 0;
# Step through the file list
while (defined(@rawTrackingFiles[$rc])){
        # Filter the . and .. designators from the list
        if (!(@rawTrackingFiles[$rc] eq "." || @rawTrackingFiles[$rc] eq "..")) {
                # Assign the filename without the extension to the list
                @unsortedMachineNames[$fc]=substr(@rawTrackingFiles[$rc],0,-4);
                $fc++;
        }
        $rc++;
}

@machineName = sort @unsortedMachineNames;

$mc = 0;


# Create the batch file GETDATA.CMD

# Set the output file name
$destFilename = $auditLogs . "\\getdata.cmd";

# Open the batch command destination file
open (DESTFILE, ">" . $destFilename ) || die scalar(localtime) . " Error opening
destination file, $destFilename: $!\n";

while (defined(@machineName[$mc])) {
        # Get the %MACHINEROLE% of the local machine by reading its local
        # machine-tracking update file
        $trackingFile = "\\\\" . @machineName[$mc] . "\\HIDDEN" . '$' . "\\" .
@machineName[$mc] . ".log";
        # There is no need to exit the program if any single machine can not
        # be properly audited, therefore the PRINT command is used instead of
        # the DIE command
        open (MACHINEFILE, $trackingFile ) || print scalar(localtime) . " Error opening
machine tracking file, $trackingFile: $!\n";
        # Clear the raw array before using it
        @raw = "";
        @raw=<MACHINEFILE>;
        close MACHINEFILE;
        ################################################################
        #                                                              #
        # The format of a record (row) in the machine-tracking file is:#
        # %COMPUTERNAME% Day-of-week Date Time %MACHINEROLE%           #
        #                                                              #
        # These fields are delimited by spaces.  The split() function  #
```

```
        # splits the record into its component fields and assigns each #
        # to a PERL variable.  The only fields used in this script are #
        # Date and %MACHINEROLE%.                                      #
        #                                                              #
        ################################################################
        ($machineTrackingName, $day, $date, $time, $machineRole) = split (' ', @raw[0]);

        # The date stamp is of the format mm/dd/yyyy in the machine-tracking
        # file. The following command translates it into the standard format
        # for the audit system yyyy-mm-dd
        $dateStamp = substr($date,-4) . "-" . substr($date,0,2) . "-" . substr($date,3,2);

        ################################################################################
        #                                                                              #
        # PERL has extremely powerful text matching and handling functions.    #
        # the $variable =~ /pattern/ command will find the first occurence of  #
        # "pattern" in $variable.                                              #
        # Moreover, once the pattern is found, everything in $variable before  #
        # "pattern" is assigned to the built-in variable $` and everything     #
        # after "pattern" is assigned to the built-in variable $'  This makes  #
        # it very easy to separate the two data fields into separate variables.#
        #                                                                      #
        # In this case, the extension is being romved from a filename, leaving #
        # only the part of the filename before the dot.a hash is created.      #
        # Since the dot has special meaning in the search sequence, its        #
        # special significance is nullified by "escaping" it using the         #
        # backslash character.                                                 #
        #                                                                      #
        ########################################################################
        $machineTrackingName =~ /\./;
        $machineName = $`;


        ########################################################################
        #                                                                      #
        # To avoid using the system() call and using operating system specific #
        # commands in PERL scripts, the commands to move the audit data to the #
        # audit server are built into a batch file, GETDATA.CMD, which is run  #
        # by the calling program, COLLECT.CMD.                                 #
        # Many of the operating system symbols, such as \ and $ have special   #
        # significance within PERL.  These symbols either had to be "escaped   #
        # or if they were a part of a PERL variable, concatenated with the     #
        # output text.                                                         #
        #                                                                      #
        ########################################################################

        # Write the batch commands to the file
        print DESTFILE "REM Check startup scripts and support files, copying them from the
audit \n";
        print DESTFILE "REM server if necessary\n";
        print DESTFILE "REM While this is not expected to be necssary, it will ease the
impact on \n";
        print DESTFILE "REM possible file replacement while processing the Site Group
Policy startup \n";
        print DESTFILE "REM script if any of the audit system files needed to be
updated.\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\localaudit.cmd (copy /y $auditScripts\\localaudit.cmd \\\\" . $machineTrackingName .
"\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\perl.exe (copy /y $auditScripts\\perl.exe \\\\" . $machineTrackingName . "\\HIDDEN" .
'$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\perlcrt.dll (copy /y $auditScripts\\perlcrt.dll \\\\" . $machineTrackingName .
"\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\perlcore.dll (copy /y $auditScripts\\perlcore.dll \\\\" . $machineTrackingName .
"\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\attest.pl (copy /y $auditScripts\\attest.pl \\\\" . $machineTrackingName . "\\HIDDEN"
. '$' . ") > NUL 2> NUL\n";
```

```
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\y-with-crlf.txt (copy /y $auditScripts\\y-with-crlf.txt \\\\" . $machineTrackingName .
"\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\audit_" .$machineRole . ".inf (COPY /Y $auditScripts\\audit_" .$machineRole . ".inf
\\\\" . $machineTrackingName . "\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\audit_" .$machineRole . ".sdb (COPY /Y $auditScripts\audit_" .$machineRole . ".sdb
\\\\" . $machineTrackingName . "\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\localaudit.pl (COPY /Y $auditScripts\\localaudit.pl \\\\" . $machineTrackingName .
"\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\checktracking.pl (COPY /Y $auditScripts\\checktracking.pl \\\\" . $machineTrackingName
. "\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\recordfqdn.pl (COPY /Y $auditScripts\\recordfqdn.pl \\\\" . $machineTrackingName .
"\\HIDDEN" . '$' . ") > NUL 2> NUL\n";
        print DESTFILE "IF NOT EXIST \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\subinacl.exe (COPY /Y $auditScripts\\subinacl.exe \\\\" . $machineTrackingName .
"\\HIDDEN" . '$' . ") > NUL 2> NUL\n";

        print DESTFILE "\n";

        print DESTFILE "REM Y-WITH-CRLF.TXT is simply a text file containing y and a
carriage \n";
        print DESTFILE "REM return/linefeed.  Feeding a \"y\" followed by a carriage
return/linefeed\n";
        print DESTFILE "REM into standard input is the same as typing it at the
keyboard.\n";
        print DESTFILE "REM This technique is used to force the CACLS command to complete
without\n";
        print DESTFILE "REM user intervention.\n";
        print DESTFILE "REM The user with permissions is the account the Windows Task
Scheduler runs \n";
        print DESTFILE "REM under on the audit server.\n";
        print DESTFILE "CACLS \\\\" . $machineTrackingName . "\\HIDDEN\$\\*.* /G SYSTEM:F
SROOT.REALM.ORG\AuditCollect:F /D EVERYONE < $auditScripts\\y-with-crlf.txt > NUL
2>NUL\n";
        print DESTFILE "\n";

        print DESTFILE "REM Move the local audit file to the audit server\n";
        print DESTFILE "XCOPY /Z /Y /V \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\" . $dateStamp . "-" . $machineTrackingName . ".raw $auditLogs > NUL 2> NUL\n";
        print DESTFILE "\n";

        print DESTFILE "REM The machine-tracking file is only updated if the local audit
completed and\n";
        print DESTFILE "REM moved data to the audit server.\n";
        print DESTFILE "REM Update the machine-tracking file with the current date\n";
        print DESTFILE "IF EXIST \"";
        print DESTFILE $auditLogs . "\\" . $dateStamp . "-$machineTrackingName.raw\" ";
        print DESTFILE " (\n";
        print DESTFILE "       TYPE  \\\\" . $machineTrackingName . "\\HIDDEN" . '$' .
"\\" . $machineName . ".log >> \"" . $auditLogs . "\\machinelist\\" .
$machineTrackingName . ".log\" 2> NUL\n";
        print DESTFILE "       )\n\n\n";

        # Increment the machine name counter
        $ mc++;
}

# Close the destination file
close (DESTFILE);
perl auditreport.pl -d:"C:\My Documents\STUDY\GIAC\Scripts\reportdefinitions.txt" -
i:"\\blazing-sroot.sroot.realm.org\auditlogs" -o:"\\blazing-
sroot.sroot.realm.org\auditreports"
```

AUDITREPORT.CMD

```
@ECHO OFF

REM   AUDITREPORT.CMD
REM
REM   Version 1.1
REM   Last revised 8/12/1
REM
REM This script runs at 8:00 AM daily
REM This script is run by the Windows Task Scheduler on the audit
server.
REM
REM This script use the audit files and the machine-tracking files
stored on
REM the audit server along with the Report Definitions file to create
REM Summary Reports and Exception Reports.
REM
REM

perl auditreport.pl -d:"\\blazing-
sroot.sroot.realm.org\auditscripts\reportdefinitions.txt" -
i:"\\blazing-sroot.sroot.realm.org\auditlogs" -o:"\\blazing-
sroot.sroot.realm.org\auditreports"
```

AUDITREPORT.PL

```
#
#       AUDITREPORT.PL
#
#       Version 1.4
#       Last revised 8/12/1
#
# This script takes the data file directory, the report output directory
#  and the exception definitions file as command line arguments.
# The usage for this script is:
#
# perl auditreport -i:input-directory -o:output-directory -d:report-definitions
#
#        input-directory:        the complete path to the directory
#                                containing the raw audit files
#        output-directory:       the complete path to the directory where
#                                the reports will be created
#        report-definitions:     the complete path and filename of the report
#                                definitions file
#
# The audit system expects to have the following directory structure
#
#  input-directory (This directory contains the audit data and is hard
#    |               coded in the localaudit.cmd, site.cmd and auditreport.cmd
#    |               scripts.  It is passed to this script to avoid putting
#    |               operating system-specific information in the script.)
#    |
#    |------machinelist (This contains the machine-tracking files.)
#
#
#  output-directory (This directory contains the audit reports after they
#                    have been created.  This directory is not referenced in
#                   any file except auditreport.cmd.  It is passed to this
#                   script to avoid putting operating system-specific
#                   information in the script.)
#
#  report-definitions  (This directory contains the report definitions used
#                       to provide user-friendly error messages in the reports.
#                      This directory is not referenced in any file except
#                      auditreport.cmd.  It is passed to this script to avoid
#                     putting operating system-specific information in the
#                     script.)
#
#
# This script:
#       Reads in each machine-tracking file and creates a machine-tracking
#          report in the Summary Report
#       Reads in each machine-specific audit data file and creates a entry
#          in the Summary Report.  If the machine failed to comform to the
#          audit policy, a Exception Report is created for that machine and
#          linked to the Summary Report using a hyperlink
#       Formats all reports using HTML
#       Tracks the length of time it takes to run


my @raw;          # Input array
my @filtered;     # Filtered array
my @exceptions;    # Array of audit failures
my @inputFileList; # Array containing the names of the raw audit files
my $temp;         # A temporary holding variable
my $rc;           # Raw array counter
my $ic;           # Inner array counter
my $fc;           # Filtered array counter
my $ec;           # Exceptions array counter
my $ac;           # Audit file counter
my $sa;           # Password policy report definitions counter
```

```
my $lo;           # Log settings report definitions counter
my $ea;           # Audit policy report definitions counter
my $pr;           # Security rights report definitions counter
my $rk;           # Registry permissions report definitions counter
my $rv;           # Registry settings report definitions counter
my $fs;           # File permissions report definitions counter
my $gm;           # Restricted groups report definitions counter
my $sg;           # System services report definitions counter

my $input;        # Name of the input data directory
my $output;       # Name of the output report directory
my $definition;          # Name of the report definition file
my $tracking;     # Name of the tracking directory
my $startTime;    # Time that the script started
my $endTime;      # Time that the script ended
my $elapsedTime;   # Elapsed time that the script ran

my %sah;          # Password policy report definitions hash
my %loh;          # Log settings report definitions hash
my %eah;          # Audit policy report definitions hash
my %prh;          # Security rights report definitions hash
my %rkh;          # Registry permissions report definitions hash
my %rvh;          # Registry settings report definitions hash
my %fsh;          # File permissions report definitions hash
my %gmh;          # Restricted groups report definitions hash
my %sgh;          # System services report definitions hash

my @sas;          # Password policy exception report section array
my @los;          # Log settings report exception report section array
my @eas;          # Audit policy report exception report section array
my @prs;          # Security rights report exception report section array
my @rks;          # Registry permissions report exception report section array
my @rvs;          # Registry settings report exception report section array
my @fss;          # File permissions report exception report section array
my @gms;          # Restricted groups report exception report section array
my @sgs;          # System services report exception report section array

my $oc;                       # The index of the record on which the last
                              # successful audit completed
my $la;                       # Used to index the array of error messages for
                              # machines that have not run the audit on the
                              # current date
my @rawTrackingFiles;         # All the files in the machine-tracking
                              # directory
my @trackingFileList;         # The sorted machine-tracking files
my @unsortedTrackingFiles;    # The unsorted machine-tracking files
my $targetDate;                     # The current date, used in machine-tracking report
my $day;                      # The day of the week, used in machine-tracking report
my @date;                     # The date, used in machine-tracking report
my $time;                     # The time, used in machine-tracking report
my @month;                    # The month, used in machine-tracking report
my @day;                      # The day of the month, used in machine-tracking report
my @year;                     # The year, used in machine-tracking report
my @machineRole;              # The machine role, used in machine-tracking report
my @machineTrackingName;      # The %COMPUTERNAME%, used in machine-tracking report
my @machineNotCurrent;        # Machine names of machines that did not run
                              # the audit on the current date
my $auditOccuredOnCurrentDate;        # A flag indicating that the last successful
                              # audit occurred on the current date
my %roleNotSet;                       # A hash of %COMPUTERNAME% and an error
                              # message for all systems whose
                              # machine-tracking files indicated that the
                              # %MACHINEROLE% was improperly set
my @lastAudit;                # A list of error messages for each machine
                              # that did not successfully completethe audit
                              # on the current date.
my %validRoles;                       # A hash containing the valid roles in lower
                              # case as the "key" and a non-zero, non-null
                              # value as the "value"

# Record the start time of the script
```

```
$startTime=time();

#Check command line arguments and assign them to local variables
# Clear counters and directory and file name to none
$rc=0;
$input="none";
$output="none";
$definition="none";

# Step through the command line arguments
while (defined(@ARGV[$rc])) {
        # If the command line argument starts with -i: it is an input directory
        if (@ARGV[$rc]=~ /^-i:/) {
                # See the note below on the usage of ||
                # The incorrectUsage() function will generate the specific
                # error message, prompt the user with the proper usage and
                # exit the script
                $input eq "none" || &incorrectUsage("Too many input directories");
                # The PERL variable $' means everything in a string following
                # the match in this case it would strip the -i: from the
                # directory name
                $input = $';
        }
        # If the command line argument starts with -o: it is an output directory
        # See the note below on the usage of ||
        if (@ARGV[$rc]=~/^-o:/) {
                $output eq "none" || &incorrectUsage("Too many output directories");
                $output = $';
        }
        # If the command line argument starts with -d: it is a definition file
        # See the note below on the usage of ||
        if (@ARGV[$rc]=~/^-d:/) {
                $definition eq "none" || &incorrectUsage("Too many definition files");
                $definition = $';
        }
        $rc++;
}

# Check to make sure that an input directory, an output directory and a
# report definitions file were specified
$input ne "none" || &incorrectUsage("No input directory specified.");
$output ne "none" || &incorrectUsage("No input directory specified.");
$definition ne "none" || &incorrectUsage("No definition file specified.");


############################
#                          #
# Machine Tracking Report  #
#                          #
############################

###############################################################################
#                                                                             #
# The Machine Tracking Report is part of the Summary Report.  It is made up of #
# two parts.  The first part of the Machine Tracking Report is an indication   #
# which appears at the end of records in the audited machines summary if that  #
# system did not have the %MAHCINEROLE% variable defined.  It is an indication #
# of an audit failure, but will not necessarily cause an Execption Report to   #
# be generated for that machine.  However, since systems without the           #
# %MACHINEROLE% variable correctly defined are audited against the strictest   #
# audit policy template, the Domain Controller template, it is likely that     #
# these systems will also generate an Exception Report.                        #
#                                                                              #
# The second part of the Machine Tracking Report is a list of systems that     #
# have provided audit data to the audit server in the past, but failed to do   #
# so on the current date.  The date of the last successful transmission of an  #
# audit data file audit data file to the audit server is listed for these      #
# machines.  Additionally, if the %MACHINEROLE% is set incorrectly on these    #
# machines, it will be indicated in this part of the report.                   #
#                                                                              #
# The information about the machines that ran audits on the current date, but  #
```

```
# have their %MACHINEROLE% set incorrectly is stored in the %roleNotSet hash.  #
# This information is included with the entry for that machine in the          #
# Summary report when it is generated.                                         #
#                                                                              #
# The information about machines that have run the audit before, but did not   #
# run the audit on the current date is stored in the @lastAudit array.         #
#                                                                              #
################################################################################


# Set up the hash of valid roles.  Any valid role should return a non-zero or
# non-null value.  Other roles could be added here
%validRoles= ('dc'=>'1','workstation'=>'1','server'=>'1');

# Process the machine-tracking files

# The machine-tracking files are in the machinelist subdirectory of the $input
# directory by design
# Set the correct directory for the machine-tracking files
$tracking = $input . "\\machinelist";


################################################################################
#                                                                              #
# The || die command is read "or" die.  In PERL logical operators can be used  #
# to link certain commands.  These commands return some type of value when     #
# they are used.  The return value often does not need to be used for anything.#
# For instance, the opendir() function returns a true value if the directory   #
# open succeeds and a false value if the directory open fails.                 #
# Since PERL stops evaluating a logical construct as soon as that contstruct   #
# is either true or false without a doubt, the second command will only ever   #
# execute if the first one is false.  This is because as long as the           #
# first component of an OR statement is true, the entire statement will be     #
# true.  If the first component of the OR statement is false, the second       #
# component will have to be evaluated in order to determine the outcome of the #
# logical statement.                                                           #
# In this case, the script exits with an error if it can't open the directory  #
# containing the machine-tracking files                                        #
#                                                                              #
################################################################################

# Open machine-tracking files directory
opendir (SOURCEDIR, $tracking) || die scalar(localtime) . " Error opening directory to
get input files, $tracking: $!\n";

# Read a list of all the files in the directory into the raw array
@rawTrackingFiles=readdir(SOURCEDIR);
close SOURCEDIR;


# Sorting the file names creates sorted output in the Summary Report
# since the input data is already sorted

$rc = 0;
$fc = 0;
# Step through the file list
while (defined(@rawTrackingFiles[$rc])){
        # Filter the . and .. designators from the list
        if (!(@rawTrackingFiles[$rc] eq "." || @rawTrackingFiles[$rc] eq "..")) {
                @unsortedTrackingFiles[$fc]=@rawTrackingFiles[$rc] . "\n";
                $fc++;
        }
        $rc++;
}

@trackingFileList = sort @unsortedTrackingFiles;

# Get target (current) date
$targetDate = &dateAsString(time());

# Clear the @raw array.  If the array is not cleared, when the data from the
```

```
# machine-tracking file is read into the array, it will be appended to the
# data from the previous machine-tracking file.
@raw = "";

# Clear the machine-tracking file list counter
$ac = 0;

# Clear the last audit array counter.
$la = 0;


# Step through the file list
while (defined(@trackingFileList[$ac])){
        $inputFile = $tracking . "\\" . @trackingFileList[$ac];
        # Open the machine-tracking file
        open (SOURCEFILE, $inputFile) || print "Error opening machine-tracking file:
$inputFile: $!\n";
        # Read the data into the @raw directory
        @raw=<SOURCEFILE>;
        close SOURCEFILE;

        # Clear the record counter
        $rc = 0;
        # Step through the records read in from the machine-tracking file
        while (defined(@raw[$rc])) {
                ###################################################################
                #                                                                 #
                # The format of a record (row) in the machine-tracking file is:#
                # %COMPUTERNAME% Day-of-week Date Time %MACHINEROLE%           #
                #                                                                 #
                # These fields are delimited by spaces.  The split() function  #
                # splits the record into its component fields and assigns each #
                # to a PERL variable.  The fields that are important are:      #
                # %COMPUTERNAME%, Date and %MACHINEROLE%.  Furthermore, the    #
                # Date field must is in the format mm/dd/yyyy and must be split#
                # into Day-of-month, Month and Year fields.  The second split()#
                # accomplishes this requirement.                               #
                #                                                                 #
                # There is a record written the the machine-tracking file upon #
                # each successful completion of the audit process, which       #
                # includes moving the audit file to the audit server.          #
                # Therefore, the relevant fields must be stored in arrays in   #
                # order to determine the last time the audit completed         #
                # successfully and whether the %MACHINEROLE% was set correctly #
                # at that time.                                                #
                #                                                                 #
                # The data in the machine-tracking file could also be used for #
                # trend analysis or troublshooting the audit process.          #
                #                                                                 #
                ###################################################################
                (@machineTrackingName[$rc], $day, @date[$rc], $time, @machineRole[$rc]) =
split (' ', @raw[$rc]);
                (@month[$rc],@day[$rc], @year[$rc]) = split (/\//,@date[$rc]);

                # Convert the machine role and machine-tracking name to lower
                # case to simplify matching
                @machineRole[$rc] = lc(@machineRole[$rc]);
                @machineTrackingName[$rc] = lc(@machineTrackingName[$rc]);

                # Increment the record counter
                $rc++;
        }



        ###########################################################################
        #                                                                         #
        # When an array in PERL is evaluated as a scalar (single value)         #
        # variable, it returns the number of objects in the array.              #
        # Since there is an object in the month, day, year, machineName and     #
        # machineRole arrays for each instance of the raw array, the number of #
```

```
        # objects in the raw array is the number of iterations necessary to    #
        # evaluate all members of the aforementioned arrays.                   #
        #                                                                      #
        ########################################################################
        $hc = scalar(@raw);
        # Clear the counter;
        $rc = 0;
        # Clear the flags
        $auditOccuredOnCurrentDate = 0;
        $machineRoleDefined = 0;
        # Get the current
        my @timearray=localtime(time());

        # Step through each record to find the last successful audit
        while ($rc < $hc) {
                ########################################################################
                #                                                                      #
                # Sometimes blank lines are read in from a file at the end of  #
                # the array.  Therefore, checking the existence of year, day   #
                # and month ensures that the blank lines are not being         #
                # considered, i.e.,processing time, is not wasted on the blank #
                # lines.                                                       #
                #                                                                      #
                ########################################################################
                if(@year[$rc] && @month[$rc] && @day[$rc]) {
                        ##########################################################
                        #                                                        #
                        # If the record is non-zero, set the values from the   #
                        # record to a variable indicating the date of the      #
                        # record currently being evaluated.  When all the      #
                        # records have been evaluated, the "last" variables    #
                        # will contain the last day, month and year that the   #
                        # audit process sucessfully completed.  This is        #
                        # indicated by the record in the machine-tracking file.#
                        #                                                        #
                        ##########################################################
                        $lastYear = @year[$rc];
                        $lastMonth = @month[$rc];
                        $lastDay = @day[$rc];
                        # Set a counter indicating the record containing the
                        # last good audit date.
                        $oc = $rc;
                        # If the date in the record currently being evaluated
                        # is the current date, set a flag indicating that the
                        # audit ran on the current date.
                        if (
                            (@year[$rc] == (@timearray[5]+1900)) &&
                            (@month[$rc] == (@timearray[4]+1)) &&
                            (@day[$rc] == @timearray[3])
                           ) {
                               $auditOccuredOnCurrentDate = 1;
                        }
                }
                # Increment the record counter
                $rc++;
        }
        # If the %MACHINEROLE was not properly defined on the machine at the
        # time of the last sucessful audit, create an error message.
        if (!($validRoles{@machineRole[$oc]})) {
                $roleNotSet{@machineTrackingName[$oc]} = "The machine role was not
properly defined on this machine.  (<FONT COLOR=\"#ff0000\">%MACHINEROLE% =
\"@machineRole[$oc]\"</FONT>) \n";
        }
        # If the last sucessful audit was not completed on the current day,
        # create an error message
        if (!($auditOccuredOnCurrentDate)) {
                @lastAudit[$la] = "The last successful audit of @machineTrackingName[$oc]
occurred on @month[$oc]/@day[$oc]/@year[$oc]\n";
                @machineNotCurrent[$la] = @machineTrackingName[$oc];
                $la++;
        }
```

```
        # Update the machine-tracking file list counter
        $ac++;

}

################################################################################
#                                                                              #
# The || die command is read "or" die.  In PERL logical operators can be used  #
# to link certain commands.  These commands return some type of value when     #
# they are used.  The return value often does not need to be used for anything.#
# For instance, the open() function returns a true value if the file open      #
# succeeds and a false value if the file open fails.                           #
# Since PERL stops evaluating a logical construct as soon                      #
# as it is either true or false without a doubt, the second command will only  #
# ever execute if the first one is false.  This is because as long as the      #
# first component of an OR statement is true, the entire statement will be     #
# true.  If the first component of the OR statement is false, the second       #
# component will have to be evaluated in order to determine the outcome of the #
# logical statement.                                                           #
# In this case, the script exits with an error if it can't open the report     #
# definition file                                                              #
#                                                                              #
################################################################################


# Read in format definititions
open (SOURCEFILE, $definition) || die scalar(localtime) . " Error opening definition
file: $!\n";
@raw=<SOURCEFILE>;
close SOURCEFILE;

#################################
# The Report Defininitions File #
#################################


################################################################################
#                                                                              #
# The Report Definitions file is a modified version of the information found   #
# within the audit template files created with the Security Configuration and  #
# Analysis MMC snap-in.                                                        #
# The Report Definitions file is made up of two fields and section headings.   #
# The section headings are enclosed with square brackets and are identical to  #
# those found in the audit templates. The information in the first field is    #
# also identical to the information in the audit templates and in organized    #
# under the same headings.  The SECEDIT.EXE tool outputs audit mismatches      #
# using the audit template file definitions.                                   #
# The second field is made up of HTML encoded user-friendly error messages     #
# corresponding to each of the entries in the first field with the exception   #
# of the headings.                                                             #
# The two fields are separated by a "tab" "pipe" "tab" pattern,                #
# "     |       ".  This was done for readiblity to ensure that there would     #
# be no collisions with the audit template output.                             #
#                                                                              #
################################################################################


################################################################################
#                                                                              #
# Create report subsection definitions                                         #
#                                                                              #
# Find section headings, thus defining the beginning and end of each section.  #
# The beginning of a section is one line after the section heading.            #
# The end of a section is one sooner than the beginning of the next section.   #
# The end of the last section is the end of the report definitions file.       #
#                                                                              #
################################################################################


# Clear the counters
$rc = 0;
```

```
$sa = 0;
$lo = 0;
$ea = 0;
$pr = 0;
$rk = 0;
$rv = 0;
$fs = 0;
$gm = 0;
$sg = 0;

################################################################################
#                                                                              #
# PERL has extremely powerful text matching and handling functions.            #
# the $variable =~ /pattern/ command will find the first occurence of          #
# "pattern" in $variable.  This uses a syntax known as a Regular Expression.   #
# However, sometimes there are difficulties finding patterns containing        #
# special characters.  Most of the special characters have a special meaning   #
# within the match command.  There are multiple ways to get the special        #
# characters recognized as part of the pattern and not as special characters.  #
# Any characters between \Q and \E are treated as part of the pattern.         #
# Another way to do this is to "Escape" each sepcial character individually.   #
# "Escaping" a special character tells the regular expression to treat it as   #
# part of the pattern and is accomplished by placing a backslash in front of   #
# the special character.                                                       #
# Therefore, in order to search for the pattern:                               #
#    [System Access]                                                           #
# The right and left square brackets have to be escaped individually.          #
# The proper regular expression ended up being:                                #
#    \[System Access\]                                                         #
#                                                                              #
################################################################################

# Step through all the report definitions, recording the begining of all
# report subsections
while(defined(@raw[$rc])){
        if (@raw[$rc]=~/\[System Access\]/) {
                # Begining of password policy subsection
                $sa=$rc+1;
        }
        if (@raw[$rc]=~/\[Logs\]/) {
                # Begining of log settings subsection
                $lo=$rc+1;
        }
        if (@raw[$rc]=~/\[Event Audit\]/) {
                # Begining of audit policy subsection
                $ea=$rc+1;
        }
        if (@raw[$rc]=~/\[Privilege Rights\]/) {
                # Begining of security rights subsection
                $pr=$rc+1;
        }
        if (@raw[$rc]=~/\[Registry Keys\]/) {
                # Begining of registry permissions subsection
                $rk=$rc+1;
        }
        if (@raw[$rc]=~/\[Registry Values\]/) {
                # Begining of registry settings subsection
                $rv=$rc+1;
        }
        if (@raw[$rc]=~/\[File Security\]/) {
                # Begining of file permissions subsection
                $fs=$rc+1;
        }
        if (@raw[$rc]=~/\[Group Membership\]/) {
                # Begining of restricted groups subsection
                $gm=$rc+1;
        }
        if (@raw[$rc]=~/\[Service General Setting\]/) {
                # Begining of system services subsection
                $sg=$rc+1;
        }
```

```
        $rc++;
}


################################################################################
#                                                                              #
# PERL has extremely powerful text matching and handling functions.            #
# the $variable =~ /pattern/ command will find the first occurence of          #
# "pattern" in $variable.                                                      #
# Moreover, once the pattern is found, everything in $variable before          #
# "pattern" is assigned to the built-in variable $` and everything after       #
# "pattern" is assigned to the built-in variable $'                            #
# This makes it very easy to separate the two data fields into separate        #
# variables.                                                                   #
#                                                                              #
# In this case, a hash is created.  A hash is like an array, but instead of    #
# being indexed by a number, (@array[1]) it is indexed on a text field.        #
# The hash is made up of a key (the index) and the value (the value being      #
# indexed.)  This technique was used to attempt to speed up the matching       #
# of the audit mismatches (failures) from the input data files and the         #
# user-friendly names.                                                         #
#                                                                              #
################################################################################

################################################################################
#                                                                              #
# A hash is created for the messages for each section, except file and         #
# registry security.  This is because of the way that the SECEDIT tool returns #
# data.  File and registry permissions are evaluted from the point in the tree #
# where they are set, down along the child branches.  This means that audit    #
# exceptions will occur that do not exactly match the definitions file.  Using #
# hash to speed matching only works when the matches are exact.  Another       #
# method is used for file and registry permissions as explained below.         #
#                                                                              #
################################################################################
while ($sa < ($lo-1)){
        @raw[$sa++] =~ /\t\|\t/;
        # Password policy hash
        $sah{$`} = $';
}
while ($lo < ($ea-1)){
        @raw[$lo++] =~ /\t\|\t/;
        # Log settings hash
        $loh{$`} = $';
}
while ($ea < ($pr-1)){
        @raw[$ea++] =~ /\t\|\t/;
        # Audit policy hash
        $eah{$`} = $';
}
while ($pr < ($rk-1)){
        @raw[$pr++] =~ /\t\|\t/;
        # Security rights hash
        $prh{$`} = $';
}
while ($rv < ($fs-1)){
        @raw[$rv++] =~ /\t\|\t/;
        # Registry settings hash
        $rvh{$`} = $';
}
while ($gm < ($sg-1)){
        @raw[$gm++] =~ /\t\|\t/;
        # Restricted groups hash
        $gmh{$`} = $';
}
while ($sg < ($rc-1)){
        @raw[$sg++] =~ /\t\|\t/;
        # System services hash
        $sgh{$`} = $';
}
```

```
# Get the current datestamp and set the Summary Report file path and name
$dateStamp = &dateAsString(time());

# The Summary Report is created in the output directory
$summaryReport = $output . "\\Summary Report-$dateStamp.html";

# Open the file for the Summary Report
open(SUMMFILE, ">" . $summaryReport) || die scalar(localtime) . " Error opening output
file: $summaryReport: $!\n";

# Write the header to the Summary Report
# All reports are formatted in HTML
print SUMMFILE "<HTML>\n";
print SUMMFILE "<HEAD>\n";
print SUMMFILE "<TITLE>Audit Summary Report - " . scalar(localtime) . "</TITLE>\n";
print SUMMFILE "</HEAD>\n";
print SUMMFILE "<center><h1>Audit Summary Report</h1></center>\n";
print SUMMFILE "<hr><h2>" . scalar(localtime) .
"\n<hr></h2><h3>Filename:\t$summaryReport</h3><hr>\n";
print SUMMFILE "<h2>Computers that are non-compliant to the audit policy\n</h2>";


# Determine input (audit data) files
# There is always a machine-tracking file for every audit data file

# Open input directory
opendir (SOURCEDIR, $input) || die scalar(localtime) . " Error opening directory to get
input files, $input: $!\n";

# Read a list of all the files in the directory into the raw array
@raw=readdir(SOURCEDIR);
close SOURCEDIR;

# Determine which files to use based on the current date.

################################################################
##                                                            ##
##   If there are no raw data files for the current date      ##
##   the report assumes that there is no data and the report  ##
##   will be empty.  This is done so that the audit report    ##
##   is timely.  This is because it is expected to be used    ##
##   to enforce compliance with written policy.               ##
##                                                            ##
################################################################

# Clear counters
$rc=0;
$fc=0;

#Get target (current) date
$targetDate = &dateAsString(time());

# Step through the file list
while (defined(@raw[$rc])){
        # If the datestamp indicates that this data file was created
        # on the current date, include it in the report
        if (substr(@raw[$rc],0,length($targetDate)) eq $targetDate && substr(@raw[$rc],-4)
eq ".raw") {
                @unsortedFileList[$fc]=@raw[$rc] . "\n";
                $fc++
        }
        $rc++;
}


# Sorting the file names creates sorted output in the Summary Report
# since the input data is already sorted
@inputFileList = sort @unsortedFileList;
```

```
#####################
#                   #
# Process Data Files #
#                   #
#####################

###############################################################################
#                                                                             #
# For each raw audit input file, run the audit report.  A summary audit report #
# will be produced, showing whether or not a particular machine is in         #
# compliance.  If there are any settings on a machine that are out of         #
# compliance, a detailed exception report will be produced for that machine.  #
# This report will be linked to the summary report with a hypertext link.     #
# All reports are encoded in HTML.                                            #
#                                                                             #
###############################################################################

# Clear the file counter
$ac=0;

# Step throught the sorted list of data files for the current date
while (defined(@inputFileList[$ac])) {

        # Clear @filtered and @exceptions since the new information would
        # otherwise simply be appended to the end of the data from the previous
        # files.
        @filtered = "";
        @exceptions = "";

        # The input file is the file in the $input directory
        $inputFile = $input . "\\" . @inputFileList[$ac];

        # Open the input file name, read the input and close the input file.
        open (SOURCEFILE, $inputFile) || print scalar(localtime) . "Error opening input
file: $inputFile: $!\n";
        @filtered = <SOURCEFILE>;
        close SOURCEFILE;


        # Read environmental variables from data
        # Use the matching techniques described about to filter out newlines
        # and trailing spaces
        @filtered[1] =~ /\n/;
        $machineName = $`;
        $machineName =~ / /;
        $machineName = $`;


        ###########################################################################
        #                                                                         #
        # Clean up the data for the execption report.  While this data is is      #
        # never actually output, it could be.  It could be used for running       #
        # other reports such as a trending analysis report or a correlation       #
        # between break-ins and out of compliance systems.                        #
        # Any setting that does not comply with the audit template generates      #
        # a line in the data file with the word "Mismatch" and the setting        #
        # from the template that did not comply.                                  #
        # Therefore, only the "Mismatch" lines are retained.  The word            #
        # "Mismatch" and any leading spaces are also cleaned out of the line.     #
        # Next, the data is converted to lower case so that the matches in the    #
        # report definition file will be easier to find.  Finally, the            #
        # trailing period and carriage return are removed for the same reason.    #
        #                                                                         #
        ###########################################################################

        # Clear the counters
        $fc=0;
        $ec=0;
        #\x2e\x0d\x0d\x0a
        # Step through the filtered data
        while (defined(@filtered[$fc])){
```

```
        # Does this line contain "Mismatch       - "?
        if (@filtered[$fc]=~ /Mismatch       - /) {
                # Assign the lower case value of everything after
                # the match
                $temp = lc($');
                # Find the pattern dot carriage return
                $temp =~ /\.\r$/;
                # Assign everything before the match to the exceptions array
                @exceptions[$ec] = $`;
                # Increment the exceptions counter
                $ec++;
        }
        # Increment the filtered data counter
        $fc++;
}

########################################################################
#                                                                      #
# If there are exceptions, write an entry to the high-level exception  #
# report showing machines that are not in compliance with a link to    #
# the detailed exception report.                                       #
# Since the exceptions array is cleared using @exceptions = "", the    #
# 1st element of the array is a null, once data is placed into the     #
# array, the 1rst element accepts data and is no longer null.          #
# Therefore, in order for the @exceptions array to have no report      #
# data, it must only contain a single entry that has no word,          #
# whitespace or non-word characters.  Within the search command,       #
# the dot means all characters except newline.                         #
#                                                                      #
########################################################################
if ((@exceptions == 1) && !(@exceptions[0] =~ /./)) {
        # If there is an entry for this machine in the %roleNotSet hash
        # print the error message.
        if ($roleNotSet{$machineName}){
                print SUMMFILE "<li><FONT COLOR=\"#ff0000\">$machineName</FONT>";
                print SUMMFILE "- Not in compliance.</FONT>\n";
                print SUMMFILE "<ul><li>";
                print SUMMFILE $roleNotSet{$machineName};
                print SUMMFILE "</ul>\n";
        }
        else {
                print SUMMFILE "<li><FONT COLOR=\"#008000\">$machineName\n</FONT>";
        }
}
if (!(@exceptions == 1) && (@exceptions[0] =~ /./)) {
         ###################################################################
         #                                                                 #
         # Determine file name of Detailed Exception Report                #
         # The name of the exception report is the name of the input       #
         # file with -Exception Report.html instead of the .raw            #
         # Since the last charater of $inputFile is a newline (\n)         #
         # the last five characters have to be replaced .raw\n.            #
         # Please note that \n is a single character.                      #
         #                                                                 #
         ###################################################################
         # Create the link to the Exception Report and write the entry
         # to the Summary Report.
         # Use a relative link for the file
         $linkName = &dateAsString(time()) . "-$machineName-Exception Report.html";

         # Set the Exception Report filename
         $exceptionReport = $output . "\\" . $linkName;

         # If there is an entry for this machine in the %roleNotSet hash
         # print the error message.
         if ($roleNotSet{$machineName}){
                print SUMMFILE "<li><FONT COLOR=\"#ff0000\"><a
HREF=\"$linkName\">$machineName<a> ";
                print SUMMFILE "- Not in compliance.</FONT>\n";
                print SUMMFILE "<ul><li>";
                print SUMMFILE $roleNotSet{$machineName};
```

```
                print SUMMFILE "</ul>\n";
        }
        else {
                print SUMMFILE "<li><FONT COLOR=\"#ff0000\"><a
HREF=\"$linkName\">$machineName<a> ";
                print SUMMFILE "- Not in compliance.</FONT>\n";
        }

#########################################################################
#                                                                       #
# Process @exceptions, sorting the data into the the proper output      #
# arrays for each section.                                              #
# Step through the @exceptions array and perform a look up on each      #
# category hash, excepting the file and registry permissions.           #
# The SECEDIT tool, which provides most of the raw data for this        #
# report does not provide detailed information on what is wrong with    #
# the file or registry permissions, simply that they are not in         #
# compliance with the audit template.                                   #
# Likewise, this report simply shows what items are out of compliance.  #
# The category arrays group the data into related sections that are in  #
# the same order the settings are found in the Security Analysis and    #
# Group Policy snap-in MMC tools.                                       #
#                                                                       #
#########################################################################

        # Clear counters for individual report sections
        $ec = 0;
        $ic = 0;
        $sa = 0;
        $lo = 0;
        $ea = 0;
        $pr = 0;
        $rk = 0;
        $rv = 0;
        $fs = 0;
        $gm = 0;
        $sg = 0;

        # Step through the execptions
        while (defined(@exceptions[$ec])){
                # Is the exception in Account Policies
                if ($sah{@exceptions[$ec]}) {
                        @sas[$sa++] = $sah{@exceptions[$ec++]};
                        next;
                }
                # Is the exception in Event Log Settings
                if ($loh{@exceptions[$ec]}) {
                        @los[$lo++] = $loh{@exceptions[$ec++]};
                        next;
                }
                # Is the exception in Audit Policy
                if ($eah{@exceptions[$ec]}) {
                        @eas[$ea++] = $eah{@exceptions[$ec++]};
                        next;
                }
                # Is the exception in User Rights Assignments
                if ($prh{@exceptions[$ec]}) {
                        @prs[$pr++] = $prh{@exceptions[$ec++]};
                        next;
                }
                # Is the exception in Security Options/Registry Values
                # - registry entry values used to set the security
                # options
                if ($rvh{@exceptions[$ec]}) {
                        @rvs[$rv++] = $rvh{@exceptions[$ec++]};
                        next;
                }
                # Is the exception in Resticted Groups
                if ($gmh{@exceptions[$ec]}) {
                        @gms[$gm++] = $gmh{@exceptions[$ec++]};
                        next;
```

```
                        }
                        # Is the exception in System Services
                        if ($sgh{@exceptions[$ec]}) {
                                @sgs[$sg++] = $sgh{@exceptions[$ec++]};
                                next;
                        }
                        ########################################################
                        #                                                      #
                        # Exception reports for registry permission entries    #
                        # always have the same format and may not be in the    #
                        # definition file since they may be a subkey under     #
                        # the key defined in the definition file and the       #
                        # template.                                            #
                        #                                                      #
                        ########################################################

                        ########################################################
                        #                                                      #
                        # Registry Permissions exceptions always follow the    #
                        # same pattern.  Furthermore, no other exceptions      #
                        # follow that pattern.  The following chart shows the  #
                        # Registry Permissions exceptions and the types.       #
                        #                                                      #
                        # Exception Starts With       Exception Type           #
                        # machine\hardware            HKLM\hardware            #
                        # machine\software            HKLM\software            #
                        # machine\system              HKLM\system              #
                        # classes_root                HKCR                     #
                        # users                       HKU                      #
                        #                                                      #
                        # The substitution operator, s/// is used to find the  #
                        # Registry Permissions exception and replace the       #
                        # beginning of the exception with the exception type   #
                        # spelled out, for instance instead of: users\.default #
                        # it would be: HKEY_USERS\.default                     #
                        #                                                      #
                        ########################################################
                        # Is the exception in:
                        # Registry Permissions - HKEY_LOCAL_MACHINE\hardware
                        if (@exceptions[$ec] =~
s/machine\\hardware/HKEY_LOCAL_MACHINE\\hardware/i) {
                                @rks[$rk] = @exceptions[$ec++];
                                @rks[$rk] = "<li>Registry permissions are incorrectly set
on <b>@rks[$rk]</b>\n";
                                $rk++;
                                next;
                        }
                        # Is the exception in:
                        # Registry Permissions - HKEY_LOCAL_MACHINE\software
                        if (@exceptions[$ec] =~
s/^machine\\software/HKEY_LOCAL_MACHINE\\software/i) {
                                @rks[$rk] = @exceptions[$ec++];
                                @rks[$rk] = "<li>Registry permissions are incorrectly set
on <b>@rks[$rk]</b>\n";
                                $rk++;
                                next;
                        }
                        # Is the exception in:
                        # Registry Permissions - HKEY_LOCAL_MACHINE\system
                        if (@exceptions[$ec] =~
s/^machine\\system/HKEY_LOCAL_MACHINE\\system/i) {
                                @rks[$rk] = @exceptions[$ec++];
                                @rks[$rk] = "<li>Registry permissions are incorrectly set
on <b>@rks[$rk]</b>\n";
                                $rk++;
                                next;
                        }
                        # Is the exception in:
                        # Registry Permissions - HKEY_CLASSES_ROOT
                        if (@exceptions[$ec] =~ s/^classes_root/HKEY_CLASSES_ROOT/i) {
                                @rks[$rk] = @exceptions[$ec++];
```

```
                                @rks[$rk] = "<li>Registry permissions are incorrectly set
on <b>@rks[$rk]</b>\n";
                                $rk++;
                                next;
                        }
                        # Is the exception in:
                        # Registry Permissions - HKEY_USERS\.default
                        if (@exceptions[$ec] =~
s/^\Qusers\.default\E/\QHKEY_USERS\.default\E/i) {
                                @rks[$rk] = @exceptions[$ec++];
                                @rks[$rk] = "<li>Registry permissions are incorrectly set
on <b>@rks[$rk]</b>\n";
                                $rk++;
                                next;
                        }

                        # Is the exception in File Permissions

                        ##########################################################
                        #                                                        #
                        # Exception reports for file permission entries always   #
                        # have the same format and may not be in the definition  #
                        # file since they may be a subdirectory under the        #
                        # directory defined in the definition file and the       #
                        # template.                                              #
                        #                                                        #
                        ##########################################################

                        ##########################################################
                        #                                                        #
                        # File Permissions exceptions always start with a        #
                        # letter, followed by a colon, followed by a backslash.  #
                        #  Example:  C:\boot.ini                                 #
                        # No other exceptions have this pattern.                 #
                        # Therefore, any exception that matches this pattern     #
                        # is a file permissions exception.                       #
                        #                                                        #
                        ##########################################################
                        if (@exceptions[$ec] =~ /^[a-z]\:\\/i) {
                                @fss[$fs] = @exceptions[$ec++];
                                @fss[$fs] = "<li>File permissions are incorrectly set on
<b>@fss[$fs]</b>\n";
                                $fs++;
                                next;
                        }
                        $ec++;
                }

                # Open the file for the Exception Report.
                open (EREPORTEFILE, ">" . $exceptionReport) || print scalar(localtime) . "
Error opening Exception Report file: $exceptionReport: $!\n";

                # Write the header to the Exception Report
                print EREPORTEFILE "<HTML>\n";
                print EREPORTEFILE "<HEAD>\n";
                print EREPORTEFILE "<TITLE>Exception Report for $machineName - " .
scalar(localtime) . "</TITLE>\n";
                print EREPORTEFILE "</HEAD>\n";
                print EREPORTEFILE "<center><h1>Exception Report for
$machineName</h1></center>\n";
                print EREPORTEFILE "<hr><h2>" . scalar(localtime) .
"\n<hr></h2><h3>Filename:\t$exceptionReport</h3><hr>\n";
                print EREPORTEFILE "<h2>The <FONT COLOR=\"#ff0000\">non-compliant</FONT>
items are grouped according to the categories below.\n</h2>";

                # Create links for each section
                print EREPORTEFILE "<ul><li><A HREF=\"#Account Policies\">Account
Policies</A>\n";
                print EREPORTEFILE "<li><A HREF=\"#Audit Policy\">Audit Policy</A>\n";
                print EREPORTEFILE "<li><A HREF=\"#User Rights Assignment\">User Rights
Assignment</A>\n";
```

```
                print EREPORTEFILE "<li><A HREF=\"#Security Options (Registry
Settings)\">Security Options (Registry Settings)</A>\n";
                print EREPORTEFILE "<li><A HREF=\"#Event Log Settings\">Event Log
Settings</A>\n";
                print EREPORTEFILE "<li><A HREF=\"#Restricted Groups\">Restricted
Groups</A>\n";
                print EREPORTEFILE "<li><A HREF=\"#System Services\">System
Services</A>\n";
                print EREPORTEFILE "<li><A HREF=\"#Registry Permissions\">Registry
Permissions</A>\n";
                print EREPORTEFILE "<li><A HREF=\"#File System Permissions\">File System
Permissions</A></ul>\n";

                # Write report sections
                print EREPORTEFILE "<hr><h2><A NAME=\"Account Policies\">Account
Policies</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @sas;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "<hr><h2><A NAME=\"Audit Policy\">Audit
Policy</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @eas;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "<hr><h2><A NAME=\"User Rights Assignment\">User Rights
Assignment</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @prs;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "<hr><h2><A NAME=\"Security Options (Registry
Settings)\">Security Options (Registry Settings)</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @rvs;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "<hr><h2><A NAME=\"Event Log Settings\">Event Log
Settings</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @los;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "<hr><h2><A NAME=\"Restricted Groups\">Restricted
Groups</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @gms;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "<hr><h2><A NAME=\"System Services\">System
Services</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @sgs;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "<hr><h2><A NAME=\"Registry Permissions\">Registry
Permissions</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @rks;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "<hr><h2><A NAME=\"File System Permissions\">File
System Permissions</A></h2>\n";
                print EREPORTEFILE "<ul>\n";
                print EREPORTEFILE @fss;
                print EREPORTEFILE "</ul>\n";

                print EREPORTEFILE "</ul><hr>End of Exception Report for $machineName on "
. scalar (localtime) ."\n</BODY></HTML>";
                close EREPORTEFILE;
```

```
        }

        # Update the raw audit input file counter
        $ac++;
}


# Write the list of machines that have successfully completed the audit in the
# past, but did not do so on the current date
print SUMMFILE "</ul><hr>\n";
print SUMMFILE "<h2>Computers which did not complete audit on " . scalar(localtime) .
"</h2>\n<ul>";
# Clear the counter
$ac = 0;
while (defined(@lastAudit[$ac])){
        # If there is an entry for this machine in the %roleNotSet hash
        # print the error message.
        if ($roleNotSet{@machineNotCurrent[$ac]}){
                print SUMMFILE "<li><FONT COLOR=\"#ff0000\">@lastAudit[$ac]</FONT>";
                print SUMMFILE "- Not in compliance.</FONT>\n";
                print SUMMFILE "<ul><li>";
                print SUMMFILE $roleNotSet{@machineNotCurrent[$ac]};
                print SUMMFILE "</ul>\n";
        }
        else {
                print SUMMFILE "<li>@lastAudit[$ac]";
        }
        $ac++;
}
print SUMMFILE "</ul>\n";

# Get the current time to calculate elapsed time
$endTime=time();

# Calculate elapsed time
$elapsedTime = ($endTime-$startTime);
$elapsedTimeMinutes = $elapsedTime/600;

# Write the start time, finished time and elapsed time to the Summary Report
print SUMMFILE "<hr>\n";
print SUMMFILE "<ul>\n";
print SUMMFILE "<li>Report Started:\t" . scalar(localtime($startTime)) . "\n";
print SUMMFILE "<li>Report Ended:\t" . scalar(localtime($endTime)) . "\n";
print SUMMFILE "<li>Elapsed time:\t\t$elapsedTimeMinutes minutes</ul><hr>\n";
close SUMMFILE;


################################################################################
#                                                                              #
#                              Subroutines                                     #
#                                                                              #
################################################################################


# Generic Subroutine
# This code demonstrates the basic structure of a PERL function.
# It has no function within the Audit System
sub functionName {
        #Function Purpose

        #Assign passed variables to local variables and declare local variables
        my $passedvariable = $_[0];    # The first values passed by the
                                       # calling function
        my @localarray;                        # The variables only have local
        my $localscalar;               # significance, having no value
                                       # outside the function

        #Function Code
                #This is where the body of the function goes

        #Return data
```

```
		#This is the value returned to the calling function
	return ($date);
}



# Create a datestamp
sub dateAsString {
	# This function takes input from the time() function or a number that
	# indicates the number of seconds since 1900 and converts it
	# into a date string of format: yyyy-mm-dd.
	# Calling dateasstring() with time() as the input, returns the
	# present date.
	# You can get other dates by manipulating the value returned by time
	# by adding or subtracting the number of seconds in a day or
	# a week and then calling dateasstring().
	# 60s/m * 60m/h * 24h/d * 7d/w = 604800 seconds in a week.
	# 60s/m * 60m/h * 24h/d = 86400 seconds in a day.

	my $passedtime = $_[0];                      # Time passed by the calling program
	my @timearray=localtime($passedtime); # Time stored in a standard array format
	my $date;                             # The return datestamp

	 ##########################################################################
	#                                                                          #
	# The standard time array format is structured                             #
	# @timearray[0] = second                                                   #
	# @timearray[1] = minute                                                   #
	# @timearray[2] = hour                                                     #
	# @timearray[3] = day of the month                                         #
	# @timearray[4] = month, 0 = January                                       #
	# @timearray[5] = years since 1900                                         #
	# @timearray[6] = day of week, 0 = Sunday                                  #
	# @timearray[7] = day of year                                              #
	# @timearray[8] = daylight savings time, 0 = no, 1 = yes                   #
	#                                                                          #
	############################################################################

	#Add the year to the datestamp
	$date=($timearray[5]+1900);
	$date .= "-";

	#Add the month to the datestamp
	if ((@timearray[4]+1) < 10){
		#Add a leading 0 if the month is a single digit
		$date .= "0" . (@timearray[4]+1) ."-";
	}
	if ((@timearray[4]+1) > 9){
		$date .= (@timearray[4]+1) ."-";
	}

	#Add the day of the month to the datestamp
	if (@timearray[3] < 10){
		#Add a leading 0 if the day is a single digit
		$date .= "0" . ($timearray[3]);
	}
	if (@timearray[3] > 9){
		$date .= (@timearray[3]);
	}
	#Return the datestamp
	return ($date);
}

# Return proper usage notation
sub incorrectUsage{
	#Function Purpose
	#This function returns the usage instructions and exits the script
	#This function is passed the specific error message

	my $errorMessage = $_[0];      # The specific error message passed
	                               # by the calling program
```

```
        my $usageText;                # The usage instructions

        # Set the useage instructions
        $usageText = "The usage for this script is: \n\nperl auditreport -i:input-
directory -o:output-directory -d:report-definitions\n\n\tinput-directory:\tthe complete
path to the directory\n\t\t\t\tcontaining the raw audit files\n\toutput-directory:\tthe
complete path to the directory where \n\t\t\t\tthe reports will be created\n\treport-
definitions:\tthe complete path and filename of the report\n\t\t\t\tdefinitions
file\n\n";


        # Function Code
        # Exit the script with the specific error message and the usage
        # instructions
        die scalar(localtime) . " \n$errorMessage\n\n$usageText\n";

        # Return data
        return ();
}
```

AUDIT_DC.INF

```
[Unicode]
Unicode=yes
[System Access]
MinimumPasswordAge = 1
MaximumPasswordAge = 90
MinimumPasswordLength = 12
PasswordComplexity = 1
PasswordHistorySize = 24
LockoutBadCount = 3
ResetLockoutCount = 15
LockoutDuration = 15
RequireLogonToChangePassword = 1
ClearTextPassword = 0
[System Log]
MaximumLogSize = 4194240
AuditLogRetentionPeriod = 2
RetentionDays = 7
RestrictGuestAccess = 1
[Security Log]
MaximumLogSize = 4194240
AuditLogRetentionPeriod = 2
RetentionDays = 7
RestrictGuestAccess = 1
[Application Log]
MaximumLogSize = 4194240
AuditLogRetentionPeriod = 2
RetentionDays = 7
RestrictGuestAccess = 1
[Event Audit]
AuditSystemEvents = 3
AuditLogonEvents = 3
AuditObjectAccess = 2
AuditPrivilegeUse = 2
AuditPolicyChange = 3
AuditAccountManage = 3
AuditProcessTracking = 0
AuditDSAccess = 2
AuditAccountLogon = 3
CrashOnAuditFull = 1
[Privilege Rights]
seassignprimarytokenprivilege =
seauditprivilege =
sebackupprivilege = *S-1-5-32-544
sebatchlogonright =
sechangenotifyprivilege = *S-1-5-11
secreatepagefileprivilege = *S-1-5-32-544
secreatepermanentprivilege =
secreatetokenprivilege =
sedebugprivilege =
sedenybatchlogonright =
sedenyinteractivelogonright =
sedenynetworklogonright =
sedenyservicelogonright =
seenabledelegationprivilege = *S-1-5-32-544
seincreasebasepriorityprivilege = *S-1-5-32-544
seincreasequotaprivilege = *S-1-5-32-544
seinteractivelogonright = *S-1-5-32-544
seloaddriverprivilege = *S-1-5-32-544
selockmemoryprivilege =
semachineaccountprivilege =
senetworklogonright = *S-1-5-32-544,*S-1-5-11
seprofilesingleprocessprivilege = *S-1-5-32-544
seremoteshutdownprivilege = *S-1-5-32-544
serestoreprivilege = *S-1-5-32-544
sesecurityprivilege = *S-1-5-32-544
seservicelogonright =
seshutdownprivilege = *S-1-5-32-544
```

```
sesyncagentprivilege =
sesystemenvironmentprivilege = *S-1-5-32-544
sesystemprofileprivilege = *S-1-5-32-544
sesystemtimeprivilege = *S-1-5-32-544
setakeownershipprivilege = *S-1-5-32-544
setcbprivilege =
seundockprivilege =
[Registry Keys]
"MACHINE\SOFTWARE\Microsoft\OS/2 Subsystem for
NT",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"machine\software",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"machine\software\microsoft\netdde",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KA;;;SY)"
"machine\software\microsoft\protected storage system provider",1,"D:AR"
"machine\software\microsoft\windows
nt\currentversion\perflib",2,"D:P(A;CI;GR;;;IU)(A;CI;GA;;;BA)(A;CI;GA;;;SY)(A;CI;GA;;;CO)
"
"machine\software\microsoft\windows\currentversion\group
policy",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CI;KA;;;SY)"
"machine\software\microsoft\windows\currentversion\installer",0,"D:PAR(A;CI;KA;;;BA)(A;CI
;KR;;;AU)(A;CI;KA;;;SY)"
"machine\software\microsoft\windows\currentversion\policies",0,"D:PAR(A;CI;KA;;;BA)(A;CI;
KR;;;AU)(A;CI;KA;;;SY)"
"machine\system",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"machine\system\clone",1,"D:AR"
"machine\system\controlset001",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset002",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset003",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset004",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset005",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset006",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset007",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset008",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset009",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\controlset010",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;
KA;;;SY)"
"machine\system\currentcontrolset\control\securepipeservers\winreg",2,"D:PAR(A;CI;KA;;;BA
)(A;;KR;;;BO)(A;CI;KA;;;SY)"
"machine\system\currentcontrolset\control\wmi\security",2,"D:P(A;CI;GR;;;BA)(A;CI;GA;;;SY
)(A;CI;GA;;;CO)"
"machine\system\currentcontrolset\enum",1,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CI;KA;;;SY
)"
"machine\system\currentcontrolset\hardware
profiles",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"users\.default",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"users\.default\software\microsoft\netdde",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KA;;;SY)"
"users\.default\software\microsoft\protected storage system provider",1,"D:AR"
"CLASSES_ROOT",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"MACHINE\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\AsrCommands",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CI;CCDCLCSWRPSDRC;;
;BO)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\PermittedManagers",2,"D:PAR(A;
CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\ValidCommunities",2,"D:PAR(A;C
I;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
[File Security]
"%SystemDrive%\Documents and Settings\All
Users",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY)"
"%SystemDrive%\ntldr",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\config.sys",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)
"
"%SystemDrive%\ntdetect.com",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\boot.ini",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
```

```
"%SystemDrive%\autoexec.bat",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;B
U)"
"%SystemRoot%\$NtServicePackUninstall$",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"c:\boot.ini",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\ntdetect.com",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\ntldr",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\ntbootdd.sys",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\autoexec.bat",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY)"
"c:\config.sys",2,"D:PAR(A;;FA;;;BA)(A;;CCSWWPLORC;;;AU)(A;;FA;;;SY)"
"%ProgramFiles%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICIIO;FA;;;CO)(A;OICI;
FA;;;SY)"
"%SystemRoot%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICIIO;FA;;;CO)(A;OICI;FA
;;;SY)"
"%SystemRoot%\CSC",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\debug",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICIIO;FA;;;CO)(A;O
ICI;FA;;;SY)"
"%SystemRoot%\Registration",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FR;;;AU)(A;OICI;FA;;;SY)"
"%SystemRoot%\repair",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\Tasks",1,"D:AR"
"%SystemRoot%\Temp",2,"D:PAR(A;OICI;FA;;;BA)(A;CI;DCLCWP;;;AU)(A;OICIIO;FA;;;CO)(A;OICI;F
A;;;SY)"
"%SystemDirectory%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICIIO;FA;;;CO)(A;OI
CI;FA;;;SY)"
"%SystemDirectory%\appmgmt",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY
)"
"%SystemDirectory%\DTCLog",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICIIO;FA;;;C
O)(A;OICI;FA;;;SY)"
"%SystemDirectory%\GroupPolicy",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;
;;SY)"
"%SystemDirectory%\NTMSData",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\Setup",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY)"
"%SystemDirectory%\ReinstallBackups",1,"D:P(A;OICI;GXGR;;;BU)(A;OICI;GXGR;;;PU)(A;OICI;GA
;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDirectory%\repl",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY)"
"%SystemDirectory%\repl\import",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;0x1
301bf;;;RE)(A;OICI;FA;;;SY)"
"%SystemDirectory%\repl\export",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;0x1
200a9;;;RE)(A;OICI;FA;;;SY)"
"%SystemDirectory%\spool\printers",2,"D:PAR(A;OICI;FA;;;BA)(A;CI;DCLCSWWPLO;;;AU)(A;OICII
O;FA;;;CO)(A;OICI;FA;;;SY)"
"%SystemDirectory%\config",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\dllcache",2,"D:P(A;OICI;GA;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDirectory%\ias",2,"D:P(A;OICI;GA;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDrive%\Documents and
Settings",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY)"
"%SystemDrive%\My Download
Files",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1201bf;;;AU)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)"
"%SystemDrive%\System Volume Information",1,"D:PAR"
"%SystemDrive%\Temp",2,"D:PAR(A;OICI;FA;;;BA)(A;CI;DCLCWP;;;AU)(A;OICIIO;FA;;;CO)(A;OICI;
FA;;;SY)"
"%SystemDrive%\",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICIIO;FA;;;CO)(A;OICI;
FA;;;SY)"
"%SystemDrive%\IO.SYS",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY)"
"%SystemDrive%\MSDOS.SYS",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY)"
"%SystemRoot%\regedit.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rcp.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\Ntbackup.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rexec.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rsh.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\regedt32.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\debug\UserMode",0,"D:PAR(A;OICI;FA;;;BA)(A;;CCDCWP;;;AU)(A;OIIO;DCLC;;;AU)(
A;OICI;FA;;;SY)"
"%SystemDrive%\Documents and
Settings\Administrator",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\Documents and Settings\All
Users\Documents\DrWatson",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICIIO;DCLCWP;
;;AU)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)"
"%SystemDirectory%\secedit.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\Inetpub",1,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI
;0x1200a9;;;BU)"
```

```
"%SystemDrive%\Documents and Settings\All
Users\Documents\DrWatson\drwtsn32.log",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1301bf;;;AU)(A;O
ICIIO;FA;;;CO)(A;OICI;FA;;;SY)"
"%SystemRoot%\Offline Web Pages",1,"D:AR(A;OICI;FA;;;WD)"
"%SystemRoot%\NTDS",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\SYSVOL",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICIIO;FA;;;CO)(A;
OICI;FA;;;SY)"
"%SystemRoot%\SYSVOL\domain\Policies",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OI
CIIO;FA;;;CO)(A;OICI;0x1301bf;;;PA)(A;OICI;FA;;;SY)"
"%SystemDrive%\Documents and Settings\Default
User",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;;;SY)"
"%SystemRoot%\security",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)"
"%SystemDrive%\Program Files\Resource Kit",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
[Version]
signature="$CHICAGO$"
Revision=1
[Group Membership]
*S-1-5-32-546__Memberof =
*S-1-5-32-546__Members =
*S-1-5-32-551__Memberof =
*S-1-5-32-551__Members =
*S-1-5-32-549__Memberof =
*S-1-5-32-549__Members =
*S-1-5-32-548__Memberof =
*S-1-5-32-548__Members =
*S-1-5-32-550__Memberof =
*S-1-5-32-550__Members =
Pre-Windows 2000 Compatible Access__Memberof =
Pre-Windows 2000 Compatible Access__Members =
[Profile Description]
Description=Domain Controller Audit Template for GIAC Practicum
[Kerberos Policy]
MaxTicketAge = 10
MaxRenewAge = 7
MaxServiceAge = 600
MaxClockSkew = 5
TicketValidateClient = 1
[Registry Values]
MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\System\LegalNoticeCaption=1,Co
nsent to Monitoring
MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\System\LegalNoticeText=1,This
is a RealmCo computer system.  This computer system, including all related equipment,
networks and network devices (specifically including Internet access), is provided only
for authorized RealmCo use.  RealmCo computer systems may be monitored for all lawful
purposes, including ensuring that their use is authorized, for management of the system,
to facilitate protection against unauthorized access and to verify security procedures,
survivability, and operation security.  Monitoring include active attacks by authorized
RealmCo employees or contractors to test or verify the security of this system.  During
monitoring, information may be examined, recorded, copied and used for authorized
purposes.  All information, including personal information, placed on or sent over this
system may be monitored.  Use of this RealmCo computer system, authorized or
unauthorized, constitutes consent to monitoring of this system.  Unauthorized use may
subject you to criminal prosecution.  Evidence if unauthorized use collect during
monitoring may be used for administrative, criminal or adverse action.  Use of this
system constitutes consent to monitoring for these purposes.
MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDriveTypeAutoRun=4,
255
machine\software\microsoft\driver signing\policy=3,1
machine\software\microsoft\non-driver signing\policy=3,1
machine\software\microsoft\windows
nt\currentversion\setup\recoveryconsole\securitylevel=4,0
machine\software\microsoft\windows nt\currentversion\setup\recoveryconsole\setcommand=4,0
machine\software\microsoft\windows nt\currentversion\winlogon\allocatecdroms=1,1
machine\software\microsoft\windows nt\currentversion\winlogon\allocatedasd=1,0
machine\software\microsoft\windows nt\currentversion\winlogon\allocatefloppies=1,1
machine\software\microsoft\windows nt\currentversion\winlogon\cachedlogonscount=1,0
machine\software\microsoft\windows nt\currentversion\winlogon\passwordexpirywarning=4,14
machine\software\microsoft\windows nt\currentversion\winlogon\scremoveoption=1,1
machine\software\microsoft\windows\currentversion\policies\system\disablecad=4,0
machine\software\microsoft\windows\currentversion\policies\system\dontdisplaylastusername
=4,1
```

```
machine\software\microsoft\windows\currentversion\policies\system\shutdownwithoutlogon=4,
0
machine\system\currentcontrolset\control\lsa\auditbaseobjects=4,1
machine\system\currentcontrolset\control\lsa\crashonauditfail=4,1
machine\system\currentcontrolset\control\lsa\fullprivilegeauditing=3,0
machine\system\currentcontrolset\control\lsa\lmcompatibilitylevel=4,5
machine\system\currentcontrolset\control\lsa\restrictanonymous=4,2
machine\system\currentcontrolset\control\print\providers\lanman print
services\servers\addprinterdrivers=4,1
machine\system\currentcontrolset\control\session manager\memory
management\clearpagefileatshutdown=4,1
machine\system\currentcontrolset\control\session manager\protectionmode=4,1
machine\system\currentcontrolset\services\lanmanserver\parameters\autodisconnect=4,30
machine\system\currentcontrolset\services\lanmanserver\parameters\enableforcedlogoff=4,1
machine\system\currentcontrolset\services\lanmanserver\parameters\enablesecuritysignature
=4,1
machine\system\currentcontrolset\services\lanmanserver\parameters\requiresecuritysignatur
e=4,0
machine\system\currentcontrolset\services\lanmanworkstation\parameters\enableplaintextpas
sword=4,0
machine\system\currentcontrolset\services\lanmanworkstation\parameters\enablesecuritysign
ature=4,1
machine\system\currentcontrolset\services\lanmanworkstation\parameters\requiresecuritysig
nature=4,0
machine\system\currentcontrolset\services\netlogon\parameters\disablepasswordchange=4,0
machine\system\currentcontrolset\services\netlogon\parameters\requiresignorseal=4,0
machine\system\currentcontrolset\services\netlogon\parameters\requirestrongkey=4,0
machine\system\currentcontrolset\services\netlogon\parameters\sealsecurechannel=4,1
machine\system\currentcontrolset\services\netlogon\parameters\signsecurechannel=4,1
MACHINE\System\CurrentControlSet\Control\Lsa\SubmitControl=4,0
MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\AutoAdminLogon=4,0
[Service General Setting]
MSFTPSVC,4,"D:AR(D;;RPWPDTWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;WPWDWO;;;WD)"
IISADMIN,4,"D:AR(D;;RPWPDTWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
SharedAccess,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RasAuto,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RasMan,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RemoteAccess,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
SMTPSVC,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
TlntSvr,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
W3SVC,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
```

AUDIT_SERVER.INF

```
[Unicode]
Unicode=yes
[System Access]
MinimumPasswordAge = 1
MaximumPasswordAge = 90
MinimumPasswordLength = 12
PasswordComplexity = 1
PasswordHistorySize = 24
LockoutBadCount = 3
ResetLockoutCount = 15
LockoutDuration = 15
RequireLogonToChangePassword = 1
ClearTextPassword = 0
[System Log]
MaximumLogSize = 4194240
AuditLogRetentionPeriod = 2
RetentionDays = 7
RestrictGuestAccess = 1
[Security Log]
MaximumLogSize = 4194240
AuditLogRetentionPeriod = 2
RetentionDays = 7
RestrictGuestAccess = 1
[Application Log]
MaximumLogSize = 4194240
AuditLogRetentionPeriod = 2
RetentionDays = 7
RestrictGuestAccess = 1
[Event Audit]
AuditSystemEvents = 3
AuditLogonEvents = 3
AuditObjectAccess = 2
AuditPrivilegeUse = 2
AuditPolicyChange = 3
AuditAccountManage = 3
AuditProcessTracking = 0
AuditDSAccess = 0
AuditAccountLogon = 3
CrashOnAuditFull = 1
[Privilege Rights]
seassignprimarytokenprivilege =
seauditprivilege =
sebackupprivilege = *S-1-5-32-544
sebatchlogonright =
sechangenotifyprivilege = *S-1-5-32-545
secreatepagefileprivilege = *S-1-5-32-544
secreatepermanentprivilege =
secreatetokenprivilege =
sedebugprivilege =
sedenybatchlogonright =
sedenyinteractivelogonright =
sedenynetworklogonright =
sedenyservicelogonright =
seenabledelegationprivilege =
seincreasebasepriorityprivilege = *S-1-5-32-544
seincreasequotaprivilege = *S-1-5-32-544
seinteractivelogonright = *S-1-5-32-544
seloaddriverprivilege = *S-1-5-32-544
selockmemoryprivilege =
semachineaccountprivilege =
senetworklogonright = *S-1-5-32-544,*S-1-5-32-545
seprofilesingleprocessprivilege = *S-1-5-32-544
seremoteshutdownprivilege = *S-1-5-32-544
serestoreprivilege = *S-1-5-32-544
sesecurityprivilege = *S-1-5-32-544
seservicelogonright =
seshutdownprivilege = *S-1-5-32-544
```

```
sesyncagentprivilege =
sesystemenvironmentprivilege = *S-1-5-32-544
sesystemprofileprivilege = *S-1-5-32-544
sesystemtimeprivilege = *S-1-5-32-544
setakeownershipprivilege = *S-1-5-32-544
setcbprivilege =
seundockprivilege =
[Group Membership]
*S-1-5-32-547__Memberof =
*S-1-5-32-547__Members =
[Registry Keys]
"MACHINE\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\AsrCommands",2,"D:PAR(A;CI;KA;;;BA)(A;CI;CCDCLCSWRPSDRC;;;BO)(A;CIIO;KA
;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"MACHINE\SOFTWARE\Microsoft\OS/2 Subsystem for
NT",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"machine\software",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"machine\software\microsoft\netdde",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KA;;;SY)"
"machine\software\microsoft\protected storage system provider",1,"D:AR"
"machine\software\microsoft\windows
nt\currentversion\perflib",2,"D:P(A;CI;GR;;;IU)(A;CI;GA;;;BA)(A;CI;GA;;;SY)(A;CI;GA;;;CO)
"
"machine\software\microsoft\windows\currentversion\group
policy",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CI;KA;;;SY)"
"machine\software\microsoft\windows\currentversion\installer",0,"D:PAR(A;CI;KA;;;BA)(A;CI
;KA;;;SY)(A;CI;KR;;;BU)"
"machine\software\microsoft\windows\currentversion\policies",0,"D:PAR(A;CI;KA;;;BA)(A;CI;
KR;;;AU)(A;CI;KA;;;SY)"
"machine\system",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"machine\system\clone",1,"D:AR"
"machine\system\controlset001",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset002",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset003",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset004",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset005",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset006",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset007",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset008",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset009",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset010",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\currentcontrolset\control\securepipeservers\winreg",2,"D:PAR(A;CI;KA;;;BA
)(A;;KR;;;BO)(A;CI;KA;;;SY)"
"machine\system\currentcontrolset\control\wmi\security",2,"D:P(A;CI;GR;;;BA)(A;CI;GA;;;SY
)(A;CI;GA;;;CO)"
"machine\system\currentcontrolset\enum",1,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CI;KA;;;SY
)"
"machine\system\currentcontrolset\hardware
profiles",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"users\.default",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"users\.default\software\microsoft\netdde",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KA;;;SY)"
"users\.default\software\microsoft\protected storage system provider",1,"D:AR"
"CLASSES_ROOT",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\PermittedManagers",2,"D:PAR(A;
CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\ValidCommunities",2,"D:PAR(A;C
I;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
[File Security]
"%SystemDrive%\Documents and Settings\All
Users",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemRoot%\Offline Web Pages",1,"D:AR(A;OICI;FA;;;WD)"
```

```
"%SystemDrive%\Documents and Settings\All
Users\Documents\DrWatson\drwtsn32.log",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;
FA;;;SY)(A;OICI;0x1301bf;;;BU)"
"%SystemDrive%\Inetpub",1,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI
;0x1200a9;;;BU)"
"%SystemDirectory%\secedit.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\Documents and Settings\All
Users\Documents\DrWatson",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OI
CIIO;DCLCWP;;;BU)(A;OICI;CCSWWPLORC;;;BU)"
"%SystemDrive%\Documents and
Settings\Administrator",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\debug\UserMode",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;;CCDCWP;;;BU)(A;
OIIO;DCLC;;;BU)"
"%SystemDirectory%\regedt32.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rsh.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rexec.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\Ntbackup.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rcp.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\regedit.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\MSDOS.SYS",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDrive%\IO.SYS",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDrive%\",0,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1200
a9;;;BU)"
"%SystemDrive%\Temp",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;CI;DCLC
WP;;;BU)"
"%SystemDrive%\System Volume Information",1,"D:PAR"
"%SystemDrive%\My Download
Files",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1201bf;;;BU)"
"%SystemDrive%\Documents and
Settings",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\ias",2,"D:P(A;OICI;GA;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDirectory%\dllcache",2,"D:P(A;OICI;GA;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDirectory%\config",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\spool\printers",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;
;SY)(A;CI;DCLCSWWPLO;;;BU)"
"%SystemDirectory%\repl\export",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;RE)(A;OICI;FA;
;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\repl\import",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1301bf;;;RE)(A;OICI;FA;
;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\repl",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\ReinstallBackups",1,"D:P(A;OICI;GXGR;;;BU)(A;OICI;GXGR;;;PU)(A;OICI;GA
;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDirectory%\Setup",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\NTMSData",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\GroupPolicy",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;
;;SY)"
"%SystemDirectory%\DTCLog",0,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;O
ICI;0x1200a9;;;BU)"
"%SystemDirectory%\appmgmt",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU
)"
"%SystemDirectory%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1
200a9;;;BU)"
"%SystemRoot%\Temp",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;CI;0x100
026;;;BU)"
"%SystemRoot%\Tasks",1,"D:AR"
"%SystemRoot%\repair",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\Registration",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;FR;;;BU)"
"%SystemRoot%\debug",0,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x
1200a9;;;BU)"
"%SystemRoot%\CSC",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1200a9
;;;BU)"
"%ProgramFiles%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1200
a9;;;BU)"
"c:\config.sys",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)(A;;0x1200a9;;;BU)"
"c:\autoexec.bat",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"c:\ntbootdd.sys",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\ntldr",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\ntdetect.com",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\boot.ini",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"%SystemRoot%\$NtServicePackUninstall$",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
```

```
"%SystemDrive%\autoexec.bat",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;B
U)"
"%SystemDrive%\boot.ini",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\ntdetect.com",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\config.sys",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)
"
"%SystemDrive%\ntldr",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\Documents and Settings\Default
User",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemRoot%\security",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)"
"%SystemDrive%\Program Files\Resource Kit",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
[Version]
signature="$CHICAGO$"
Revision=1
[Profile Description]
Description=Server Audit Template for GIAC Practicum
[Service General Setting]
MSFTPSVC,4,"D:AR(D;;RPWPDTWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;WPWDWO;;;WD)"
IISADMIN,4,"D:AR(D;;RPWPDTWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
SharedAccess,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RasAuto,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RasMan,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RemoteAccess,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
SMTPSVC,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
TlntSvr,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
W3SVC,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
DNS,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
NtFrs,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
IsmServ,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
kdc,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
LicenseService,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
LPDSVC,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
[Registry Values]
MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDriveTypeAutoRun=4,
255
MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\AutoAdminLogon=4,0
machine\system\currentcontrolset\services\netlogon\parameters\signsecurechannel=4,1
machine\system\currentcontrolset\services\netlogon\parameters\sealsecurechannel=4,1
machine\system\currentcontrolset\services\netlogon\parameters\requirestrongkey=4,0
machine\system\currentcontrolset\services\netlogon\parameters\requiresignorseal=4,0
machine\system\currentcontrolset\services\netlogon\parameters\disablepasswordchange=4,0
machine\system\currentcontrolset\services\lanmanworkstation\parameters\requiresecuritysig
nature=4,0
machine\system\currentcontrolset\services\lanmanworkstation\parameters\enablesecuritysign
ature=4,1
machine\system\currentcontrolset\services\lanmanworkstation\parameters\enableplaintextpas
sword=4,0
machine\system\currentcontrolset\services\lanmanserver\parameters\requiresecuritysignatur
e=4,0
machine\system\currentcontrolset\services\lanmanserver\parameters\enablesecuritysignature
=4,1
machine\system\currentcontrolset\services\lanmanserver\parameters\autodisconnect=4,30
machine\system\currentcontrolset\control\session manager\protectionmode=4,1
machine\system\currentcontrolset\control\session manager\memory
management\clearpagefileatshutdown=4,1
machine\system\currentcontrolset\control\print\providers\lanman print
services\servers\addprinterdrivers=4,1
machine\system\currentcontrolset\control\lsa\restrictanonymous=4,2
machine\system\currentcontrolset\control\lsa\lmcompatibilitylevel=4,5
machine\system\currentcontrolset\control\lsa\fullprivilegeauditing=3,0
machine\system\currentcontrolset\control\lsa\crashonauditfail=4,1
machine\system\currentcontrolset\control\lsa\auditbaseobjects=4,1
machine\software\microsoft\windows\currentversion\policies\system\shutdownwithoutlogon=4,
0
machine\software\microsoft\windows\currentversion\policies\system\dontdisplaylastusername
=4,1
machine\software\microsoft\windows\currentversion\policies\system\disablecad=4,0
machine\software\microsoft\windows nt\currentversion\winlogon\scremoveoption=1,1
machine\software\microsoft\windows nt\currentversion\winlogon\passwordexpirywarning=4,14
machine\software\microsoft\windows nt\currentversion\winlogon\cachedlogonscount=1,0
machine\software\microsoft\windows nt\currentversion\winlogon\allocatefloppies=1,1
```

```
machine\software\microsoft\windows nt\currentversion\winlogon\allocatedasd=1,0
machine\software\microsoft\windows nt\currentversion\winlogon\allocatecdroms=1,1
machine\software\microsoft\windows nt\currentversion\setup\recoveryconsole\setcommand=4,0
machine\software\microsoft\windows
nt\currentversion\setup\recoveryconsole\securitylevel=4,0
machine\software\microsoft\non-driver signing\policy=3,1
machine\software\microsoft\driver signing\policy=3,1
MACHINE\System\CurrentControlSet\Services\LanManServer\Parameters\EnableForcedLogOff=4,1
MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\System\LegalNoticeText=1,This
is a RealmCo computer system.  This computer system,including all related
equipment,networks and network devices (specifically including Internet access),is
provided only for authorized RealmCo use.  RealmCo computer systems may be monitored for
all lawful purposes,including ensuring that their use is authorized,for management of the
system,to facilitate protection against unauthorized access and to verify security
procedures,survivability,and operation security.  Monitoring include active attacks by
authorized RealmCo employees or contractors to test or verify the security of this
system.  During monitoring,information may be examined,recorded,copied and used for
authorized purposes.  All information,including personal information,placed on or sent
over this system may be monitored.  Use of this RealmCo computer system,authorized or
unauthorized,constitutes consent to monitoring of this system.  Unauthorized use may
subject you to criminal prosecution.  Evidence if unauthorized use collect during
monitoring may be used for administrative,criminal or adverse action.  Use of this system
constitutes consent to monitoring for these purposes.
MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\System\LegalNoticeCaption=1,Co
nsent to Monitoring
```

AUDIT_WORKSTATION.INF

```
[Unicode]
Unicode=yes
[System Access]
MinimumPasswordAge = 1
MaximumPasswordAge = 90
MinimumPasswordLength = 12
PasswordComplexity = 1
PasswordHistorySize = 24
LockoutBadCount = 3
ResetLockoutCount = 15
LockoutDuration = 15
RequireLogonToChangePassword = 1
ClearTextPassword = 0
[System Log]
MaximumLogSize = 2097152
AuditLogRetentionPeriod = 1
RetentionDays = 7
RestrictGuestAccess = 1
[Security Log]
MaximumLogSize = 2097152
AuditLogRetentionPeriod = 1
RetentionDays = 7
RestrictGuestAccess = 1
[Application Log]
MaximumLogSize = 2097152
AuditLogRetentionPeriod = 1
RetentionDays = 7
RestrictGuestAccess = 1
[Event Audit]
AuditSystemEvents = 3
AuditLogonEvents = 3
AuditObjectAccess = 2
AuditPrivilegeUse = 2
AuditPolicyChange = 3
AuditAccountManage = 3
AuditProcessTracking = 0
AuditDSAccess = 0
AuditAccountLogon = 3
CrashOnAuditFull = 0
[Version]
signature="$CHICAGO$"
Revision=1
[Privilege Rights]
seassignprimarytokenprivilege =
seauditprivilege =
sebackupprivilege = *S-1-5-32-544
sebatchlogonright =
sechangenotifyprivilege = *S-1-5-32-545
secreatepagefileprivilege = *S-1-5-32-544
secreatepermanentprivilege =
secreatetokenprivilege =
sedebugprivilege =
sedenybatchlogonright =
sedenyinteractivelogonright =
sedenynetworklogonright =
sedenyservicelogonright =
seenabledelegationprivilege =
seincreasebasepriorityprivilege = *S-1-5-32-544
seincreasequotaprivilege = *S-1-5-32-544
seinteractivelogonright = *S-1-5-32-544,*S-1-5-32-545
seloaddriverprivilege = *S-1-5-32-544
selockmemoryprivilege =
semachineaccountprivilege =
senetworklogonright = *S-1-5-32-544,*S-1-5-32-545
seprofilesingleprocessprivilege = *S-1-5-32-544
seremoteshutdownprivilege = *S-1-5-32-544
serestoreprivilege = *S-1-5-32-544
```

```
sesecurityprivilege = *S-1-5-32-544
seservicelogonright =
seshutdownprivilege = *S-1-5-32-544,*S-1-5-32-545
sesyncagentprivilege =
sesystemenvironmentprivilege = *S-1-5-32-544
sesystemprofileprivilege = *S-1-5-32-544
sesystemtimeprivilege = *S-1-5-32-544
setakeownershipprivilege = *S-1-5-32-544
setcbprivilege =
seundockprivilege = *S-1-5-32-544,*S-1-5-32-545
[Group Membership]
*S-1-5-32-547__Memberof =
*S-1-5-32-547__Members =
[Registry Keys]
"MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\ValidCommunities",2,"D:PAR(A;C
I;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\PermittedManagers",2,"D:PAR(A;
CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"MACHINE\SOFTWARE\Microsoft\OS/2 Subsystem for
NT",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)"
"machine\software",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"machine\software\microsoft\netdde",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KA;;;SY)"
"machine\software\microsoft\protected storage system provider",1,"D:AR"
"machine\software\microsoft\windows
nt\currentversion\perflib",2,"D:P(A;CI;GR;;;IU)(A;CI;GA;;;BA)(A;CI;GA;;;SY)(A;CI;GA;;;CO)
"
"machine\software\microsoft\windows\currentversion\group
policy",0,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CI;KA;;;SY)"
"machine\software\microsoft\windows\currentversion\installer",0,"D:PAR(A;CI;KA;;;BA)(A;CI
;KA;;;SY)(A;CI;KR;;;BU)"
"machine\software\microsoft\windows\currentversion\policies",0,"D:PAR(A;CI;KA;;;BA)(A;CI;
KR;;;AU)(A;CI;KA;;;SY)"
"machine\system",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"machine\system\clone",1,"D:AR"
"machine\system\controlset001",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset002",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset003",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset004",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset005",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset006",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset007",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset008",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset009",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\controlset010",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;
KR;;;BU)"
"machine\system\currentcontrolset\control\securepipeservers\winreg",2,"D:PAR(A;CI;KA;;;BA
)(A;;KR;;;BO)(A;CI;KA;;;SY)"
"machine\system\currentcontrolset\control\wmi\security",2,"D:P(A;CI;GR;;;BA)(A;CI;GA;;;SY
)(A;CI;GA;;;CO)"
"machine\system\currentcontrolset\enum",1,"D:PAR(A;CI;KA;;;BA)(A;CI;KR;;;AU)(A;CI;KA;;;SY
)"
"machine\system\currentcontrolset\hardware
profiles",0,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"users\.default",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"users\.default\software\microsoft\netdde",2,"D:PAR(A;CI;KA;;;BA)(A;CI;KA;;;SY)"
"users\.default\software\microsoft\protected storage system provider",1,"D:AR"
"CLASSES_ROOT",2,"D:PAR(A;CI;KA;;;BA)(A;CIIO;KA;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
"MACHINE\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\AsrCommands",2,"D:PAR(A;CI;KA;;;BA)(A;CI;CCDCLCSWRPSDRC;;;BO)(A;CIIO;KA
;;;CO)(A;CI;KA;;;SY)(A;CI;KR;;;BU)"
[File Security]
"%SystemDrive%\Program Files\Resource Pro Kit",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
```

```
"%SystemRoot%\security",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)"
"%SystemDrive%\Documents and Settings\Default
User",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDrive%\ntldr",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\config.sys",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)
"
"%SystemDrive%\ntdetect.com",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\boot.ini",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\autoexec.bat",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;B
U)"
"%SystemRoot%\Offline Web Pages",1,"D:(A;OICI;GA;;;WD)"
"%SystemDrive%\Documents and Settings\All
Users\Documents\DrWatson\drwtsn32.log",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;
FA;;;SY)(A;OICI;0x1301bf;;;BU)"
"%SystemRoot%\$NtServicePackUninstall$",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"c:\boot.ini",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\ntdetect.com",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\ntldr",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\ntbootdd.sys",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)"
"c:\autoexec.bat",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"c:\config.sys",2,"D:PAR(A;;FA;;;BA)(A;;FA;;;SY)(A;;0x1200a9;;;BU)"
"%ProgramFiles%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1200
a9;;;BU)"
"%SystemRoot%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1200a9
;;;BU)"
"%SystemRoot%\CSC",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\debug",0,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x
1200a9;;;BU)"
"%SystemRoot%\Registration",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;FR;;;BU)"
"%SystemRoot%\repair",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\Tasks",1,"D:AR"
"%SystemRoot%\Temp",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;CI;0x100
026;;;BU)"
"%SystemDirectory%",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1
200a9;;;BU)"
"%SystemDirectory%\appmgmt",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU
)"
"%SystemDirectory%\DTCLog",0,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;O
ICI;0x1200a9;;;BU)"
"%SystemDirectory%\GroupPolicy",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;AU)(A;OICI;FA;
;;SY)"
"%SystemDirectory%\NTMSData",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\Setup",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\ReinstallBackups",1,"D:P(A;OICI;GXGR;;;BU)(A;OICI;GXGR;;;PU)(A;OICI;GA
;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDirectory%\repl",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\repl\import",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1301bf;;;RE)(A;OICI;FA;
;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\repl\export",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;0x1200a9;;;RE)(A;OICI;FA;
;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDirectory%\spool\printers",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;
;SY)(A;CI;DCLCSWWPLO;;;BU)"
"%SystemDirectory%\config",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\dllcache",2,"D:P(A;OICI;GA;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDirectory%\ias",2,"D:P(A;OICI;GA;;;BA)(A;OICI;GA;;;SY)(A;OICI;GA;;;CO)"
"%SystemDrive%\Documents and
Settings",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDrive%\My Download
Files",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1201bf;;;BU)"
"%SystemDrive%\System Volume Information",1,"D:PAR"
"%SystemDrive%\Temp",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;CI;DCLC
WP;;;BU)"
"%SystemDrive%\",0,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;0x1200
a9;;;BU)"
"%SystemDrive%\IO.SYS",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemDrive%\MSDOS.SYS",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
"%SystemRoot%\regedit.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rcp.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\Ntbackup.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rexec.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDirectory%\rsh.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
```

```
"%SystemDirectory%\regedt32.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\Documents and
Settings\Administrator",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemDrive%\Documents and Settings\All
Users\Documents\DrWatson",2,"D:PAR(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OI
CIIO;DCLCWP;;;BU)(A;OICI;CCSWWPLORC;;;BU)"
"%SystemDirectory%\secedit.exe",2,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)"
"%SystemRoot%\Debug\UserMode",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;;CCDCWP;;;BU)(A;
OIIO;DCLC;;;BU)"
"%SystemDrive%\Documents and Settings\All
Users",0,"D:PAR(A;OICI;FA;;;BA)(A;OICI;FA;;;SY)(A;OICI;0x1200a9;;;BU)"
[Profile Description]
Description=Workstation Audit Template for GIAC Practicum
[Service General Setting]
MSFTPSVC,4,"D:AR(D;;RPWPDTWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;WPWDWO;;;WD)"
IISADMIN,4,"D:AR(D;;RPWPDTWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
SharedAccess,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RasAuto,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RasMan,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
RemoteAccess,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
SMTPSVC,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
TlntSvr,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
W3SVC,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
DNS,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
NtFrs,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
IsmServ,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
kdc,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
LicenseService,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
LPDSVC,4,"D:AR(D;;RPWDWO;;;WD)(A;;CCLCSWLOCRRC;;;AU)S:AR(AU;FA;RPWDWO;;;WD)"
[Registry Values]
MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDriveTypeAutoRun=4,
255
MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\AutoAdminLogon=4,0
machine\system\currentcontrolset\services\netlogon\parameters\signsecurechannel=4,1
machine\system\currentcontrolset\services\netlogon\parameters\sealsecurechannel=4,1
machine\system\currentcontrolset\services\netlogon\parameters\requirestrongkey=4,0
machine\system\currentcontrolset\services\netlogon\parameters\requiresignorseal=4,0
machine\system\currentcontrolset\services\netlogon\parameters\disablepasswordchange=4,0
machine\system\currentcontrolset\services\lanmanworkstation\parameters\requiresecuritysig
nature=4,0
machine\system\currentcontrolset\services\lanmanworkstation\parameters\enablesecuritysign
ature=4,1
machine\system\currentcontrolset\services\lanmanworkstation\parameters\enableplaintextpas
sword=4,0
machine\system\currentcontrolset\services\lanmanserver\parameters\requiresecuritysignatur
e=4,0
machine\system\currentcontrolset\services\lanmanserver\parameters\enablesecuritysignature
=4,1
machine\system\currentcontrolset\services\lanmanserver\parameters\enableforcedlogoff=4,1
machine\system\currentcontrolset\services\lanmanserver\parameters\autodisconnect=4,30
machine\system\currentcontrolset\control\session manager\protectionmode=4,1
machine\system\currentcontrolset\control\session manager\memory
management\clearpagefileatshutdown=4,1
machine\system\currentcontrolset\control\print\providers\lanman print
services\servers\addprinterdrivers=4,0
machine\system\currentcontrolset\control\lsa\restrictanonymous=4,2
machine\system\currentcontrolset\control\lsa\lmcompatibilitylevel=4,5
machine\system\currentcontrolset\control\lsa\fullprivilegeauditing=3,0
machine\system\currentcontrolset\control\lsa\crashonauditfail=4,0
machine\system\currentcontrolset\control\lsa\auditbaseobjects=4,1
machine\software\microsoft\windows\currentversion\policies\system\shutdownwithoutlogon=4,
0
machine\software\microsoft\windows\currentversion\policies\system\dontdisplaylastusername
=4,1
machine\software\microsoft\windows\currentversion\policies\system\disablecad=4,0
machine\software\microsoft\windows nt\currentversion\winlogon\scremoveoption=1,1
machine\software\microsoft\windows nt\currentversion\winlogon\passwordexpirywarning=4,14
machine\software\microsoft\windows nt\currentversion\winlogon\cachedlogonscount=1,1
machine\software\microsoft\windows nt\currentversion\winlogon\allocatefloppies=1,1
machine\software\microsoft\windows nt\currentversion\winlogon\allocatedasd=1,0
machine\software\microsoft\windows nt\currentversion\winlogon\allocatecdroms=1,1
```

```
machine\software\microsoft\windows nt\currentversion\setup\recoveryconsole\setcommand=4,0
machine\software\microsoft\windows
nt\currentversion\setup\recoveryconsole\securitylevel=4,0
machine\software\microsoft\non-driver signing\policy=3,1
machine\software\microsoft\driver signing\policy=3,1
```

REPORTDEFINITONS.TXT

```
[System Access]
minimumpasswordage     |       <li><b>Minimum password age</b> is incorrectly set
maximumpasswordage     |       <li><b>Maximum password age</b> is incorrectly set
minimumpasswordlength  |       <li><b>Minimum password length</b> is incorrectly set
passwordcomplexity     |       <li><b>PASSFLT.DLL is not enabled</b>
passwordhistorysize    |       <li><b>Password history</b> is incorrectly set
lockoutbadcount        |       <li><b>Account lockout threshold</b> is incorrectly set
resetlockoutcount      |       <li><b>Reset lockout account</b> counter is incorrectly set
lockoutduration        |       <li><b>Account lockout duration</b> is incorrectly set
requirelogontochangepassword  |       <li><b>Logon is not required in order to change
pasword</b>
cleartextpassword      |       <li><b>Passwords are being stored using reversible
encryption for all users in the domain</b>
maxticketage   |       <li><b>Maximum lifetime for user ticket</b> is incorrectly set
maxrenewage    |       <li><b>Maximum lifetime for user ticket renewal</b> is incorrectly
set
maxserviceage  |       <li><b>Maximum lifetime for service ticket</b> is incorrectly set
maxclockskew   |       <li><b>Maximum tolerance for computer clock synchronization</b> is
incorrectly set
ticketvalidateclient   |       <li><b>User logon restrictions are not being enforced</b>
[Logs]
maximumlogsize |       <li><b>Maximum log size</b> incorrectly configured
auditlogretentionperiod        |       <li><b>Log retention method</b> incorrectly
configured
retentiondays  |       <li><b>Log retention period</b> incorrectly configured
restrictguestaccess    |       <li><b>Guest access to log</b> incorrectly configured
[Event Audit]
auditsystemevents      |       <li><b>Audit system events</b> is incorrectly set
auditlogonevents       |       <li><b>Audit logon events</b> is incorrectly set
auditobjectaccess      |       <li><b>Audit object access</b> is incorrectly set
auditprivilegeuse      |       <li><b>Audit privilege use</b> is incorrectly set
auditpolicychange      |       <li><b>Audit policy change</b> is incorrectly set
auditaccountmanage     |       <li><b>Audit account management</b> is incorrectly set
auditprocesstracking   |       <li><b>Audit process tracking</b> is incorrectly set
auditdsaccess  |       <li><b>Audit directory services access</b> is incorrectly set
auditaccountlogon      |       <li><b>Audit account logon events</b> is incorrectly set
crashonauditfull       |       <li><b>Crash system if security audit log</b> is full is
incorrectly set
[Privilege Rights]
seassignprimarytokenprivilege |       <li>The <b>replace a process level token</b> user
privilege right is inccorectly assigned
seauditprivilege       |       <li>The <b>generate security audits</b> user privilege
right is inccorectly assigned
sebackupprivilege      |       <li>The <b>backup files and directories</b> user privilege
right is inccorectly assigned
sebatchlogonright      |       <li>The <b>log on as a bactch job</b> user privilege right
is inccorectly assigned
sechangenotifyprivilege        |       <li>The <b>bypass traverse checking</b> user
privilege right is inccorectly assigned
secreatepagefileprivilege      |       <li>The <b>create a pagefile</b> user privilege
right is inccorectly assigned
secreatepermanentprivilege     |       <li>The <b>create permanent shared objects</b> user
privilege right is inccorectly assigned
secreatetokenprivilege |       <li>The <b>create a token object</b> user privilege right
is inccorectly assigned
sedebugprivilege       |       <li>The <b>debug programs</b> user privilege right is
inccorectly assigned
sedenybatchlogonright  |       <li>The <b>deny logon as a batch job</b> user privilege
right is inccorectly assigned
sedenyinteractivelogonright    |       <li>The <b>deny logon locally</b> user privilege
right is inccorectly assigned
sedenynetworklogonright        |       <li>The <b>deny network logon</b> user privilege
right is inccorectly assigned
sedenyservicelogonright        |       <li>The <b>deny login as a service</b> user
privilege right is inccorectly assigned
seenabledelegationprivilege    |       <li>The <b>enable computer and user accounts to be
trusted for delegation</b> user privilege right is inccorectly assigned
```

```
seincreasebasepriorityprivilege       |       <li>The <b>increase scheduling priority</b>
user privilege right is inccorectly assigned
seincreasequotaprivilege      |       <li>The <b>increase quotas</b> user privilege right
is inccorectly assigned
seinteractivelogonright       |       <li>The <b>logon locally</b> user privilege right is
inccorectly assigned
seloaddriverprivilege  |       <li>The <b>load and unload device drivers</b> user
privilege right is inccorectly assigned
selockmemoryprivilege  |       <li>The <b>lock pages in memory</b> user privilege right is
inccorectly assigned
semachineaccountprivilege     |       <li>The <b>add workstations to domain</b> user
privilege right is inccorectly assigned
senetworklogonright    |       <li>The <b>access this computer from the network</b> user
privilege right is inccorectly assigned
seprofilesingleprocessprivilege       |       <li>The <b>profile single process</b> user
privilege right is inccorectly assigned
seremoteshutdownprivilege     |       <li>The <b>force shutdown from a remote system</b>
user privilege right is inccorectly assigned
serestoreprivilege     |       <li>The <b>restore files and directories</b> user privilege
right is inccorectly assigned
sesecurityprivilege    |       <li>The <b>manage auditing and security log</b> user
privilege right is inccorectly assigned
seservicelogonright    |       <li>The <b>log on as a service</b> user privilege right is
inccorectly assigned
seshutdownprivilege    |       <li>The <b>shut down the system</b> user privilege right is
inccorectly assigned
sesyncagentprivilege   |       <li>The <b>synchronize directory service data</b> user
privilege right is inccorectly assigned
sesystemenvironmentprivilege  |       <li>The <b>modify firmware environment values</b>
user privilege right is inccorectly assigned
sesystemprofileprivilege      |       <li>The <b>profile system performance</b> user
privilege right is inccorectly assigned
sesystemtimeprivilege  |       <li>The <b>change the system time</b> user privilege right
is inccorectly assigned
setakeownershipprivilege      |       <li>The <b>take ownership of files or other
objects</b> user privilege right is inccorectly assigned
setcbprivilege |       <li>The <b>act as part of the operating system</b> user privilege
right is inccorectly assigned
seundockprivilege      |       <li>The <b>remove computer from docking station</b> user
privilege right is inccorectly assigned
[Registry Keys]
machine\software\microsoft\os/2 subsystem for nt     |       <li>Registry permissions are
incorrectly set on <b>HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\OS/2 Subsystem for NT</b>
machine\software       |       <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\software</b>
machine\software\microsoft\netdde     |       <li>Registry permissions are incorrectly set
on <b>HKEY_LOCAL_MACHINE\software\microsoft\netdde</b>
machine\software\microsoft\protected storage system provider        |       <li>Registry
permissions are incorrectly set on <b>HKEY_LOCAL_MACHINE\software\microsoft\protected
storage system provider</b>
machine\software\microsoft\windows nt\currentversion\perflib        |       <li>Registry
permissions are incorrectly set on <b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\perflib</b>
machine\software\microsoft\windows\currentversion\group policy      |       <li>Registry
permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\group policy</b>
machine\software\microsoft\windows\currentversion\installer  |       <li>Registry
permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\installer</b>
machine\software\microsoft\windows\currentversion\policies   |       <li>Registry
permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\policies</b>
machine\system |       <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system</b>
machine\system\clone   |       <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\clone</b>
machine\system\controlset001   |       <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset001</b>
machine\system\controlset002   |       <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset002</b>
```

```
machine\system\controlset003  |      <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset003</b>
machine\system\controlset004  |      <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset004</b>
machine\system\controlset005  |      <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset005</b>
machine\system\controlset006  |      <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset006</b>
machine\system\controlset007  |      <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset007</b>
machine\system\controlset008  |      <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset008</b>
machine\system\controlset009  |      <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset009</b>
machine\system\controlset010  |      <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\controlset010</b>
machine\system\currentcontrolset\control\securepipeservers\winreg   |      <li>Registry
permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\securepipeservers\winreg</b>
machine\system\currentcontrolset\control\wmi\security       |      <li>Registry
permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\wmi\security</b>
machine\system\currentcontrolset\enum |      <li>Registry permissions are incorrectly set
on <b>HKEY_LOCAL_MACHINE\system\currentcontrolset\enum</b>
machine\system\currentcontrolset\hardware profiles   |      <li>Registry permissions are
incorrectly set on <b>HKEY_LOCAL_MACHINE\system\currentcontrolset\hardware profiles</b>
machine\software\microsoft\windows nt\currentversion\asrcommands    |      <li>Registry
permissions are incorrectly set on <b>HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\AsrCommands</b>
machine\system\currentcontrolset\services\snmp\parameters\permittedmanagers        |
       <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\PermittedManagers
</b>
machine\system\currentcontrolset\services\snmp\parameters\validcommunities  |
       <li>Registry permissions are incorrectly set on
<b>HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SNMP\Parameters\ValidCommunities<
/b>
users\.default |      <li>Registry permissions are incorrectly set on
<b>HKEY_USERS\.default</b>
users\.default\software\microsoft\netdde     |      <li>Registry permissions are
incorrectly set on <b>HKEY_USERS\.default\software\microsoft\netdde</b>
users\.default\software\microsoft\protected storage system provider |      <li>Registry
permissions are incorrectly set on <b>HKEY_USERS\.default\software\microsoft\protected
storage system provider</b>
classes_root   |      <li>Registry permissions are incorrectly set on
<b>HKEY_CLASSES_ROOT</b>
[Registry Values]
machine\software\microsoft\windows\currentversion\policies\system\legalnoticecaption
       |      <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\System\LegalNoti
ceCaption</b>
machine\software\microsoft\windows\currentversion\policies\system\legalnoticetext  |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\System\LegalNoti
ceText</b>
machine\software\microsoft\windows\currentversion\policies\explorer\nodrivetypeautorun
       |      <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDrive
TypeAutoRun</b>
machine\software\microsoft\driver signing\policy     |      <li>Incorrect value for the
following registry key: <b>HKEY_LOCAL_MACHINE\software\microsoft\driver
signing\policy</b>
machine\software\microsoft\non-driver signing\policy |      <li>Incorrect value for the
following registry key: <b>HKEY_LOCAL_MACHINE\software\microsoft\non-driver
signing\policy</b>
machine\software\microsoft\windows nt\currentversion\setup\recoveryconsole\securitylevel
       |      <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\setup\recoveryconsole\securitylevel</b>
machine\software\microsoft\windows nt\currentversion\setup\recoveryconsole\setcommand
       |      <li>Incorrect value for the following registry key:
```

```
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\setup\recoveryconsole\setcommand</b>
machine\software\microsoft\windows nt\currentversion\winlogon\allocatecdroms       |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\winlogon\allocatecdroms</b>
machine\software\microsoft\windows nt\currentversion\winlogon\allocatedasd  |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\winlogon\allocatedasd</b>
machine\software\microsoft\windows nt\currentversion\winlogon\allocatefloppies    |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\winlogon\allocatefloppies</b>
machine\software\microsoft\windows nt\currentversion\winlogon\cachedlogonscount    |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\winlogon\cachedlogonscount</b>
machine\software\microsoft\windows nt\currentversion\winlogon\passwordexpirywarning
       |      <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\winlogon\passwordexpirywarning</b>
machine\software\microsoft\windows nt\currentversion\winlogon\scremoveoption       |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows
nt\currentversion\winlogon\scremoveoption</b>
machine\software\microsoft\windows\currentversion\policies\system\disablecad       |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\policies\system\disableca
d</b>
machine\software\microsoft\windows\currentversion\policies\system\dontdisplaylastusername
       |      <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\policies\system\dontdispl
aylastusername</b>
machine\software\microsoft\windows\currentversion\policies\system\shutdownwithoutlogon
       |      <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\policies\system\shutdownw
ithoutlogon</b>
machine\system\currentcontrolset\control\lsa\auditbaseobjects       |      <li>Incorrect
value for the following registry key:
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\lsa\auditbaseobjects</b>
machine\system\currentcontrolset\control\lsa\crashonauditfail       |      <li>Incorrect
value for the following registry key:
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\lsa\crashonauditfail</b>
machine\system\currentcontrolset\control\lsa\fullprivilegeauditing  |      <li>Incorrect
value for the following registry key:
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\lsa\fullprivilegeauditing</b>
machine\system\currentcontrolset\control\lsa\lmcompatibilitylevel   |      <li>Incorrect
value for the following registry key:
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\lsa\lmcompatibilitylevel</b>
machine\system\currentcontrolset\control\lsa\restrictanonymous      |      <li>Incorrect
value for the following registry key:
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\lsa\restrictanonymous</b>
machine\system\currentcontrolset\control\print\providers\lanman print
services\servers\addprinterdrivers    |      <li>Incorrect value for the following
registry key:
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\print\providers\lanman print
services\servers\addprinterdrivers</b>
machine\system\currentcontrolset\control\session manager\memory
management\clearpagefileatshutdown    |      <li>Incorrect value for the following
registry key: <b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\session
manager\memory management\clearpagefileatshutdown</b>
machine\system\currentcontrolset\control\session manager\protectionmode     |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\control\session manager\protectionmode</b>
machine\system\currentcontrolset\services\lanmanserver\parameters\autodisconnect   |
       <li>Incorrect value for the following registry key:
<b>HKEY_LOCAL_MACHINE\system\currentcontrolset\services\lanmanserver\parameters\autodisco
nnect</b>
machine\system\currentcontrolset\services\lanmanserver\parameters\enableforcedlogoff
       |      <li>Incorrect value for the following registry key:
```

HKEY_LOCAL_MACHINE\system\currentcontrolset\services\lanmanserver\parameters\enableforcedlogoff
machine\system\currentcontrolset\services\lanmanserver\parameters\enablesecuritysignature | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\lanmanserver\parameters\enablesecuritysignature
machine\system\currentcontrolset\services\lanmanserver\parameters\requiresecuritysignature | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\lanmanserver\parameters\requiresecuritysignature
machine\system\currentcontrolset\services\lanmanworkstation\parameters\enableplaintextpassword | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\lanmanworkstation\parameters\enableplaintextpassword
machine\system\currentcontrolset\services\lanmanworkstation\parameters\enablesecuritysignature | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\lanmanworkstation\parameters\enablesecuritysignature
machine\system\currentcontrolset\services\lanmanworkstation\parameters\requiresecuritysignature | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\lanmanworkstation\parameters\requiresecuritysignature
machine\system\currentcontrolset\services\netlogon\parameters\disablepasswordchange | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\netlogon\parameters\disablepasswordchange
machine\system\currentcontrolset\services\netlogon\parameters\requiresignorseal | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\netlogon\parameters\requiresignorseal
machine\system\currentcontrolset\services\netlogon\parameters\requirestrongkey | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\netlogon\parameters\requirestrongkey
machine\system\currentcontrolset\services\netlogon\parameters\sealsecurechannel | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\netlogon\parameters\sealsecurechannel
machine\system\currentcontrolset\services\netlogon\parameters\signsecurechannel | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\system\currentcontrolset\services\netlogon\parameters\signsecurechannel
machine\system\currentcontrolset\control\lsa\submitcontrol | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SubmitControl
machine\software\microsoft\windows nt\currentversion\winlogon\autoadminlogon | Incorrect value for the following registry key: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\AutoAdminLogon
[File Security]
%systemdrive%\documents and settings\all users | File permissions are incorrectly set on %SystemDrive%\Documents and Settings\All Users
%systemdrive%\ntldr | File permissions are incorrectly set on %SystemDrive%\ntldr
%systemdrive%\config.sys | File permissions are incorrectly set on %SystemDrive%\config.sys
%systemdrive%\ntdetect.com | File permissions are incorrectly set on %SystemDrive%\ntdetect.com
%systemdrive%/boot.ini | File permissions are incorrectly set on %SystemDrive%/boot.ini
%systemdrive%\autoexec.bat | File permissions are incorrectly set on %SystemDrive%\autoexec.bat
%systemroot%\$ntservicepackuninstall$ | File permissions are incorrectly set on %SystemRoot%\$NtServicePackUninstall$
c:\boot.ini | File permissions are incorrectly set on c:\boot.ini
c:\ntdetect.com | File permissions are incorrectly set on c:\ntdetect.com
c:\ntldr | File permissions are incorrectly set on c:\ntldr
c:\ntbootdd.sys | File permissions are incorrectly set on c:\ntbootdd.sys

```
c:\autoexec.bat        |      <li>File permissions are incorrectly set on
<b>c:\autoexec.bat</b>
c:\config.sys  |      <li>File permissions are incorrectly set on <b>c:\config.sys</b>
%programfiles% |      <li>File permissions are incorrectly set on <b>%ProgramFiles%</b>
%systemroot%   |      <li>File permissions are incorrectly set on <b>%SystemRoot%</b>
%systemroot%\csc       |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\CSC</b>
%systemroot%\debug     |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\debug</b>
%systemroot%\registration      |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\Registration</b>
%systemroot%\repair    |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\repair</b>
%systemroot%\tasks     |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\Tasks</b>
%systemroot%\temp      |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\Temp</b>
%systemdirectory%      |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%</b>
%systemdirectory%\appmgmt      |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\appmgmt</b>
%systemdirectory%\dtclog       |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\DTCLog</b>
%systemdirectory%\grouppolicy  |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\GroupPolicy</b>
%systemdirectory%\ntmsdata     |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\NTMSData</b>
%systemdirectory%\setup        |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\Setup</b>
%systemdirectory%\reinstallbackups     |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\ReinstallBackups</b>
%systemdirectory%\repl |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\repl\import</b>
%systemdirectory%\repl\import  |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\repl\import</b>
%systemdirectory%\repl\export  |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\repl\export</b>
%systemdirectory%\spool\printers       |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\spool\printers</b>
%systemdirectory%\config       |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\config</b>
%systemdirectory%\dllcache     |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\dllcache</b>
%systemdirectory%\ias  |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\ias</b>
%systemdrive%\documents and settings  |      <li>File permissions are incorrectly set on
<b>%SystemDrive%\Documents and Settings</b>
%systemdrive%\my download files        |      <li>File permissions are incorrectly set on
<b>%SystemDrive%\My Download Files</b>
%systemdrive%\system volume information       |      <li>File permissions are incorrectly
set on <b>%SystemDrive%\System Volume Information</b>
%systemdrive%\temp     |      <li>File permissions are incorrectly set on
<b>%SystemDrive%\Temp</b>
%systemdrive%\ |      <li>File permissions are incorrectly set on <b>%SystemDrive%\</b>
%systemdrive%\io.sys   |      <li>File permissions are incorrectly set on
<b>%SystemDrive%\IO.SYS</b>
%systemdrive%\msdos.sys        |      <li>File permissions are incorrectly set on
<b>%SystemDrive%\MSDOS.SYS</b>
%systemroot%\regedit.exe       |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\regedit.exe</b>
%systemdirectory%\rcp.exe      |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\rcp.exe</b>
%systemdirectory%\ntbackup.exe         |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\Ntbackup.exe</b>
%systemdirectory%\rexec.exe    |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\rexec.exe</b>
%systemdirectory%\rsh.exe      |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\rsh.exe</b>
%systemdirectory%\regedt32.exe         |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\regedt32.exe</b>
```

```
%systemroot%\debug\usermode   |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\debug\UserMode</b>
%systemdrive%\documents and settings\administrator   |      <li>File permissions are
incorrectly set on <b>%SystemDrive%\Documents and Settings\Administrator</b>
%systemdrive%\documents and settings\all users\documents\drwatson   |      <li>File
permissions are incorrectly set on <b>%SystemDrive%\Documents and Settings\All
Users\Documents\DrWatson</b>
%systemdirectory%\secedit.exe |      <li>File permissions are incorrectly set on
<b>%SystemDirectory%\secedit.exe</b>
%systemdrive%\inetpub  |      <li>File permissions are incorrectly set on
<b>%SystemDrive%\Inetpub</b>
%systemdrive%\documents and settings\all users\documents\drwatson\drwtsn32.log     |
       <li>File permissions are incorrectly set on <b>%SystemDrive%\Documents and
Settings\All Users\Documents\DrWatson\drwtsn32.log</b>
%systemroot%\offline web pages       |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\Offline Web Pages</b>
%systemroot%\ntds      |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\NTDS</b>
%systemroot%\sysvol    |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\SYSVOL</b>
%systemroot%\sysvol\domain\policies   |      <li>File permissions are incorrectly set on
<b>%SystemRoot%\SYSVOL\domain\Policies</b>
%systemdrive%\documents and settings\default user    |      <li>File permissions are
incorrectly set on <b><b>%SystemDrive%\Documents and Settings\Default User</b>
%systemroot%\security  |      <li>File permissions are incorrectly set on
<b><b>%SystemRoot%\security</b>
%systemdrive%\program files\resource kit     |      <li>File permissions are incorrectly
set on <b><b>%SystemDrive%\Program Files\Resource Kit</b>
[Group Membership]
*s-1-5-32-546__memberof       |      <li>The <b>Guests</b> group is forbidden by policy
to belong to any groups
*s-1-5-32-546__members |      <li>The <b>Guests</b> group is forbidden by policy to
contain any members
*s-1-5-32-547__memberof       |      <li>The <b>Power Users</b> group is forbidden by
policy to belong to any groups
*s-1-5-32-547__members |      <li>The <b>Power Users</b> group is forbidden by policy to
contain any members
*s-1-5-32-551__memberof       |      <li>The <b>Backup Operators</b> group is forbidden
by policy to belong to any groups
*s-1-5-32-551__members |      <li>The <b>Backup Operators</b> group is forbidden by
policy to contain any members
*s-1-5-32-549__memberof       |      <li>The <b>Server Operators</b> group is forbidden
by policy to belong to any groups
*s-1-5-32-549__members |      <li>The <b>Server Operators</b> group is forbidden by
policy to contain any members
*s-1-5-32-548__memberof       |      <li>The <b>Account Operators</b> group is forbidden
by policy to belong to any groups
*s-1-5-32-548__members |      <li>The <b>Account Operators</b> group is forbidden by
policy to contain any members
*s-1-5-32-550__memberof       |      <li>The <b>Print Operators</b> group is forbidden by
policy to belong to any groups
*s-1-5-32-550__members |      <li>The <b>Print Operators</b> group is forbidden by policy
to contain any members
pre-windows 2000 compatible access__memberof         |      <li>The <b>Pre-Windows 2000
Compatible Access</b> group is forbidden by policy to belong to any groups
pre-windows 2000 compatible access__members   |      <li>The <b>Pre-Windows 2000
Compatible Access</b> group is forbidden by policy to contain any members
[Service General Setting]
msftpsvc       |      <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>FTP Publishing Service</b>
iisadmin       |      <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>IIS Admin Service</b>
sharedaccess   |      <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Internet Connection Sharing</b>
rasauto |      <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Remote Access Auto Connection Manager</b>
rasman  |      <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Remote Access Connection Manager</b>
remoteaccess   |      <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Routing and Remote Access</b>
```

```
smtpsvc |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Simple Mail Transport Protocol (SMTP)</b>
tlntsvr |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Telnet</b>
w3svc   |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>World Wide Web Publishing Service</b>
dns     |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>DNS Server</b>
ntfrs   |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>File Replication Service</b>
ismserv |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Intersite Messaging</b>
kdc     |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Kerberos Key Distribution Center</b>
licenseservice |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>License Logging Service</b>
lpdsvc  |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>TCP/IP Print Server (LPD)</b>
termservice     |       <li>The following service is either disabled and/or has incorrectly
configured permsissions: <b>Terminal Services</b>
```