



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

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

Interested in learning more?

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

Centralized Auditing of a Windows NT Computer

by Steven Toy

Disclaimer

The information contained in this paper is not advice and should not be construed as an endorsement of any specific product or company.

Introduction

System auditing is important in order to capture general information about system use as well as attempted security violations. It is also important to centralize the management and reporting of system auditing in order to maximize the usefulness of the information. The goal of this paper is to explain step-by-step how to enable Windows NT auditing and how to collect and examine the audit logs. For the purpose of the paper the scope of the auditing is a Windows NT network running Windows NT Workstation and Server with medium security requirements.

Windows NT has extensive built-in auditing. The challenge when auditing in an NT network is not enabling system level auditing, but managing the file level auditing and figuring out what to do with the information once it is logged. Auditing can create an enormous amount of data that is located on thousands of computers. To yield the greatest results, audit data should be compiled to as few machines as possible and then used to create reports that are understandable.

First Step: Deciding What Auditing You Need

Auditing in Windows NT is broken down into seven different categories. Auditing can be enabled for all of the seven categories for both successful attempts and/or failed attempts. The following are the seven categories and an explanation of what they are for:

Logon and Logoff: These events describe a single logon or logoff attempt, whether successful or unsuccessful. Included in each logon description is an indication of what type of logon was requested or performed (that is, interactive, network, or service).

File and Object Access: These events describe both successful and unsuccessful accesses to protected files and objects.

Use of User Rights: These events describe both successful and unsuccessful attempts to use privileges. It also includes information about when some special privileges are assigned. These special privileges are audited only at assignment time, not at time of use.

User and Group Management: These events describe high-level changes to the user accounts database, such as User Created or Group Membership Change. Potentially, a more detailed, object-level audit is also performed (see Object Access events).

Security Policy Changes: These events describe high-level changes to the security policy database, such as assignment of privileges or logon capabilities. Potentially, a more detailed, object-level audit is also performed (see Object Access events).

Restart, Shutdown, and System: These events indicate something affecting the security of the entire system or audit log occurred.

Process Tracking: These events provide detailed subject-tracking information. This includes information such as program activation, handle duplication, and indirect object access.

The best method for deciding what auditing is needed in your environment is to decide how you would use the information obtained through the auditing. Remember when deciding what auditing you need that the more auditing you do, the more CPU and disk resources are used. Especially since your auditing should be done network wide the impact to CPU utilization and disk usage can really add up.

We will assume we are enabling auditing on an NT network with medium security. I will go step by step and explain to you the reasons I am using the auditing settings I use throughout the rest of this paper.

Logon and Logoff: Successful logon and logoff audit logs can be helpful in placing people at certain locations when doing security investigations, therefore I am going to enable successful logon and logoff auditing. I also would like to know when/if someone is trying to break into a computer, therefore I will also enable logon and logoff failures.

File and Object Access: File Access auditing can be useful for files where multiple people may be editing the file, and for cases when sensitive data needs more security than just NTFS file permissions. In some instances there may be regulation within a certain industry saying that certain types of data have to be auditing when they are accessed. In either case, auditing of file access is only done on files that specifically have auditing turned on, so I am going to turn on both success and failure auditing of file accesses.

Use of User Rights: This could possibly create a lot of log entries, but it is important enough to audit. One example of an item that are audited when this is enabled is taking ownership of a file. I am going to audit both success and failure of use of user rights because I feel it is important to know when a user is making the kind of changes associated with use of user rights.

User and Group Management: I certainly want to know when people are put into or taken out of a group. Furthermore I don't believe that this will happen frequently enough

to cause a lot of entries in the logs. I am going to audit both success and failure of User and Group Management.

Security Policy Changes: This is another category that is worth some impact to CPU and disk space. I want to know when security policy is changed and I am going to enable both successful and failed attempts to change security policy.

Restart, Shutdown and System: In a high security situation you may want to know every time a system is shut down and restarted, but in the case of my medium security situation, I don't need to know this and it would just cause additional log entries. I am going to leave this turned off. You may have a specific application or situation that may require you to enable this.

Process Tracking: This is the audit category that is the most granular and also the one that causes the most log entries. This is only needed in high security implementations and should be enabled with caution.

Second Step: Enabling Auditing on the System

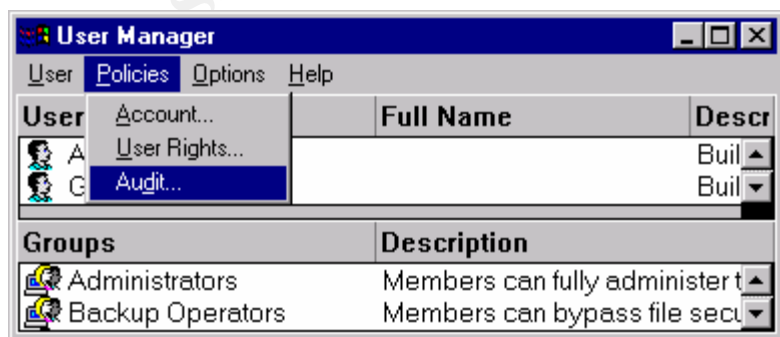
Enclosed are three different ways to enable auditing on a Windows NT system. The first is the manual enabling of auditing using the Regular NT User Manager. The second is the manual enabling of auditing using User Manager for domains. This allows you to remotely enable auditing. The third way is using a WinBatch script and the User Manager for domains.

Note: You have to have Administrator privileges to enable auditing in Windows NT.

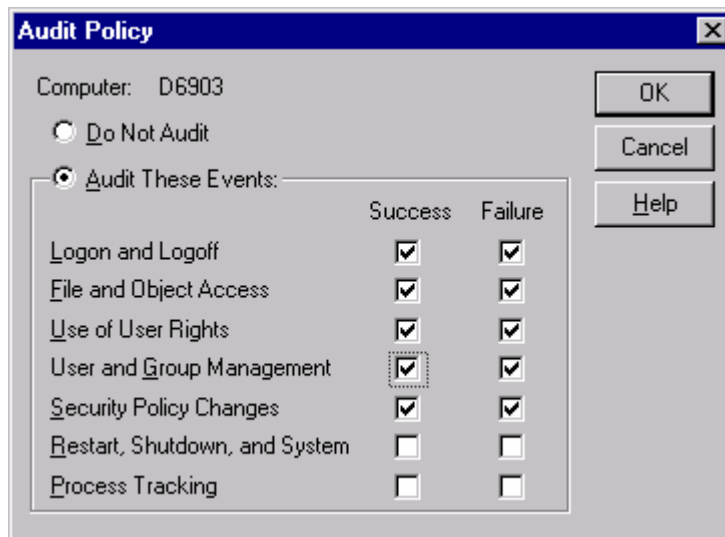
Using Regular User Manager

First, run the User Manager (start -> run -> musrmgr.exe):

Next, go to Policies -> Audit as shown below



Next click on 'Audit These Events' and then check which events you want to audit.

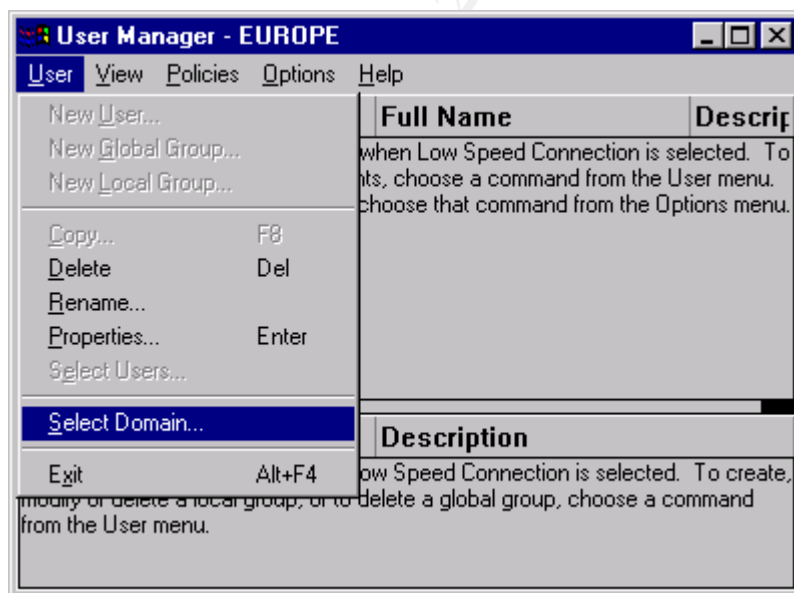


You are done.

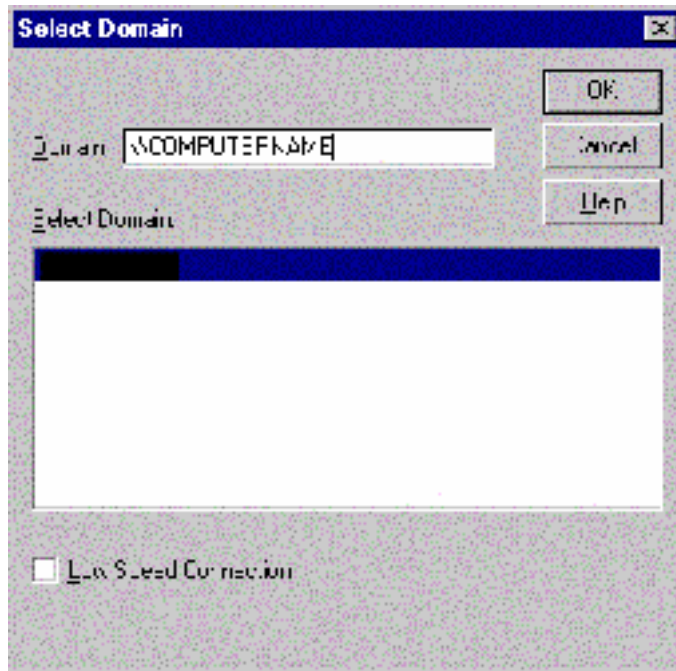
Using User Manager for Domains

First, run User Manager for Domains (start -> run -> usrmgr.exe):

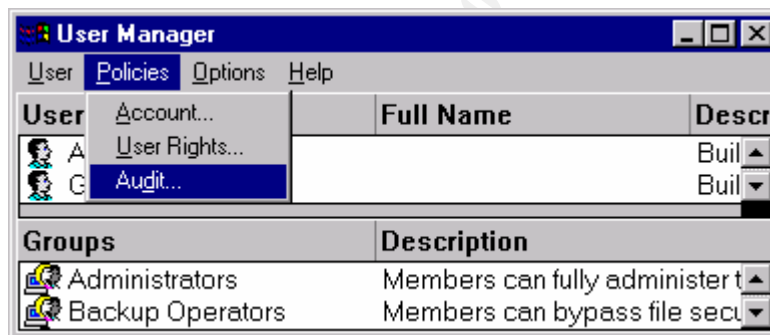
Next, go to User -> Select Domain



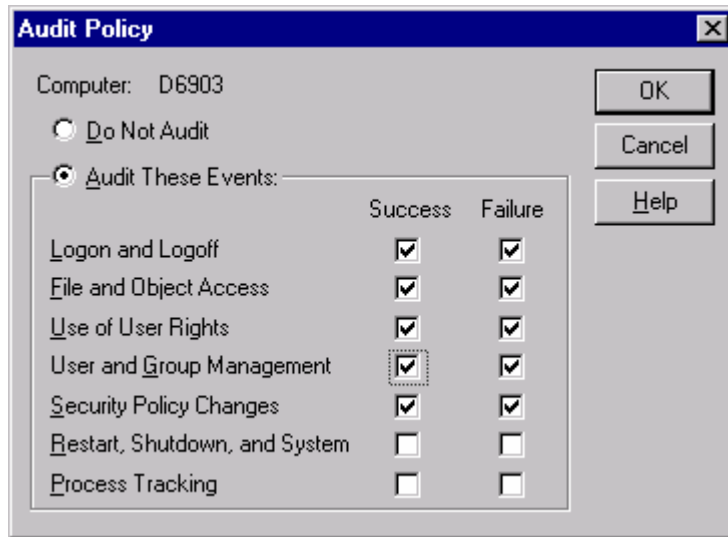
Type in the name of the computer you want to change in the form \\computername, remember that you have to have Administrator privileges on that machine.



Go to Policies -> Audit



Next click on 'Audit These Events' and then check which events you want to audit.



Using a WinBatch Script and User Manager for Domains

The following is a WinBatch script that uses the User Manager for Domains to enable auditing. You have to have WinBatch to run the script. WinBatch is a program that allows you to do Batch style processing within Windows GUI applications. You can even compile the WinBatch scripts and use them on any Windows computer, even one that doesn't have WinBatch loaded. This script takes input from a file (C:\filename.txt) and goes to each computer in the file and enables auditing. The file has one computer name per line in plain text.

```
;this is a .wbt file that can be run by the WinBatch program or
;compiled into a .exe

Run("usrmgr.exe","")

nodenames=fileopen("c:\filename.txt","READ") ; this opens a file
;to read in computer names

while @TRUE
    x=FileRead(nodenames) ; this reads in the computer names
    ; from the above file
    If x == "*EOF*" Then Break

sendkeysto("User Manager","!us") ; brings up the Domain field
sendkeysto("Select Domain","\\") ; types in \\
sendkeysto("Select Domain", x) ; this types in the name of the
```

```

;computer
sendkeysto("Select Domain","{enter}")
sendkeysto("User Manager","!pd") ; this brings up the auditing
;window

sendkeysto("Audit","d")
sendkeysto("Audit","a") ; this enabled auditing
sendkeysto("Audit","l") ; this goes to the log on log off
;auditing tab
sendkeysto("Audit","!c{+}") ; this enables success auditing
sendkeysto("Audit","{tab}")
sendkeysto("Audit","!c{+}") ; this enables failure auditing
sendkeysto("Audit","f") ; this goes to the file access auditing
;tab
sendkeysto("Audit","!c{+}") ; this enables success auditing
sendkeysto("Audit","{tab}")
sendkeysto("Audit","!c{+}") ; this enables failure auditing
sendkeysto("Audit","u") ; this goes to the Use of User rights
;auditing tab
sendkeysto("Audit","!c{+}") ; this enables success auditing
sendkeysto("Audit","{tab}")
sendkeysto("Audit","!c{+}") ; this enables failure auditing
sendkeysto("Audit","g") ; this goes to the User and Group
;Management auditing tab
sendkeysto("Audit","!c{+}") ; this enables success auditing
sendkeysto("Audit","{tab}")
sendkeysto("Audit","!c{+}") ; this enables failure auditing
sendkeysto("Audit","s") ; this goes to the Security Policy
;Changes auditing tab
sendkeysto("Audit","!c{+}") ; this enables success auditing
sendkeysto("Audit","{tab}")
sendkeysto("Audit","!c{+}") ; this enables failure auditing
sendkeysto("Audit","r") ; this goes to the Restart, Shutdown, and
;System auditing tab
sendkeysto("Audit","!c{-}") ; this disables success auditing
sendkeysto("Audit","{tab}")
sendkeysto("Audit","!c{-}") ; this disables failure auditing
sendkeysto("Audit","p") ; this goes to the Process Tracking
;auditing tab
sendkeysto("Audit","!c{-}") ; this disables success auditing
sendkeysto("Audit","{tab}")
sendkeysto("Audit","!c{-}") ; this disables failure auditing

sendkeysto("Audit","{enter}") ; this closes the window and
;accepts the changes

endwhile

sendkeysto("User","!ux")

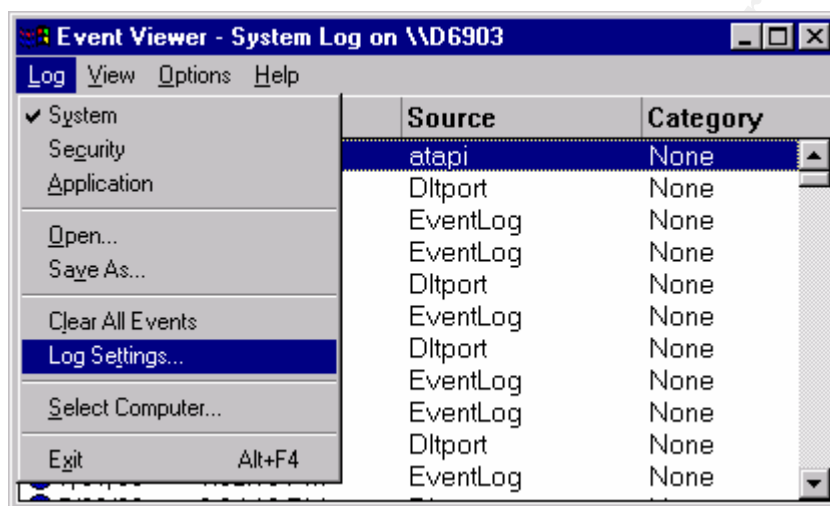
```


Third Step: Configuring the Event Log

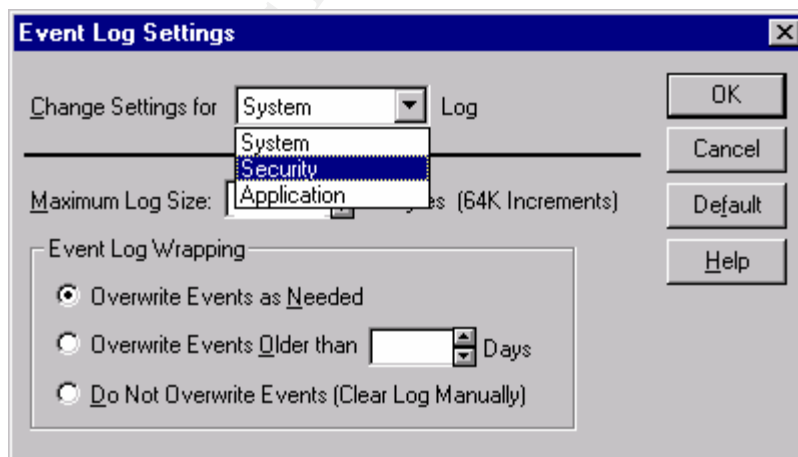
It is important to configure the event log after enabling auditing. Auditing writes events to the event log and can cause the event log to fill up much more quickly than system events can. It is important to make sure you have enough disk space for the event log and it is important to configure what happens when the event log fills up. The event log settings will largely be dependent on how much auditing you have enabled, how secure you want your system to be and how often you plan on retrieving and/or examining the event log on the computer.

To configure the Event Log First open the Event Viewer (Start -> Run -> eventvwr.exe)

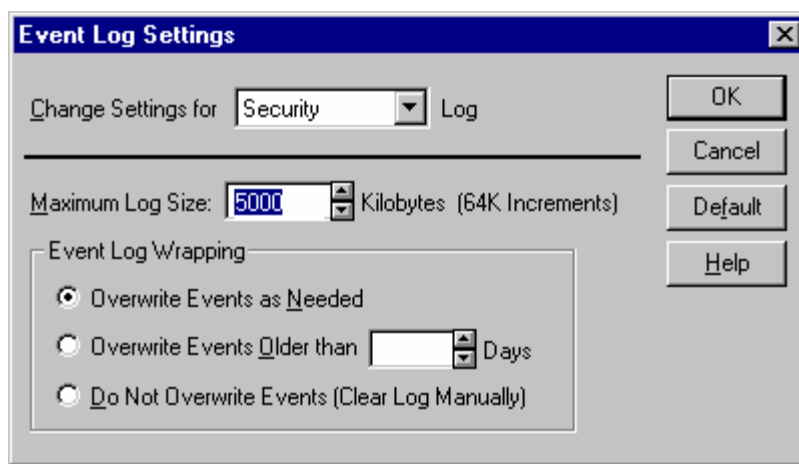
Go to Log -> Log Settings...



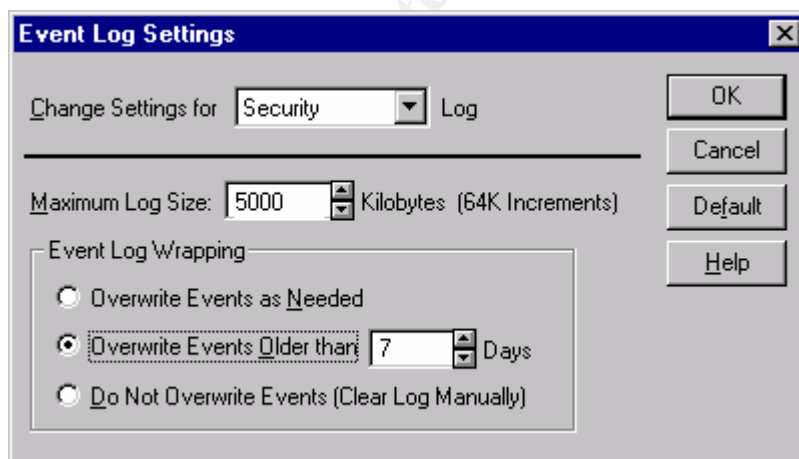
Select 'Change Settings for' Security Log



Select Maximum Log Size and choose a value that you think will be sufficient to hold your Audit logs. Note: If the log size is not in 64K increments it will be rounded to the nearest 64K increment when you click on the OK button.



Select the type of Event Log Wrapping you want. Overwrite events as needed will allow someone to cover up certain audit events by creating a bunch of random audit events. Overwrite events Older than X days will allow audit events to be missed if the log fills up before X days are over. Do not overwrite events means that you will have to manually clear the log when it fills up. There is also a registry setting that will allow you to have the computer crash when the audit log is full. This should only be used for highly secure computers, and still should not be done lightly. Someone can use this feature in a DOS attack by causing a log of audit events. For our case I am going to select Overwrite Events Older than 7 days.



Again, we can do this with a WinBatch script very simply because the Event Viewer allows you to change the settings on remote computers (with the appropriate privileges). Here is an example WinBatch script:

```
;this is a .wbt file that can be run by the WinBatch program or
;compiled into a .exe

Run("eventvwr.exe","")

nodenames=fileopen("c:\filename.txt","READ") ; this opens a file
;to read in computer names

while @TRUE
    x=FileRead(nodenames) ; this reads in the computer names
    ;from the above file
    If x == "*EOF*" Then Break

    sendkeysto("Event Viewer","!ls") ; brings up the Computer field
    sendkeysto("Select","\\") ; types in \\
    sendkeysto("Select", x) ; this types in the name of the
    ;computer
    sendkeysto("Select","{enter}")
    sendkeysto("Event","!lt") ; this brings up the Log Settings
    ;window
    sendkeysto("Event Log","{UP}") ;this brings up the Security
    ;settings
    sendkeysto("Event Log","{UP}") ;this brings up the Security
    ;settings
    sendkeysto("Event Log","{DOWN}") ;this brings up the Security
    ;settings
    sendkeysto("Event Log","{tab}") ; this brings up the Max log size
    sendkeysto("Event Log","5056") ; this types in max log size
    sendkeysto("Event Log","{tab}") ; this brings up Event Log
    ;wrapping
    sendkeysto("Event Log","o") ; this sets overwrite events older
    ;than
    sendkeysto("Event Log","{tab}") ; this brings up the number of
    ;days
    sendkeysto("Event Log","7") ; this types in 7

    sendkeysto("Event Log","{enter}") ; this closes the window and
    ;accepts the changes

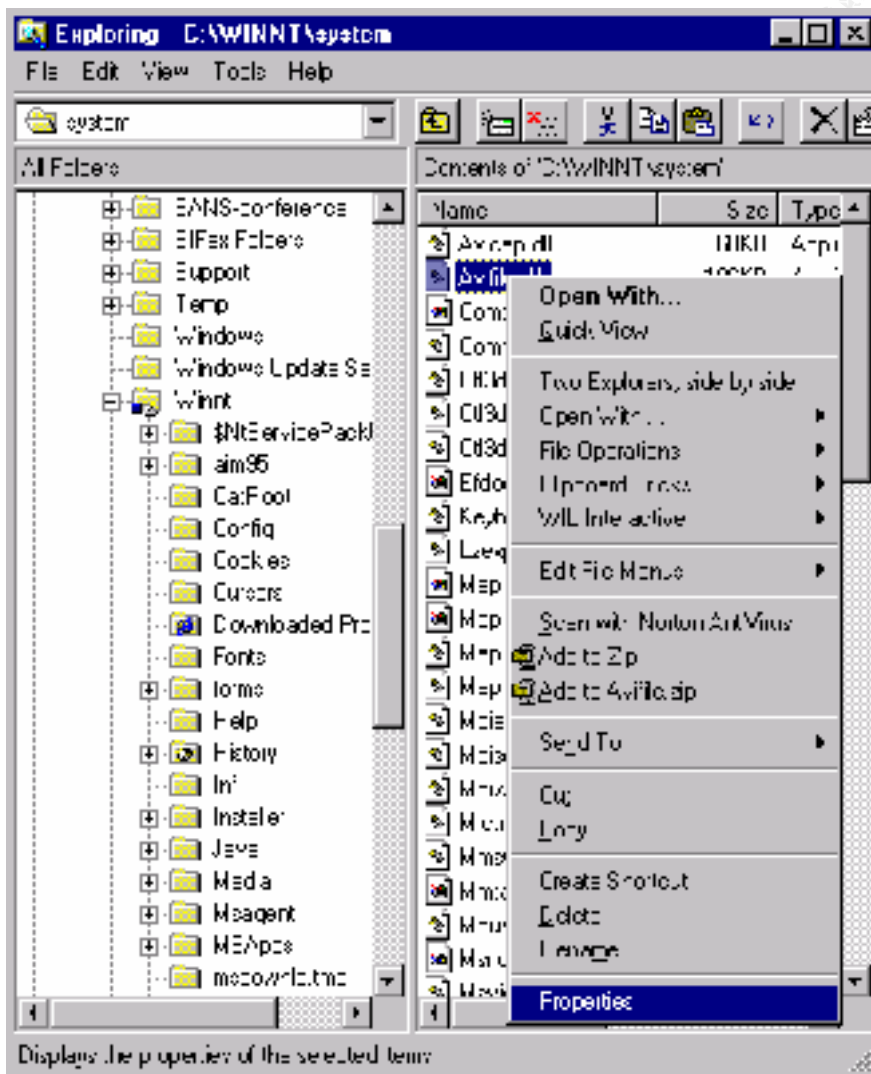
endwhile

sendkeysto("Event","!lx")
```

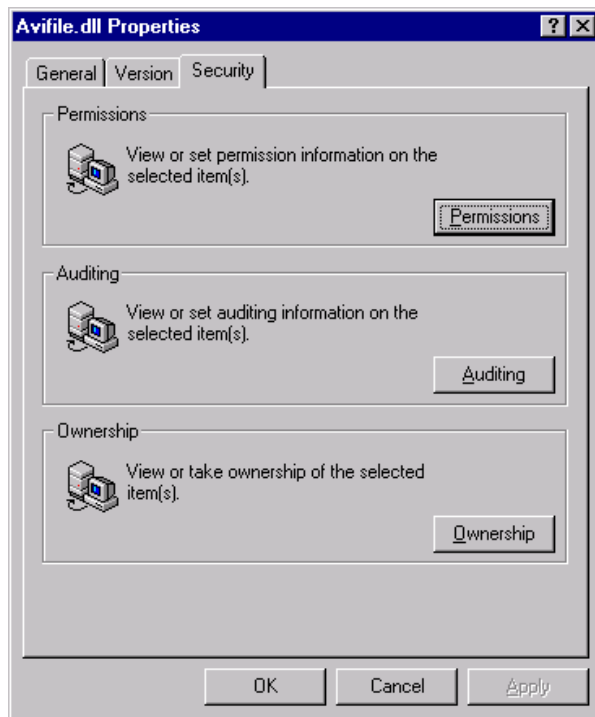
Fourth Step: Enable Auditing on any files, folders, printers and registry entries that you want Audited

If you enabled file and object auditing, you have to additionally set up auditing on the files, directories, printers and registry entries that you want audited. This can be done manually or with command line tools from the Windows Resource Kit or from third party software vendors.

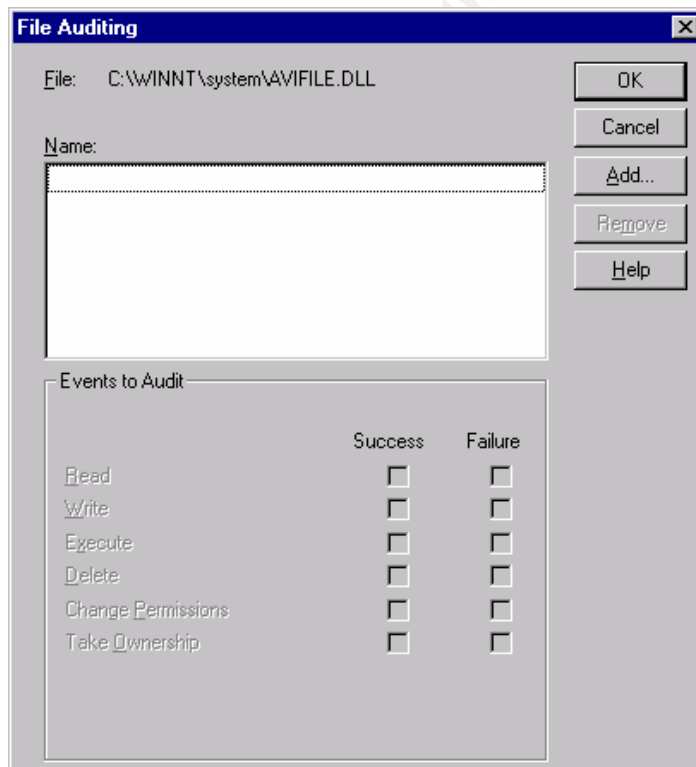
To enable or change auditing on a file, folder or printer, go to Windows NT Explorer and right click on the file, directory or printer you want to change and go to properties



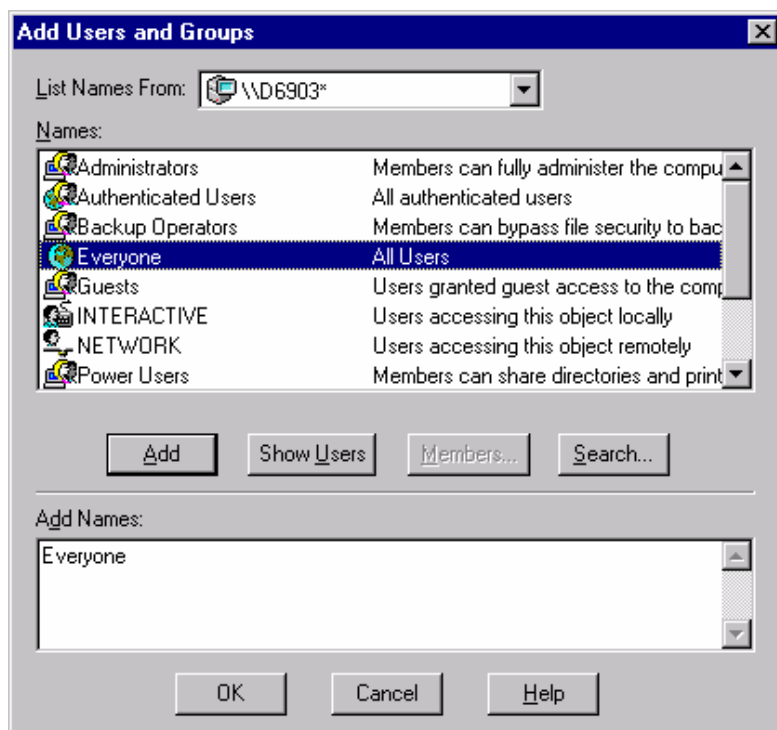
Go to the Security Tab:



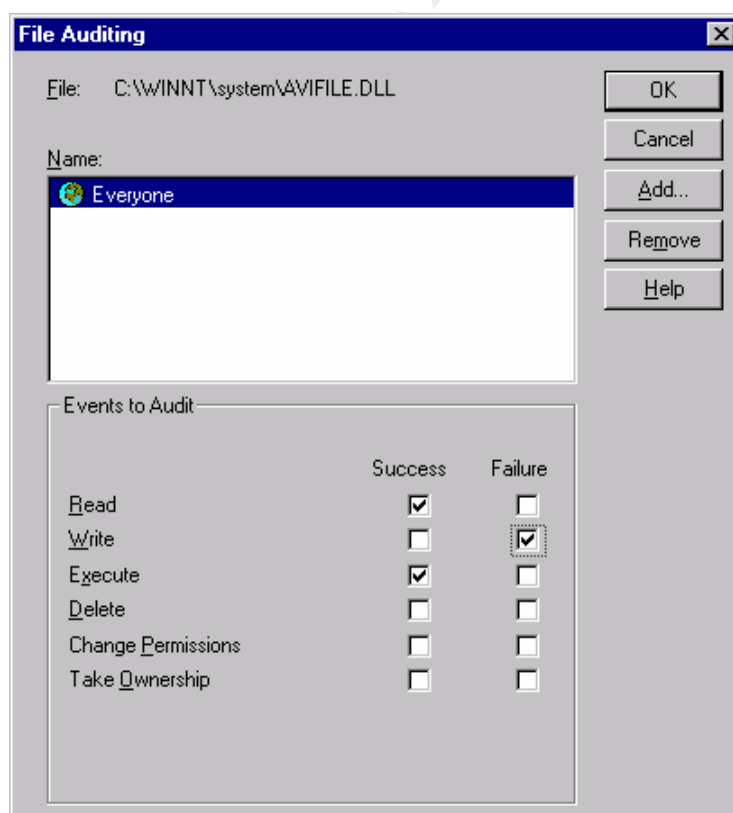
Click on the Auditing Button



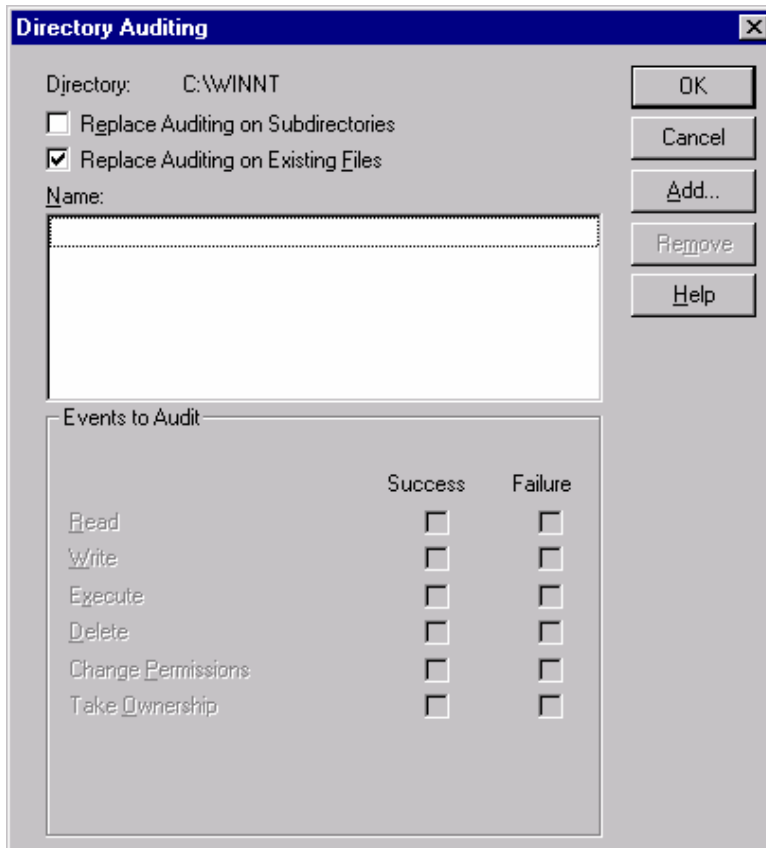
Click on the Add button and Select the Users you want to Audit and click the Add button



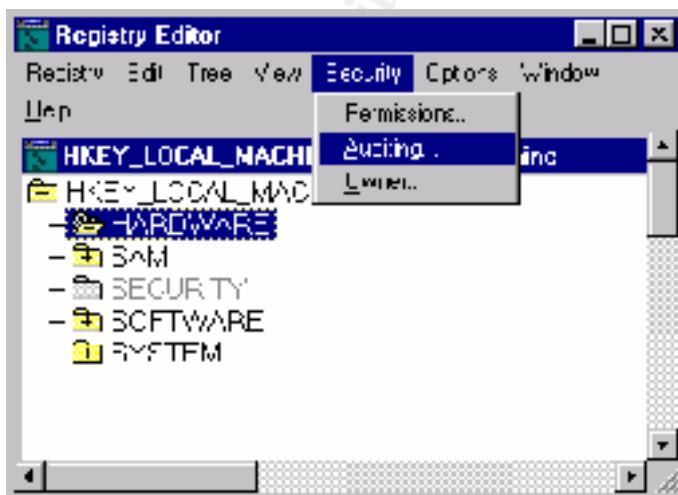
Then Select what actions you want to audit for that file



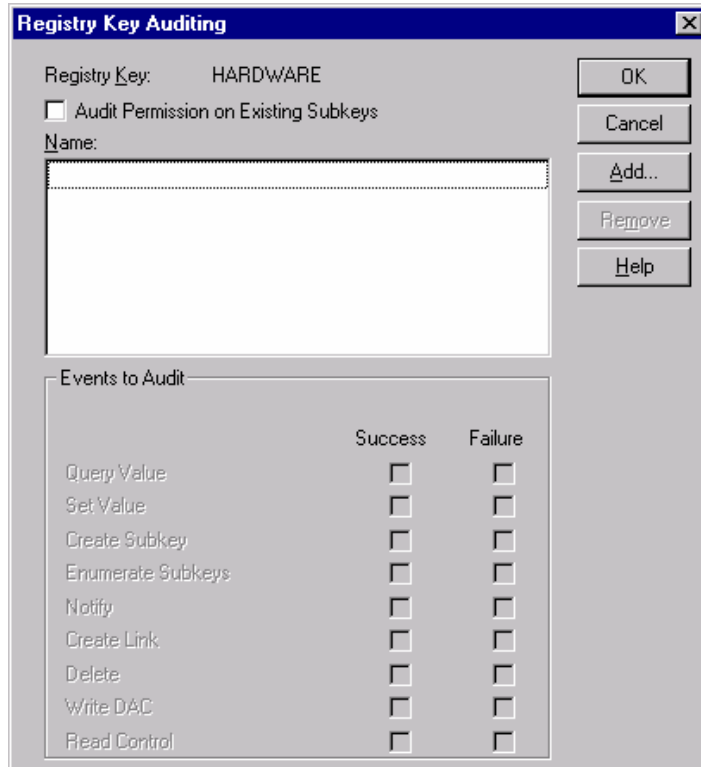
When enabling or changing auditing on a directory you have two extra options, replace auditing on subdirectories and replace auditing on existing files



To enable or change auditing on registry entries, go to regedt32 and click on Security -> Auditing and then follow the steps above



There is an additional setting for Registry key auditing also



Because file Auditing is independent for each file, it can be difficult to manage independent file audit settings. Both setting and listing file audit settings from the GUI tools is tedious in situations where one person is managing multiple machines. For this reason there are NT Resource Kit and third party software tools that can be used at the command line and therefore can be scripted to allow for effective management of file audit settings.

Two specific third part tools are Super ACLs by Trusted Systems (www.trustedsystems.com) and NT Command Line Security Tools by Pedestal Software (www.pedestalsoftware.com). The following are some examples of the Super ACLs tools from their website:

Assure that all *.EXE and *.DLL files (not directories) in a large directory tree (like WINNT) allow at most "Read" access to the built-in "Everyone" group (that is, set them to Read if they currently allow more):

```
modacl /s /f C:\WINNT /nm *.EXE *.DLL /am "everyone:>read" /ar everyone:read
```

Take ownership of all items owned by a user "BBye" who has left your organization (making sure that you have Full Access in the item's ACL's):

```
takeown /s C: D: E: /om bbye /add
```


Find all directories in which a user named JJones has at least read ("RX") access to the directory itself or files newly created in the directory (the file default permissions):

```
pracl /s /d ProjectDir /am "jjones:>rx>rx"
```

Fifth Step: Doing something with the Audit Logs

This step is the most important and the most difficult. It is important to examine the audit logs regularly to detect any misuse and it is also important to save the audit logs for security investigations. This paper does not contain an exhaustive list of the possible solutions for collecting and analyzing audit logs. In some cases audit log collection and analysis may be able to be incorporated into some existing software already in place at a company. In other cases a company may decide that they want such detailed analysis of the audit logs that they need to write a custom application to do the event analysis. To deploy a successful audit collection and analysis engine a company needs to consider all the options for audit collection and analysis.

To reduce the risk of tampering and to increase the usefulness of the data, it is essential to centralize the audit data and reporting. The following steps should be taken to make sure the audit logs are secure:

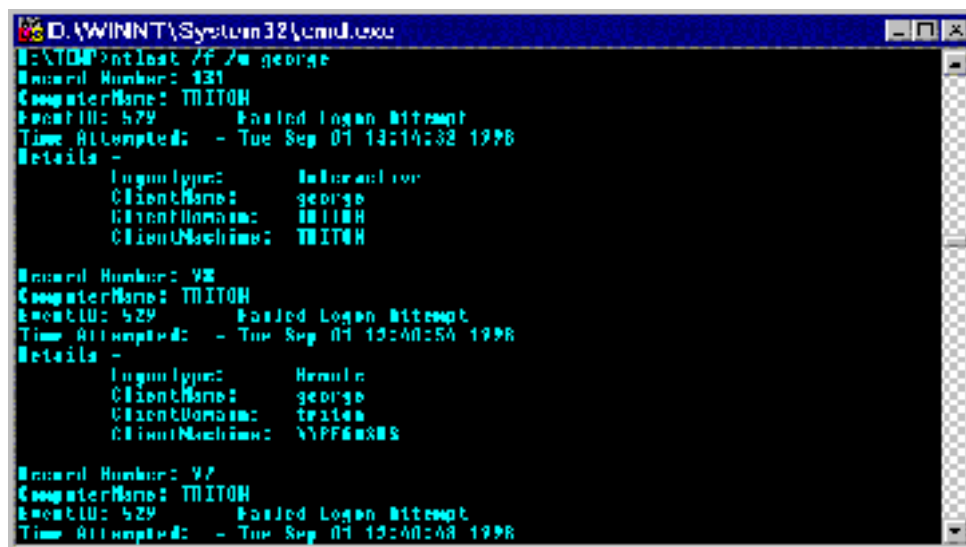
- Compile the audit logs on a secure machine
- Run the audit log analysis and collection on a dedicated machine that does nothing else
- If you print audit information make sure you are printing to a secure printer

Command Line Tools

There are some command line tools that help in analyzing audit logs. Command line tools are especially useful if you want to send simple customized email or other notifications to system administrators. Command line tools can be use in scripts to alert administrators when certain conditions occur.

One command line tool that is useful is NTLast by NTOBJECTives, Inc.. This tools searches Event Logs for logon and logoff audit entries. It can search .evt files that are output by the Event Logs, or it can search the Event Logs themselves on a computer over the network. This tool can report on successful and failed logons as well as tell the difference between interactive and remote logons. This tool can also list logons from a specific machine or user.

Here is a search for failed logons by a user george:

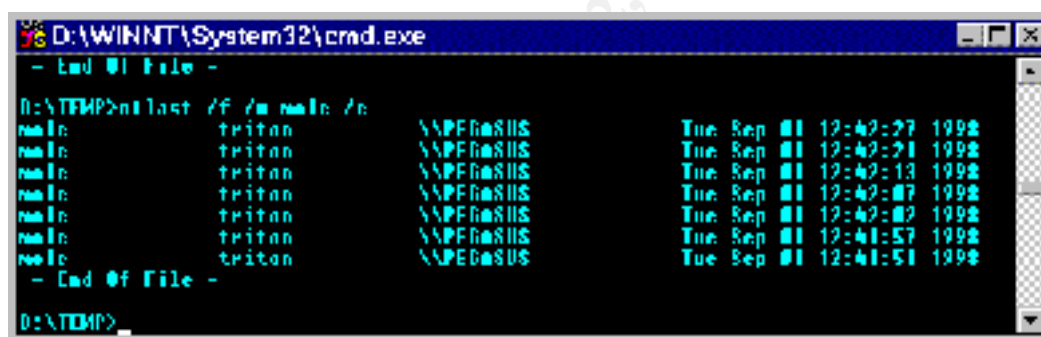


```
D:\WINNT\System32\cmd.exe
D:\TEMP>ntlast /f /u george
Record Number: 131
ComputerName: TRITON
EventID: 529      Failed Login Attempt
Time Attempted:  - Tue Sep 01 12:44:32 1998
Details -
  LogonType:      Network
  ClientName:     george
  ClientDomain:   TRITON
  ClientMachine:  TRITON

Record Number: 92
ComputerName: TRITON
EventID: 529      Failed Login Attempt
Time Attempted:  - Tue Sep 01 12:40:54 1998
Details -
  LogonType:      Network
  ClientName:     george
  ClientDomain:   triton
  ClientMachine:  \\\PELAGUS

Record Number: 92
ComputerName: TRITON
EventID: 529      Failed Login Attempt
Time Attempted:  - Tue Sep 01 12:40:48 1998
```

Here is a search for failed logons in condensed mode, this helps see that there are multiple logon attempts with the same username over a short period of time:



```
D:\WINNT\System32\cmd.exe
- End Of File -

D:\TEMP>ntlast /f /u main /n
Main In      triton      \\\PELAGUS      Tue Sep 01 12:42:27 1998
Main In      triton      \\\PELAGUS      Tue Sep 01 12:42:21 1998
Main In      triton      \\\PELAGUS      Tue Sep 01 12:42:13 1998
Main In      triton      \\\PELAGUS      Tue Sep 01 12:42:07 1998
Main In      triton      \\\PELAGUS      Tue Sep 01 12:42:02 1998
Main In      triton      \\\PELAGUS      Tue Sep 01 12:41:57 1998
Main In      triton      \\\PELAGUS
- End Of File -

D:\TEMP>
```

Using this command line tool it would be very easy to search audit logs of computers and report periodically. This tool is limited because it only searches the audit logs for logon and logoff information. This tool does not search for other audit information.

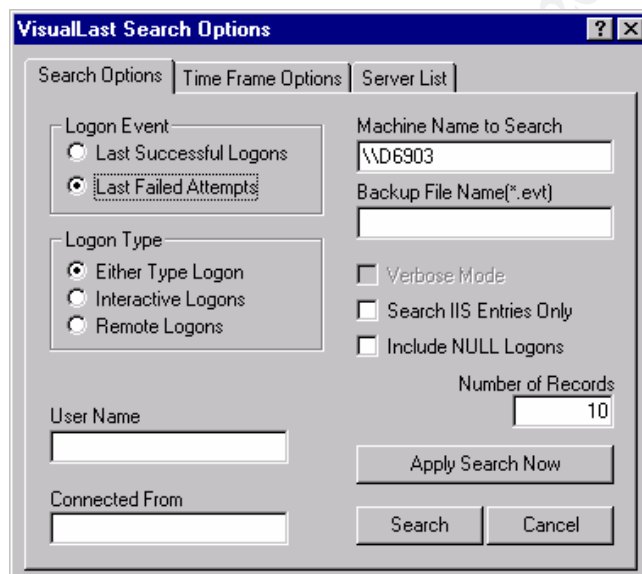
Visual Last by NTOBJECTives, Inc.

NTOBJECTives also makes a GUI version of NTLlast called Visual Last. This is a GUI tool that searches event logs for logon and logoff audit information. Again, this tool is limited to only logon and logoff information.

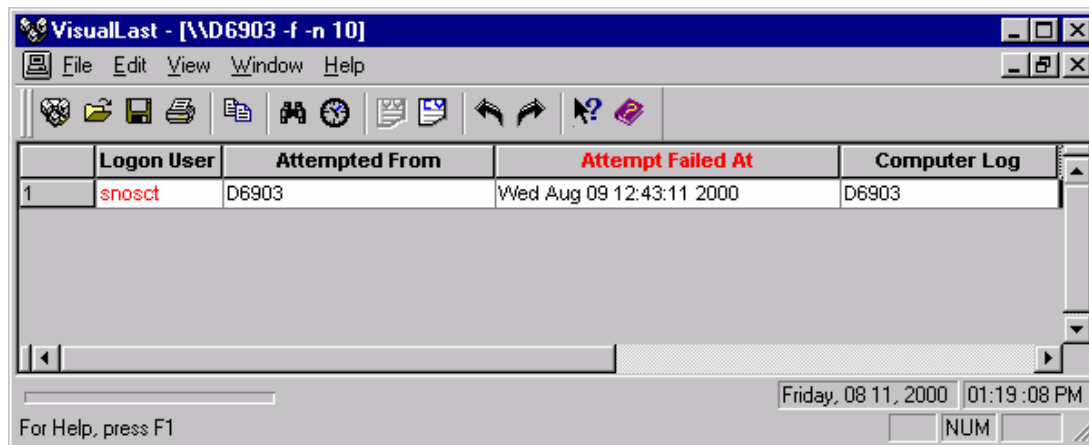
Here is the first screen, you have to point Visual Last to a domain controller to get the names of the machines on your network:



Once you select a machine you want to search, it gives you the options for the search:



Here is what the report looks like:



The screenshot shows the VisualLast application window. The title bar reads "VisualLast - [\\D6903 -f -n 10]". The menu bar includes File, Edit, View, Window, and Help. The toolbar contains various icons for file operations and navigation. The main display area shows a table with the following data:

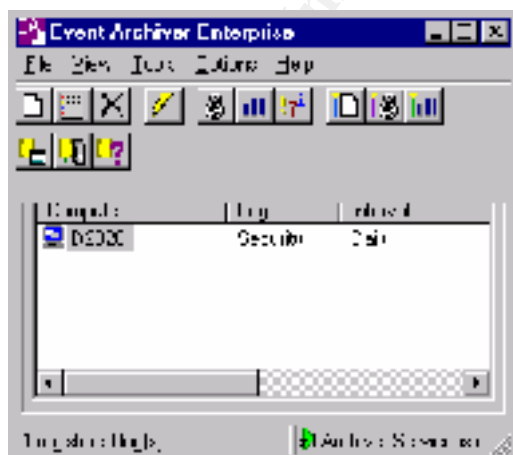
	Logon User	Attempted From	Attempt Failed At	Computer Log
1	sno3ct	D6903	Wed Aug 09 12:43:11 2000	D6903

At the bottom of the window, there is a status bar with the text "For Help, press F1" on the left and a date/time display "Friday, 08 11, 2000 01:19:08 PM" on the right, along with a "NUM" button.

The NTLlast command line tool may be more useful than Visual last for some applications. Remember that both of these tools only report on logon and logoff information. If you want a simple visual tool to report on logon and logoff information, Visual Last may be just the ticket. If you want to do more with the information that just view it one computer at a time, like script it and email reports, the command line tool is probably better suited.

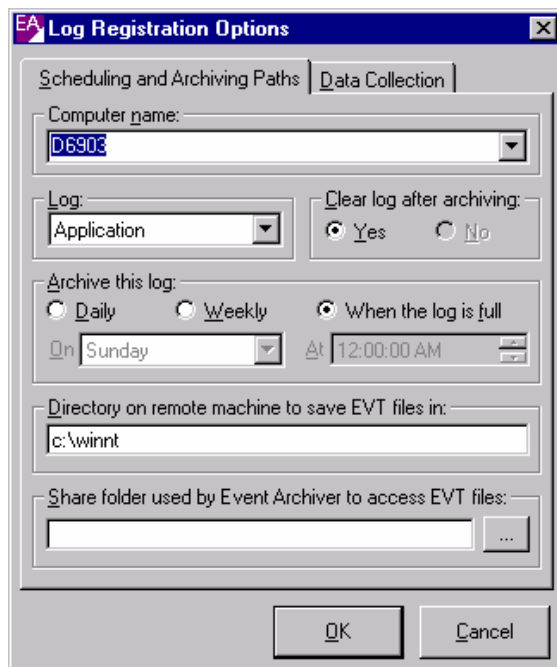
Event Archiver by Dorian Software Creations, Inc.

A third party software product called Event Archiver allows for the collection of Event Logs and also allows for global administration of Event log settings and Audit settings. This product does not analyze the data, but it does archive the data and allow exporting of the data to Access or SQL databases.

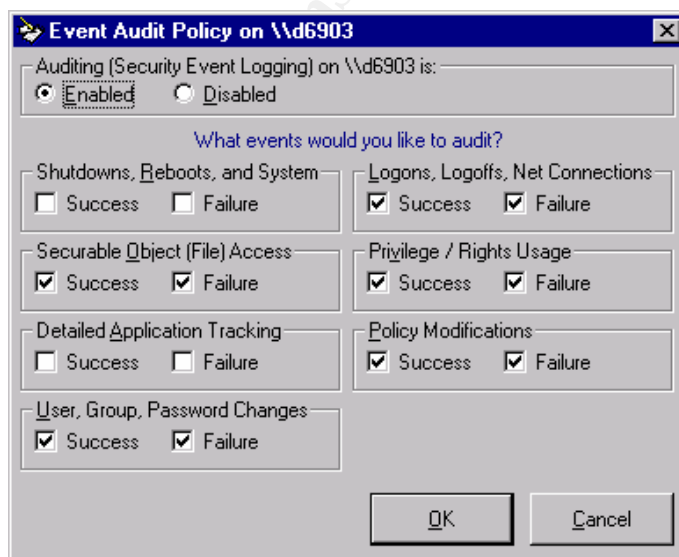


This product does have limitations. It only allows you to select one computer at a time to be audited. If you wanted to audit on hundreds of computers it would take substantial time to set up. Also, it only allows for event logs to be collected daily at most. If you want to collect the event logs more often than daily, you will not be able to use this software. One reason you would want to collect event logs more often is if you want an indication that someone is deleting part of your audit log. If you collect audit logs in real time, then if someone tries to go back and delete the audit log, it will already be archived to another machine.

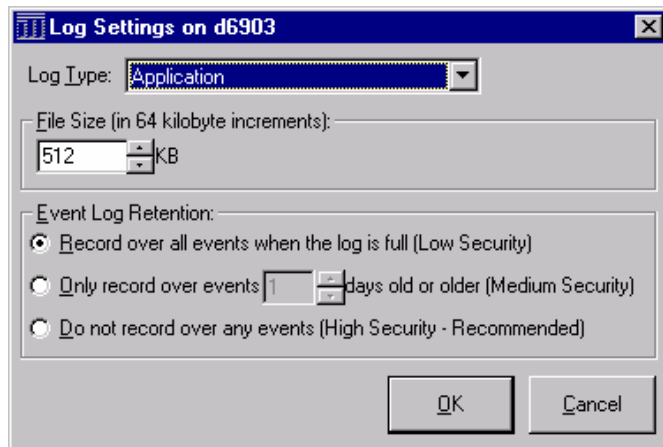
Here is the screen that allows you to add a computer to be logged:



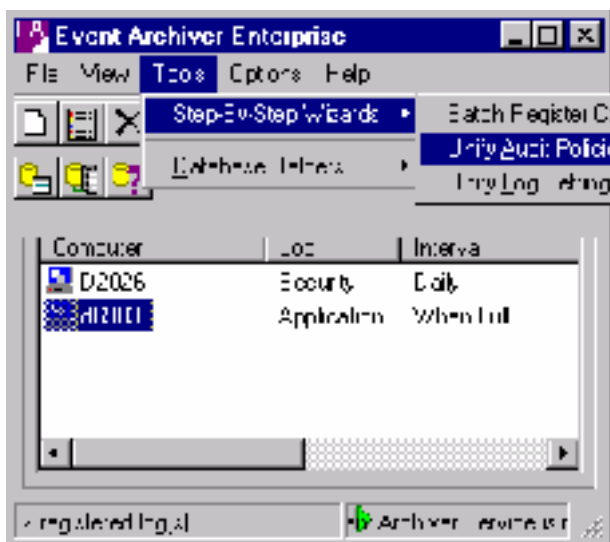
Here is the screen that allows you to manage the audit policy on a machine:



Here is the screen that allows you to manage the log settings on a machine:



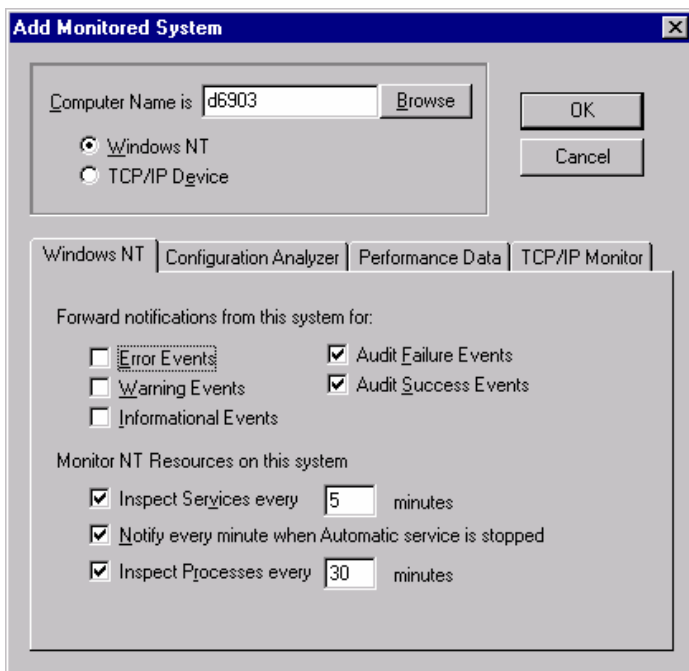
As I mentioned, there are options to unify all the audit and log settings between computers:



Dorian Software is releasing an Event Log analyzer later this summer.

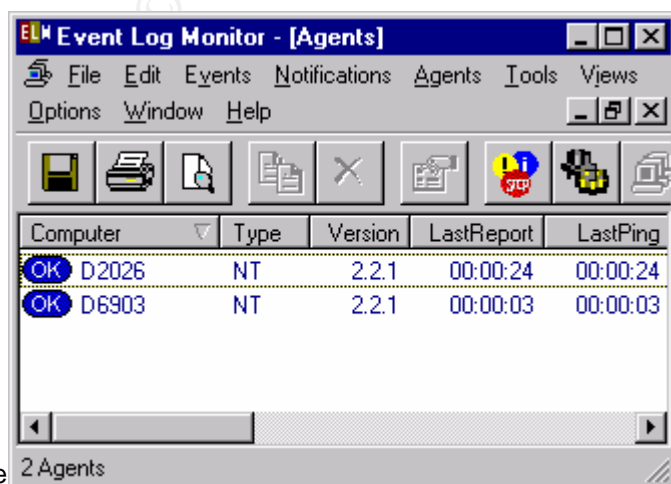
Event Log Monitor by TNT Software

Another third party software package that does Event Log Management is Event Log Monitor by TNT Software. This software package does event log collection and analysis in real time. It can be configured to alert on certain events and it can create detailed reports of specific types of events. It is easy to configure. When you first launch the Event Log Monitor it will ask for a machine to be monitored.

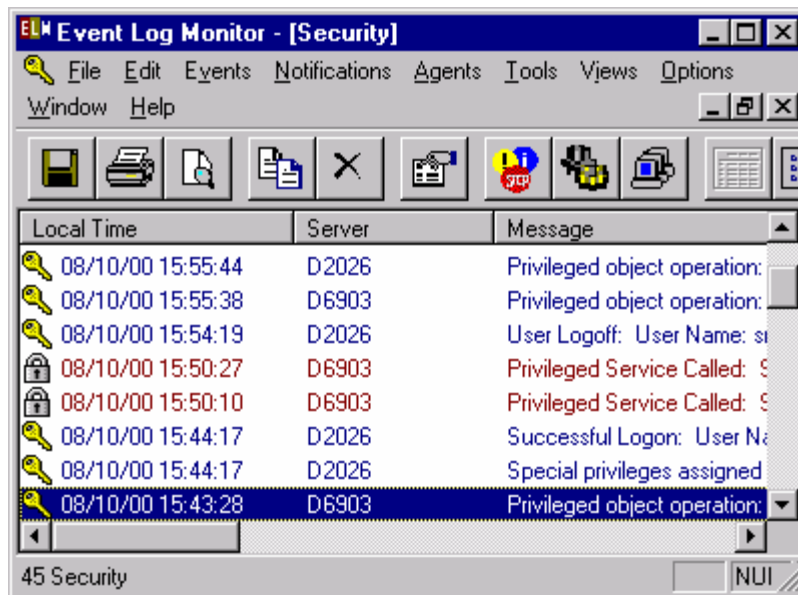


It is somewhat limited in the setup because it does not allow adding multiple machines at once. You can select that kind of notifications you want on this screen and then click on the OK button and it will set up the machine to be monitored. Be careful about the amount of disk space that you have available on the machine that you are using. Just the configuration file takes 35KB per machine for this software. Also, you may run into bandwidth problems if you try to monitor too many machines with one server.

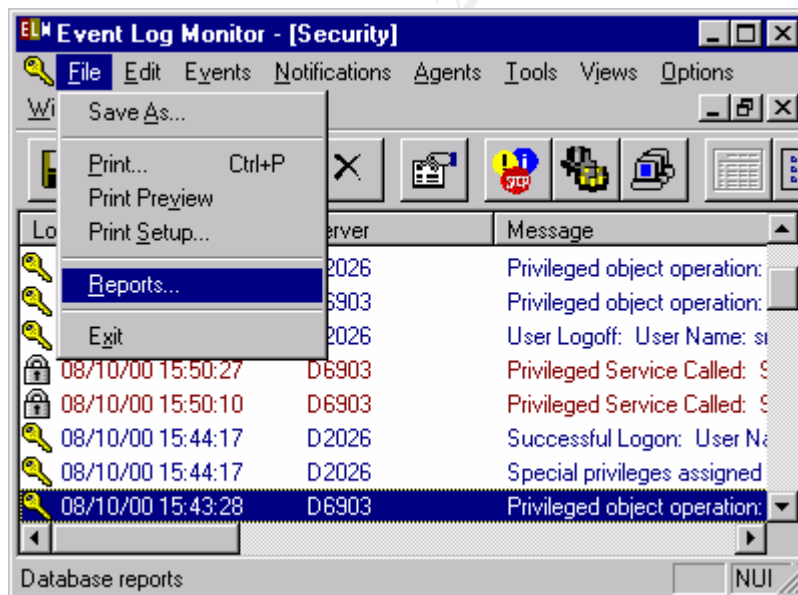
The user interface for this software is fairly straight forward. There is a window where you can see all the Agents you have installed. It looks like this:



There is also a window where you can see all the security events that occur:



One thing this software doesn't have that would be helpful is some post-event log processing of the messages. This specific software package just compiles the event log entries into effectively one large event log. You can do some filtering, but it would be helpful for it to break out the specific event log entries and give more explanation of what they are. The reporting does help with this a little bit. There are reports available from the file menu:



They include some general use reports, like a daily logon report and a daily logon failure report, but most people will need to customize them to make them more useful.

The Event Log Monitor is a very powerful tool but does require some customization to make it useful.

Conclusion

Auditing on NT can give system administrators invaluable information about use of system resources. Enabling auditing is simple and can easily be done company wide. Managing file level auditing and analyzing audit logs is more difficult, and often requires third party software to simplify the process. With the correct third party software and possibly some customized scripts and reports, system administrators should be able to effectively manage system level audit information.

© SANS Institute 2000 - 2002, Author retains full rights.

References:

Fossen, Jason. (2000) *Windows NT & Windows 2000 Security: Step by Step*. Washington DC: The SANS Institute.

Gollmann, Dieter. (1999) *Computer Security*. New York: John Wiley & Sons.

Microsoft Corporation. *MSDN Library*. Retrieved August 7, 2000 from the World Wide Web: <http://msdn.microsoft.com/library/default.asp>

NT OBJECTives, Inc. *NT Last version 2.85*. Retrieved August 11, 2000 from the World Wide Web: <http://www.ntobjectives.com/ntlastv2.htm>

Schultz, Eugene. (1999) *Windows NT Security: Basic/Intermediate*. New Orleans: The SANS Institute.

Trusted Systems. *Super CACLs Overview*. Retrieved August 9, 2000 from the World Wide Web: <http://www.trustedsystems.com/scaclsintro.htm>

Vacca, John. (1997) *Intranet Security*. Rockland: Charles River Media, Inc.

© SANS Institute 2000 - 2002. All rights reserved. Author retains full rights.