Global Information Assurance Certification Paper

Centralized Windows 2000 Event Logging

A Step – By – Step Guide

Written by: Scott Richardson
Version Number: 1.4b
Course: Security Essentials (GSEC)

Abstract

So much takes place on corporate networks these days that Administrators and IT staff are often completely un-aware of. I know that on my network there was a lot taking place that was contrary to company policy, and that opened up security vulnerabilities. There were even problems that I was un-aware of because the end users did not want to file Help Desk tickets. I think the key to having a smooth running, secure network, is awareness. If the IT team is aware of the goings on of the network they can deal with small issues when they arise instead of waiting for the problem is grow out of control and cause system failures. One way to raise the awareness is with the built-in Event Logging in Windows 2000. While the Event Viewer has it's benefits it also has one major downfall: Each computer holds it's own logs and there is no built-in way to centralize the logging. The purpose of this paper is to show you how to setup a centralized logging system for your Windows 2000 Corporate Network. The tools I have chosen to do this task are as follows:

- Dumpel.exe (http://microsoft.com/windows2000/techinfo/reskit/tools/existing/dumpel-0.asp)
- Ntotools2.zip (http://packetstorm.widexs.nl/NT)
- Microsoft SQL Server 2000
- IIS 5.0 or higher
- A Windows 2000 Network

At the end of this paper you should be able to successfully setup a centralized logging station on your Windows 2000 network. You will also have an understanding of what events you need to audit to ensure that you are getting the right information, the information that will help you make your network more secure and run better. Knowledge is power and the more you know what is happening on your network, the better prepared you are to protect it.

The Need

Like I just mentioned, the need for centralized logging is huge, especially for companies where there is a large number of computers and a limited number of IT Staff. Because Windows 2000 does not offer a centralized logging solution, if an Administrator wanted to view the logs from multiple workstations he/she would need to log on to each computer. This can be very time consuming and frustrating for the Administrator. A centralized console where the Administrator can view the logs from all the workstations would be a very large benefit to the IT Department of any sized organization. Take this incident for example: One morning I get a call from my manager asking if I could look into a possible attempted un-authorized network use over the weekend. I guess he had noticed something a miss in his office. At the time I did not have a logging solution in place so my only method of investigation was to visit the workstation and try to see what I could get from what logs there were. After some time I had to tell my

manager that I could not find any information to either support or discredit his assumptions. That is when I got to thinking that it would be very valuable to have that sort of information right at my fingertips. So, to solve that problem, I implemented an Event Logging Policy on my network and now… if someone asks, I am able to provide them with the information they desire quickly and with great accuracy.

In the world of Event Logging there are nine main event categories that can be logged. They are:

- Account Logon Events
- Account Management
- Directory Service Access
- Logon Events
- Object Access
- Policy Change
- Privilege Use
- Process Tracking
- System Events.

These 9 categories can be set to log ether successes, failures or both. When designing your companies logging policy you want to make sure you select only the events that are of relevance and importance. Later on in this paper I will give you my recommendations for your Logging Policy. But for now, lets get into the nitty gritty of how to make it all happen.

The Process

The first thing you need to do is designate a computer to be the central logging database server, and install Windows 2000 Server, IIS and SQL on that machine. You will also need to make the following directory structure and then copy the dumpel.exe and ntolog.exe files into the "C:\Central Logging\Tools" directory.

- C:\Central Logging\Logs
- C:\Central Logging\Logs\Application
- C:\Central Logging\Logs\Security
- C:\Central Logging\Logs\System
- C:\Central Logging\Tools
- C:\Central Logging\Scripts

Now that we have created our directory structure and put the tools in the correct place it is time to start making the batch files that will pull the information from the computers. What you will need to do is create a list of all the computers you want to obtain logging information from. The first file we will create is a batch file called getlog.bat that has the following commands in it.

getlog.bat

```
@echo off
c:\central logging\tools\dumpel.exe –f c:\central logging\logs\system\%1.txt –s \\%1 -l system –format sTCISctd
c:\central logging\tools\ntolog.exe \\%1 /c /sys
c:\central logging\tools\dumpel.exe –f c:\central logging\logs\application\%1.txt –s \\%1 -l application –format sTCISctd
c:\central logging\tools\ntolog.exe \\%1 /c /app
c:\central logging\tools\dumpel.exe –f c:\central logging\logs\security\%1.txt –s \\%1 -l security –format sTCISctd
c:\central logging\tools\ntolog.exe \\%1 /c /sec
```

The syntax for using this batch file is \>getlog *workstationname* where the workstation name is the name of the computer to pull the logs from. The batch file then gets the logs from the computer specified and puts the appropriate logs in the correct directories. The name of the file being saved is the name of the computer the log was taken from. After dumping the log file, the remote logs are removed using the ntolog.exe tool.

Now that we can get the logs from one computer we need to create another batch file that will get the logs from all of the computers that we specify. This file will be called collectlogs.bat

collectlogs.bat

```
@echo off
echo.
echo.
echo This batch file will gather the Event Logs for the computers specified in this file.
pause
echo.
echo.
Call getlog.bat first-computer
Call getlog.bat second-computer
.
.
Call getlog.bat last-computer
```

When the collectlogs.bat file is run, all of the computers specified will have their logs collected, deposted into a directory on the logging computer and then the remote logs will be cleared. Now that we have all the logs in one place the next thing we need to do create the SQL database that will hold the information.

To create the Database and the Tables execute the following queries in the SQL Query Viewer.

1st: Create Database

```
CREATE DATABASE EventLogs
```

2nd: CreateTables

```
CREATE TABLE Eventlogs.dbo.application
(
	LogStrings	nvarchar(1000),
	Type		nvarchar(255),
	Category	nvarchar(255),
	EventID		nvarchar(255),
	EventSource	nvarchar(255),
	Computer	nvarchar(255),
	LogTime		nvarchar(255),
	LogDate		nvarchar(255)

)

CREATE TABLE Eventlogs.dbo.[system]
(
	LogStrings	nvarchar(1000),
	Type		nvarchar(255),
	Category	nvarchar(255),
	EventID		nvarchar(255),
	EventSource	nvarchar(255),
	Computer	nvarchar(255),
	LogTime		nvarchar(255),
	LogDate		nvarchar(255)

)

CREATE TABLE Eventlogs.dbo.security
(
	LogStrings	nvarchar(1000),
	Type		nvarchar(255),
	Category	nvarchar(255),
	EventID		nvarchar(255),
	EventSource	nvarchar(255),
	Computer	nvarchar(255),
	LogTime		nvarchar(255),
	LogDate		nvarchar(255)

)
```

3rd: Add User to Database (requires knowledge of SQL 2000 Administration)

Using the SQL Enterprise Manger create any username / password combination you like, and give that user db_datareader and db_datawriter permissions on the EventLogs database. For simplicity I created a user named EventLogger with the password *logger*.

4th: Create Stored Procedure for importing text file into database. (**NOTE: you must repeat each BULK INSERT Section for every computer on your list.)

```
CREATE PROCEDURE [dbo].[LoadData] AS

BULK INSERT [EventLogs].[dbo].application from 'c:\Central Logging\logs\application\firstcomputer.txt'
WITH (FIELDTERMINATOR = '\t', ROWTERMINATOR = '\n')
BULK INSERT [EventLogs].[dbo].security from 'c:\ Central Logging\logs\security\firstcomputer.txt'
WITH (FIELDTERMINATOR = '\t', ROWTERMINATOR = '\n'  )
BULK INSERT [EventLogs].[dbo].[system] from 'c:\ Central Logging\logs\system\firstcomputer.txt'
WITH (FIELDTERMINATOR = '\t', ROWTERMINATOR = '\n'  )
.
.
.

BULK INSERT [EventLogs].[dbo].application from 'c:\ Central Logging\logs\application\lastcomputer.txt'
WITH (FIELDTERMINATOR = '\t', ROWTERMINATOR = '\n')
BULK INSERT [EventLogs].[dbo].security from 'c:\ Central Logging\logs\security\lastcomputer.txt'
WITH (FIELDTERMINATOR = '\t', ROWTERMINATOR = '\n'  )
BULK INSERT [EventLogs].[dbo].[system] from 'c:\ Central Logging\logs\system\lastcomputer.txt'
WITH (FIELDTERMINATOR = '\t', ROWTERMINATOR = '\n'  )


GO
```

What this stored procedure will do is take the text files that were created by getlogs.bat and import them into the Database. Each complete import takes three commands. The first command:

```
BULK INSERT [EventLogs].[dbo].application from ‘c:\ Central Logging\logs\application\firstcomputer.txt'
WITH (FIELDTERMINATOR = '\t', ROWTERMINATOR = '\n')
```

Imports the application logs, the second the security logs and the third, the system logs.

So we now have all the basics that we need to get the logs from the client computers to a central location, and then import them into an SQL Database. The only thing left to do is set it up so that it does it all by itself. We obviously don’t want to have to manually run these scripts every time we want to look at the logs, so in this next section we are going to talk about automation. Automation means: *“Automatic; as opposed to human; operation or control of a process”*[*] What we want to do is make it so that the computer does all the work for us.

For the first step, we are going to use Windows Scheduler to schedule the collection of the logs. Windows Scheduler can be found under Start -> Programs -> Accessories -> System Tools -> Scheduled Tasks. Click on “Add Scheduled Task” to start the task scheduler wizard. Follow the wizard through selecting the

[*] www.hyperdictionary.com/dictionary/automation

frequency you would like to collect the logs. For this example I chose to collect the logs every day at 11pm. The next step is to import the text files we collected into a database. For this we are going to use the SQL Server Agent to schedule the task. First, open the SQL Enterprise Manager and create a new job under the SQL Server Agent. (You will need to make sure that the SQL Server Agent service is set to run automatically) You should set the new job to run the stored procedure LoadData once a day at 12 Midnight (or whatever time you choose, just make sure you leave enough time after the start of the 1st scheduled task).

So now the hard work is over. We've taken what used to be a mess of rogue logs, an un-manageable sea of information, and consolidated it into one simple location. Now we need to decide what to do with the information. If we wanted, we could open our SQL database and look through all our logs manually and try to isolate logs of importance….. or, we could create a simple ASP website that displayed the information for us in a nice way. This web site would allow us to view the logs in an assortment of ways, and to get information out of them that we would not have been able to before. I am not going to post all the code for the website, but simple ASP web pages and SQL Queries can easily be written to pull all the necessary information from the database.

Event Logging Policies

Once we have the system in place to hold the logs, we need to decide what information we want to keep track of. As mentioned earlier, there are many different events you can log. But keep in mind that even though there are a lot of things you can log you don't want to go crazy and log EVERYTHING, yet at the same time you don't want to log too little so that the information is too scarce to get anything good out of it. A good rule of thumb is *to "Audit only the events that really matter"*[*] Based on my experience my recommendation I would recommend that you audit the following events for the following reasons.

Account Management: Logs all changes to accounts, additions, deletions, group membership, etc.
Take the following example: A hacker exploits a currently unknown vulnerability on your IIS Server giving him full access to your server. He then proceeds to create a user for himself and put that user into the administrators group. That hacker now has administrative privileges on your network. With the ability to monitor account creation and group membership, a quick look through your logs would reveal the hacker and allow you to put a stop the intrusion.

Logon Evens: Logs all failed and successfully logon events.

[*] Posey, Brien M. "Creating an Audit Policy" November 2, 2000
URL: http://networking.earthweb.com/netos/print.php/10951_624801_1 (Aug 28, 2003)

This is an especially important event to log because it will show you if someone is trying to brute force hack you network. It is especially important to log both success and failures. Some people only think the only need to log failed attempts, but what happens if your company does not have an account lockout policy, and the intruder is able to guess the correct password after 20 or 30 tries? You wouldn't know because you are not logging successful logons.

Policy Change: Logs when users rights have been changed.
My recommendation to log Policy Changes only applies to organizations where user access is restricted and it is necessary to know if users are trying to usurp policy and change their rights. Sometimes it is possible for a user to change their local privileges to circumvent domain policy, while this change is not permanent and is set back to the default domain policy at logon, it is good to know if your users are taking matters into their own hands.

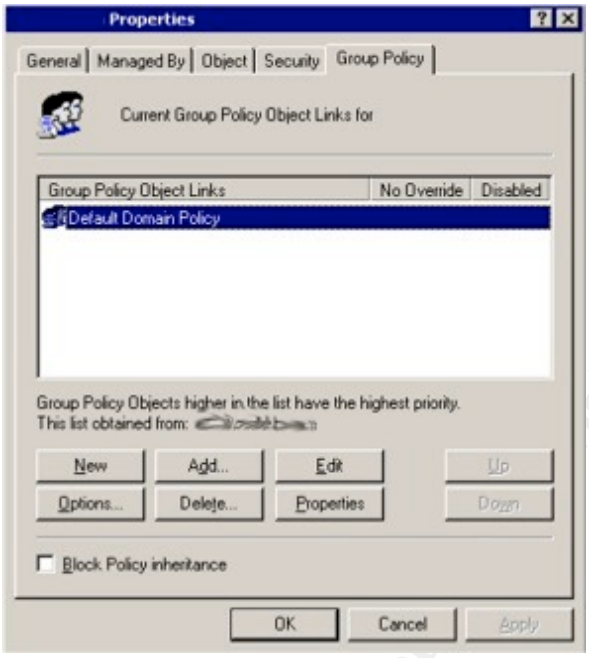
Privilege Use: Are your users successfully gaining privileged access. Are they trying? Logging privilege use is similar to logging policy change, but it just lets you know if your users are finding ways to perform actions that require security permissions above what has been assigned to their account

System Events: System Startup and Shutdown. Was software installed, any other system related events.
Logging system events will help you keep tabs on things such as application hangs, are illegal programs being installed and any other important information relating to the system such as depleted Hard Drive space, etc.

I think it is important to make special note that it is imperative to log all your servers, especially your domain controllers, as that is where a lot of the security related transactions take place. Failure to log your Domain Controller can result in valuable information slipping away and could make the difference between catching an intruder and letting an intruder get away.

By logging the suggested events you will be logging valuable information that will assist you in verifying that your network security policies are being followed. To enable logging of these events you need to modify the group policy for your Windows 2000 Domain. This is done through the Active Directory Users and Computer plug-in.

Open the Active Directory Users and Computer plug-in and open the Default Domain Policy by selecting the Domain Name, right clicking and choosing properties. On the dialogue box that opens, select the group policy tab. It should look like this:



Highlight the Default Domain Policy and select the Edit button. When the Group Policy Editor opens, you want to drill down to the "Computer Configuration + Windows Settings + Security Settings + Local Policies" and then finally select Audit Policy. On the right hand preview pane, you will see the nine categories that you can audit. Make the appropriate changes based on the suggestions provided in this documents, or whatever your companies Audit Policy calls for. Once you have made the change to the Default Domain Policy, you need to make sure that your group policies are setup to update on a regular basis, so that all of your machines get the new policy.

Other Logging

The auditing we have discussed so far, are the basic windows functions. We can however audit a much larger number of things. For example we can audit files and folders, printers and even the registry.

The auditing of files and folders can be very useful if you want to keep track of access to confidential information, or if you to monitor who is trying to access information that has been restricted. In order to audit file and directory access, you must be using the NTFS file system. To activate file level auditing right click on the file or folder and select Properties -> Security -> Advanced -> Auditing. When you get to that screen you can select which events you want to audit.

Auditing printers is equally as easy. The auditing of printers can be especially handy, particularly if there are users who want to know if people are using their

printer when they are not around. To enable Auditing on a printer attached to a Windows 2000 computer, open the Printer properties of the printer and then select Security -> Advanced -> Auditing. Make sure you Audit the “Everyone” group, that way you can tell if people who do not have Domain Accounts are trying to print to your printers.

Auditing of the registry is a bit more tricky and is only recommended for computers with which Security is of the utmost importance, namely production servers, etc. To enable registry auditing on a computer, you need to open the registry using “regedt32” not “regedit” and select key by key what you want to audit, see what I mean by a bit more tricky? Once you have selected a key you want to audit, select “Security -> Permissions -> Advanced -> Auditing.” You can then make you selection for what you want to Audit.

All of these additional Auditing items will generate Security Logs in the Event Viewer, which you can then view on your centralized logging server.

Summary

Auditing and Event Logging can be a very powerful tool. Centralize those logs, and the tool becomes even more powerful. This tool gives the Administrator the ability to quickly look at his/her logs and see exactly what is happening on each workstation. They can see the attempted security breaches, the violation of policy, etc. The opposite is also true, in that it is possible to confirm that the security measures in place on the network are working. If an IT Team were to spend a lot of time and money developing and implementing a defense in depth plan, but they had no way of monitoring weather or not it was working, what was the point? You need to know what is going on with your network at all times and centralized logging is an easy, efficient way of doing that.

References

Posey, Brien M. “Creating an Auditing Policy” November 2, 2000 URL: http://networking.earthweb.com/netos/print.php/10951_624801_1 (Aug 29, 2003)

Minasi, Mark “Mastering Windows 2000 Server, Second Edition”, Alameda, CA. © 2000 Sybex Inc, Pages 1291-1300

Burke, Paul J, et al. “Professional SQL Server 2000 XML” Birmingham, UK. © 2001 Wrox Press Ltd.®

Hyperdictionary © 2000-2003 WEBNOX CORP. URL: Http://www.hyperdictionary.com

The following URL’s are Copyright © 2003 Microsoft Corporation, One Microsoft Way, Redmond, Washington

Event Viewer Overview - Microsoft http://www.microsoft.com/windows2000/en/server/help/default.asp?url=/windows2000/en/server/help/event_overview_01.htm

Windows 2000 Security Event Descriptions http://support.microsoft.com/?kbid=299475

Account Logon Events: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/datacenter/515.asp

Audit Account Management: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/datacenter/516.asp

Audit directory service access http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/datacenter/517.asp

Audit Logon Events: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/datacenter/518.asp

Audit Object Access:

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/datacenter/519.asp

Audit Policy Change:
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/datacenter/520.asp

Audit Privilege Use
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/datacenter/521.asp