Global Information Assurance Certification Paper

Wednesday, April 13, 2005

GIAC Certified Windows Security Administrator (GCWN) Practical Assignment
Version 5.0 (July 16, 2004) Option 1

Securing Your Environment Through Log Analysis

By: David Cox

Table Of Contents

Executive Summary

Almost all operating systems, appliances, and devices create some form of log. These logs typically contain historical information about the health of the device. As a security professional, one of your duties is to find some way to pull all of these logs together for both archiving and analyzing. Many times nefarious activity can be detected in log files, but they must be looked at to be of any use. Useful information such as date and time, or success and failure of events can be analyzed to reveal an attack. My paper focus's on the Windows event log and how it pertains to security. Windows operating systems come equipped with the ability to log security events. Events such as failed logon attempts or successful changes to the auditing policies need to be scrutinized carefully. My paper details what an event is, which ones are important and why, and some tools we can use to collect and generate alerts and reports with.

Section 1
Monitoring The Windows Event Log Is A Serious Security Issue

The Windows event log is an invaluable tool for securing any network. As a Windows Server System Administrator, I learned early on how valuable reading the event logs was. By taking event ID's and event text, I have been able to fix the majority of my problems with Google and Microsoft Knowledgebase searches. From a security standpoint, event logs can be used to detect intrusions and serve as evidence in a court of law. The problem is that Microsoft's event logging is very cryptic and will not alert you to a possible intrusion. In this paper I will outline the basics of the event log, some applications that can be used to monitor it, and how we can use it to alert us of potential security breaches.

It has been my experience that many engineers do not give enough if any attention to event logs. I can't blame them. It is not flashy or glamorous. Event logs can also be tedious and time consuming to read through. Most of us want instant gratification. A reboot might conveniently fix your problem right now, but a thorough inspection of the log may reveal why you had to reboot in the first place. That revelation may be more serious than a hardware failure or blue screen. It may be a hacker finalizing the installation of his root kit. If that is not enough, consider new legislation which is holding institutions accountable of computer neglect legally responsible.

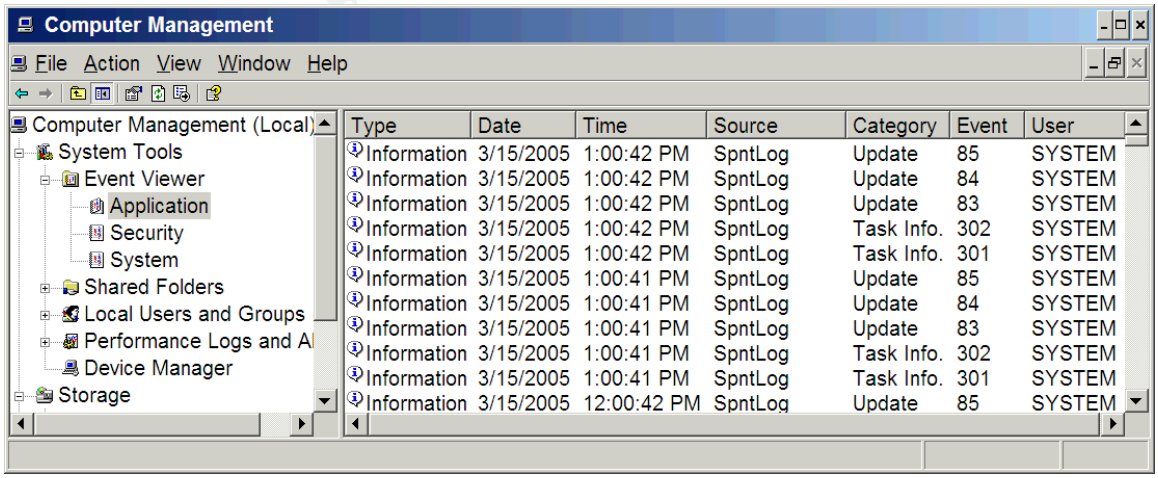
Let's say you have a web server running Windows 2000 Server and IIS which you use for selling diabetic medical equipment. It holds consumer information such as credit card numbers and health records. The administrator for the web server is lucky if he gets time to put a service pack on much less read through log files. Now let's say that his web server gets hacked. All the user data including credit card numbers are stolen. The hacker puts his back doors on the server and uses it to share illegal copies of copyrighted music, software, and movies. The hacker also uses the server as a hopping point to hack other servers. The unknowing administrator simply doesn't have time to investigate why the server is slowing down and rebooting more often than not. This scenario goes on for some time until the men in dark suits show up. They confiscate his server and ask about his security policies and procedures. They inform him that his server had been hacked and used to distribute illegal child porn, music files, and was used as a hop point to hack a top secret Department of Defense server. A forensic analysis reveals the hackers files and finds that the server had not been logging for six months and was missing several hot fixes. When asked about log retention, the administrator admits to not having time for the task. At this point the company web server is gone and they will not get it back for a very long time. The administrator will probably be fired but still have to go to court as a witness to the laws that have been broken.

Computer logs can be used in a court of law, and as such can be invaluable to a prosecution as well as the defense. In this somewhat exaggerated example, several laws have been broken. First, since a Department of Defense computer was attacked, and medical records were

involved, the Federal Computer Fraud & Abuse Act was probably broken. Second, the improper protection of medical records is a violation of the Health Insurance Portability and Accountability Act (HIPAA). Thirdly, the child pornography is a violation of statute 18 U.S.C. 2252A and the copyrighted files a violation of 18 U.S.C. 2319. Finally, if the hacker installed a sniffer, then the Electronic Communications Privacy Act was broken. This is summarized in the old saying; an ounce of prevention is worth a pound of cure.

As you can see in my fictitious example, event log monitoring is much more than an inconvenience. It can show the initial footsteps of hacker's movements on a server. Being alerted to these events can allow a quick defensive action against the attack. Because every Microsoft operating system has event logs, this issue applies to all of them. Most organizations will not have the resources to monitor servers and desktops, but all internet facing computers would be a good start. We have also seen by my example the consequences of not doing anything. Both for your career and your companies bottom line. What we need is a tool which will read through the event logs in real-time and make intelligent alerting decisions on events which may indicate an intrusion.

Let's take a look at what the event log is. When I try to understand something, I like to start from ground zero and work my way up. I will give examples from my own Windows Server 2003 Standard Edition server which is a HP Compaq DC7100 CMT. The event log put simply is a service which runs on all Microsoft operating systems and writes to three or more log files. You view the event logs with the event viewer. To access the event viewer, right click on "My Computer" and choose "Manage". Expand System Tools and Event Viewer to see the different log files. You typically will have Application, Security, and System as seen in the screen shot below.



You may have more logs underneath Event Viewer if you have installed DNS,

Active Directory, etc. Each log consists of a file located in %SYSTEMROOT%\system32\config. The Application, Security, and System log files are AppEvent.Evt, SecEvent.Evt, and SysEvent.Evt respectively. The logs grow as more events are written to them. You can define how large you want the logs to grow, and what to do if they reach the maximum size. In my screenshot I have the Application log selected and you can see that most of the events are being logged by my virus scanning software. This is indicated by "SpntLog" under the Source column. The Application log primarily logs events related to installed software. The Security log logs events relating to security events such as successful and failed logon attempts. Finally, the System log will log events relating to the health and operation of the operating system. Here you would find events such as low disk space and low memory errors.

There are three types of events. Above is the Informational type event. This is the least serious of the events. They are generally just informational. For example, my virus software is logging when it scans and updates. This is indicated by the white call out with the blue "I" inside it. A warning event indicated by a yellow triangle with an exclamation point inside it represents an event more serious than the informational but not as serious as the error event. You may receive such a warning when the server encounters transmission errors on the network. The error event is the most serious of the event types. Its symbol is a red circle with a white x inside. If your server completely loses its network connection, you may receive an error event stating such. Since we are primarily concerned with the security log, I will focus on that. When looking at the security log, you will have two symbols. The key represents a successful audit, such as a user logging on successfully. The padlock symbol represents a failure audit, such as a user attempting to log in with the wrong password. Note that these symbolic representations may vary between operating system versions.

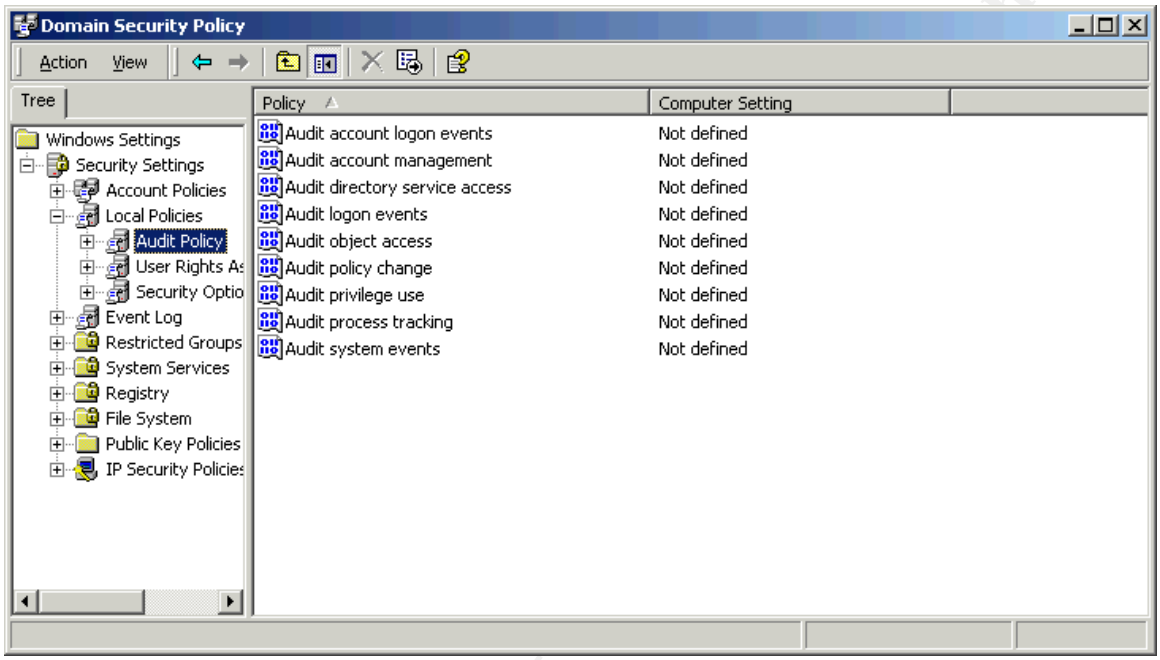
The auditing policy of the local machine and domain level group policy objects define what will be logged into the security log. On a domain controller this could easily be overwhelming. Choosing which actions to audit will be an art that will vary depending on your level of paranoia. Some events are more important than others, and some are only important when logged under certain circumstances.

Let's look at what an important event is. Ultimately this is what we want to know in order to sift through the piles of logs that will be generated. There are several articles and best practices which help us determine this. I've yet to find an exhaustive document which details event log analysis. I will use the information gathered in my research to form a baseline of events for which we can search through the logs.

To illustrate event analysis, I have created a fictitious domain using VMware Workstation version 3.2.0 to create a virtual sandbox environment. Virtual machines are true operating system environments which can be saved as a file to a local computer. They make it very easy to create, destroy, and re-create test environments.

My first virtual machine is a Windows 2000 Service Pack 4 Domain

Controller for the 321TSET.COM domain. My second virtual machine is a Windows XP Service Pack 1 client joined to the 321TSET.COM domain. By default no security logging is enabled in either the Domain Security Policy or the Default Domain Controller Policy. This can be seen depicted in the following screen shot of the Domain Security Policy:



The default Security Policy on my XP client also has the auditing categories set to Not Defined, therefore it does not log security events either. The Domain Security Policy is the Group Policy which applies to all users and computers in the domain. "In the Windows 2000 operating system, a Group Policy Object (GPO) is a collection of settings that define what a system will look like and how it will behave for a defined group of users. Microsoft provides a program snap-in that allows you to use the Group Policy Microsoft Management Console (MMC). The selections result in a Group Policy Object. The GPO is associated with selected Active Directory containers, such as sites, domains, or organizational units (OUs). The MMC allows you to create a GPO that defines registry-based polices, security options, software installation and maintenance options, scripts options, and folder redirection options" (searchWindowsSecurity.com). In essence, if we make a change here such as enable successful Audit Account Logon Events, we enable it for every logon from every machine in the domain. The domain controllers have their own GPO named the Default Domain Controllers Policy.

The choices you make for auditing will have a direct affect on what events get logged, how large the log file will grow, and system performance. Whereas the default logs the least amount of events, to enable each of these for success and failure would generate the most events. The larger your enterprise environment, the less you may want to log. You will have to decide what best fits your needs. With the above configuration, my test client did not generate events on the Domain Controller for either successful or failed logon attempts. I

would suggest the following settings for both the Default Domain Policy and the Default Domain Controller Policy:

Audit Account Logon Events	Success, Failure
Audit Account Management	Success, Failure
Audit Directory Service Access	Failure
Audit Logon Events	Success, Failure
Audit Object Access	Failure
Audit Policy Change	Success, Failure
Audit Privilege Use	Failure
Audit Process Tracking	Not Defined
Audit System Events	Success, Failure

These settings closely mimic what you would see in a default Windows Server 2003 domain controller installation. If you only make these changes on the Default Domain Policy, then only member servers and workstations will log events. Also, if you only edit the Default Domain Controller Policy, only the domain controllers will log events.

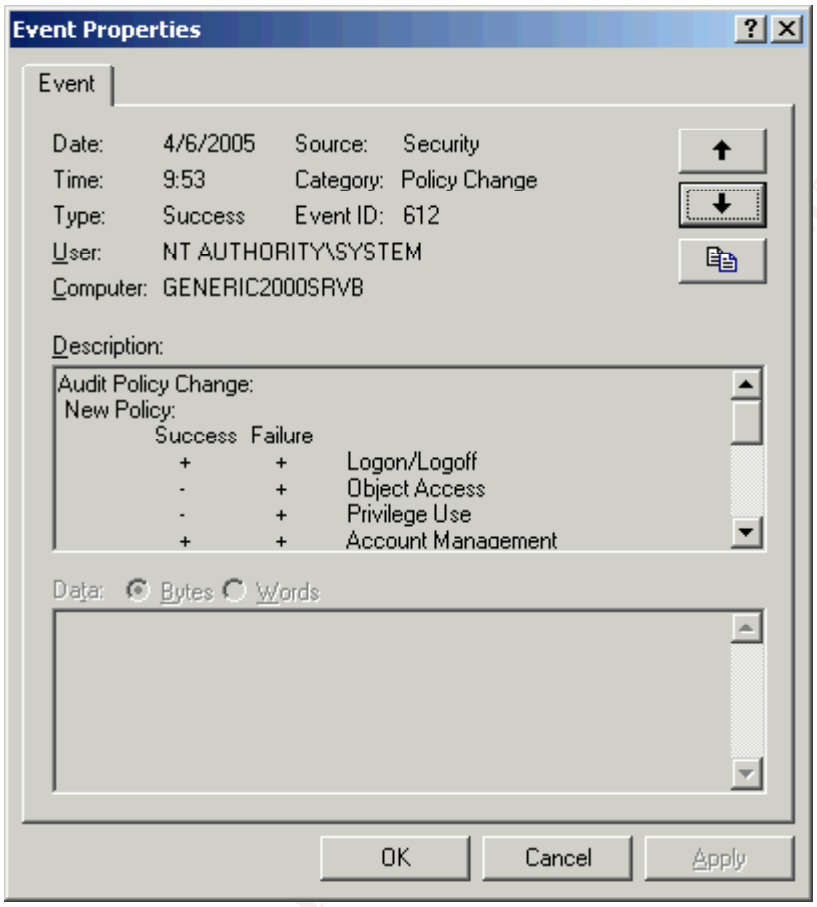
It is important to understand what these categories represent. As the name implies, Audit Account Logon Events create events for successful and unsuccessful logons as it pertains to domain level authentication. This can easily be confused with Audit Logon Events which generates events when a user logs on and off a workstation. Audit Account Logon Events was created to fill some of the holes that the Audit Logon Events had. "The problem with Audit logon events and logon and logoff is that Windows 2000 and NT record these events on the system on which the logon occurs. When a user logs on interactively at a workstation, Windows 2000 and NT record the logon event in the local workstation's Security log—if you've turned on audit policy at the workstation. When a user connects to a server over the network (e.g., by using a drive mapping), Windows 2000 and NT record the network logon on the server's Security log. As a result, logon and logoff activity events are scattered across every system in your network. Microsoft heard our complaints and added the Audit account logon events category, which tracks user authentication at centralized points: the DC's in your domain" (Smith). Furthermore, the client's workstation has to authenticate to the domain controller also. This will generate similar authentication events as seen by user authentication.

Audit Account Management is also another important category to define. These events help you track the creation, deletion, and modification of users and groups. Some important events to look for here are user additions to administrative groups, and account lockouts.

Object Access and Directory Service Access are similar in that when combined they allow you to audit any object. Generally Object Access allows you to audit such things as files on a file server while Directory Service Access allows you to audit Active Directory objects such as users. Both of these auditing categories will require you to set or modify the auditing properties on

the objects themselves as well as turning on Object Access and Directory Service Access.

The Audit Policy Change category allows you to monitor the success and failure of changes to any of the nine Audit Policy categories. A wise attacker will try to change the Audit Policy so that his movements do not get recorded. The following screen shot is of the one generated when I set the audit policy earlier:



As you can see, there are two columns with plus signs indicating that an auditing category has been enabled, and a minus sign which indicates a setting of not defined.

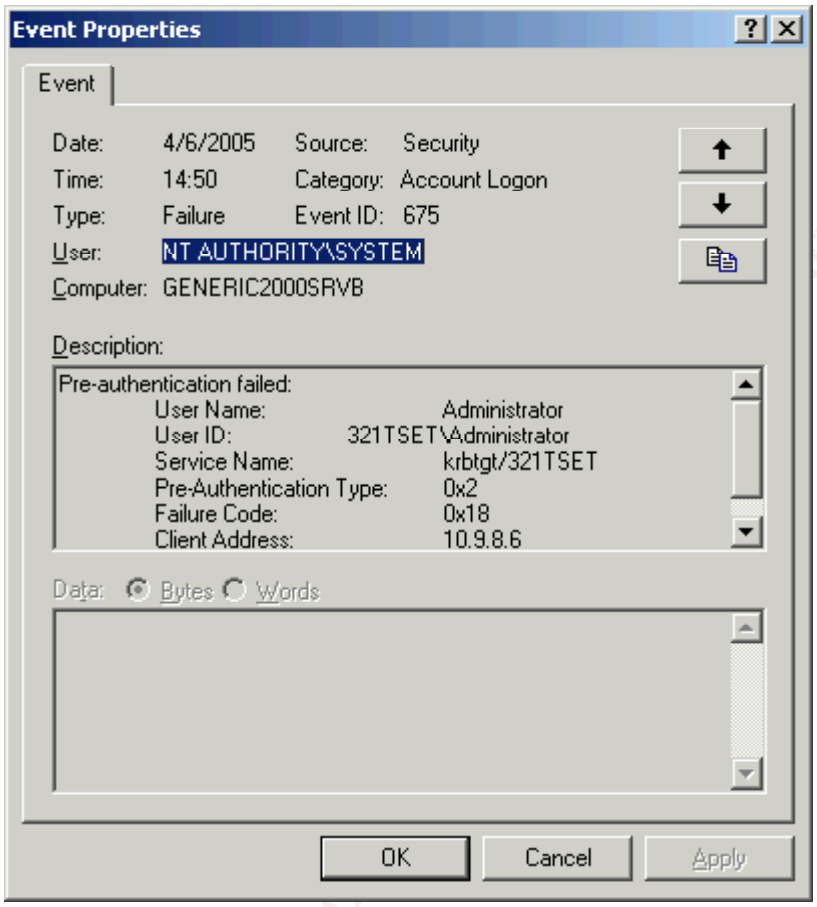
Audit Privilege Use will indicate when a user successfully or unsuccessfully uses a right. This can generate a lot of events. You may want to leave Success not defined and look only for Failed attempts, particularly for the Act As Part Of The Operating System right.

Audit Process Tracking will generate events which show when a process on the local computer is started and stopped. This can also generate a high volume of events and should be used only for specific troubleshooting.

The Audit System Events policy will enable you to track when a computer was rebooted and if the security log was cleared. You want to focus on unscheduled reboots, and investigate any occurrences of the security log being

cleared.

Now that we know more about how the event gets created, let's look more closely at the event itself. An event is something that happens on the server which gets put into one of the logs seen in event viewer. Below is a screenshot of an event.



As you can see, there is a lot of good information in this event. The information is organized by fields which contain data about the event. Let's analyze this event by starting with the Date and Time fields.

Date and Time is important for several reasons. The time an event happened gives us a forensic starting point for when an attack may have started, or how it may have progressed and ended. If the events are happening at night when no one is working, that raises a red flag. Or if many of the same event are happening seconds apart, this might be an indication of an automated attack. As you can see from the screenshots Event ID field, this is a 675 event which has been logged to the security log as indicated in the Source field. It is a failure audit of a failed attempt to log in as Administrator to the 321TSET domain. The User and Computer fields inform us that the event was logged on GENERIC2000SRVB, my domain controller, by the local System account. In this case, the Category field indicates that this event applies to Account Logon activity, and the Type field indicates it is a Failure Audit. A failure like this

should raise suspicion for several reasons. First, best practices state that administrators should not use the built in administrator accounts, so if you are following best practices you might wonder who is trying to do that. Second, if you have renamed the Administrator account to something else, the logons are being initiated by someone who doesn't know the new user name. If you were to see many consecutive events such as this, it may indicate a brute force attack against the Administrator account. If you see many of these events with a user from your domain, it may indicate a forgotten password, recently changed password, hack attack, or a virus infected machine. If you have an employee with a virus infected machine, then the Client Address fields IP information in the event description field is invaluable to tracking down and cleaning the machine. The event description field provides other useful information as well. Not only does it give the name of the user and IP of the client's workstation, it also has two hex codes which are very useful. When reading these codes, anything prefixed by "0x" is meant to be read as a hexadecimal number. The Pre-Authentication Type field value of 0x2 tells us that the logon was of the interactive nature. This helps determine whether a live body was doing something at a computer to generate this, or if an automated script or mapped drive was the culprit. The Failure code of 0x18 when converted to decimal equals 24. This means the user was valid, but the password was incorrect.

The 675 event would be an important one to audit. It falls under Audit Account Logon Events category. The following are some other ones commonly audited along with what they represent.

Events Generated By The Audit Logon Events Policy

Event ID 529 - This is caused by a failed logon attempt due to a bad username or password. Treat this event as you would the 675.

Event ID 539 - This is generated because an attempt was made to log in with a locked out account. You want to make sure this event wasn't generated because of repeated 529 events.

Events Generated By The Audit Account Management Policy

Event ID 624 - A user account was created. This may be hard to audit in large organizations. Some red flags would be users created during off hours, and newly created users which then are added to an administrative group (event 632).

Event ID 632 - This is generated because membership to a global group was modified. If you see this event, make sure it wasn't because someone added themselves to a group with administrative credentials such as Domain Admins.

Event ID 633 - This represents the removal of membership to a global group. If the account is removed from an administrative group, you want to make sure it's

not someone who got elevated privileges long enough to give themselves a backdoor, and then removed themselves from the group so as to not be recognized.

Event ID 636 - This gets generated when a user is added to a local group. You may want to use this to monitor additions to the local administrator group on your member servers.

Events Generated By The Audit Policy Change Policy

Event ID 608 & 609 - Respectively, these are generated by adding and removing rights to a user account. This may indicate a user has elevated his privileges. Events with the addition of the Act As Part Of The Operating System right should be investigated.

Event ID 610 & 611 - These events are generated when a domain trust relationship is created (610) and removed (611). You want to make sure you don't have trust relationships with unknown or unnecessary domains.

Event ID 612 - This event occurs if an audit policy was modified. Hackers may change the policy to exclude auditing activities they want to hide.

Events Generated By The Audit System Events Policy

Event ID 512 & 513 - These system events are created when a server starts up and when it shuts down. If you see these events at unscheduled times you may want to investigate. Some hackers upon first compromising a server will write startup batch files which configure the server to their needs. Then they will reboot the server.

Event ID 514 - An authentication package has been loaded. You can have more than one authentication package which can authenticate user requests. Ensure that the ones being loaded are valid. Attackers may operate at this level to steal passwords. The standard authentication package is Kerberos for domain authentication and MSV1_0 for local authentication and can be checked in the registry at HKEY_LOCAL_MACHINE\SYSTEM\CURRENTCONTROLSET\CONTROL\LSA.

Event ID 516 - This event is generated when events are being overwritten by newer ones. A large deluge of events may indicate an attack.

Event ID 517 - An audit log was cleared. Hackers may clear the logs to cover their tracks. Sometimes attackers will disable security logging altogether. Generally there should always be daily events in the security log.

Event ID 518 - A notification package has been loaded by the Security Account

Manager. The standard notification packages are scecli, kdcsvc, and rassfm. These packages can be verified in the registry at HKEY_LOCAL_MACHINE\SYSTEM\CURRENTCONTROLSET\CONTROL\LSA.

This is not an exhaustive list of event ID's. I have tried to pick out some of the more important ones that you may want to have alarms tied to. Different environments will dictate if any or all of these events are important.

Section II
Products Which Offer Solutions To Capturing And Monitoring Event Logs

There are many tools and third-party programs available to monitor event logs. I will present my own custom built solution and several third-party programs which can be bought retail. The thing to remember is that there is a solution in your price range. Money should not be a reason to not archive and monitor event logs.

My solution is made of readily available and free tools. I used simple batch techniques to automate the process. I limited the batch files to simple commands which should run in almost any Windows 2000 and above environment. The primary tool I used to collect the logs was psloglist.exe. This is a freeware utility distributed by Sysinternals which collects log files from Windows machines. This tool is very good at retrieving log files. Its biggest limitation is the fact that it can only filter up to 10 event ID's. I didn't find this limitation overwhelming since ten events turned out to be all that most people would need anyway. You can find psloglist.exe and more freeware tools at http://www.sysinternals.com. One obstacle you will need to overcome when dealing with log files is their shear size. Even when filtering on ten event ID's, I was still pulling down seven to ten megabyte log files from my busy domain controllers. I solved this quite simply with another freeware utility from http://www.gzip.org called gzip.exe. I found this very easy to use from the command line which made it easy to script. I achieved extremely good compression on my log files using this utility. With just a few hours work I made a batch file which used these two utilities to collect and compress log files from my domain controllers. My last step was to use windows task scheduler to kick it off at night. As far as log archiving, this solution achieves everything you would want it to do. One of its major shortcomings is the inability to filter on more that ten event ID's at a time. This can be easily overcome by writing another batch file which polls for another ten event ID's. There is also no alerting in my solution. This also would not be too difficult to overcome. I found several white papers which would put this functionality into my solution. I decided that it would have created too much e-mail traffic based on the event ID's I was interested in.

The next product I would like to cover which I use myself is called GFiLANguard Security Event Log Monitor (SELM) by http://www.gfi.com/. We currently use this product to keep about a weeks worth of logs from our two root domain controllers. This product has an alerting system which we use to notify

us if any of the administrative group memberships are altered. This product requires either an Access database, an MSDE database, or a SQL Server database. I found the Access database to run fine for small networks. If you are going to have a database over 2 GB, then you will have to go with SQL Server. One downside to this product is that it slows down tremendously when the database starts reaching this size, even with SQL Server. The product is simple to set up and use. It also offers many features beyond what you would normally need for archiving and monitoring log files. You will need administrative rights to read the security logs on your target machines. There is a "Connect As" option which is used for this purpose. No agent is required on the target machines and communications is done via normal Microsoft networking ports. This makes it a less obtrusive alternative to other programs. By default it is set up to monitor for most of the common event ID's previously mentioned in section I. All that is required is for you to point it to the machines you want to monitor. You can assign any monitored machine as low, medium, or high security. This determines the default settings, which may vary depending on the machines role.

There are four main consoles you will need to become familiar with. The first and most important is the Configuration console. This is where you define your rules, computers to monitor, alerting, and database options. The second is the Event viewer console. This is where you will view the events which matched the criteria in the Configuration console and hence were logged. The third console is the Reporting console. This is where you can create html formatted reports for the events which have been captured. The final console is the Status Monitor console. This is just a small application that runs in the system tray. It gives a summary of actions taken. This utility can be installed stand-alone on any workstation and act as a monitoring tool. It will allow the administrator to see current scans, important events logged, operational history, and a graphical pane showing logging activity.

This product comes with noise filters already enabled, so you will not be pulling down the entire logs by default. Disabling the noise filters is quite easy but will cause your databases to grow exponentially. In my test lab, the database went from 350 Kilobytes to 20 Megabytes when noise filtering was turned off. The product is not designed well in respect to managing the databases it will create. You can schedule the databases to compact, but the database size will not shrink nearly as much as my compression solution. You can have the program back up events older than a specified time period, so you would only need to script the compression of these backup databases. You can schedule log reads in either real-time, or increments of hours or days.

The power of this program is in the event processing rules. These define what to look for and what to do about it. A great feature is the ability to search on not only the event ID, but all the fields including the description. Furthermore, you can define your organizations business hours which will allow you to have different classifications for the same event based on time of day. You can get even more granular by filtering on the computer name and operating system. Rules can also have classifications in this product. You can keep the default

classifications or change them. The choices are Critical, High, Medium, Low, and Unclassified. These classifications are helpful when you go to view events. You may only want to see Critical events, so they would be separated in their own category. This is also useful when using the reporting tool. I found the reporting tool very useful and easy to use. It comes with many reports already created for you. Each rule can have its own action. The different actions are Archive, E-mail, Inform Status Monitor, and Run Command/File. Typically critical events will e-mail the administrator while the others simply log.

Languard SELM will also pull the application, system, DNS, Active Directory, and File Replication logs. All of these features allow for very granular event log filtering, but can be problematic when you create your own rules. The rule creation process is somewhat problematic since you will need to know which event ID you are looking for. Sometimes this means making the event happen so you can record the event ID.

I also looked at a collection of event logging software by Dorian Software Creations, Inc. (http://www.doriansoft.com). I didn't find this software to be quite as refined as GFI's, but it certainly got the job done. They take an interesting approach to their logging software in that it consists of multiple installable programs which can all be used together or independently. The first program is Event Archiver Enterprise. This program will retrieve log files from servers. I found the interface to be very simple, flexible, and able. One downside is the manner in which you initiate the log retrieval process. With most log archiving programs, the software simply connects directly to the logging service on the server and pulls the events straight from there. Dorian's program copies the events to a log "staging area" on the server. After logs are generated here, you must then tell the program where you want to put it. In my case I told the program to copy it to a file share on my personal server. You could also have the file copied to an FTP server. A great feature that is extremely helpful is the ability to automatically have the log files compressed. Once you have your log files archived, you are ready to analyze them.

Event Analyst is the next installable program available from Dorian. You use this tool to analyze events much like in the Windows Event Viewer. I found this tool prone to hang and was very slow at viewing large log files. I later found that this program was much more reliable on Windows XP than Windows 2000. The filtering options were not as refined as GFI's. The option to output your searches to HTML was slow, but very useful. The reporting function came with good canned queries and was very useful.

The final piece of the Dorian logging suite is the Event Alarm. This program monitors logs in real time and sends alerts on specified criteria. There are five alert options this product shares with GFI; e-mail, pager, popup, database, and syslog. Event Alarm has convenient canned alert options which keeps you from having to know specific event ID's, but does not offer the granularity of the GFI product. For example, with GFI, I could specify to filter on any field of an event and combinations thereof. With Dorian I was limited to general event information. This was particularly bothersome when multiple hits of my event were recorded and I was sent a flood of e-mails. The alarms are

supposed to be real-time, but in my experience several hours passed before receiving alarms.

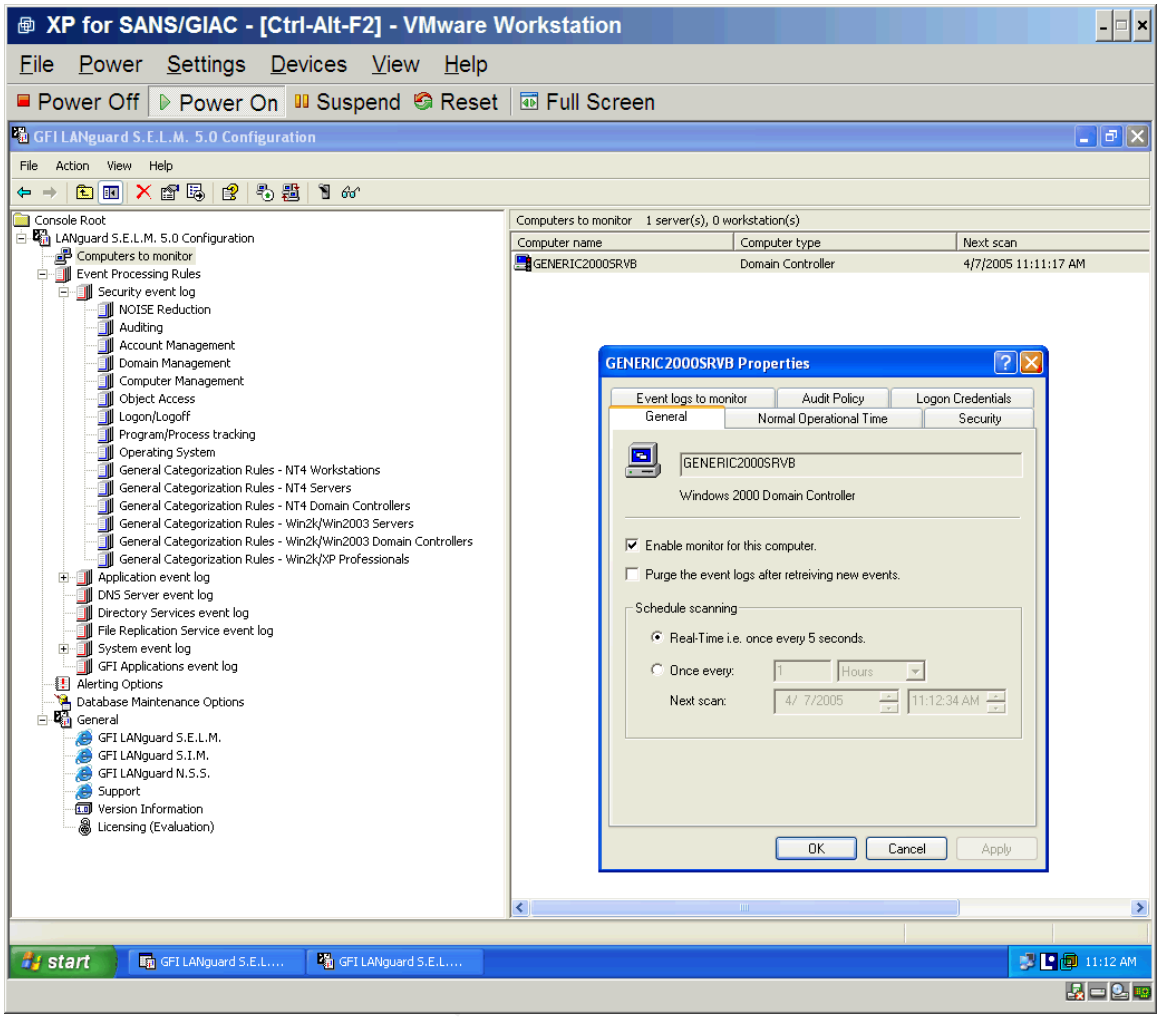
I conclude that although there are many options available to archive, filter, and alert on events, GFI's product was by far the most superior in almost every aspect. I would not hesitate to recommend the product for large and small organizations.

Section III
Implementing GFiLANguard Security Event Log Monitor (SELM)

To show how to install and configure GFI's SELM, I will create a sandbox environment on my machine consisting of a vanilla Windows XP SP1 installation running on VMWare. The first step is acquiring the software from the GFI webpage at http://www.gfi.com/ and executing it. Currently the installation package is named languardselm5.exe and is approximately 12 MB in size.

The install routine is much like any windows based program. Choose "Next" until you reach the license agreement to which you should read and accept. If you do not have Microsoft Message Queuing Service (MSMQ), then you will need to install it before you can continue. At this point you will be asked to specify an account with domain administrative privileges for the service to run under. Afterwards you will need to specify a password for a service account which will be created on the local machine. Finally, you will be asked to use either Microsoft Access or SQL Server as the database engine. From here you can use the wizard based configuration, or configure the product manually.

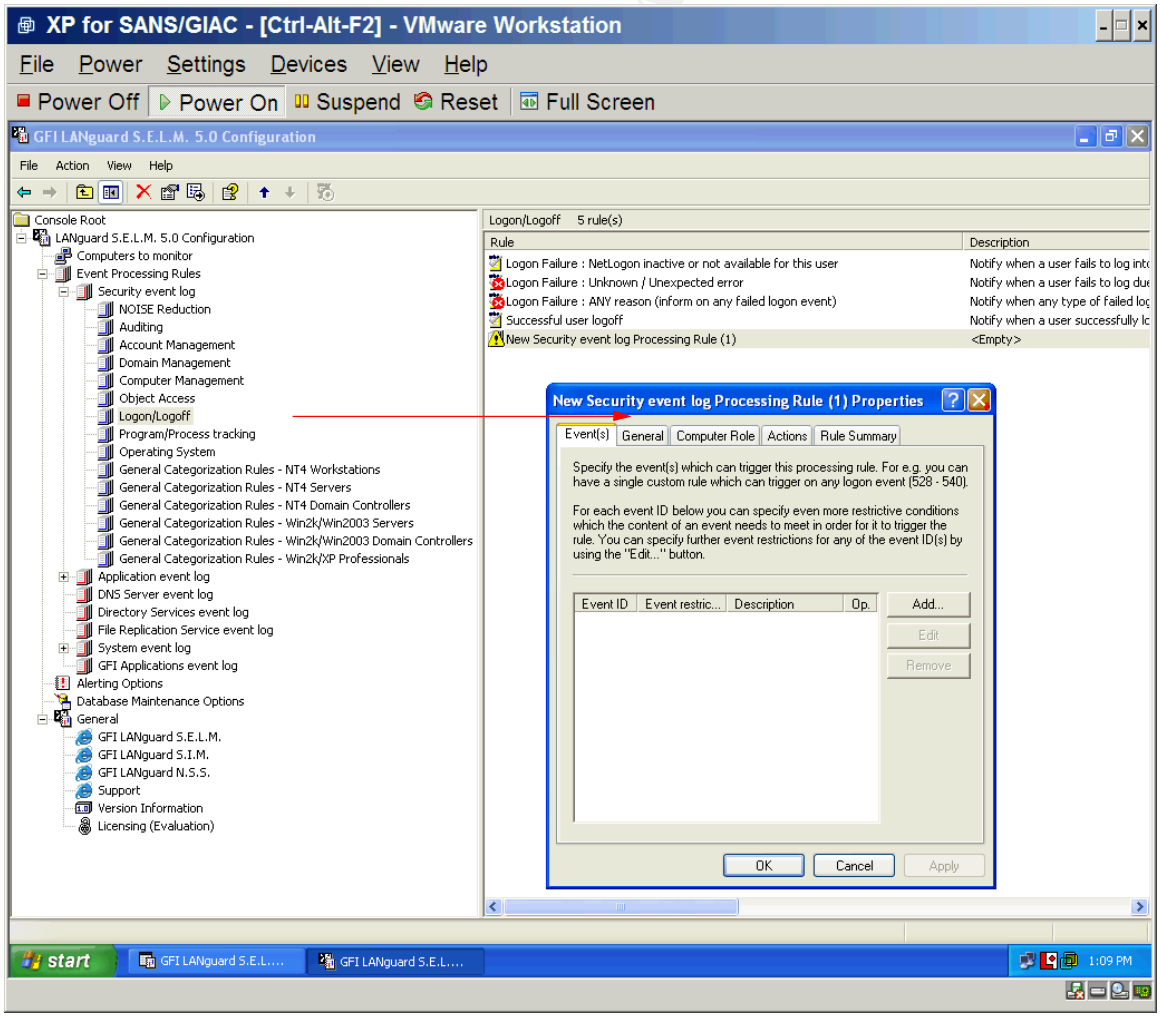
The install routine will kick off the Initialization Wizard which offers the easiest method for getting up and running. Your first step in the wizard is to configure the alerter settings. I configured the e-mails to come to my personal account via an internal SMTP server; you could alternatively create a generic mail account for multiple administrators to monitor. At this point you will need to either enter a license number or use the evaluation version of the software. Choose the option to "specify computers to monitor later"; otherwise it will try to enumerate all the hosts in your domain. Choose "No, I will enable the auditing policies manually". By now you should have a good idea of what you want to audit. Keep in mind that if your audit policy does not force workstations and servers to log important events, then GFI will never pull the event or send an alarm on it. Finally, select to open the SELM Configuration dialog box. This is the main configuration console as depicted below:



This console has the same look and feel as a normal Microsoft Management Console. Within the configuration window you have LANguard S.E.L.M. 5.0 Configuration under which are the configuration categories. The first step is to right click on Computers to monitor and choose New, then Single computer entry. This will open a browse window in which you can select the server you wish to monitor. Once the server is selected, it is added to the right pane of the configuration window and a properties box is displayed. The default is to not monitor, so check the box to enable monitor for this computer. After you apply and close this screen you will be monitoring the security log for that server with the default settings. This process is depicted with the red arrows in the diagram above. Most likely you will want to tweak the default settings for your environment. High security environments may want to choose the Real-time schedule so that administrators are alerted as soon as possible. Adjust the Normal Operational Time to match your enterprise. This has a direct affect on how the rules you create get processed. A rule that is triggered during business hours may be low priority, whereas the same rule triggered after hours would be considered high priority. The rule for Audit Policy Changes is one such rule. Changing the security level has a direct impact on how the rules are handled also. Events that would be low priority on a low security server may be high

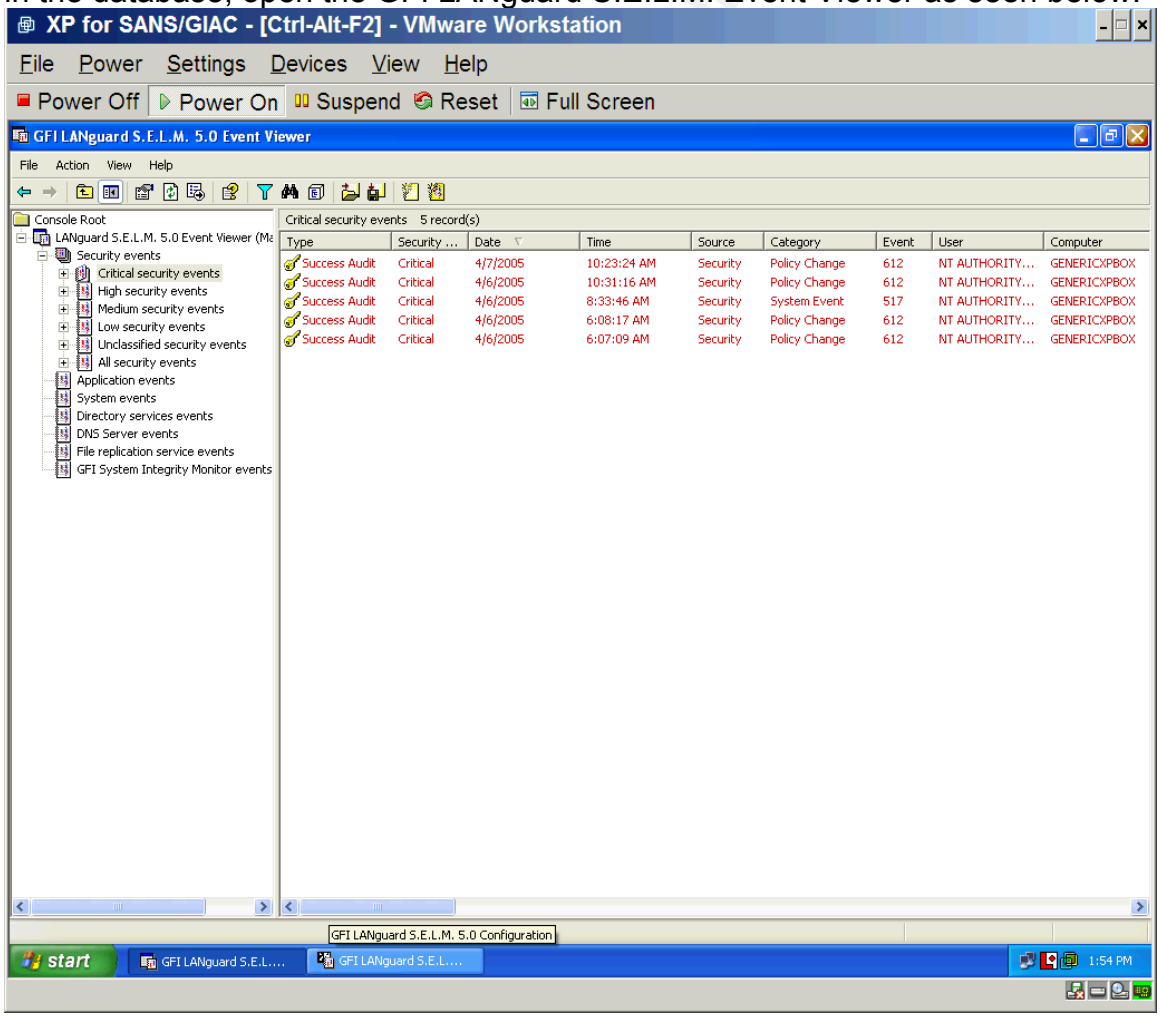
priority on a high security server. By moving the slider to the High security end, you change the properties of the default rule set. Where the rule for a low security server may simply log and archive for an event, a high security server may log, archive, and send an administrative alert.

Underneath the Computers to monitor are the Event Processing Rules. This product has the capability to collect and act on other log files, but I will be focusing on the Security event log. When you expand the Security event log you will see several categories of rules. They have organized the rules into sections that closely match the nine auditing policies covered earlier. The noise reduction category is a set of rules designed to ignore common low priority events in an attempt to reduce the log size. Disable all the rules in this category. The general Categorization Rules is where you will find most of the rules that will apply to your environment. For my test example, that would be the Win2k/Win2003 Domain Controllers. You can use these as they are, modify them, or create your own rules. To create a rule, right click on the category which most closely matches the type of event to audit and select New, Processing rule. You should have a window similar to the one in the screen shot below:



Click Add to begin the creation of the new rule. You will need to know the event ID number of the event you wish to monitor. As you type it in, the Event short description field will populate with the events type. The next window allows you to define the values of the event fields you want matched. This allows for a high degree of granularity and will help to keep unwanted events from being alerted on. The General tab allows you to assign this rule to a specific computer or any computer as well as time of day. You can also filter by operating system on the next tab titled Computer Role. The Actions tab is where you define what you want to happen when an event triggers your rule and how to classify that rule. The classifications are arbitrary and will depend on your own enterprise environment. The classifications are Critical, High, Medium, Low, and Unclassified. Your React by choices are Archive, Email, Inform LANguard S.E.L.M. Monitor, and Run command/file. The final Rule Summary tab gives a simplistic description of the rule you have created.

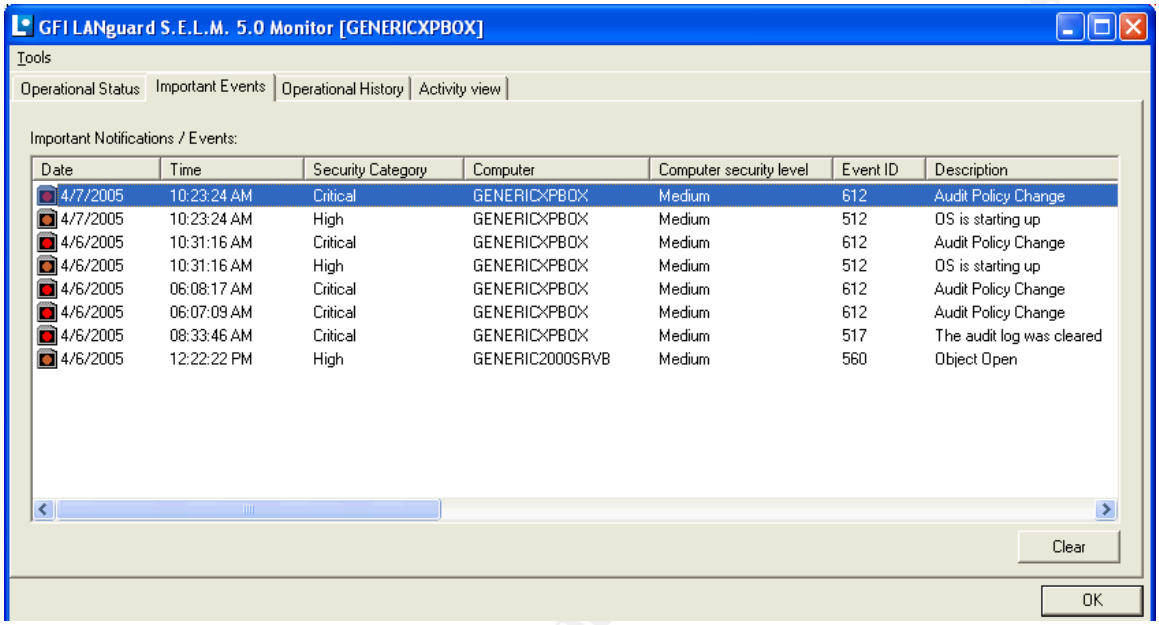
At this point LANguard is pulling events and matching them against your rule set. It is also saving the events to a database. To view the archived events in the database, open the GFI LANguard S.E.L.M. Event Viewer as seen below:



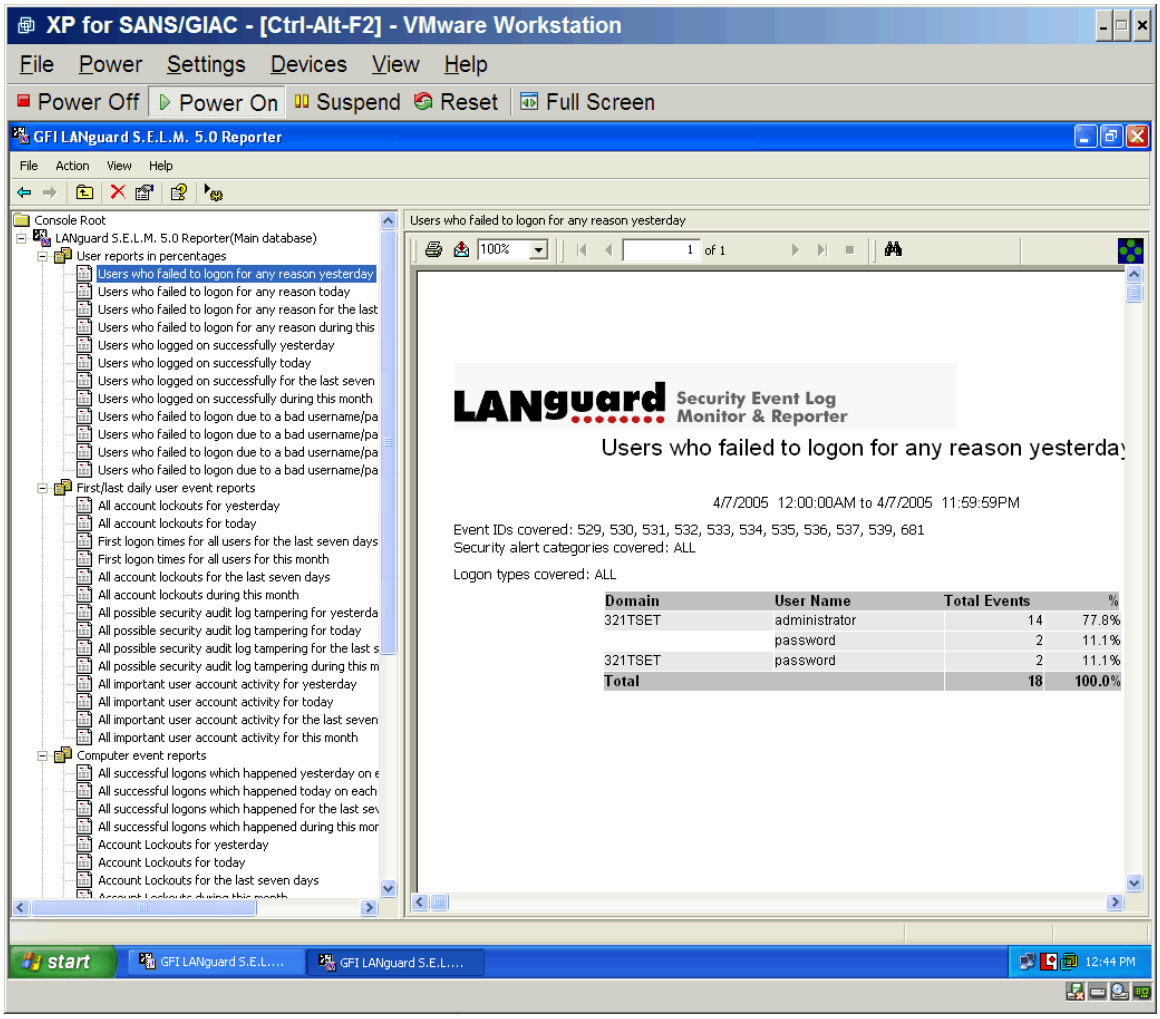
The security events have been broken down into classifications which were defined earlier in the rule creation. This separation of events helps you

separate and easily view the more important events.

The GFI LANguard S.E.L.M. Monitor is a useful tool which provides summary information for the rule processing. It can be loaded independently on a workstation and is customizable. Below is a screen shot of the GFI LANguard S.E.L.M. Monitor:

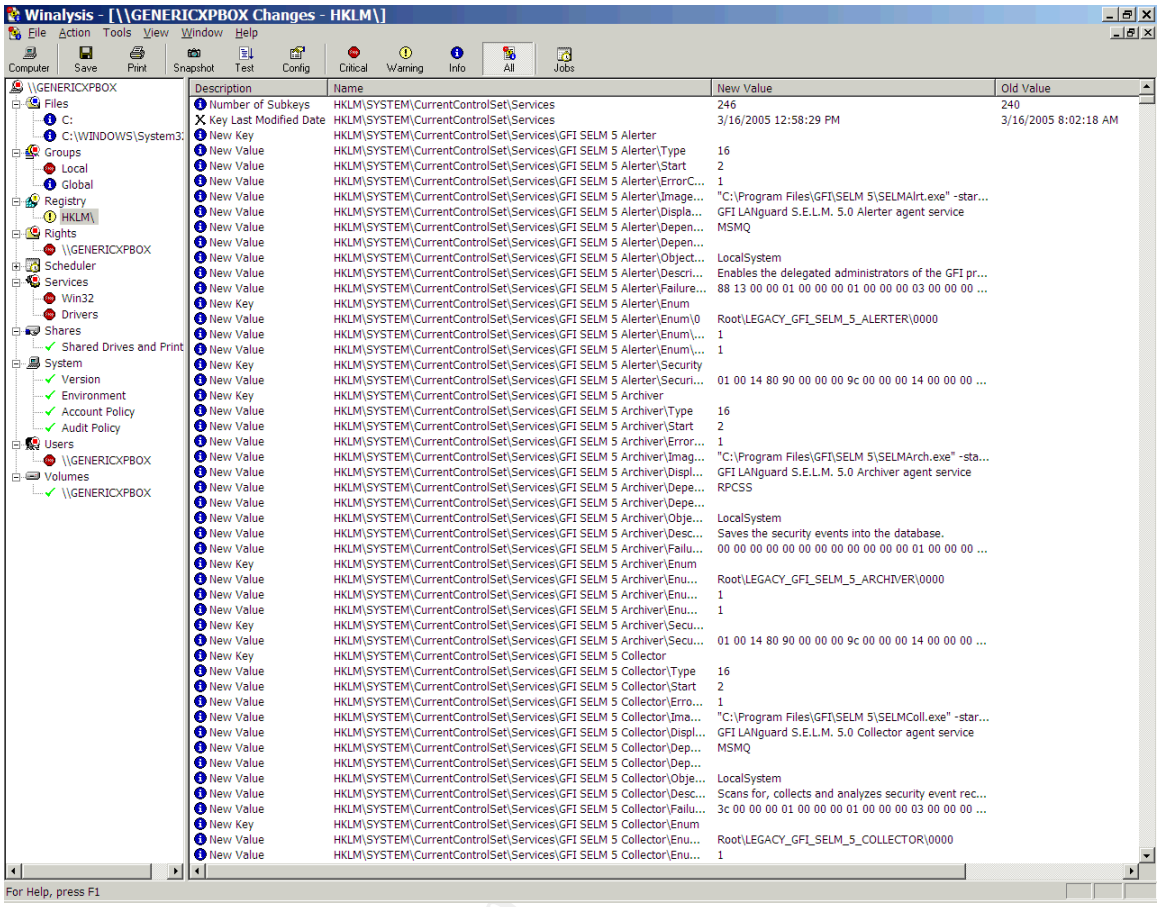


The final tool which makes up the LANguard suite is the Reporting tool. I found this to be extremely easy to use and very informative. Below is a screenshot of the main reporting window:



The reporter tool comes with many canned reports. Simply right click on the report you wish to view, and choose to generate the report. In the example above, you will see the results of the Users who failed logon for any reason yesterday report.

The installation of this product does have security implications. Using a tool called Winalysis by Winalysis software (http://www.winalysis.com/) I was able to summarize the changes it made to my local machine. SELM creates a local user named “GFI_MONITOR_USR” and adds it to the local administrators group and gives it the “SeBatchLogonRight” right. Hundreds if not thousands of registry entries are created during the installation. The following snapshot shows a sample of them as displayed by Winalysis:



Four new services, Message Queuing, GFI Alerter agent service, GFI Archiver agent service, and GFI Collector agent service are added to the local machine. These services will be configured to run as a domain account with administrative privileges. You could however create a custom service account to use which has only explicit rights to read the security log. Distributed Transaction Coordinator, Remote Access Connection Manager, and Telephony are all services which were changed from the "Stopped" state to "Started". I ran a comparison of a full port scan before installing the software and afterwards. TCP ports 1054, 1056, 1801, 2103, 2107, and 2105 were all opened. UDP ports 3527 and 1055 were also opened. Services and applications spawn processes which listen for calls by opening ports on the local computer. If a vulnerability for the application comes out, you could be at risk of virus infection or hack attacks. Consider controlling which servers will be communicating over these ports using a technology such as IPSec. "Internet Protocol Security (IPsec) is a framework of open standards for ensuring private, secure communications over Internet Protocol (IP) networks, through the use of cryptographic security services. IPsec supports network-level peer authentication, data origin authentication, data integrity, data confidentiality (encryption), and replay protection" (Microsoft.com).

Conclusion

I hope I have shown how critically valuable the event log is. If you want to get started right away but need more time to research vendors, simply right click the log in event viewer and do a “Save As”. You can open the log later with the event viewer on your local computer and analyze it at your convenience. If you want to go back and retrieve past log files, see if your backup solution can easily restore the logs from the system directory. Although my main focus was on the Windows security log, there are many other devices on your network that generate log files. Consider a centralized logging solution which will help you paint a more thorough picture of traffic moving through your network.

References

"searchWindowsSecurity.com Definitions." Searchwindowssecurity.techtarget.com. 2005. 12 April, 2005. <http://searchwindowssecurity.techtarget.com/sDefinition/0,,sid45_gci847626,00.html>.

"Technology Centers." Microsoft.com. 2005. 12 April, 2005. <http://www.microsoft.com/windowsserver2003/technologies/networking/ipsec/default.mspx>

Smith, Randy Franklin. "Tracking Logon and Logoff Activity in Windows 2000." Windows 2000 Magazine (2001)