Windows Event Logs - Real Time Management and Administrative Alerts

Submitted by: Michael Stines
Date Submitted: July 25, 2004

A wealth of information may be found within the event logs of a Microsoft Windows server. Depending upon how the administrator of a given system has configured the server, the contents of these logs may range from information about basic system events to output from numerous audits. Additionally, event log entries may be generated by applications residing upon the server as well as roles and services provided by the server. The events contained within these logs may be used for a range of purposes, from troubleshooting a specific application error to confirming replication between domain controllers. This information provides essential data for an administrator. If used proactively, this data may be key in identification and deterrence of potential system, application or security issues. It provides insight into historical events that have occurred, allowing the administrator to establish a baseline of sorts, to be used for future reference. In summary, this data provides insight into the overall health of server, as well as that of any applications or services resident upon the server.

In order to be effective, these logs must be monitored and maintained. If a critical event is logged on a server that is not being regularly monitored, the first sign of trouble may be when the server, service or application fails to function properly. Although unique events contained within the logs may provide vital information to the administrator, the number of events specific to a particular server can grow into the thousands in a short period of time – thereby raising the distinct possibility that even the most vigilant administrator may overlook a key log entry. Furthermore, in a networked environment processes and systems often span numerous physical systems – in which events among numerous servers must be correlated, event log management can quickly become an overwhelming task. Obviously, the task of monitoring and managing Windows event logs in this fashion within an environment consisting of numerous Windows servers is not an easy one. With event logs distributed across numerous systems, timely detection and reaction to a critical event presents itself as a challenge. Rather than attempt to monitor numerous systems each as an individual entity, a centralized collection point of significant events is proposed. From this collection point, three key elements could be addressed:

1. Incident detection
2. Alerts generated in response to critical events
3. Log archival

As a typical Windows server maintains it's own set of event logs, and does not have the built-in ability to write events to a central collection point, incident detection quickly surfaces as a key factor of log centralization. Distributed event logging poses a challenge for the management of large numbers of servers, due to the time required to manually review the details of events for each server in an appropriate fashion and due to the fact that each log may contain related although segregated data. Furthermore it raises a security risk. Events of a given server often need to be considered in context with the events recorded in other servers. For example, should three servers each record a login failure event for the same username – and a fourth server records a login success, did an incident occur? Observed individually, the three single login failure events and subsequent single login success event may fail to raise alarm. Yet when reviewed in context with the cumulative records of each of the four servers, the administrator may see that a possible unauthorized user has gained access to the fourth server. Through the use of event log centralization of significant events; recorded events spanning multiple servers may be viewed as a whole, at which point analysis may be performed upon the cumulative collection of logged events. This information would be of particular value in the review of security events, as well as events of distributed processes which span multiple servers. Event log centralization allows the administrator to make the best use of the data otherwise contained within distributed locations.

The preceding describes the need to examine security event logs in context with records from other servers. Yet there is another challenge involved with manual event log review that may not be readily apparent. An issue with this type of event log analysis is time. Everything is historical. Even if an administrator reviews logs - distributed or centralized - on a frequent, scheduled basis – all the logs analyzed are records of past events. Windows event logs are just what the name implies – a compilation of events over time. Depending upon the nature of event, prompt response to a specific incident recorded within the logs may often be critical. Should event 2013 (The <disk drive> disk is at or near capacity. You may need to delete some files.) appear after hours Friday evening, the administrator must be alerted of the event in order to respond on a timely basis. Otherwise, following this example, should the c:\ partition reach capacity, this server may well be nonfunctional the following Monday when the staff returns to work.

As security events can compromise the integrity of a given server, critical security events need to be communicated promptly to the administrator and/or proper staff. Without capability for alerting, should an administrator be tasked with manual review of log files of numerous servers, it is conceivable that discovery of such an event may be hours; perhaps days after the event was originally logged. While Windows Performance Logs and Alerts allows the administrator to configure the server to log counters of specific events, and alerts be issued based upon these events, it is resource intensive – as each server must be manually configured as described. This process, addressed in Microsoft Knowledge Base Article – 300504 may serve to fulfill some needs, but it is not a scalable solution. Centralized logging offers the ability to realize real time alerting – in the form of email notification for specific events such as audit lapses, administrator login failure/success, audit policy modifications, administrative group membership changes, and more. Event log centralization of significant events, combined with an alerting mechanism addresses the time sensitivity of detection and response to a critical event.

With the advent of rules and regulations associated with items such as HIPAA, GLBA, and Sarbanes-Oxley among others that are almost certain to follow, data and log retention and archival are becoming increasingly common topics of conversation. (At this point it is necessary to present a disclaimer. If your organization is subject to any one or more of the above - or subject to any governing authority that addresses log retention – please consult with your company's legal department. This document does not dispense legal information, not does it imply that these procedures will satisfy legal requirements). Having stated that, log centralization of significant events does offer advantages in regard to archival. Native logging by individual servers presents scalability problems in larger environments. By default, every Windows 2000 Server logs Application, Security and System events. These logs are set to maximum log size of 512KB, and overwrite after 7 days. A domain controller is also likely to have logs for DNS, Directory Service and File Replication Service. The administrator will probably change the default log size to meet the needs or requirements of their organization's policy. If using local event logging, adequate consideration and planning must be dedicated to the creation and compliance with a log retention policy, as it is necessary to formulate a combination of log settings (size and overwrite) and archival procedures to avoid losing events. Loss of logged events is possible should logs fill and begin overwriting, prior to archival process. Alternatively, using centralized logging of significant events, all flagged events may simply be written to a central logging server, and archived daily.

Having identified a need for a centralized logging facility and benefits it offers, we must examine possible methods of achieving this need. This is necessary to compare risk levels of our proposed solution to possible alternatives, and to make the business case that our proposed solution is the correct solution. Three potential methods are:

1. Native tools built-in to Windows 2000, including Resource Kit Tools
2. Open-Source Products
3. Commercial Off The Shelf Software

Windows 2000 and the Windows 2000 Resource offers tools to achieve most, if not all of our goals of log centralization, alerting and retention. In his document *"Centralized Windows 2000 Event Logging, A Step-By-Step Guide"* (http://www.sans.org/rr/paper/67/1245.pdf), Scott Richardson has detailed the use the dumpel.exe and ntolog.exe tools to accomplish our goal of log centralization. Mr. Richardson clearly defines the process of scripting dumpel.exe to collect logs from remote servers, followed by their removal utilizing the ntolog.exe tool. Similarly dumpevt.exe, an open-source product is available at http://www.somarsoft.com/somarsoft_main.htm allows for remote retrieval of event logs. While these are solutions for log centralization and retention, it does not address our need for real time or near real time alerting. These centralized logs could indeed be queried or parsed, and alerts issued based upon the findings, but timeliness of these detects and subsequent alerts do not meet our requirement of real time alerting. Alternatively, rather than centralize logs we could simply prepare a script or sched task using dumpel to dump events to local

.txt file(s), then backup the servers, including the .txt files. Scripts again could be written to parse the information contained within the .txt files, and issue email alerts based upon findings. If our backup rotation is in accordance with legal requirements for log retention, we have succeeded in addressing log retention, but incident detection and alerting is not timely, as we cannot parse the .txt files until after the scheduled dump of the event logs.

By using select open-source software, we may build upon the native tools contained within Windows 2000 in order to achieve our goals. To meet our requirements, we need a centralized logging server, and a tool that will allow flagged events to be moved to the logging server in real time, or near real time. From this central server, events would be examined and alerts issued if a specific criterion is met. There are several products available in the open source community which will meet the above needs. Centralized logging may be accomplished with products such as Kiwi Syslog Daemon, or Winsyslog, among others. Notable events may be moved in real time or near real time using products such as SNARE Agent For Windows. Both Kiwi Syslog Daemon and Winsyslog are capable of generating administrative alerts – in the form of email or text pages, in response to specific, pre-programmed events. Events recorded by the central Syslog server would then be archived daily and then backed up to backup media on a pre-scheduled basis. This combination of products and systems meets all our goals of:

1. Centralization of significant events
2. Alerting in response to key events
3. Retention of these events.

The final consideration is Commercial Off-The-Shelf-Software. The need for event log management and alerting is widespread, and there are numerous commercial products available to meet this need. These products range in complexity, functionality, and price. Business needs, legal requirements, and functionality may require a commercial product as a solution for event log management. Products such as NetIQ’s Security Manager, GFI LANguard Security Event Log Manager, and Event Sentry are among the many products that may meet this need. Consideration must be given to verifying the potential product meets business and technical needs, functionality requirements, and are user-friendly. Additionally, consideration must be given to the question of “Can the current infrastructure support the product?” Attention must be given to items such as bandwidth requirements, resource requirements of host servers and workstations, and perhaps most importantly - storage requirements. If the organization is required to store the cumulative event logs of a significant amount of systems for an extended period, the data could grow to terabytes over a relative short span of time.

Having reviewed various options of meeting our goal of centralization of significant events, alerting of critical events and archival of these events; and having reviewed potential shortcomings and risks, we realize a balance of needs vs. rewards must be reached. The business must review needs and assign a value to them. How important is timely administrative alerting of critical events? What, if any legal or regulatory requirements in regards to event log retention must the business meet? Is the business committed to assigning staff to manage event logs, either manually at every server – or through a centralization process? Organizational or departmental management must be alerted to the risks addressed by of each of our three identified methods and make a business decision accordingly. Regardless of the size of business at hand, the benefits that event log centralization, alerting, and archival offers are comparable. The primary point of contention may be legal matters, primarily with regard to event log retention and archival. If the company has to comply with stringent regulations imposed by HIPAA, GLBA, Sarbanes-Oxley, or other regulatory requirement, it may be necessary to purchase an off-the-shelf product. The business must clearly understand any legal or regulatory directives and make a decision based upon these requirements. A number of organizations may not necessarily have the business need or funding available to purchase a commercial off the shelf product, yet have a need for event log centralization and subsequent administrative alerting - and given the fact the use of open source products will allow us these goals, we will examine the products and steps necessary to achieve this end.

As stated previously, Windows servers each maintain their own set of event logs, and do not have the built-in ability to write events to a central collection point. Understanding this fact, we need to locate a product that will allow our Windows servers to send data to a centralized collection point. For this purpose, Snare Agent for Windows will be used. Manufactured by InterSect Alliance, Snare Agent for Windows is described as *"… a Windows NT, Windows 2000, Windows XP, and Windows 2003 compatible service that interacts with the underlying Windows Eventlog subsystem to facilitate remote, real-time transfer of event log information."* In addition to the Snare Agent for Windows, Intersect Alliance also provides Snare Agent for Linux, Snare Agent for Solaris, Snare Agent for Lotus Notes, Snare Agent for IIS Web Servers, and Snare Agent for ISA Servers – all open source products. It is the Snare Agent for Windows that will be used to interface with the Windows event logging systems to read, filter and send select events to the remote logging host. The Snare Agent has the ability to read, filter and send specified logs from the DNS Server, File Replication, and Directory Service logs as well as from the System, Application and Security logs. Events within these logs are filtered in real-time according to parameters configured by the administrator and reported to the remote syslog server via the UDP protocol.

In addition to the core functionality of the product, a number of additional features are packed into the Snare Agent for Windows. The Agent itself may be installed locally from a downloadable installation file, or it may be installed remotely over the network onto multiple systems using tools provided by Intersect Alliance. Remote installation tools are RSNARE, which is a batch file that may be used to install the Snare Agent to the ADMIN$ share, another is SnareInstaller, a VBS script which performs a similar function. Once installed, the Snare Agent is fully configurable remotely via a web browser. The service may be stopped, started and all configuration parameters are available to the administrator via the browser. There is no need to touch all the servers to be monitored either for installation or for administration. Documentation and download information for the Snare Agent for Windows may be found at http://www.intersectalliance.com

Having identified a method of retrieving the events from our Windows servers in real time, we now require a syslog server that will serve as a central collection point. Choices for a syslog server are varied, ranging from the syslog daemon running on a Unix or Linux system to installable applications such as Winsyslog or Kiwi Syslog Daemon. It would be recommended to begin with a product in which the technical staff or administrators have familiarity. For this reason, we will utilize the Kiwi Syslog Daemon, as it is a commonly used application and is highly configurable. The Kiwi Syslog Daemon is a freeware product available from Kiwi, and may be found at www.kiwisyslog.com. The Kiwi Syslog Daemon is described as *"…a freeware Syslog Daemon for Windows. It receives, filters, logs, displays and forwards Syslog messages and SNMP traps from hosts such as routers, switches, Unix hosts and any other syslog enabled device."* The Kiwi Syslog Daemon is freeware, and there is no timeout on the freeware version. The features of the free version are numerous, key elements of interest are:

1. Easy to use GUI based manager
2. Runs as a service
3. Messages are displayed, and tasks are performed in real-time as they are received
4. Message logging or forwarding is available, based upon priority
5. Receives messages via UDP, TCP or SNMP
6. Automatic log file archival based upon a custom schedule
7. Syslog message buffering to ensure messages are not dropped when under a heavy load

The shareware version is free to use for as long as you wish. Kiwi offers a free, 30-day trial period of the full-featured version of Kiwi Syslog Daemon. After which time, it will lapse into the reduced functionality of the shareware version. However for the license fee of $99, the registered version and all it's additional features becomes a value. The licensed version offers numerous additional features, is more versatile and will be needed in this project. Key features of the licensed version are:

1. Filtering Options – Filters may be built based upon IP Address, Hostname, or Message Text. Unwanted messages may be filtered out and dropped. Actions may be performed in response to a message having specified keywords
2. Powerful scripting engine for filtering, parsing, and performing actions

3. Pass information such as message text, hostname, facility, or logging level from the received Syslog messages to an external program – such as email
4. Additional buffering of incoming messages, to further insure that messages are not dropped or lost when the Syslog server is under a heavy load

The licensed version of the Kiwi Syslog Daemon is highly configurable, allowing the administrator to create rules, filters, and actions to sort and manage incoming event log data. The GUI manager features a display of syslog messages received. It has 10 independently configurable display screens, and rules may be created to send specific messages to a particular display screen. This provides an instant filtering capability. It is through the use of such rules that incoming messages are processed. Message handling is further defined through the use of filters. Filters may be created and applied to the rules, specifying which message values and attributes are to be analyzed. Actions may be created to serve as a response when a specific condition is met. Available actions include the sounding of an audio alarm, issuance of email messages, and forward messages to another host. Up to 100 rules may be defined within a Kiwi syslog server. For each rule, up to 100 filters and 100 actions may be applied. Additionally, the licensed version of the Kiwi syslog server offers powerful archival tools for messages received. Log files may be separated hourly, daily, weekly, monthly, or on a schedule determined by the administrator. Furthermore, logs may be archived in response to rules and filters – archives can be created based upon priority, host name, message text, IP address, and/or domain name.

While necessary steps and configuration examples will be provided to achieve our stated goals, a comprehensive knowledge of the capabilities and use of the Kiwi Syslog Daemon is encouraged. An excellent, and highly recommended resource is the document “Effective Logging & Use of the Kiwi Syslog Utility” by Brian R. Wilkins. www.sans.org/rr/papers/33/201.pdf
In his document, Mr. Wilkins begins by walking the reader through an introduction to syslog and an overview of the Kiwi Syslog Daemon, and goes into an in-depth explanation of its configuration and potential uses.

So, let put this all together and make it work. We’ll begin with the server that is to host the Kiwi Syslog Daemon. The Kiwi Syslog Daemon may be installed on Windows XP, Windows 2000 Professional, Windows 2000 Server, or Windows 2003 Server. For this exercise, we’ll install it on Windows 2000 Server, as this is likely to be the version administrators are most familiar with. Aside from the compatible operating systems, Kiwi does not post their minimum hardware requirements. This is likely to be a subjective value, as the load will vary from one installation site to another. This server should have adequate memory to buffer incoming messages, the CPU speed should be enough to handle the needs of the syslog service and the OS, and storage space should be plentiful.

In other words, the server may not need to be the “latest and greatest”, but you also probably don’t want to use an antiquated system.

Consideration must be given to the security of the server that is to host the Kiwi Syslog Daemon, as this server will be storing critical information captured from a number of servers, and this data must be protected.   For this reason, the syslog server should be a dedicated server, and not fill other functions – such as print server, fileserver, etc. Also give thought to the name assigned to this server. It should be something nonspecific that does not define the role it plays. Should an intruder, internal or external, gain unauthorized access to the network – the syslog server is likely to be targeted. An intruder will often kill the syslog services, and attempt to alter or destroy the log history to hide their activities, and we do not wish to assist such an intruder by assigning a name such as SyslogServer – or something similar - to this machine. This server should be physically secure, behind a firewall in a location not readily accessible. This server may be either a member server of your domain, or it may be an out-of-band server in a single workgroup. Administrators should review their security policy and implement the syslog server in a fashion deemed appropriate. It should be mentioned that GPO’s might be applied to this server if it is a member server, otherwise security templates may be applied if installed in a workgroup configuration. In this exercise, we’ll install Kiwi on a member server.

An outline of the installation and configuration procedures follows, as well as reasons for each step.

1. Initial Server build – Install OS, Service Pack, and software updates
2. Download and install Kiwi Syslog Daemon
3. Basic Kiwi configuration – set up essential service configurations to receive messages from logged servers, and allow for testing
4. Install the Snare Agent for Windows on a logged server
5. Basic Snare Agent configuration – set up essential configuration to direct messages to the Kiwi Syslog Server
6. Test to verify Kiwi Syslog Server is receiving messages from the Snare Agent
7. Harden the Syslog server – stop unnecessary services, configure security settings
8. Verify Kiwi still functions as expected after server OS hardening
9. Configure the Snare Agent on the logged server to capture and send desired events to the Syslog server
10. Advanced configuration of Kiwi Syslog Daemon – configure rules, filters and actions
11. Test to verify functionality and operability of both the Snare Agent and Kiwi Syslog Daemon.

Initial Server Build

As previously mentioned, The Kiwi Syslog Daemon may be installed on Windows XP, Windows 2000 Professional, Windows 2000 Server, or Windows 2003 Server. Presuming the reader has experience installing Windows Server, we will not focus a significant amount of attention to this step. Verify that the latest applicable service pack and all appropriate software updates are installed.

Download and Install the Kiwi Syslog Daemon

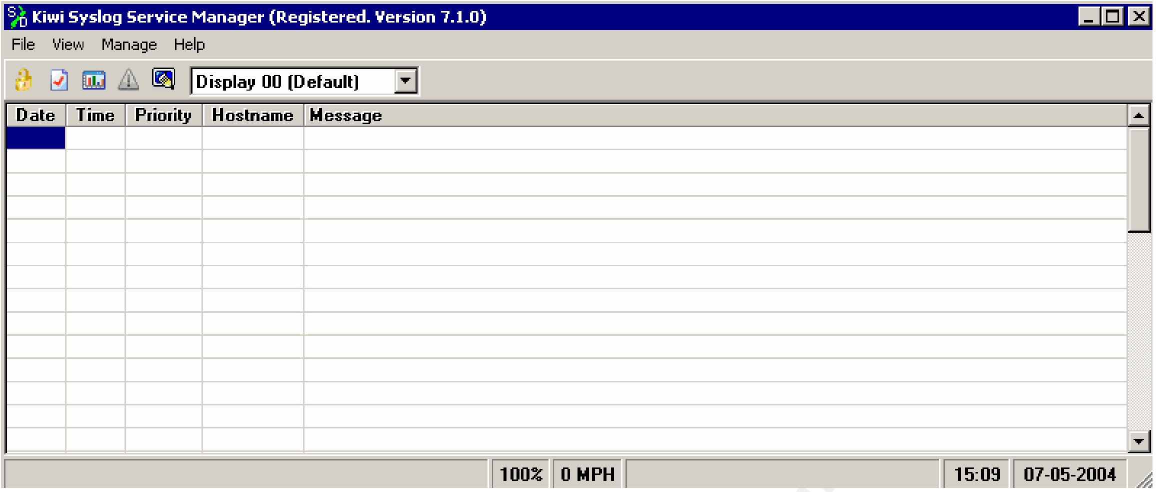
Available at www.kiwisyslog.com/ At the time of this writing, the most recent version is 7.1.0. Please note there are two types of Kiwi Syslog Daemon. One is the Standard Version, the other is the Service Edition Version.

> The Service Edition Version will only run on Windows NT (Server or Workstation), Windows 2000 (Server or Professional), Windows XP Professional, and Windows 2003 Server; and is made of 2 parts – The Service Manager and the Service itself. The Service Manager is the interactive GUI that controls the Service (start, stop, etc). It handles audible alarms, display properties, and sets properties of the service. The Service receives and processes incoming messages. It starts automatically at system startup without requiring the user to be logged in. The Service Manager controls the Service.
>
> The Standard Version will run on Windows 95, Windows 98, ME, NT, 2000, 2003, and XP. Additionally, the Standard Version requires the user to be logged in for the Daemon to function.

Download the most recent release of the Service Version. We wish to use the service version, as it does not require the user to be logged in at all time. Also should the server require rebooting, the service will start automatically and messages will be received and logged without interruption. At the time of download, be sure to request a trial key that will enable the full functionality of the Kiwi Syslog Daemon. The install process is straightforward. Agree to the License Agreement; accept the default install configuration and select install.

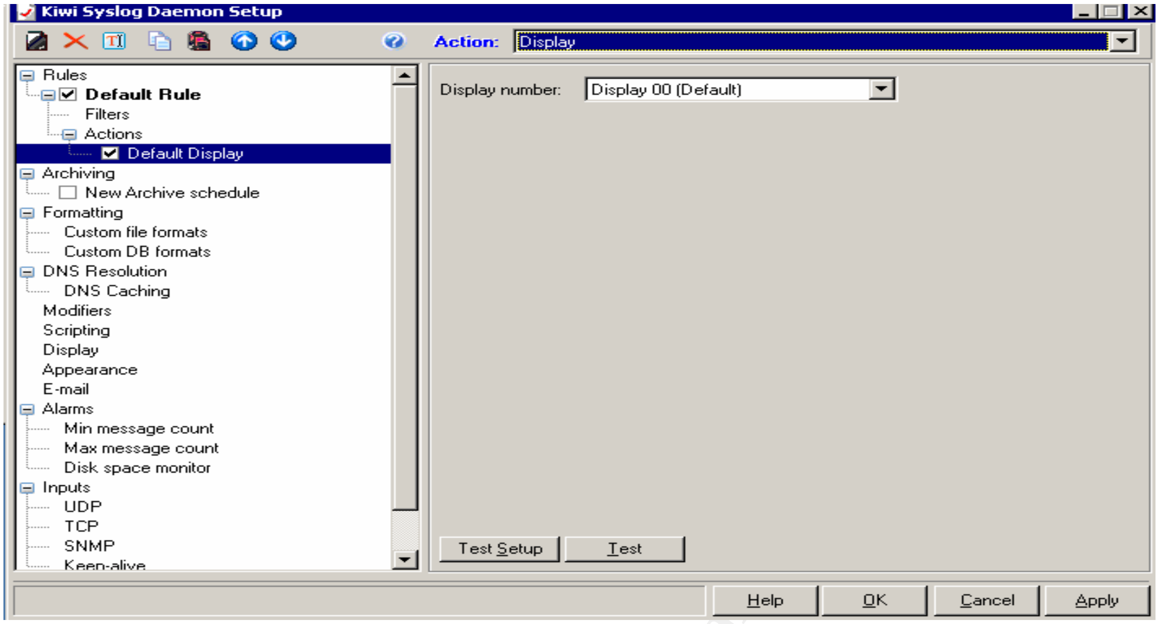
Basic Kiwi Configuration

The intent at this stage of installation is to configure only the essential items necessary to bring the Kiwi Syslog Daemon online and enable basic functionality. Once installed, open the Kiwi Service Manager. The installation process will add a shortcut to the desktop, as well as place an entry in the Start menu. The service manager should appear and the following is displayed.



Select “Manage”, then select “Install Syslogd Service”. Following installation of the service, again select manage, then select “Start Sylogd Service”. The service should now be installed and running. Check services and scroll down to Kiwi Syslog Daemon. Verify the service is installed and running. Startup Type should be set to automatic.

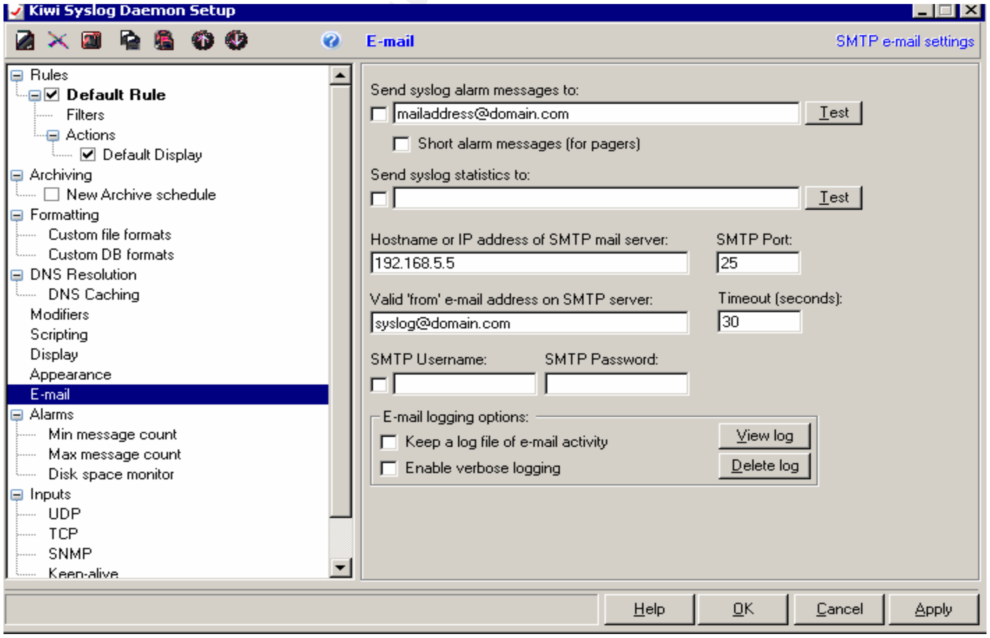
Now that the service is installed and running – a bit of quick configuration is required. From the main Kiwi Syslog Service Manager console, click the “Setup” icon. This icon is the one with the red check within the box. Once the Setup screen appears we need to create a rule so that we can later quickly verify communications with the Snare Agent that is to be installed on another server. Right click “Rules” and select “Add Rule”, and note a new rule appears, appropriately named “New Rule”. This may be renamed to any title of preference, but in this example it will be called “Default Rule”. Next, right click actions and select “Add Action”. This will create an action titled “New Action”, which may be renamed to any title of preference. In this case, we’ll title it “Default Display”. In the upper right drop-down box titled “Action”, select “Display” and accept Display 00 (Default), and select OK. At this point, your setup page should appear as pictured below:



Action: Display

Display number: Display 00 (Default)

To prepare for alerting, we need to configure the appropriate mail server settings. From the Kiwi Syslog Daemon Setup screen, click "Email". There are three fields that will require appropriate information. First, note the field labeled "Send syslog alarm messages to". Enter the pager or email address or pager mailing address to which you wish to have alerts sent. Second, find the field labeled "Hostname or IP address of SMTP mail server". Enter the IP or Hostname of your mail server. Third and finally, is the field labeled "Valid from e-mail address on SMTP server". Enter the desired sender's name for future alerts. The configured E-mail settings appears as follows:



Send syslog alarm messages to:
□ mailaddress@domain.com

□ Short alarm messages (for pagers)

Send syslog statistics to:
□

Hostname or IP address of SMTP mail server: 192.168.5.5

SMTP Port: 25

Valid 'from' e-mail address on SMTP server: syslog@domain.com

Timeout (seconds): 30

SMTP Username: □

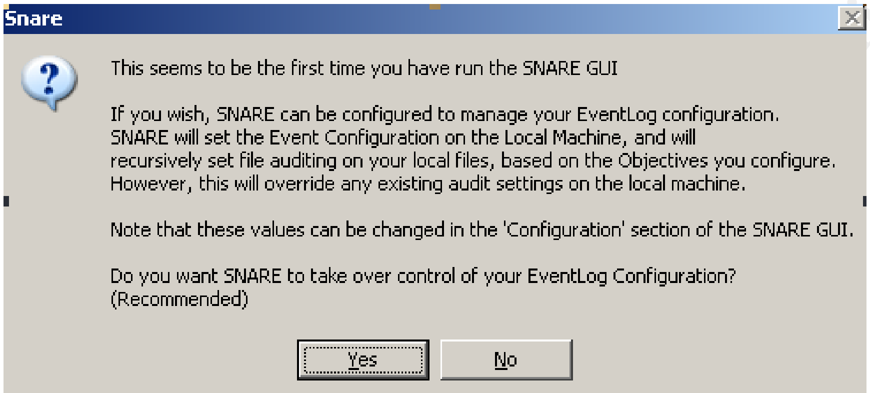
SMTP Password:

E-mail logging options:
□ Keep a log file of e-mail activity
□ Enable verbose logging

Install the Snare Agent

Following the install and basic configuration of the Kiwi Syslog Daemon, we now need to install the Snare Agent for Windows on the logged server. Download the most recent version of Snare Agent for Windows at http://www.intersectalliance.com/projects/

Once downloaded the install process is straightforward. You will be prompted for an install directory, and to place a shortcut in the start menu, followed by a prompt (as appears below) asking if you wish to allow the Snare Agent to manage your event log settings. Select Yes at this prompt.



This seems to be the first time you have run the SNARE GUI

If you wish, SNARE can be configured to manage your EventLog configuration.
SNARE will set the Event Configuration on the Local Machine, and will
recursively set file auditing on your local files, based on the Objectives you configure.
However, this will override any existing audit settings on the local machine.

Note that these values can be changed in the 'Configuration' section of the SNARE GUI.

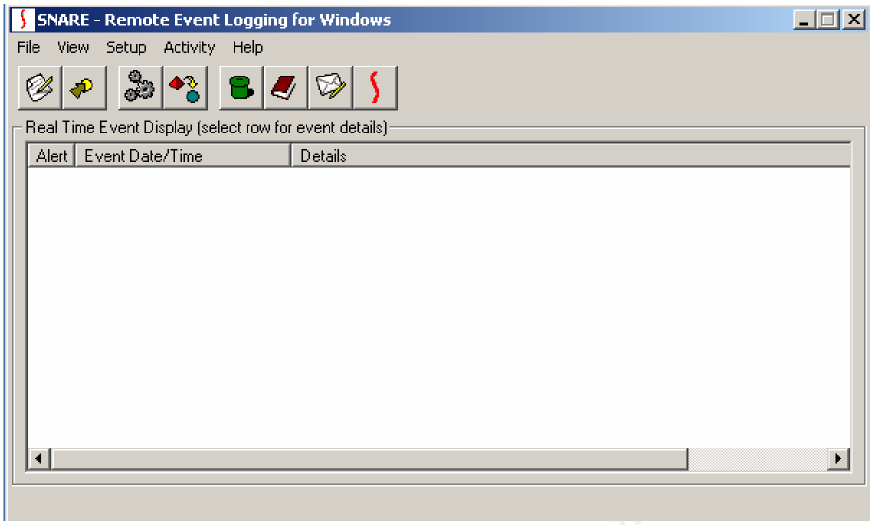
Do you want SNARE to take over control of your EventLog Configuration?
(Recommended)

At this point, the Snare Agent Installation Wizard will finalize the installation, place a shortcut to the Agent configuration on both the desktop and within the start menu and will present the readme.txt file upon completion.

Basic Snare Agent Configuration

Following the installation of the Snare Agent, we will continue with basic configuration. This will allow for testing to insure that communication with the Kiwi Syslog server is established, and that events are being directed to the Kiwi Syslog server. In order for events to be passed to the Kiwi Syslog server, the Snare Core service must be installed and running. This service is installed during the Snare Agent installation. Verify the installation by looking for Snare within services. Confirm the service is running and set to automatic so that the service will start automatically when the hosting server is rebooted.

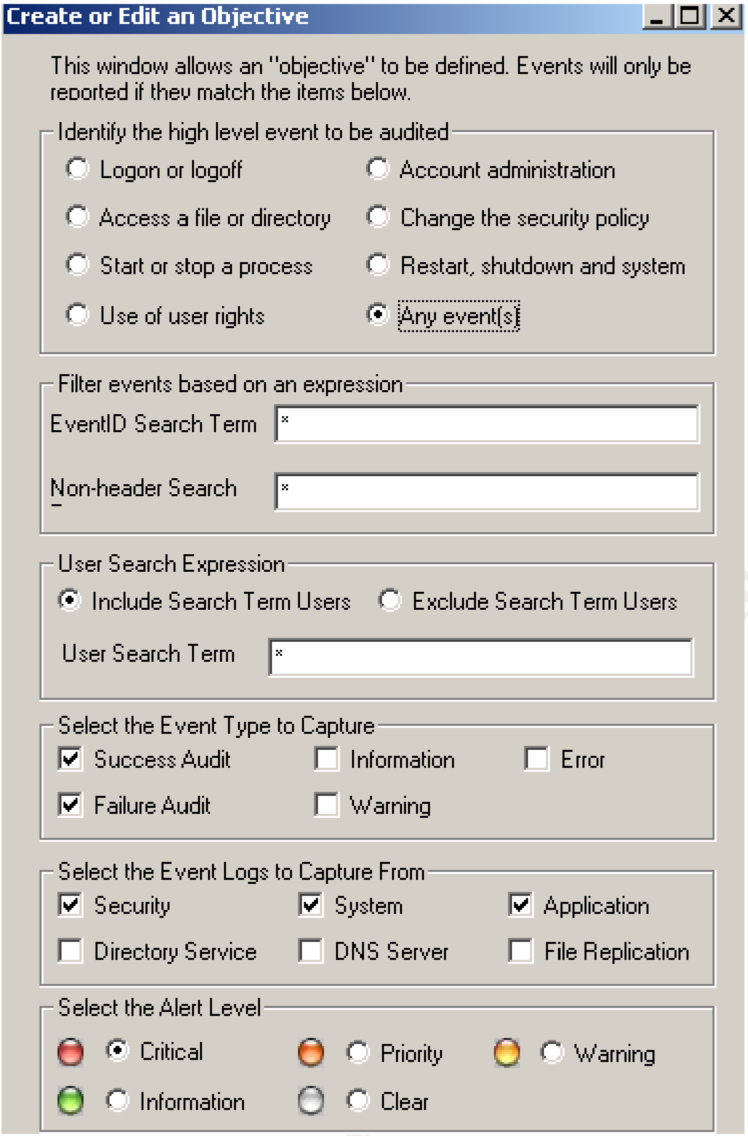
Using the Intersect Alliance shortcut on the desktop or within the start menu, open the Snare Agent Console. The Snare Agent Console appears as follows:


SNARE - Remote Event Logging for Windows
File View Setup Activity Help
Real Time Event Display (select row for event details)
Alert
Event Date/Time
Details

From the Snare Agent Console we will configure the initial audit configuration parameters by selecting “Setup”, then from the dropdown menu – select “Audit Configuration”. From the Audit Configuration Console, the initial parameters to configure are:

1. Local Host Name – enter the hostname of this server, which is hosting the Snare Agent.
2. Enter the remote IP or DNS address – enter the IP address of the server hosting the Kiwi Syslog Daemon.
3. Verify each of the following are selected, and has a check in the box preceding each entry:
    a. Enable SYLOG header – this will allow the event log records to be written in an acceptable format for the Kiwi Syslog Server
    b. Automatically set audit configuration – enabling this setting will alleviate the possibility of the Windows event log filling up. Should the event log fill up, no further events may be read – and auditing function effectively stops. By selecting this option, the Snare Agent will configure the event logs to overwrite as required, thus preventing the event log subsystem from stopping.
    c. Automatically set file system configuration – by enabling this setting, the Snare Agent configures Windows to collect file accesses as well as configure file system parameters for proper auditing functionality.

The next step in the Snare Agent configuration is to actually capture events, and direct them to the Kiwi Syslog server. From the Audit Configuration Console, click “Add An Objective”, and the Create or Edit an Objective Console appears as seen below.



Create or Edit an Objective

This window allows an "objective" to be defined. Events will only be reported if they match the items below.

Identify the high level event to be audited

○ Logon or logoff ○ Account administration
○ Access a file or directory ○ Change the security policy
○ Start or stop a process ○ Restart, shutdown and system
○ Use of user rights ● Any event(s)

Filter events based on an expression

EventID Search Term: *

Non-header Search: *

User Search Expression

● Include Search Term Users ○ Exclude Search Term Users

User Search Term: *

Select the Event Type to Capture

☑ Success Audit □ Information □ Error
☑ Failure Audit □ Warning

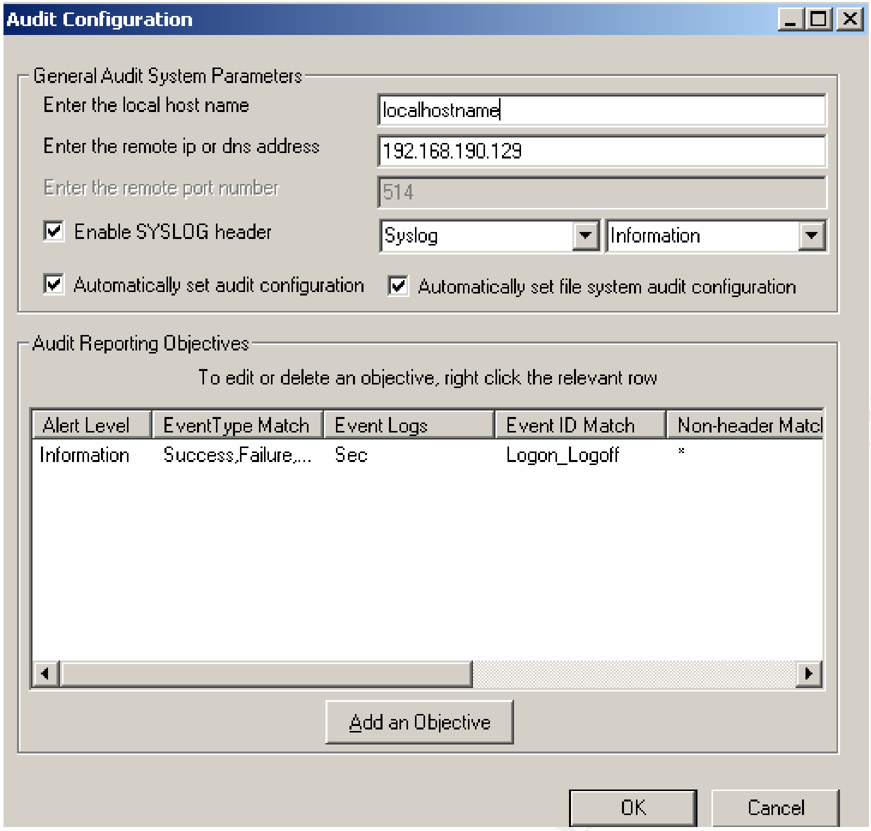
Select the Event Logs to Capture From

☑ Security ☑ System ☑ Application
□ Directory Service □ DNS Server □ File Replication

Select the Alert Level

● Critical ○ Priority ○ Warning
○ Information ○ Clear

It is from this screen that most of the remaining configuration of the Snare Agent will take place. At this point of setup, we want to verify configuration of both the Snare Agent and the Kiwi Syslog server is correct, and that the Snare Agent is capturing events and sending these events to the Kiwi Syslog server. With this in mind, select the "Logon or Logoff" radio button, found under the "Identify High Level Event To Be Audited" category at the top of page. Under the "Select the Event Type to Capture" heading, select all options – Success, Failure, Information, Warning, and Error. Finally, under the "Select the Alert Level" heading – select Information. Click "OK" and you will be returned to the Audit Configuration Console. Your configured Audit Configuration Console should appear as follows:



Audit Configuration

General Audit System Parameters

Enter the local host name: localhostname

Enter the remote ip or dns address: 192.168.190.129

Enter the remote port number: 514

☑ Enable SYSLOG header: Syslog, Information

☑ Automatically set audit configuration ☑ Automatically set file system audit configuration

Audit Reporting Objectives

To edit or delete an objective, right click the relevant row

Alert Level	EventType Match	Event Logs	Event ID Match	Non-header Matcl
Information	Success,Failure,...	Sec	Logon_Logoff	*

Select OK, and exit the Snare Agent configuration.

Test to Verify Kiwi Syslog Server is receiving messages from the Snare Agent

At this point, verify the Kiwi Syslog Service Manager is displayed, and Display00(Default) is the selected. Log off the server hosting the Snare Agent and then log back on. The Kiwi Service Manager should now display logon events. If these events are not being displayed, review your configuration settings. Also determine if a firewall is placed between the Kiwi Syslog server and the server hosting the Snare Agent. Should a firewall be present, UDP port 514 must be open inbound, that is the server hosting the Snare agent will attempt to transmit messages to the Kiwi Syslog Daemon via UPD port 514.

Harden the Syslog server

Presuming the initial configuration of both the Kiwi Syslog Daemon and the Snare Agent resulted in successful communication of events to the Kiwi Syslog server, we now need to redirect our attention to the server hosting the Kiwi Syslog Daemon. As mentioned previously, the Kiwi Syslog server will be storing critical information captured from other servers' event logs, and this data must be protected. It is important the Kiwi Syslog server be a dedicated server, and not fill other duties such as print server, file server, etc. Security Policies of your organization should be reviewed, and insure the Kiwi Syslog server is in accordance with these policies. Presuming this is a clean operating system install, all appropriate service packs and software updates are applied, it is strongly recommended to follow SANS best practices in hardening this server. Items to consider include, but are not limited to:

1. Physical security – insure this server is in a secure location, and is inaccessible to unauthorized personnel
2. Insure Antivirus Software is installed, properly configured, and kept up to date.
3. Disable unused services
4. Secure the registry
5. Restrict Anonymous connections
6. Apply Group Policy Objects in accordance with your Security Policy to further strengthen this server.

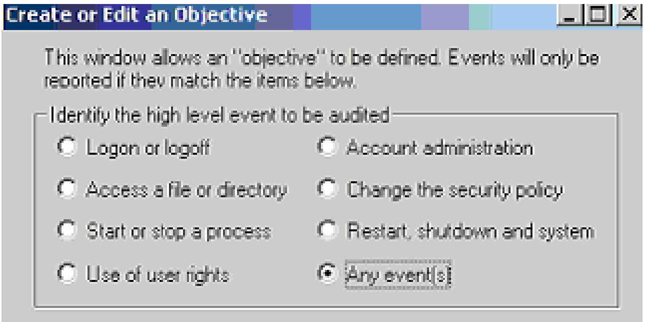
The details of hardening a Windows 2000 Server are extensive and are out of the scope of this document. However, SANS offers an excellent resource to meet this need in *"Securing Windows 2000 – Step By Step"* which is available at https://store.sans.org. Microsoft, with their Trustworthy Computing Initiative also has large amounts of information available online concerning security of Microsoft products. An excellent starting point may be found at www.microsoft.com/security. From this point, there are volumes of information to be found. An excellent resource is *"Hardening Systems and Servers: Checklists and Guides"* that may be found at the time of this writing at http://www.microsoft.com/technet/security/topics/hardsys/default.mspx

Verify Kiwi still functions as expected after server OS hardening

Presuming you have now taken appropriate actions in securing and hardening the Kiwi Syslog server, we need to again verify the Snare Agent is communicating properly with the Kiwi Syslog server. Once again, verify the Kiwi Syslog Service Manager is displayed, and Display00(Default) is the selected. Log off the server hosting the Snare Agent and then log back on, and verify events are still displayed by the Kiwi Syslog server. Should these events not be displayed, first verify the Kiwi service is still running. Then carefully review the server hardening processes followed above. Identify the details of each change made, undo the change and repeat the test until the problem has been identified.

Configure the Snare Agent on the logged server to capture and send desired events to the Syslog server

At this point, a further analysis of the Snare Agent is necessary. From this point forward – final configuration of both the Snare Agent and the Kiwi syslog server will be subjective. In order to configure both the Kiwi Syslog server and the Snare agent to meet your individual needs, an in-depth understanding of each is required. One of the primary functions of the Snare Agent is to filter events. This filtration is controlled through the use of "objectives". Remember, we have already created an objective for testing purposes, one that records logons and logoffs. It is this and related processes that will be examined in greater depth. Objectives define the event(s) to be captured and provide a general level of control over these events. Recall in our testing phase, we chose to capture Logon or Logoff from the “Identify the high level event to be audited” heading (see below). It is through the selection of a specific “high-level” event and through the use of additional filters that captured events may be further refined and sorted. It should be noted that while only Windows Security Event Log events are contained within the “high-level” groups, other event types could be captured as well through the use of the “any-event” option.



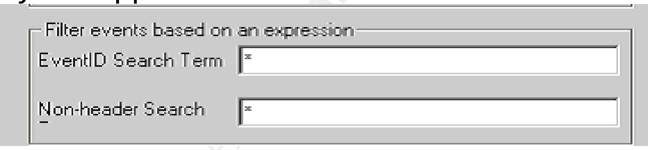
Create or Edit an Objective

This window allows an "objective" to be defined. Events will only be reported if they match the items below.

Identify the high level event to be audited

○ Logon or logoff ○ Account administration
○ Access a file or directory ○ Change the security policy
○ Start or stop a process ○ Restart, shutdown and system
○ Use of user rights ● Any event(s)

The field labeled “Filter events based on an expression” is for filtration of events, based upon the high-level option chosen for audit within the “Create or Edit and Objective” category. The “EventID Search Term” allows for a specific event ID or range of event ID’s to be audited and reported to the Kiwi Syslog server. Options for inclusion in this field are the wildcard “*” – capture all event ID’s; a unique event ID; or a number of ID’s may be entered as a comma separated string – such as 562,457,897 would capture any events matching these specific values. The field labeled “Non-header Search” allows for a search to be refined based upon the event record contents. This will examine all fields of event records, except the header. Options for inclusion in this are specific data or the wildcard “*” which will capture all matching events. If specific data is entered, there is no need to either prepend or append the data with a wildcard “*” as this is presumed by the application.



Filter events based on an expression

EventID Search Term: *

Non-header Search: *

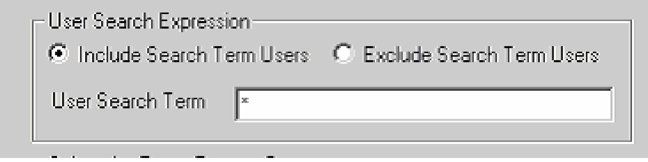
Microsoft offers several resources that may prove useful in the configuration of the Snare Agent to capture specific events. The following links provide reference to Event ID’s and their corresponding description.
Event ID’s Part 1 of 2
http://support.microsoft.com/?kbid=299475

Windows Event ID’s Part 2 of 2
http://support.microsoft.com/?kbid=301677

Events and Error messages center
http://www.microsoft.com/technet/support/eventserrors.mspx

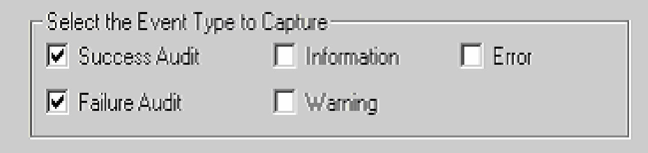
Windows 2003 Security Events
http://www.microsoft.com/technet/security/guidance/secmod128.mspx

Following the “Filter events based on an expression”, note the additional option to further filter auditable events through the use of the “User Search Expression” field. This option allows for an auditable event to be selected or discarded, based upon a user id or partial match of user id. Options for inclusion within this field are specific user id’s; partial user id’s, using the wildcard “*”; and the wildcard “*”. To audit all users, select “Include Search Term Users” and insert the “*” as the variable within the “User Search Term” field.



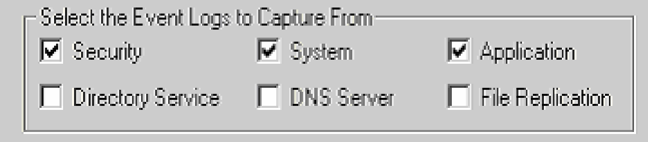
User Search Expression
● Include Search Term Users ○ Exclude Search Term Users
User Search Term: *

The Snare Agent offers further filtration capability through the use of the “Select the Event Type to Capture” option. This allows for the selection of a single event type, or combination of event types. Options are Success Audit, Failure Audit, Information, Warning, and Error. If no options are selected, then no events will be logged. All options or a combination of options may be selected to meet the administrators’ audit needs.



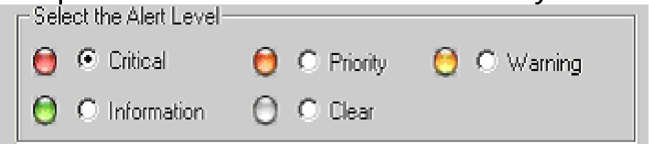
Select the Event Type to Capture
☑ Success Audit ☐ Information ☐ Error
☑ Failure Audit ☐ Warning

“Select event logs” allows the administrator to further refine their audit needs by specifying either specific event logs, or a combination of event logs to monitor. The Snare Agent is capable of capturing events from the Security Event Log, Directory Service Event Log, System Event Log, DNS Server Event Log, Application Event Log, and File Replication Event Log.



Select the Event Logs to Capture From
☑ Security ☑ System ☑ Application
☐ Directory Service ☐ DNS Server ☐ File Replication

The final option for configuration within the Snare Agent is “Select Alert Level”. This allows the administrator to assign audited events to levels of criticality. Options are: Critical, Information, Priority, Clear, and Warning. This also allows for quick identification of event severity.



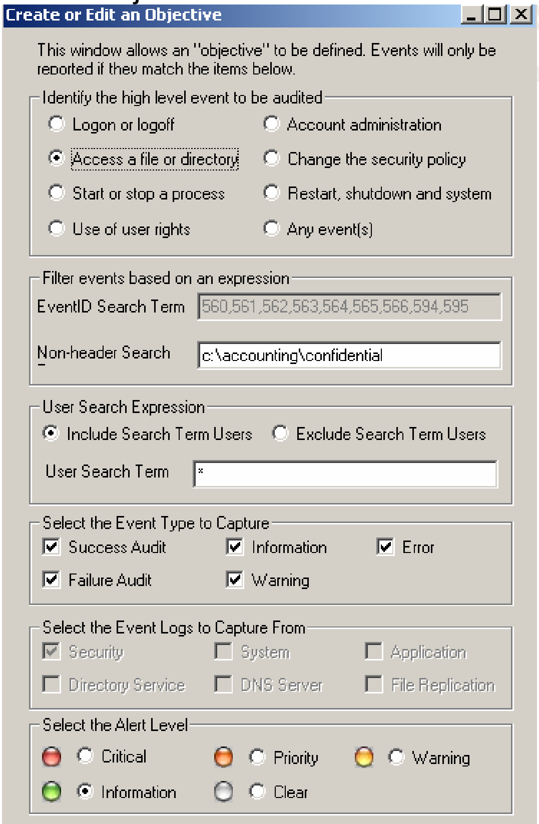
Select the Alert Level

● Critical ○ Priority ○ Warning

○ Information ○ Clear

Following is an example configuration. Presuming we have a folder “c:\accounting\confidential” – and we wish to audit and capture events pertaining to this audit.

1. We begin by identifying the high level event to be audited, in this case – “Access a file or directory”.
2. We then identify the folder we wish to audit, and capture related events. We enter “c:\\accounting\confidential”.
3. We wish to capture events pertaining to all users, so the wildcard “*” is entered in the “User Search Term” field.
4. To capture all event types, we choose all selections within the “Select Event Type to Capture” heading.
5. Finally, we choose to assign an alert level of “Information” to this particular objective.



Create or Edit an Objective

This window allows an "objective" to be defined. Events will only be reported if they match the items below.

Identify the high level event to be audited

○ Logon or logoff ○ Account administration

● Access a file or directory ○ Change the security policy

○ Start or stop a process ○ Restart, shutdown and system

○ Use of user rights ○ Any event(s)

Filter events based on an expression

EventID Search Term: 560,561,562,563,564,565,566,594,595

Non-header Search: c:\accounting\confidential

User Search Expression

● Include Search Term Users ○ Exclude Search Term Users

User Search Term: *

Select the Event Type to Capture

☑ Success Audit ☑ Information ☑ Error

☑ Failure Audit ☑ Warning

Select the Event Logs to Capture From

☑ Security ☐ System ☐ Application

☐ Directory Service ☐ DNS Server ☐ File Replication

Select the Alert Level

○ Critical ○ Priority ○ Warning

● Information ○ Clear

Advanced configuration of Kiwi Syslog Daemon

Now that we have a good understanding of the Snare Agent and the process required to capture and filter events, we need to return our attention to the Kiwi Syslog Server. Kiwi Syslog Daemon is configured through the Kiwi Syslog Service Manager – as previously explained. In a fashion similar to the Snare Agent, the Kiwi Syslog Manager is used to filter and manipulate incoming data. Our objective at this point is to capture the events issued by the Snare Agent, and create filters to act upon this information.

It is important to understand that in a production environment, numerous servers may each be hosting a Snare Agent. Each Snare Agent in turn may be configured with numerous objectives. All data meeting these objectives will then be directed to the Kiwi Syslog Server. The Kiwi Syslog Server must be configured to receive, arrange, and act upon this data in an orderly fashion. This is conducted through the use of rules, filters, and actions. Recall the GUI manager has 10 independently configurable display screens, and rules must be created to either send specific messages to a particular display screen, to a specific log file, or otherwise acted upon. Incoming messages are processed through the use these rules, and filters that are assigned to these rules. Filters specify which message values and attributes are to be analyzed. Options for filtering incoming data are:

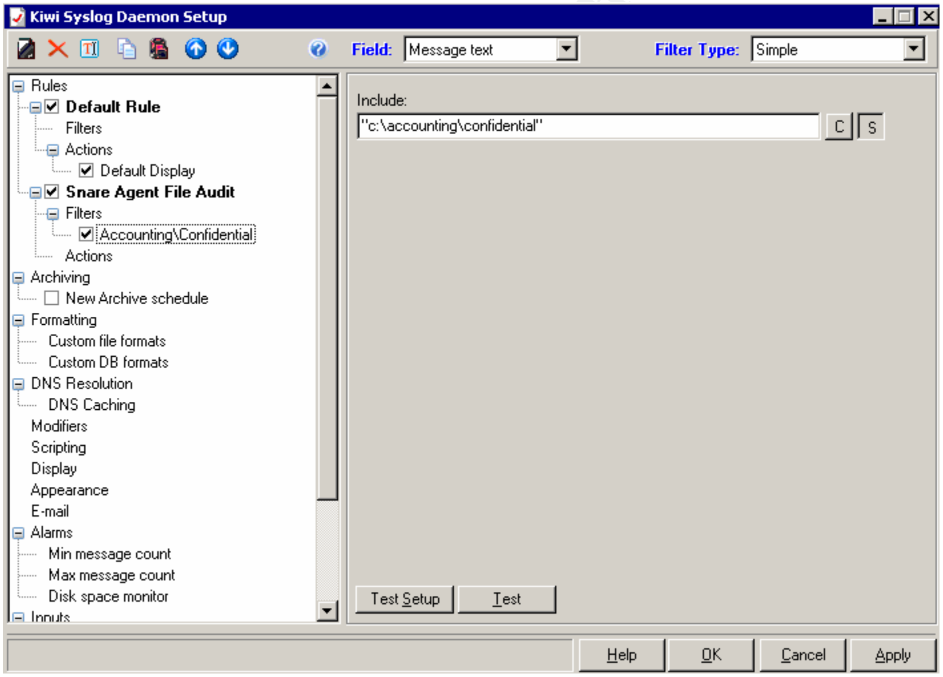
1. None
2. Priority
3. IP address
4. Hostname
5. Message Text
6. Time of Day
7. Flags / Counters

Actions may be created to serve as a response when a specific condition is met. Available actions are:

1. None
2. Display
3. Log to file
4. Forward to another host
5. Play a sound
6. Run external program
7. E-mail message
8. Send Syslog message
9. Log to ODBC database
10. Log to NT event log
11. Send SNMP trap
12. Stop processing message
13. Send ICQ instant message
14. Run script
15. Sent message via NotePager Pro

Up to 100 rules may be defined within a Kiwi syslog server. For each rule, up to 100 filters and 100 actions may be applied.

Having reviewed the essentials of rules, actions, and filters – we now can proceed to put these to practice. After launching the Kiwi Syslog Service Manager, we select the "Setup" icon and enter the Kiwi Syslog Daemon Setup Window. To create a new rule, right click "Rules" within the leftmost pane and select "Add Rule". You will then note a new rule; appropriately titled "New Rule" is present. The new rule may be renamed, by right clicking and selecting "rename". For this example, we will name the rule "Snare Agent File Audits". It is now time to create a filter to capture only this traffic from the stream of event log information that may be flowing to the Kiwi Syslog server. Add a filter by right-clicking "Filters" which appears beneath our "Snare Agent File Audits" rule. A new filter appears, titled "New Filter". This may be renamed by right clicking and selecting "rename". In this example, we will rename the filter "Accounting\Confidential". We now have created a rule and a corresponding filter. Now, we must define the filter. In the "Field" drop-down menu, select "Message Text". We now define the text the filter is to capture, by entering "c:\accounting\confidential". As this is a variable, the string must be enclosed in quotes. At this point your Kiwi Syslog Daemon screen should appear as follows:



Kiwi Syslog Daemon Setup

Field: Message text

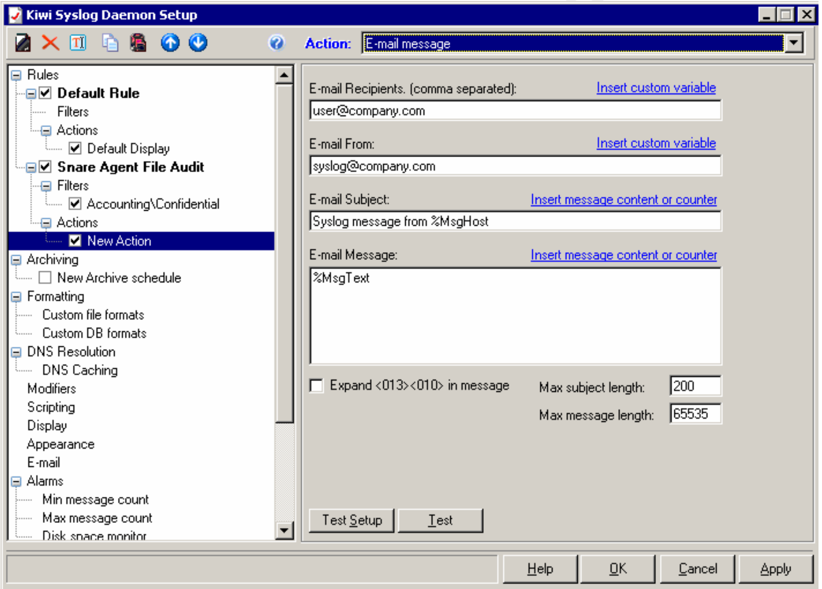
Filter Type: Simple

Include: "c:\accounting\confidential"

We have now succeeded in creating a rule, and we have created and defined a filter to capture specific information necessary for this rule to function. The next step we must take is to create an action(s) we wish to take place when desired event log information is captured. The most commonly used actions are display, log to file, and e-mail message. We will now examine the steps necessary to create each action. During the preliminary testing, we created the default rule and logged all information received to the default display. Following the same process, we create an action to log corresponding information to Display01.

Next, we wish to log to file all events received concerning the audit of c:\accounting\confidential. This is in part a repeat of previous steps. Create a new action, and select “Log to File”. You will be prompted with a recommended path and filename, which may be altered to fit your unique requirements. Rename the action “Log to file”, and click “Apply”.

The final type of action we will review is “E-mail message”. It is this action that will be used to issue administrative alerts in response to critical events captured by the Snare Agent, which are in turn forwarded to the Kiwi Syslog server where all captured events are then sorted by rules and filters. To maximize effectiveness and reduce the total number of administrative alerts issued, it is recommended that such alerts be issued in response only to the most critical of events. It may be unlikely in a production environment that we would wish to issue administrative alerts for audits of a specific folder, therefore the following is provided only for informative purposes only. Following the previously defined process of creating an action – we create a new action under the “Snare Agent File Audit” rule. From the “Action” dropdown selection, choose “E-mail Message” – and the following display appears:



Action: E-mail message

E-mail Recipients. (comma separated): user@company.com

E-mail From: syslog@company.com

E-mail Subject: Syslog message from %MsgHost

E-mail Message: %MsgText

□ Expand <013><010> in message

Max subject length: 200

Max message length: 65535

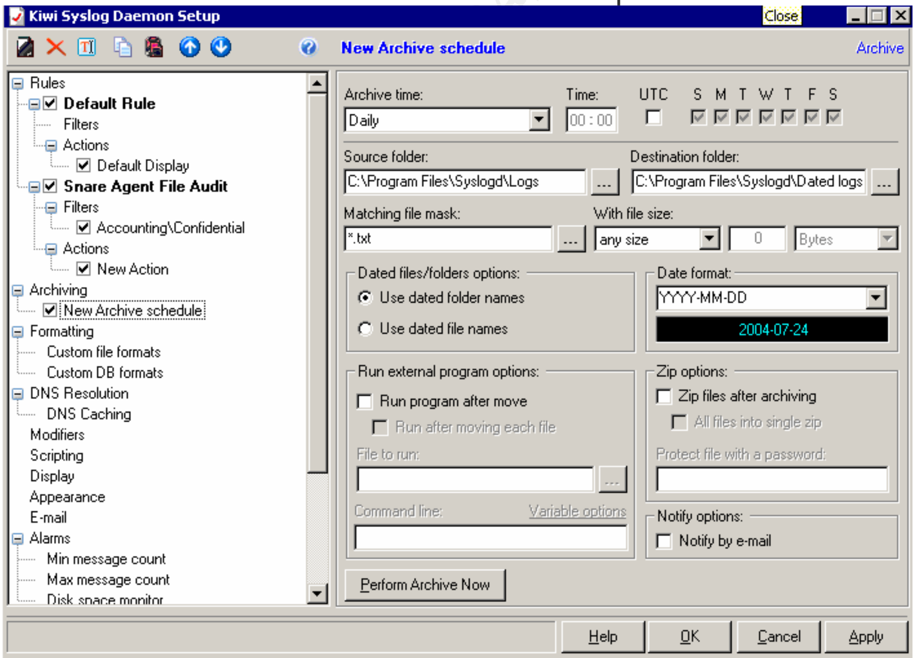
Within the field labeled “E-mail Recipients”, insert the email address or appropriate information for the administrator’s cell phone or pager. Within the “E-mail From” field, enter preferred information to serve as the email sender’s address. All other fields may be left as their default value. Select “OK” and all configuration settings are saved and the Kiwi Syslog Daemon Service Manager is presented.

At this point, it is necessary to test overall functionality of both the Kiwi Syslog Daemon and the Snare Agent. As all processes have previously been tested – it is the alerting feature that must be verified. Should the designated administrator not receive the alert – troubleshooting techniques will need to be followed to determine the cause. Items to verify include:

1. Using a mail client, can the server hosting the Kiwi Syslog Daemon issue an email to the alert address
2. Is the phone/pager capable of receiving such alerts
3. Verify the Kiwi Syslog configuration, within the "E-mail" header selection.

The final topic for consideration is file management. As the Snare Agent/Kiwi Syslog Daemon combination is configured to capture, and log significant events – these items should be archived for a period of time. The Kiwi Syslog Daemon includes highly configurable archiving capabilities. During the creation of rules and actions within the Kiwi Syslog Daemon, one of the possible actions is "Log to File". Kiwi offers the administrator the option of selecting specific items to log. Should the administrator wish to select only specific rules for log retention, a new action of "Log to File" would only need to be created for these specific rules. Alternatively, should the administrator wish to log all recorded events as a single file, an action may be created under the previously created "Default Rule". After creating the "Log to File" action, archiving may be enabled.

Archiving is not enabled by default within the Kiwi Syslog Daemon. To enable archiving, select the "Archiving" option and right-click to create a new schedule. The New Archive Schedule window is then presented.



Archive time: Daily
Time: 00 : 00
UTC □
S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S ☑

Source folder: C:\Program Files\Syslogd\Logs
Destination folder: C:\Program Files\Syslogd\Dated logs

Matching file mask: *.txt
With file size: any size 0 Bytes

Dated files/folders options:
● Use dated folder names
○ Use dated file names

Date format: YYYY-MM-DD
2004-07-24

Run external program options:
□ Run program after move
□ Run after moving each file
File to run:
Command line: Variable options

Zip options:
□ Zip files after archiving
□ All files into single zip
Protect file with a password:

Notify options:
□ Notify by e-mail

The Kiwi Syslog Daemon is very versatile in archive scheduling, up to 100 archive schedules may be created. The "Archive Time" drop-down field allows the administrator to select from Hourly, Daily, Weekly, Monthly, or Custom Schedule – should a unique schedule be desired. Additional items for selection are the source and destination folders for archives, time of day to perform the archive process, zip the folder after archive, issue an email notification upon archival completion, and the ability to run an external program after archival. For added security, the Kiwi Syslog Daemon does allow the administrator to select a shared network folder as the destination folder. As the data within these logs provides an audit trail, and someone attempting to gain unauthorized access may attempt to manipulate data to cover their actions - it is recommended that all logs be moved off the logging server.

For better utilization of storage space, and ease of file management, it is advantageous to perform a monthly consolidation of the daily archives. Using the “tar” command for Windows (available at http://gnuwin32.sourceforge.net/packages/tar.htm) the following batch file may be scheduled as a monthly task to concatenate all the daily .zip files into a single monthly file:

Monthly Archive Batch File

```
REM Connect to folder containing the daily .zip kiwi files
net use x: \\servername\sharename
x:
REM Concatenate and Compress all .zip files into a single tar file
c:\path_to_tar_command\tar -cf KiwiEndOfMonth.tar *.zip
REM Move the .tar file to the designated monthly archival folder
move *.tar x:\archives
REM Delete the original daily .zip files
del *.zip
cd archives
REM add todays date to the .tar file for future reference
c:\path-to-datename.bat\datename KiwiEndOfMonth.tar
exit
```

The “datename” command in the above batch file is actually a batch file itself. The syntax of the datename batch file is as follows:

```
@Echo OFF
TITLE DateName
REM DateName.CMD
REM takes a filename as %1 and renames as %1_YYMMDDHHMM
REM
REM ------------------------------------------------------------
IF %1.==. GoTo USAGE
Set CURRDATE=%TEMP%\CURRDATE.TMP
Set CURRTIME=%TEMP%\CURRTIME.TMP

DATE /T > %CURRDATE%
TIME /T > %CURRTIME%
```

```
Set PARSEARG="eol=; tokens=1,2,3,4* delims=/, "
For /F %PARSEARG% %%i in (%CURRDATE%) Do SET
YYYYMMDD=%%l%%k%%j

Set PARSEARG="eol=; tokens=1,2,3* delims=:, "
For /F %PARSEARG% %%i in (%CURRTIME%) Do Set HHMM=%%i%%j%%k

Echo RENAME %1 %1_%YYYYMMDD%%HHMM%
RENAME %1 %1_%YYYYMMDD%%HHMM%
GoTo END

:USAGE
Echo Usage: DateName filename
Echo Renames filename to filename_YYYYMMDDHHMM
GoTo END

:END
REM
TITLE Command Prompt
```

An added security measure for consideration is to enable auditing of the folder(s) containing the daily and archived files. Following the same process as described in the previous example, the Snare Agent may be configure to monitor access attempts to this data, and report to the Kiwi Syslog server.

If conveyed to the administrator in a timely fashion, the value of information found within the Windows Event Logs increases greatly. Timely data may aid in detecting security issues, hardware or software configuration issues, and replication problems – among others – at the time of occurrence. Equipped with such a tool, the administrator is more informed of the events taking place within the environment. We have examined the processes and software necessary to capture these events and provide just such information to the administrator and appropriate staff. Using these tools, we can capitalize on the information which otherwise would reside in distributed, static log files. The Snare Agent for Windows and the Kiwi Syslog Server allows us to centralize the data from these distributed logs, analyze this data, and react as necessary to critical events – all in real time. Additionally, we have reviewed the process for daily log archival and the movement of archived data to another network resource. This combination of tools and processes are important elements of the *Defense-In-Depth* concept – that of layering security measures - wherever and whenever possible, to better safeguard and secure the data and resources of any given organization.

References:

Events and Error messages Center - Microsoft
URL: http://www.microsoft.com/technet/support/eventserrors.mspx

Gibbs, Mark. "Down Under Syslog"
Network World June 24, 2002
URL: http://www.nwfusion.com/columnists/2002/0624gearhead.html

Microsoft "Hardening Systems and Servers: Checklists and Guides"
URL: http://www.microsoft.com/technet/security/topics/hardsys/default.mspx

Microsoft Knowledge Base Article – 299475 "Windows 2000 Security Event Descriptions (Part 1 of 2)"
URL: http://support.microsoft.com/?kbid=299475

Microsoft Knowledge Base Article – 301677 "Windows 2000 Security Event Descriptions (Part 2 of 2)"
URL: http://support.microsoft.com/?kbid=301677

Richardson, Scott. "Centralized Windows 2000 Event Logging, A Step-By-Step Guide".
URL: http://www.sans.org/rr/paper/67/1245.pdf

Sans Press. "Securing Windows 2000 – Step By Step"
URL: https://store.sans.org/store_item.php?item=22

Wilkins, Brian R. "Effective Logging & Use of the Kiwi Syslog Utility"
URL: www.sans.org/rr/papers/33/201.pdf

Windows 2003 Security Events - Microsoft
URL: http://www.microsoft.com/technet/security/guidance/secmod128.mspx

Software References

Dumpel.exe, Microsoft
http://www.microsoft.com/windows2000/techinfo/reskit/tools/existing/dumpel-o.asp

DumpEvt, Somarsoft
http://www.somarsoft.com/somarsoft_main.htm

Kiwi Syslog Daemon
http://www.kiwisyslog.com/

Snare Agent for Windows
http://www.intersectalliance.com/projects/SnareWindows/index.html#Download

Tar for Windows, Sourceforge.net
http://gnuwin32.sourceforge.net/packages/tar.htm