Global Information Assurance Certification Paper

Centralized Collection of Logs in Windows2000/Linux Mixed Environment

Hikaru Nishino
2004-02-09

GIAC Security Essentials Certification Practical Assignment
Version 1.4b Option1

Abstract

Computer system logs are very important source of information for security administrators. Yet in many cases, administrators must handle various operating systems and deal with logs in different formats. For example, Linux (UNIX) uses syslog, and Windows uses event log which are incompatible in format. To effectively analyze incidents of computer systems, an ability to centralize log collection is needed. In this research paper, I will discuss how to set up log-sender and log-receiver facilities in Windows2000/Linux mixed environment to centralize collection of various logs, and evaluate its conclusion.

1. The Role of Logs

The log is also called “audit trail” and Gopalakrishna[1] says that “an audit trail is a series of records of computer events, about an operating systems, an application, or user activities.”
Likewise, office[2] says that the following are the most important events for audit trail in network systems (Translation of Japanese to English is mine):

-Remote login failures and the time and date of occurrence
-Multiple authentication failures
-Traffic that violates security at gateways
-Attempted access to access logs from remote location
-System administration security alerts such as the existence of duplicate IP addresses.

Additionally, ITsecurity[3] says:

> The audit trail is a record of all events that take place on a system, and across a network. It should provide a trace of user actions so that security events can be related to the actions of a specific individual, and is therefore the basis of the accountability requirement for a secure system.

Thus when incidents need to be investigated, the availability of audit trail with its recorded events allows for effective analysis of these incidents. For this reason, it is crucial that one must thoughtfully choose what information to record in the log for it to be most effective.

2. The Differences in Log Content Between Windows2000 and Linux

The storage method and information content of logs are different among operating systems. In Linux, the process called “syslog” captures events and records into various logs. The configuration file for syslog is /etc/syslog.conf. The general file format of syslog.conf is:

<selector> <action>

Note also that the “selector” element consists of “facility” and “severity level” delimited by a period, and the “action” element represents the destination of logs. Tables 2-1 and 2-2 list the keywords used in “facility” and “severity level” elements respectively[4].

Table 2-1 Facility

auth	security/authorization messages(DEPRECATED Use LOG_AUTHPRIV instead)
authpriv	security/authorization messages (private)
cron	clock daemon (cron and at)
daemon	system daemons without separate facility value
kern	kernel messages
lpr	line printer subsystem
mail	mail subsystem
news	USENET news subsystem
ftp	ftp daemon
ntp	network news subsystem
syslog	messages generated internally by syslogd
user	generic user-level messages
uucp	UUCP subsystem
local0-7	reserved for local use

Table 2-2 Severity Level

emerg	system is unusable
alert	action must be taken immediately
crit	critical conditions
err	error conditions
warning	warning conditions
notice	normal, but significant, condition
info	informational message
debug	debug-level message

Following sample shows the content of an actual syslog.conf file.

```
---------------------------------------------------------------------------------------------
#kern.*                                          /dev/console
*.info;mail.none;news.none;authpriv.none;cron.none
                                                 /var/log/messages
authpriv.*                                       /var/log/secure
mail.*                                           /var/log/maillog
cron.*                                           /var/log/cron
# Everybody gets emergency messages
*.emerg                                          *
---------------------------------------------------------
```

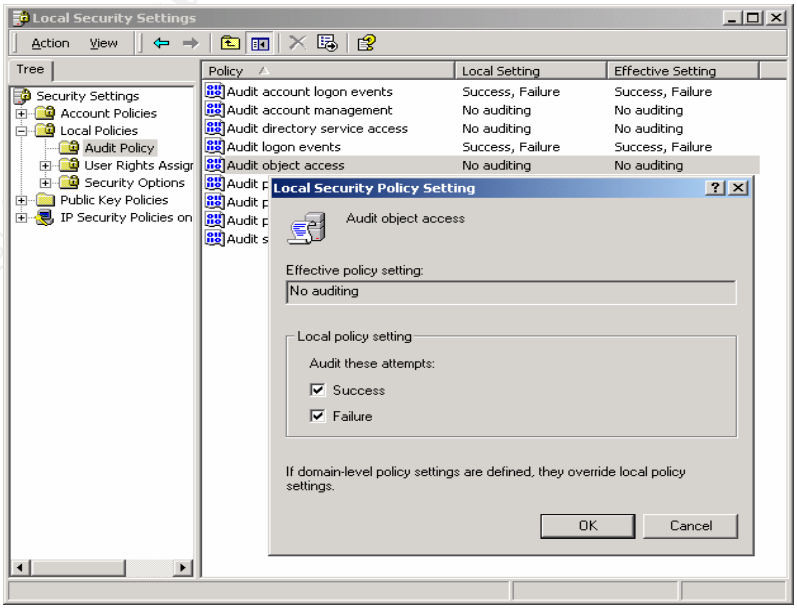
As shown above, the “selector” element consists of the “facility” element left to the period, and the “severity level” to the right of the period. In addition, we can use “*” as a wild-card. In the “action” element, we can specify not only the name of the destination file but also the IP address of the remote computer that will receive the syslog data via UDP port 514. This remote computer is called the “log server”. When we configure a log server, one simply adds a parameter “-r” in the startup scripts for the syslog process.

In Windows2000, logs are stored into a binary file called the “event-log”. Because of this, a proprietary program is needed to display the event-log information. This program is called the “event-viewer”.
Basically, event-log is classified into following three categories:

- Application log
- System log
- Security log.

Information relating to various application behaviors is recorded in the ap plication log, and information relating to OS behaviors is recorded in the system log. Finally, information such as authentication results is recorded in the security log. As the name implies, the security log is very important from computer security stand point. However, out-of-the-box, the security log is not enabled by default. One must configure the audit policy in the Local Security Policy or the Group Policy to enable security logging. Figure 2-1 shows an example of the audit policy setting. In this screen, we can change the effective policy from “No Auditing” to “Success or Failure” auditing.

Figure 2-1 Audit Policy Setting



Another important event-log setting is the “over write” setting. This setting specifies the action to take when the log size reaches its maximum. The options are as follows:

a) Overwrite events as needed
b) Overwrite events older than *n* days
c) Do not overwrite events.

Specifically, a) means automatically overwriting the oldest entries, b) means overwriting the oldest entries older than specified number of days, and c) means that new log entries will not be recorded beyond the maximum size. To avoid interruptions in recording events, one should choose either a) or b). In addition, it is important to backup event-log on a regular schedule.

3. The Merits of Centralizing Log Collection

Centralized collection of logs means that various logs in network computers are sent to one log server. The benefits of doing so are:

a) Protection from log being erased maliciously on a compromised system
b) Easy analysis of log after incidents.

Protection from log being erased

A computer intruder will most likely erase the evidence of activity after compromising the system. Commonly, that evidence includes:

- Files and programs used for intrusion and malicious activity
- History of actions (log).

For a computer intruder, these evidences are inconvenient to leave behind because his/her method of intrusion and record of malicious activity could be analyzed to identity the intruder. Following is a sample log file (/var/log/messages) of a Linux computer after the online password crack attack.

```
----------------------------------------------------------------------------------------------------
Dec 12 02:59:05 serv00 login[18914]: FAILED LOGIN 1 FROM
192.168.0.90 FOR root, Authentication failure
Dec 12 02:59:22 serv00 login[18914]: FAILED LOGIN 2 FROM
192.168.0.90 FOR root, Authentication failure
------------------------------------------------------------
```

In this log, we can see when and from which computer the intruder tried to login, and whether it was successful or not. This log will be used to analyze the intrusion or the intrusion attempt. Because of this evidentially value, it is very important to transfer logs to the remote log server to avoid being destroyed by the intruder.

Easy analysis of log after incidents

By centralizing log collection, it will be easily to analyze events on the network systems in a chronological order. Moreover, because all the various log formats are converted into one unified log format, all the events can be easily consolidated and analyzed together to give a holistic view of the events.

4. Time Synchronization

In order to centralize logs, all hosts on the network need to synchronize their system times. This enables one to analyze events in chronological order on the central log server. The various methods (commands) of time synchronization are as follows:

--Linux--

```
ntpdate [NTP Server]
```

--Windows2000--

```
net time /setsntp: [NTP Server]
```

--Windows XP--

Configured by GUI.

Here the “NTP Server” refers to a server that synchronizes the time using the NTP (Network Time Protocol). The definition of NTP as defined in [5] is:

> The Network Time Protocol (NTP) is used to synchronize the time of a computer client or server to another server or reference time source, such as a radio or satellite receiver or modem. It provides accuracies typically within a millisecond on LANs and up to a few tens of milliseconds on WANs relative to Coordinated Universal Time (UTC) via a Global Positioning Service (GPS) receiver, for example.

It is very important to regularly synchronize the system time of computers so that their logs reflect accurate event times for later analysis.

5. Architecture of Centralized Collection of Logs

There are two possible options on how to centralize logs in Windows2000 and Linux mixed environment. These two options are as follows:

a) Centralize on a Windows2000 host
b) Centralize on a Linux host

Option a)

In this option, a Windows2000 host will be a log server. But the event-log facility in Windows2000 does not have a “log-sender” feature to send log information to a

remote host nor a “log-receiver” feature to receive log information from a remote host like Linux. Because of this, it is difficult to centralize logs to a Windows2000 host as is. However there are some free software programs which provide syslog server features for Windows2000 hosts. With these programs, one can setup a Windows2000 host as a log server (syslog server). Moreover, there are some programs that enable log format conversion and log transfer to the syslog server. By using these programs, we can configure a systems as follows:

- Implement syslog server on a Windows2000 host (using free software)
- Implement syslog transfer on Windows2000 hosts (using free software)
- Configure Linux hosts to transfer syslog data to the Windows2000 syslog server

Option b)

As mentioned above, Linux supports the features of syslog server and syslog transfer by default. Thus, we can configure a system as follows:

- Configure Linux host to be a syslog server
- Implement syslog transfer on Windows2000 hosts (using free software)
- Configure Linux hosts to transfer syslog data to the Linux syslog server.

Comparison of Options a) and b)

Compared to Option a), the Option b) is thought to be more viable. This is because many free syslog-analyzing utilities such as SWATCH are not available on the Windows2000 platform. The main reason for this is because the syslog was originally developed and implemented for UNIX. Thus in this research paper, I will focus on Option b) and examine various tools to transfer logs from Windows2000 host to a Linux syslog server.

6. Log Information to Transfer

In transferring logs from Windows2000 hosts to a Linux syslog server, the information that should be transferred are as follows:

- Application logs
- System logs
- Security logs
- IIS logs
- Packet filtering logs.

First three types of logs are stored as one event-log and hence it is possible to transfer these logs at the same time if we can convert the event-log into a syslog format.

IIS logs are also important for web administrators and FTP administrators. However unlike the other three logs, the IIS logs are stored under the directory of

%SYSTEMROOT%\system32\logfiles\W3SVC1

as text files. Thus it is possible to transfer the IIS logs to the syslog server if we can convert the text file into a syslog format.

Packet filtering logs contain the information of forwarded/dropped packets, and will become a subject of discussion because Windows2000 systems are not able to record that information (in WindowsXP, the built-in personal firewall can record that information). For example, “netsh.exe” and “IPSec filter” have packet filtering features[6], but they cannot record the results as logs. Consequently I investigated free software tools that can produce packet filtering output in Windows2000 systems. In this research, I focused on packet capture tools rather than packet filtering tools. The packet capture tools are able to capture packets in promiscuous mode and output the details of these packets. There are a number of tools with that feature and I surveyed the tools with the additional features of filtering and running in the command line mode. With these features, the tool will be able to boot up automatically (i.e. command line batch), picks up packets that meet the filtering rules, and then output packet details to display (stdout : standard output device). If this result in stdout could be converted into a syslog format and transferred to the syslog server, then we can collect the same type of information as the packet filtering logs. However, note that we cannot collect the information of “actual” forwarded/dropped packets, but collect the information of packets that “should be” forwarded/dropped. I will not go into the discussion of actual packet filtering because it is out of the scope of the present paper. Following is a result of a study on a typical packet capture tool “windump”.

”Windump” is a free software and is a Windows-version of tcpdump utility running on UNIX[7]. Thus the parameters used are the same as tcpdump [8]. I examined this tools to see if it can specify the same filtering rules as netsh.
The result of examination is below:

”Windump” has the following parameter options for filtering:

- Destination IP Address ▪ or network address ▪
- Destination port#
- Source IP address ▪ or network address ▪
- Source port#
- TCP flag
- Protocol.

I examined “windump” using these options under a test condition to see whether it can be used to produce the desired packet filtering log. Following is the result of this test.

- Condition -

Access to an internal web server form the WAN

Source IP address:any , Source port#:any, Destination IP address : 192.168.0.100(test01) , destination port : 80

- Execution -

```
windump tcp and dst host 192.168.0.100 and dst port 80
```

Here “tcp” means tcp protocol, “dst host” means target host and “dst port” means target port number.

- Result -

```
15:23:40.604450 192.168.0.7.1038 > test01.80: P 1398463450:
1398463776(326)ack 1651096554 win 10720 <nop,nop,timestamp 1255639
42398> (DF)
15:23:40.613873 192.168.0.7.1038 > test01.80: P 0:326(326) ack 1 win
10720 <nop,nop,timestamp 1255639 42398> (DF)
```

The output format of this result is as follows[8]:

(Time)(source address.port) > (destination address.port)(TCPflag)(sequence#) (ack)(ack#)(window-size)<options>(Don’t Fragment).

Looking at the result, it is clear that “windump” can be used to produce the desired packet filtering log. What remains is to convert the “windump” file to a syslog format, and transfer it to the syslog server.

7. Investigation of syslog Conversion Tools

In order to transfer the log data from Windows2000 host to a Linux syslog server, one must first convert the Windows2000 logs into a format compatible with syslog. As was mentioned previously, Windows2000 does not have this functionality out of the box. Because of this, I looked into using freely available software to accomplish the format conversion. The results of this investigation are as follows:

Tools to convert and send event-log data to syslog

Following is a list of software commonly used to convert Windows2000 event-log data

into syslog format:

- EventReporter [9]
- Eventlog to Syslog Utility[10]
- BackLog [11]
- NTsyslog [12].

In discussing how these tools work, I have selected NTsyslog for the purpose of this research paper. I will describe a brief overview of setting up this tool below.

First, NTsyslog takes the event-log (Application, System, and Security log) produced by Windows2000 and converts it into syslog format.

Next, NTsyslog transmits the converted file to a syslog server. The NTsyslog consists of the following programs:

a) NTsyslog.exe

b) NTsyslogCtrl.exe.

The (a) NTsyslog.exe is used to convert event-log to syslog format and send to syslog server. It is a service program and installed from a command line as shown below:

```
ntsyslog -install
```

The (b) NTsyslogCtrl.exe is a GUI program used to identify the target syslog server and to set the “facility” and “severity level” information sent to the server. Figure 7-1 depicts the window where the type of event-log is selected, and Figure 7-2 depicts the window where the “facility” and the “severity level” are selected for transmission.

Figure 7-1 Event-log Configuration

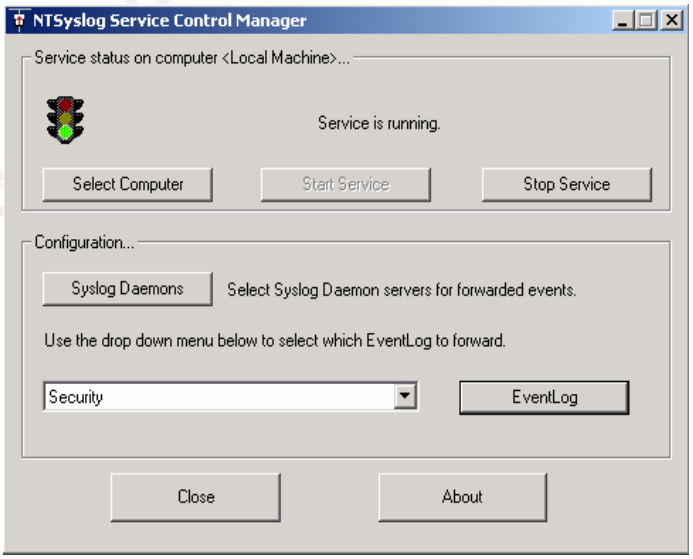
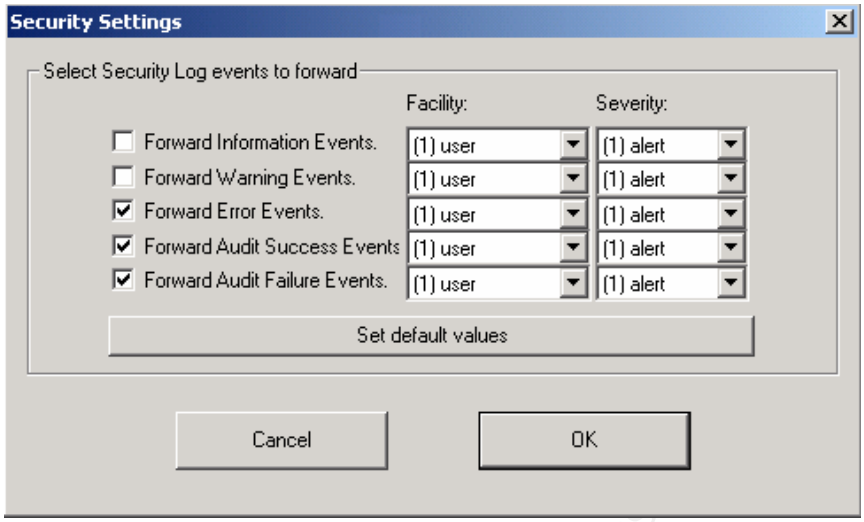


Figure 7-2 Facility and Severity Level Settings



Tools to convert and send IIS log data to syslog

After researching for syslog conversion tools for the IIS log, the following tool was found to exist:

- Monitorware Agent [13].

This is a GUI-based tool where one selects the desired text files (i.e. IIS log files), and the tool converts these files into syslog format, and sends them to the syslog server. In this feature, it is important to send only the new entries of the text file, not the entire file. This means that this tool needs to scan the text files on a regular schedule and pick up the new entries. Regarding this point, this tool has adequate feature called "Check Interval". This feature provides a regular scan, which interval is specified on the setting screen, and selection of new entries to be converted. I will describe a brief overview of setting up this tool below.

In order to convert the IIS log files to syslog format, use the File Monitor Service Window (Figure 7-3) to select the IIS log files to convert by specifying its file name and its path. Then, using the same window, fix the "Check Interval" to make a regular scan, and select "W3C WebServer Log File" for the "Logfile Type", and specify "Syslog Facility" and "Syslog Priority" in the "General Values" panel. After that, use the Agent Configuration Client Window (Figure 7-4) to specify the syslog server. This completes the configuration, and the values are saved as the default setting. To start the conversion and transmission, one simply selects this configuration as the "Rule Sets/Running Services" and restart the service from [File] menu.

Figure7-3 IIS Log Monitoring Service Settings

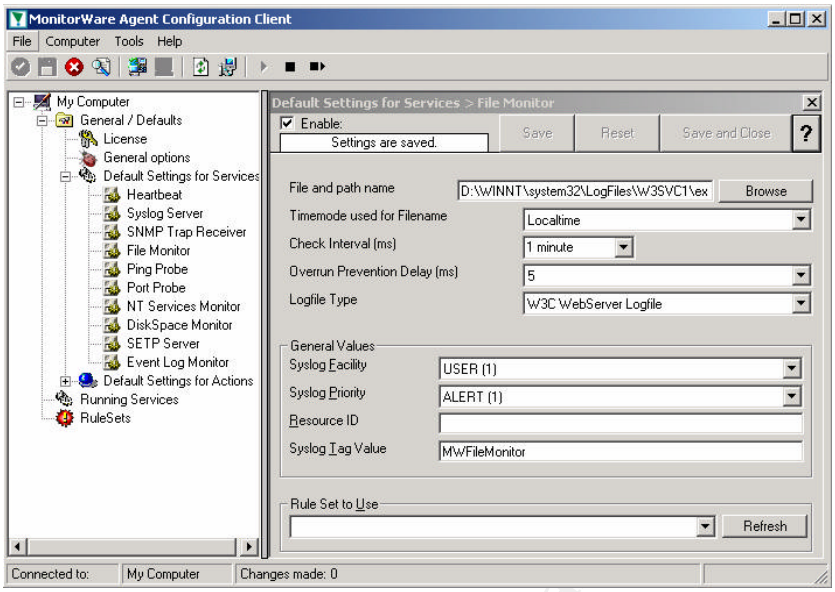
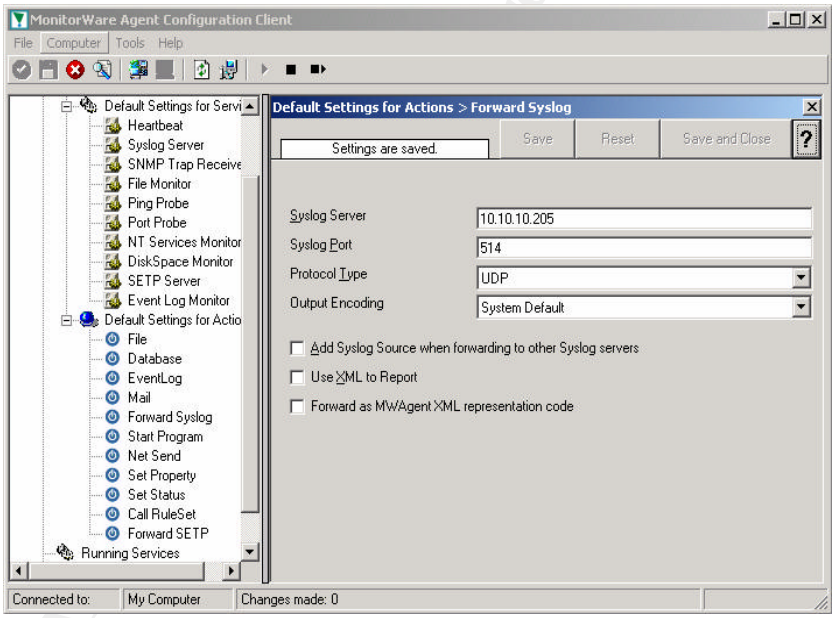


Figure7-4 Syslog Server Settings



Tools to convert and send packet capture data to syslog

Next, method of converting packet capture log to syslog format was evaluated.

As was mentioned in Section 6, the tool windump is able to collect packet capture data needed for this purpose. The problem of converting this data to syslog format and then transmitting it to a syslog server still exists.

In selecting a tool to resolve this problem, following three criteria were used:

1. Ability to transmit data in real-time (or as close to real-time as possible),
2. Ability to execute the program from the command line, and

3. Ability to configure the location of the file output (windump) and the file input (the tool) in a same directory.

Regarding Criteria 1, the real-time transmission is necessary to allow for collection of continuous record of captured packet data, with minimum of gaps in data. Criteria 2 is necessary because the windump itself is a command line tool. The need for Criteria 3, is, hopefully, self-evident. After meeting these criteria, the particular tool must have the functionality to accept input data from standard output device (i.e. display or stdin). With this, one can use commands to pipe the windump output into the conversion tool's input as shown below:

```
windump [condition expression] | convert.exe out=[syslog server]
```

The "convert.exe" used here is a format conversion tool, and the output of windump is piped into the input of the conversion tool. "syslog server" is a target host that receives syslog.

After researching for such a tool, the following tool was found to exist:

- Kiwi Logger[14]

Following are the most often used parameters for this tool:

-h -> syslog server name
-f -> facility#
-l -> level#
-i -> use stdin for message text.

By using the "-i" parameter, it is possible to construct a command as shown below:

```
windump [condition expression]  |
             klog.exe –h syslog server –F local1 –L notice  –i
```

Here "klog.exe" is the Kiwi Logger program. The "-h" option is used to specify the syslog server and "-F" option is used to set the "facility" value, and "-L" option is used to set the "severity level" value. By the "-i" option, klog.exe can get piped input data from windump. By structuring the command as shown, the resulting data from windump filtering should be converted to syslog format and sent to the syslog server.

8. Preparation and Settings

The following testing environment was used to evaluate the tools mentioned above:

Windows2000 (log-sender)

Following filtering rules were configured on the Windows2000:

-Capture telnet traffic coming into this host

-Capture ICMP Echo traffic coming into this host.

The NTsyslog was configured to:

-Transmit Audit Failure events

-The facility is set to “user” and severity level is set to “alert”

-Use default values for all other options.

The Monitorware Agent was configured so that:

-Target IIS log file(s) was selected and the File Monitor service was configured

-Facility was set to “user” and severity level was set to “alert”

-Receiving syslog server was selected (Linux server).

Next, based on the filtering rules, the batch file to execute windump was coded as shown below:

```
windump (dst host IPaddress and icmp[0] = 8) or (dst host IPaddress
and dst port 23) · klog.exe -h syslog server -F user -L alert –i
```

The “syslog server” used here is a Linux syslog server, and the “IPaddress” is the address for the Windows2000 log-sender.

Linux syslog server (log-receiver)

The Linux host was configured as the syslog server. The configuration settings were as shown below:

First, the initialization files used for syslog (/etc/init.d/syslog and /etc/sysconfig/syslog) were modified so that server is able to receive syslog data from remote hosts. The modifications were:

Add “-r” to the syslog initialization option (SYSLOGD_OPTIONS):

e.g. `SYSLOGD_OPTIONS· "-r -m 0"`

Next, an empty log file was created (i.e. windows.log) so that this file would be used by the syslog to save Windows2000 log data. Then the syslog configuration file was changed to use this file. The command used for creating new file was:

```
touch    /var/log/windows.log
```

The modifications of syslog configuration file were adding below entry:

```
user.alert     /var/log/windows.log
```

Next, the free software SWATCH[15] was installed as the syslog monitoring tool. SWATCH functions to monitor syslog, and when a specified keyword appears in syslog record, a warning is sent. By using this tool, one is able to monitor for critical

events and be alerted by email. Since the packet filtering log, this will be generated heavily, is being used in this study, one needed to be warned when certain packets considered “dangerous” were detected in the log.

To use SWATCH to monitor syslog, a list of keywords needed to be added to the configuration file. In this study, the following keywords were used:

- Keyword relating to event-log
  - “Logon Failure” to monitor failed logon attempts
- Keyword relating to IIS log
  - “cmd.exe” to detect host attacks such as shown below[16]:
    http://hostname/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir+c:\
- Keyword(s) relating to packet capture
  - “sec17.23” (host name and port number) to monitor telnet traffic, and
  - “echo request” to monitor for ping (icmp echo) requests.

Then, SWATCH was configured to sound a bell and send an alert email to root when any of the keywords were detected. The actual configuration file looks as follows:

```
## Personal Swatch configuration file#
watchfor  /Logon Failure/
       echo bold
       bell 1
       mail=root,subject=Login_Failed_SWATCH
watchfor /cmd.exe/
       echo bold
       bell 1
       mail=root,subject=cmd.exe_SWATCH
watchfor   /sec17.23/
       echo bold
       bell 1
       mail=root,subject=telnet_SWATCH
watchfor  /echo request/
       echo bold
       bell 1
       mail=root,subject=ping_SWATCH
```

After completing configuration, SWATCH was started as follows:

```
swatch -c /etc/swatch.conf -t /var/log/windows.log
```

Here, “/etc/swatch.conf” is the name of the SWATCH configuration file, and “/var/log/windows.log” is the syslog file being monitored.

Finally, the following tests were conducted from a remote host to evaluate the results.
a) “net use” command to simulate failed remote login attempts
b) IIS directory traversal attack (UNICODE attack)
c) telnet into Windows2000 host
d) ping the Windows2000 host

9. Conclusion

Following are the results of the test conducted:

“net use” command to simulate failed r emote login attempts

I used the following command and tried more than once to simulate failed remote login attempts:

```
net use x: \\computer name\share [password] /user:[username]
```

A portion of the syslog sent from NTsyslog to the Linux syslog server is shown below. Carriage returns were inserted for ease of viewing.

```
Jan 16 17:39:18 10.10.10.121 security[failure] 681 NT AUTHORITY\SYSTEM
The logon to account: Administrator  by:
MICROSOFT_AUTHENTICATION_PACKAGE_V1_0  from workstation: SEC555
failed. The error code was: 3221225578

Jan 16 17:39:18 10.10.10.121 security[failure] 529 NT AUTHORITY\SYSTEM
Logon Failure: Reason:Unknown user name or bad password User
Name:Administrator  Domain:SEC555  Logon Type:3  Logon Process:NtLmSsp
Authentication Package: MICROSOFT_AUTHENTICATION_PACKAGE_V1_0
Workstation Name:SEC555
```

Following is the alert email sent to root by SWATCH:

```
From: root root@localhost.localdomain
Message-Id: 200401160842.i0G8gnT06689@localhost.localdomain
To: root@localhost.localdomain
Subject: Login_Failed_SWATCH

Jan 16 17:39:18 10.10.10.121 security[failure] 529 NT AUTHORITY\SYSTEM
Logon Failure:  Reason:Unknown user name or bad password  User
```

```
Name:Administrator  Domain:SEC555  Logo n Type:3  Logon Process:NtLmSsp
Authentication Package: MICROSOFT_AUTHENTICATION_PACKAGE_V1_0
Workstation Name:SEC555
```

As can be seen from these results, the event-log was indeed sent from NTsyslog, and SWATCH detected the keyword “Logon Failure” and produced an alert for this event.

IIS directory traversal at tack (UNICODE attack)

I used the following URL to make UNICODE Directory Traversal attack[16].

```
http://hostname/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir+c:\
```

The content of syslog sent from MonitorWare Agent to the Linux syslog server when the above UNICODE directory traversal attack was conducted on IIS is shown below:

```
Jan 16 19:54:36 10.10.10.121 SEC250 MWFileMonitor: RealSource:"SEC250"
2004-01-16 10:53:53 127.0.0.1 - 127.0.0.1 80 GET
/scripts/../../winnt/system32/cmd.exe /c+dir+c:\ 200
Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+NT+5.0)
```

Following is the alert email sent to root by SWATCH:

```
From: root root@localhost.localdomain
Message-Id: 200401161058.i0GAw0l07820@localhost.localdomain
To: root@localhost.localdomain
Subject: cmd.exe_SWATCH

Jan 16 19:54:36 10.10.10.121 SEC250 MWFileMonitor:
RealSource:"SEC250" 2004-01-16 10:53:53 127.0.0.1 - 127.0.0.1 80
GET /scripts/../../winnt/system32/cmd.exe /c+dir+c:\ 200
Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+NT+5.0)
```

The result shows that Monitorware Agent indeed sent the IIS log to the Linux syslog server, and that SWATCH detected the keyword “cmd.exe” and produced an alert for the event.

telnet into Windows2000 host

The batch file introduced in Section 8 was used to execute windump and klog, but the telnet logs were not successfully sent to the syslog server. This is because the output

from windump was not properly fed into klog as input. After checking the windump (tcpdump) manual, it was found that the “-l” parameter could be used in conjunction to pipe command to feed data. This procedure is explained in the windump manual [8] as follows:

Make stdout line buffered. Useful if you want to see the data while capturing it.

E.g., ``tcpdump -l | tee dat'' or ``tcpdump -l > dat & tail -f dat''. Based on this, the batch file was modified as shown below, and the test was tried again.

```
windump -l (dst host IPaddress and icmp[0] = 8) or (dst host IPaddress
and dst port 23) · klog -h syslog server -F user -L alert -i
```

Following is an extract of the syslog file sent from the Windows2000 host to the Linux syslog server by windump and klog:

```
Jan 16 18:44:30 10.10.10.151 18:42:07.022471 SEC555.1040 > sec17.23: P
0:3(3) ack 19 win 17502 (DF)
 Jan 16 18:46:51 10.10.10.151 18:42:51.770755 SEC555.1048 > sec17.23: .
ack 1112 win 16410 (DF)
Jan 16 18:46:51 10.10.10.151 18:42:51.771113 SEC555.1048 > sec17.23: F
290:290(0) ack 1112 win 16410 (DF)
```

Following is the alert email sent to root by SWATCH:

```
From: root root@localhost.localdomain
Message-Id: 200401160946.i0G9ktp07314@localhost.localdomain
To: root@localhost.localdomain
Subject: telnet_SWATCH

Jan 16 18:46:51 10.10.10.151 18:42:51.771113 SEC555.1048 >
sec17.23: F 290:290(0) ack 1112 win 16410 (DF)
```

As can be seen from the result, the telnet packet captured by windump and processed by klog was indeed sent to the Linux syslog server, and that SWATCH detected the keywords for telnet and produced an alert for the event.

ping the Windows2000 host

The same batch file used in the previous test was used to execute windump and klog.

The following is an extract of the syslog file sent from the Windows2000 host to the Linux syslog server by windump and klog when the Windows2000 host was pinged from a remote host:

```
Jan 16 18:49:43 10.10.10.151 18:47:28.716283 SEC555 > sec17: icmp: echo
request
Jan 16 18:49:43 10.10.10.151 18:47:29.716286 SEC555 > sec17: icmp: echo
request
Jan 16 18:49:43 10.10.10.151 18:47:30.537786 SEC555 > sec17: icmp: echo
request
```

Following is the alert email sent to root by SWATCH:

```
From: root root@localhost.localdomain
Message-Id: 200401160954.i0G9sBF07512@localhost.localdomain
To: root@localhost.localdomain
Subject: ping_SWATCH

Jan 16 18:49:43 10.10.10.151 18:47:30.537786 SEC555 > sec17: icmp: echo
request
```

As can be seen from the result, the ping packets (icmp echo request packets) captured by windump and processed by klog was indeed sent to the Linux syslog server, and that SWATCH detected the keywords for ping and produced an alert for the event.

10. Evaluation

Through tests conducted in this study, it was confirmed that the event-log produced by Windows2000 host could be processed and transmitted using NTsyslog to the Linux syslog server in near real-time. It was also confirmed that the IIS log data, often a crucial piece of evidence in forensics analysis, could also be converted and transmitted using Monitorware Agent to the Linux syslog server successfully. Additionally, test showed that packet captured by windump could be sent to the Linux syslog server using the klog(Kiwi Logger) tool. By combining these various tools, one could setup an environment where both Windows2000 and Linux hosts possess nearly equivalent syslog (transmit) functionality. Additionally, these various logs could be collected centrally at the Linux syslog server, and managed and operated by using

syslog monitoring tools. However, some concerns were also discovered from the test results, and they are listed below:

a) The tool, Monitorware Agent, used to transmit the IIS log has a “Check Interval” setting, where the default is set to 1 minute. However, with this default value, an excessive time passes before the syslog server receives the data, thus unacceptable time gaps could show up in the log. Because of this, the “Check Interval” setting should be shortened for best result (15 seconds were used for the current testing).

b) There were unexplainable data loss (some log data not sent to the Linux syslog server) when using Monitorware Agent. It is not clear if this is a software bug, error in configuration, or incompatibility in Windows2000 OS version (The current test used Windows2000 with no patches applied). This issue needs further investigation.

c) When windump and klog were used to convert/transmit data to syslog, an excessive time passes before the Linux syslog server receives the data. This maybe due because the windump is buffering the standard output. However, in this case, the result is similar to a) where unacceptable time gaps could show up in the log. It would have been ideal if the buffering could have been controlled by some parameter. However, the study into this found that there are no such parameters. Also, the current study could not confirm the internal workings of buffering done by windump.

d) In using the combination of windump and klog, there were unexplainable data loss (log data not sent to the Linux syslog server) similar to b). This could be caused, not only by windump, but also by the versions of Winpcap and/or Windows2000. Further study needs to be conducted on these issues as well as the reliability of the klog tool.

11. References

[1] Gopalakrishna,Rajeev. “What is an Audit Trail ?” April 2000
http://www.cerias.purdue.edu/homes/rgk/at.html#intro
[2] office. “Audit Trail.” Guideline of Network Security No.9. July 15, 2003
http://shop.ns-research.jp/smi/sample.html
[3] ITsecurity. “Audit Trail(Audit Log)” Feb 01, 2004
http://www.itsecurity.com/dictionary/audtrail.htm
[4]syslog(3)-Linux man page
http://www.die.net/doc/linux/man/man3/syslog.3.html

[5] “The Network Time Protocol (NTP) Distribution.” October 13, 2003
http://www.eecis.udel.edu/~mills/ntp/html/intro
[6]Microsoft Technet. ”Netsh commands for Internet Protocol security(IPSec)”
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windowsserver2003/proddocs/standard/netsh_ipsec.asp
[7]”Description” Windump:tcpdump for Windows. January 02,2004
http://windump.polito.it/default.htm
[8] ”Windump Manual” Windump:tcpdump for Windows. March 14, 2002
http://windump.phttp://windump.polito.it/default.htmhttp://windump.polito.it/docs/manual.htmlito.it/docs/manual.htm
[9]EventReporter Web site
http://www.eventreporter.com/en/
[10]Smith,Curtis. “Eventlog to Syslog Utility” Purdue University Engineering Computer Network. http://engineering.purdue.edu/ECN/Resources/Documents/UNIX/evtsys/
[11]”BackLog-providing Syslog services for WindowsNT.” October 02, 2001
http://www.securiteam.com/tools/5YP0C003GA.html
[12]SourceForge Web Site
http://ntsyslog.sourceforge.net/
[13]MonitorWare Web Site. http://www.mwagent.com/en/
[14]Kiwi Enteprises Product Information Web Site.
http://www.kiwisyslog.com/products.htm
[15]Parker,Chris. “Using swatch for log analysis.” November 14, 2000
http://www.linuxsecurity.com/tips/tip-27.html
[16] ”Microsoft IIS and PWS Extended Unicode Directory Traversal Vulnerability.” SecurityFocus Vulnerability Web Site.
http://www.securityfocus.com/bid/1806/exploit/