Global Information Assurance Certification Paper

Title: The Importance of Logging and Traffic Monitoring for Information Security

Author: Seham Mohamed GadAllah

Version: GSEC Practical Assignment (v.1.4b)
(Option1)
30 December, 2003

INDEX

The Importance of Logging and Traffic Monitoring for Information Security

Abstract

This paper discusses one of the important aspects in any security model, which is the monitoring of the network and systems.

If you ask your self how you can get a complete view for your network, the answer will be almost through using a complete logging system and through using almost all the available traffic monitoring tools. All of them can combine with each other to give you a complete and a clear picture about the traffic passing through your network, as we will see.

From monitoring you can detect hacking attempts, virus or worm infections and propagation, configuration problems, exploits, hardware problems and many others. Monitoring is an important factor to maintain stability for the network.

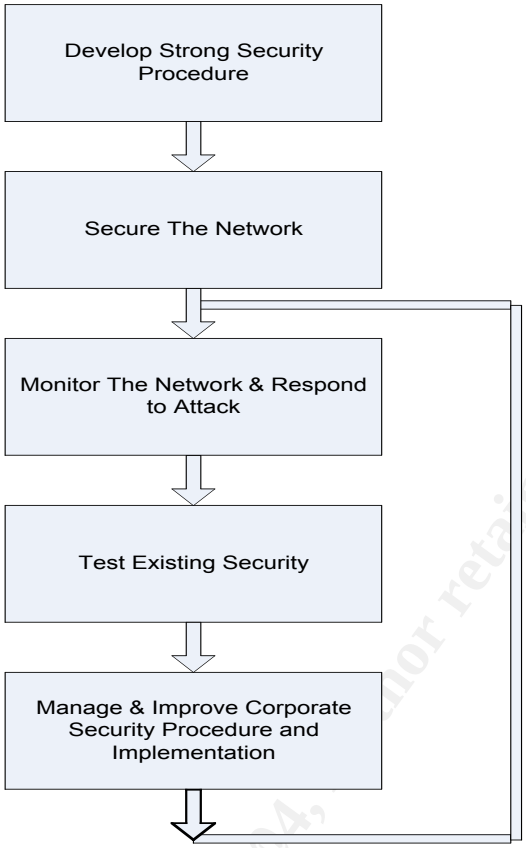
Information security focuses on ensuring confidentiality, integrity and availability. From network monitoring you can detect attempts to access forbidden information or resources such as unauthorized access, which in turn ensure confidentiality. You can detect attempts to change or alter information such as file modification, which ensure integrity. And you can detect any kind of problems that can affect the availability of the information such as DOS or DDOS attack.

The main goal of this paper is to give an idea about some of the benefits that any one can get from the complete monitoring of the network. Using both of logging for almost all the devices and the different types of network monitoring tools including bandwidth monitoring, packet sniffing and IDSs.

To have a clear understanding of this paper, it is better to have some basic knowledge about system and network administration.

1- Introduction

If we look at the following security process model [1]



Security Process Model

It shows the steps required to implement secure information system and to maintain it as secure as possible. The steps are as follow

1- Develop a security procedure that is suitable for your network. And that covers all the security issues related to your network.
2- Secure the network by implementing the developed security procedure.
3- Monitor the network & respond to attacks.
4- Test the existing network security. Check the security holes that you detect through monitoring.
5- Manage and improve the security procedure and the implementation.
6- Then you have to return back to step (3) to monitor, test and improve.

It is clear that the security process is a dynamic process. You must keep yourself updated with any new technology that can assist your security. And you must always check that your security policy is good implemented and in effect by periodic monitoring and testing.

From the above model we will concentrate our talk to be about network monitoring. To show how the complete network monitoring can give a clear view for the network security.

Network monitoring gives the ability to monitor the activities of the applications and the devices to ensure expected and normal operations. On the other hand it helps to detect problems and take the necessary actions to correct them. It can guide you to discover the security holes opened through your network intentionally by attackers or unintentionally such as disabled or unused suspicious services that may be enabled by mistake.

Any one can easily notice that attacks have become more sophisticated in the last several years as the level of attack automation has increased. You can obtain Sample and fully functional attack software easily from the Internet. Precompiled and ready to use programs allow any user to launch relatively large-scale attacks with little knowledge of the underlying security exploits. Because of that, Attack monitoring is a crucial part of the information system operation. The attack monitoring and detection can be achieved through network monitoring [2].

Network monitoring could be achieved through the following:
- Using an accurate and complete logging system for almost all devices.
- Using almost all the available traffic monitoring tools including bandwidth monitoring, packet sniffing, IDSs.

Logging can give detailed information about any access or change for any of the network resources. Frequently, uses of traffic monitoring tools help you to distinguish between normal traffic and suspicious one.

There are many free network-monitoring tools. That can help you to easily enhance your security; you do not have to care too much about the budget. The free tools are such as kiwi syslog daemon [3], backlog [4] for logging purposes and ethereal [5], MRTG [6], Snort (IDS) [7], for traffic monitoring purposes. We will give a hint about some of them later.

It is not an easy job to perfectly monitor the network. At the beginning you may face many difficulties in understanding and analyzing your logs and your network traffic. You also consume a lot of time to do that. By time you can gain the required experience to do your job quickly and easily. This can be achieved through being familiar with the normal logs and traffic passed through your network.

The order of the paper will be as follow:
- Review some issues related to logging such as logging management technique (Logging and Network Time Protocol (NTP), Choosing logging level, logging archival), and Syslog Servers.
- Review some issues about monitoring tools such as bandwidth monitoring, packet sniffing, network-based IDS.
- Show some detected cases from complete network monitoring.
- Finally suggest a procedure for the security or the network administrator that can be used to monitor the network.

2- Logging

“Logging can be a security administrator’s best friend. It’s like an administrative partner that is always at work, never complains, never gets tired, and is always on top of things. If properly instructed, this partner can provide the time and place of every event that has occurred in your network or system” [8].

Each network device or system has its own logging system such as UNIX servers, Windows servers, firewalls, routers, cache engines, IDSs, applications. You must monitor and analyze almost all the logs from your network devices and systems.

Centralized logging facilitates the process of monitoring and analyzing log messages. It is good practice to use a centralized syslog server for each type of devices, as an example:

- syslog server for all the UNIX servers
- syslog server for all the windows servers
- syslog server for all the firewalls

But at the end the decision depends on the size of the monitored network.

It is important for every security or network administrator to review the content of log files for suspicious entries indicating that a potential attack has occurred, or in the process of occurring in daily basis. Doing that will help him to enhance and maintain the security process [9]

2-1 Logging Management Technique

Logging management is very important, to have a good event logs you must have two main characteristics [10]

- Synchronized time stamp for each event.
- Sufficient logging level activity to produce detailed events of system activity.
- Sufficient archived logging information to be available if needed.

2-1-1 Logging and Network Time Protocol (NTP)

Logs are dependant on time, it is very important that your network devices, systems and your logging servers have an accurate time. To help ensure this, the Network Time Protocol (NTP) service is used [11].

Time synchronization is a must to have accurate and useful logging system, all your systems and network devices must have synchronized time stamp. If you look at any log message you will find that the time stamp is a basic part of it. If you have an accurate time for your logs you will be able to relate logging messages from different systems and network devices. In case of attack or any other network problem you could be able to analyze the logs from all the

systems and the network devices based on correct time. This could help to detect the attack or the cause of the problem and to solve it.

2-1-2 Choosing Logging Level

There are different logging levels (severity0-severity7), which are defined as Emergency, Alert, Critical, Error, Warning, Notice, Informational and Debug. The level of the syslog message specifies the type of messages sent to the syslog host.

Logging level is an important parameter that must be taken into consideration. You must choose the minimum level that gives you sufficient information. Choosing the suitable logging level helps you to maintain the stability of the network and to ensure that there will not be losses for required logging information. As an example if you set the logging level to be the highest logging level, which is debugging, there will be huge number of logging messages that require a substantial amount of disk space, high performance system, and adequate network bandwidth (specially if there is remote logging). In normal situation the error or the warning logging level could be enough. In situations that depend on or require more information you can increase the logging level as needed.

2-1-3 Logging Archival

Log server needs to have adequate amount of disk storage in order to hold all of the log messages that it is going to receive. As you look at your logging hosts you will notice that they are starting to fill up your storage media quickly, especially if you are in an active environment. Archiving your logs will help prevent your logging hosts from crashing due to storage limits being reached; which in turn leads to loss logging information. Using industry standard archive software, in combination with tape storage devices, network shares, CD-R, CD-RW, or Zip/Jazz drives; you can easily archive the logs [11]. Logging archival is one step to have good and complete logs. It helps you to return back to the old logs if you are in situation requires reviewing them.

2-2 Syslog Servers

The log message is a useful mean to view troubleshooting messages and to watch for network events such as attacks, service denials.

The syslog server is a server listen to different log messages from different servers. Using the syslog server you can establish a centralized logging system.

The RFC 3164 states that the syslog protocol provides a transport to allow a machine to send event notification messages across IP networks to event message collectors, also known as syslog server. Syslog server uses the user datagram protocol (UDP) as its underlying transport layer mechanism. The UDP port that is assigned to the syslog is 514 [12]

As an example of syslog servers is Kiwi Syslog Daemon. Kiwi syslog site defines it as a freeware server for Windows (you can use the basic features for free). It receives, logs, displays and forwards syslog messages from hosts such as routers, switches, UNIX hosts and any other syslog-enabled device. Using it you can archive the received messages into files in daily, weekly, monthly or custom basis [3].

We will briefly talk about how to configure a syslog server for Cisco PIX firewall and windows servers using Kiwi syslog daemon.

To use it with PIX firewall you need to install Kiwi syslog daemon on any Windows 2000 professional PC in your protected network (the syslog server specifications depends on the amount of traffic to be collected and the size of the network), and then to configure the firewall to enable the logging and to send the logging to Kiwi syslog server. The PIX firewall generates syslog messages for system events, such as security alerts and resource depletion [13].

To configure the Cisco firewall you need to add the following firewall commands (configuration mode):

Logging host inside x.x.x.x *; inside is the interface name*
Logging trap levelx *; levelx is the logging level*
Logging on

The first line is the logging host command. It designates a host "x.x.x.x" to receive the logging messages. This host has a syslog daemon running on it (Which is Kiwi syslog in our example). And it is connected to the inside interface (the most secure interface) of the PIX firewall.

The second line is the logging trap command. It sets the logging level to the required one. Cisco recommends that you use the debugging level during initial setup and during testing. There after set the level from debugging to error for production use.

The third line is the logging on command. It starts sending the messages to the syslog daemon. At any time if you want to stop sending the messages use the no logging command [13].

Next example is to demonstrate how to configure Kiwi syslog daemon to be used as syslog server for windows 2000 servers. First you need to install Kiwi syslog as above. For each windows server you need to install another component that is used to send the event log messages to Kiwi syslog daemon. There is free software that can do that which is backlog [4].

Backlog homepage states that it is a Windows NT service that facilitates the real time central collection and processing of Windows NT Event Log information. All event logs, application, system, security, Directory Service (for AD servers), File Replication, and DNS Server (for DNS servers) are monitored, and event information is converted to comma delimited text format,

and then delivered over UDP to a remote logging server. BackLog is currently configured to deliver audit information to a SYSLOG server running on a remote or local machine. It is also called Snare Agent for windows [4].

After the installation of the Backlog on Windows server, you need to configure Backlog to send messages to Kiwi syslog daemon, this will be done by configuring it to send the log messages to the IP address of Kiwi syslog server, also you need to configure the Backlog with the logging levels that suites your needs.

The following note must be mentioned here, you must be cautious when installing any software on a production server; you must take a complete backup for it before installation. In our case you must take a complete backup before installing Backlog on a production server.

3- Traffic Monitoring

There are many types of traffic monitoring tools that can be combined to monitor the traffic for the network devices, systems and applications. Frequently use of traffic monitoring tools helps you to be familiar with your normal traffic and as a result you can detect any suspicious traffic passing through your network.

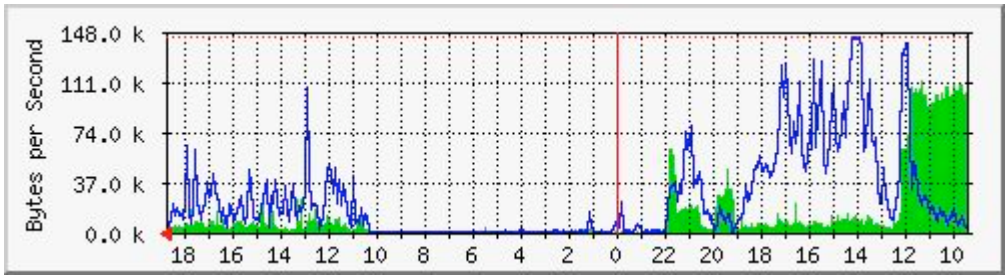
The following are some types of the traffic monitoring tools.

3-1 Bandwidth Monitoring

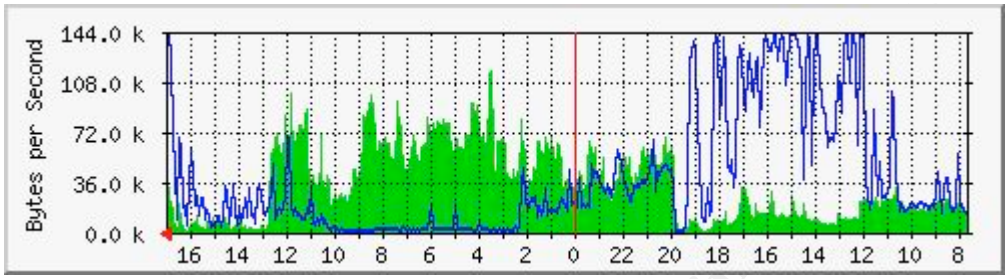
As an example of the bandwidth monitoring tools is the Multi Router Traffic Grapher (MRTG). MRTG homepage states that it is a tool to monitor the traffic load on network-links. MRTG generates HTML pages containing graphical images, which provide a live visual representation of this traffic. MRTG consists of a Perl script which uses SNMP to read the traffic counters of your routers and a fast C program which logs the traffic data and creates beautiful graphs representing the traffic on the monitored network connection. These graphs are embedded into WebPages, which can be viewed, from any modern Web-browser [6]

As an example the following are two graphs for the daily utilization of 1Mb bandwidth network. If you monitor the graph frequently, you can distinguish between normal and abnormal traffic.

The X-axis represents the hours of the day and the Y-axis represents the bandwidth utilization in Kbytes.



This picture shows the normal use for the network bandwidth



This picture shows abnormal use for the network bandwidth. There is a suspicious traffic passing through the network. You can notice the difference easily the shape of the graph gives a clear view.

Using MRTG gives a global overview for your network traffic. It could be used as a first step to check and monitor your traffic. From which you can take further steps to detect suspicious traffic or problems.

3-2 Packet Sniffing

As an example of the packet sniffing tools is Ethereal. Ethereal homepage states that it is a free network protocol analyzer for UNIX and Windows. It allows you to examine data from a live network or from a capture file on disk. You can interactively browse the capture data, viewing summary and detail information for each packet [5].

If you have doubt that there is suspicious traffic passing through your network or you have a problem and you want to know the source address, the destination address, the source port, the destination port, and the protocol for that traffic or problem, Ethereal could be of great help to you. At that point you could have all the information required to detect the attack and stop it or to detect the cause of the problem and correct it.

3-3 Network-based IDS

Intrusion Detection is the process of monitoring the events occurring in an IT system and analyzing them for signs of intrusions. These intrusions are the results of attackers accessing systems from the Internet, authorized users of the systems who attempt to gain additional unauthorized privileges, and authorized users who misuse the privileges given to them.

There are different types of Intrusion Detection Systems (IDS). There are Network-based IDS (NIDS), Network Node IDS (NNIDS), and Host-based IDS (HIDS). We will talk briefly about one of them, which is the Network-based ID system. It works like burglar alarm, alerting security people if an attack is taking place so that they can respond accordingly. It can detect intrusion by using signature/pattern analysis (signatures that are characteristics of an attack), or by using anomaly/heuristic analysis.

It is very important to notice that the IDS must be updated with the new attack signatures as soon as possible; most of the hackers try to use the new attacks to be able to compromise the systems before the administrator can patch them or apply the new updates.

IDS can do the following

- Identify attacks that firewall legitimately allow through (such as http attacks against web servers).
- Identify attempts such as port scan or ping sweep.
- Notice insider hacking.
- Provide additional checks for holes/ports opened through firewalls, intentionally or unintentionally

Snort is an example of the free available IDS [7]. The IDS capture each packet passing through a network segment and detects suspicious ones. It gives information about the source address, the destination address, the source port, the destination port, the attack type, the time of attack, it could also give advice about the required software upgrade or patches that could be used to prevent this attacks, and more other useful information.

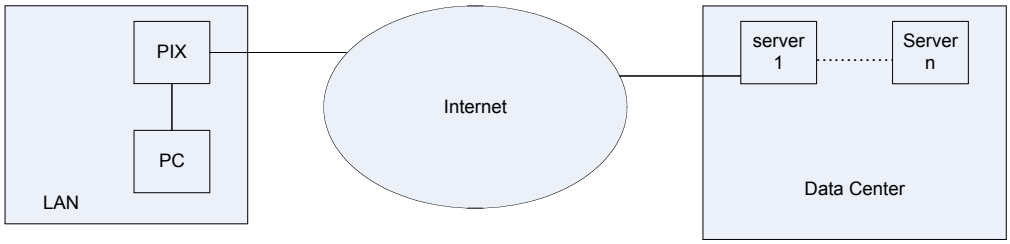
From the above you can see that the IDS can help you to detect the attacks that are already happened or in the process of happening through your network. It can help also to detect failed trials of attackers. And this can lead you to take the required actions to protect your network.

4- Detected Cases from Network Monitoring

The following are some examples of problems that already happened. From discussing the different cases you will notice that the main factor that helps to explain all the situations and to solve the problems was network monitoring. Also you will notice the importance of enabling the logging for all of the network devices and systems.

Without the complete network monitoring no one could be able to assure the security of the network. There may be many suspicious things that are happening through your network and you aren't aware with, the only thing that can give you the help is the complete network monitoring. It gives you a very clear and accurate picture.

4-1 Penetration Test Sessions



We used to do penetration tests periodically. We did the penetration tests from our LAN to test our Data Center servers; they are totally two different networks. We used a PC behind the firewall to do the test.

The steps toward detecting the problem were as follow:

- From the daily monitoring to the firewall logs (Using Kiwi syslog daemon with log files archived daily), It was found that the log file size is double the usual one this gave an indication for an abnormal situation. Searching the log file for abnormal events resulted in finding a huge number of events saying that there was a memory allocation error. No one from the users noticed any problems at that time.
- The error was repeated from time to time.
- The error occurred again and the users at that time were unable to connect to any external servers. The PIX firewall was hung up.

The period between the first and the last occurrence of the error was about 4 months. It was very important to detect the cause of the error. Returning back to the archived logs, it was found that the dates of the errors was exactly the same dates of the penetration tests that we used to do for our data center servers, using one PC from our protected LAN “behind the PIX firewall”.

The problem was that the penetration test opens a number of sessions greater than what is allowed through the firewall (although the number of tested servers was about 18 servers only).

As a result we moved the penetration test pc to be outside the firewall. And we started to run the penetration tests on non rush hours or at low load times.

From the above example it is very clear that:

- Logging
- Daily monitoring
- Archived logs

They are the only things that help to detect the problem and to solve it.

Another thing to notice, which is very important, you must know your system and network devices capacity to protect them from being overloaded. The penetration test tools opens a large number of sessions so you must monitor your network performance and utilization during the penetration test to be

sure that you will not affect your network performance. This will maintain the availability of the network.

4-2 Worm Detection

Every day we start by checking the MRTG graph for our bandwidth utilization. The shape of the graph oscillations gives an indication about our network traffic if it is normal or not. One day there is a great difference for the graph oscillation for both the inbound and the outbound traffic

The steps toward detecting the problem were as follow:

- Check the MRTG graph
- The next step was to check the firewall log file size. The firewall log file was about 9 times multiple of the usual size.
- Using Ethereal to be able to quickly detect the cause of the problem, there was a high percentage of ICMP traffic passing through the network. The source address of the traffic was one of the notebooks IP addresses that were connected to the LAN, and the destinations were external consecutive ranges of IP addresses “contiguous blocks of IP addresses”.

It was definitely the known worm “W32.Welchia.Worm”, one of its payload is to scan for active machines to infect by sending an ICMP echo request (or PING), resulting in increased ICMP traffic, localized network latency and widespread denial of service. The solution was to update the antivirus and to patch the notebook.

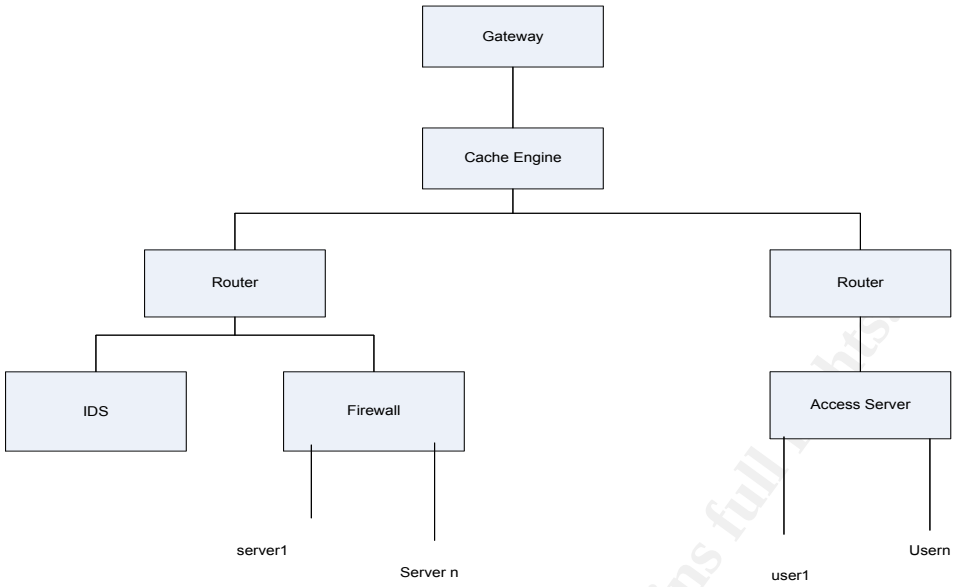
The detection of the worm was done through the network monitoring:

- Using bandwidth monitoring tool
- Using logging, daily monitoring
- Using packet sniffing tool

The three of them are combined to give a clear view for the problem and to solve it before it was distributed through the LAN.

There must be more attention and control for all the notebook devices that are connected to the LAN. They are vulnerability sources. They can transport the viruses and worms from one network to another.

4-3 Cache Engine Nimda Worm detection



After installing our IDS system by exactly two days the IDS was down. This was because it receives a huge number of events that leaded to the IDS's Database crash. We reinstalled the system again. And start to investigate the problem to stop it from reoccurring.

The steps toward detecting the problem were as follow:

- Check the IDS; the first thing to look for was the source of the huge number of events. It was the IP address of our cache engine and the destination was one of our web servers.
- We started to follow the logging step by step.
- We checked the firewall log file we found that there are access requests from the cache engine directed to the web server. From the firewall logs we got the complete requests paths.
- Also we checked the server logs to be sure of the requests paths and to get as complete information as possible. Till that time we did not know the exact source address of the attack.
- We had to check the cache logs to have a complete picture for the situation. Checking the cache log file we found that there was someone used different dialup access points to hack our servers. There was someone trying to hack our servers and to consume our bandwidth using the Nimda worm hiding itself behind the cache IP address.

From reviewing the logs we found different combinations of the following log entries directed to port 80/tcp for the web server:

http://x.x.x.x/scripts/root.exe?/c+dir
http://x.x.x.x/MSADC/root.exe?/c+dir
http://x.x.x.x/c/winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/d/winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/scripts/..%5c../winnt/system32/cmd.exe?/c+dir

http://x.x.x.x/_vti_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/_mem_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/msadc/..%5c../..%5c../..%5c/..\xc1\x1c../..\xc1\x1c../..\xc1\x1c../winnt /system32/cmd. exe?/c+dir
http://x.x.x.x/scripts/..\xc1\x1c../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/scripts/..\xc0/../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/scripts/..\xc0\xaf../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/scripts/..\xc1\x9c../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/scripts/..%35c../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/scripts/..%35c../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/scripts/..%5c../winnt/system32/cmd.exe?/c+dir
http://x.x.x.x/scripts/..%2f../winnt/system32/cmd.exe?/c+dir

This is produced by the scanning activity of the Nimda worm. At that time we could apply an access list on our cache to drop any requests containing any of the Nimda paths. That was the solution for the detected problem.

It is very obvious that the above problem was a complex problem and it could not be detected or solved without complete network monitoring, without the cache logging we couldn't be able to know the source address of the attack. That is why we say that we must log almost all the network devices and servers.
The network monitoring in this case was done by:

- Checking the IDS events
- Checking the firewall logfile
- Checking the server logfile
- Checking the cache logfile

All the monitoring tools were combined to detect the problem and to find the suitable solution for it.

5- Suggested Method for Network Monitoring

After the above overview about the logging and the traffic monitoring tools, and about some of the detected cases from the network monitoring, it is time to suggest a method that could be used daily to monitor the network.

1- Use a traffic-monitoring tool like MRTG to have a general idea about your bandwidth utilization. Bandwidth monitoring gives you a quick overview of the traffic leaving and entering your network. It can guide you to the next step that you can follow in case of detecting abnormal traffic.

2- Take a quick look for your logs, check the log file size if possible it can give an indication about attacks or errors. Start by looking at the firewall logs and then the servers.

3- Keep monitoring the IDS logs.
There is much more information that we can't mention about how to analyze your IDS logs. We will talk briefly about some of the guidelines that you can follow to monitor your IDS events.

First of all you must have a good archiving mechanism for your IDS logging. Professional attackers always try to hide their steps and this can be done by scheduling the process of attacking to be executed on different days, good archiving enables you to easily monitor the events and to detect previous hacking attempts.

It is better to care about the following when you monitor your IDS alerts:

- The total number of alarms coming from one source address
- The number of attack types coming from one source address
- The total number of destination addresses receiving attacks from one source address
- The new attack types
- The attacks that is rarely or not appear before.
- The time period in which the attack events happened.

If the total number of alarms received from one source address is high this may be an indication of attack from that source address. If the number of attack signature types coming from one source address is high this may be an indication of attack from that source address, some well known attack programs have a certain set of signature types that you can detect by frequently monitoring the IDS. If there is a number of destinations receive the same type of attack signatures from one source address this may be an indication of attack. Most professional attackers use the new attacks to compromise systems so you must care about new attacks, update your IDS with the new signature. Also you must notice the old attacks that you do not use to see them when monitoring your IDS. The time at which the attack happened could give some help, most attackers trying to hack systems in the nonworking hours to be sure that there is no one monitoring or watching them. The time period also is an important factor it can help you to guess if the attacker is using automated tool to hack the system, usually it takes small period of time if they use automated tools. All of the above factors can be combined to help you to detect if the alarms are false ones or they are indication of actual attacks.

4- Use a packet-sniffing tool like Ethereal to see if your traffic is normal or abnormal. Using packet sniffing tools can help you to be aware of the different protocols passing through your network, you can know the percentage traffic omitted from each protocol, and you can know the source and the destination addresses for all traffic. Packet sniffing tools require great amount of disk storage because it receives a copy of each packet passing through the network. You do not have to operate the packet sniffing software all the time, but it is preferable to have it available when you need to use it.

Conclusion

The network monitoring is a very important factor in information security systems. It enables you to check that your network is healthy. It helps you to maintain the stability of your network.

You can monitor the network by collecting the logging for almost all the network devices, systems and applications. And by using the different packet monitoring tools like the bandwidth monitoring, the packet sniffing and the IDS…

Each monitoring tool can give you a further step to complete the picture for your network security status. It can give you more helpful information that guides you to detect any attack or problem and to take the suitable action or solution. So you mustn't neglect the importance of each of them.

Frequently use of network monitoring tools familiarize you with the normal traffic passing through your network which in turn simplify the process of detecting attacks and virus or worm propagation and infection.

There are many free tools that can do the job for you. This is a good thing that reduces the need to increase the budget to implement the security solutions.

The monitoring process can guide you to discover any suspicious behavior before affecting your network or before propagating through it. The monitoring process can guide to further enhancement for your security system. It can give you an idea about the required tools that you can use to get more information that help to maintain confidentiality, integrity, and availability.

The network monitoring can guide you to the next step. To think about how to deal with results obtained from the network monitoring, which means the Incident handling phase that follow the detection of the attack or the problem.

References

[1] "Intrusion detection planning guide.", Cisco Systems,Inc, 1999, URL: http://www.cisco.com/en/US/products/sw/secursw/ps2113/products_maintenance_guide_chapter09186a008007d254.html, page 2-3

[2] Glenn, Michael, "A Summary of DOS/DDOS Prevention, Monitoring and Mitigation Techniques in a Service Provider Environment.", SANS Institute, August 2003, URL: http://www.sans.org/rr/papers/70/1212.pdf

[3] "Kiwi syslog daemon.", URL: http://www.kiwisyslog.com/index.htm, 1 December 2003.

[4] "Backlog: Snare Agent for Windows.", 1 December 2003URL: http://www.intersectalliance.com/projects/index.html

[5] "Ethereal.", URL: http://www.ethereal.com ,1 December 2003.

[6] "Multi Router Traffic Grapher (MRTG).", URL: http://www.mrtg.com,1 December 2003.

[7] "Snort The Open Source Network Intrusion Detection System.", URL: http://www.snort.org,1 December 2003.

[8] Stout, Kent, "Central Logging with a Twist of COTS in a Solaris Environment.", SANS Institute, March 2002, URL: http://www.sans.org/rr/papers/52/540.pdf

[9] Mendez, William, "Windows NT/2000 Event Logs.", SANS Institute, April 2002, URL: http://www.sans.org/rr/papers/67/290.pdf

[10] Stansbury, Jim, "Archiving Event Logs.", SANS Institute, August 2002, URL: http://www.sans.org/rr/papers/30/1002.pdf

[11] Allen, Stewart, "Importance of Understanding Logs from an Information Security Standpoint.", SANS Institute, 2001, URL: http://www.sans.org/rr/papers/33/200.pdf

[12] "Request for Comments:RFC 3164,The BSD Syslog Protocol.", August 2001, URL: http://www.faqs.org/rfcs/rfc3164.html , 1 December 2003.

[13] "Configuration Guide for the Cisco Secure PIX Firewall Version 5.3.", Cisco Systems,Inc, December 2000, page 2-46