Global Information Assurance Certification Paper

GSEC Practical Assignment Version 1.3
Rick Logan

Topic: Information overload, techniques and tools.

Abstract:

This article deals with the issue of security information overload. In today's technically advanced world, there is an expectation that the Security professional will be competent in managing vast amounts of information from a variety of devices and/or vendors. The professional's scope should be broad enough to encompass corporate wide business impacts right down to the structural integrity of the actual packets used to communicate. In this paper we will identify the problem and the relevant issues as well as examine an open source and a commercial solution to this issue.

Setting the scene.

Information security has evolved dramatically over the last few years as vendors and the user community have driven the requirements of systems and networks forward. Benchmarks such as CPU speed, disk capacity and network bandwidth have increased at an order of magnitude every few years. As both the quantity and importance of today's network traffic has increased so has the problem of managing and securing its safe passage.

Before the proliferation of the Internet and the desktop workstation, computer security was reasonably simple; users required a physical connection to the system and a valid username and password combination. Administration was relatively easy and centralised. Networks were typically bound to corporate environments with dedicated links for wide area connectivity, making the task of securing data storage and transfer relatively simple. The main chink in the amour was the humble modem, where anyone from anywhere in the world could establish a connection into your network accessing and or corrupting sensitive files and information.

Then along came the desktop system in the form of Unix workstations and the Windows or Macintosh Personal Computers (PC), with the users lapping up the local compute power and visually appealing programs such as office applications, email and browsers.

Users were either blissfully ignorant or willing to ignore the lack of security in the design of these products. For example, by default on the non-NT based versions of Windows it is as simple as clicking "Cancel" on the login screen and you are granted full access to the system and user information.

Several technologies evolved to meet these needs, but for the administrator of a heterogeneous network, the task became enormous. User authentication was available in centralised forms but only on a per platform basis. File access provided more complication when trying to manage both centralised and local workstation resources, to the point where some administrators deliberately disabled the users from accessing local resources. Novell's NDIS / eDIR and Microsoft's Active Directory Services address this growing identity management issues.

Typically the operating system's of these workstations supported auditing of user and system activity but since they were local and used up precious disk space, they were often left disabled or set to overwrite. Even if enabled they were very rarely checked, typically only in response to another situation, such as an employee termination.

Note: A well-defined procedure for examining and documenting a suspect system is an essential component of a security professional's toolkit, as the investigation may often lead to legal action many months or years down the track.

Even though disk and tape technologies had dramatically improved in terms of capacity, the introduction of local network cards to the desktop PC kept the volume of network traffic in excess of the logging and storage capacity.

Dedicated systems "sniffers" consisting of a promiscuous network card and packet decoder application were used to capture limited amounts of network activity. These sniffers were initially used for diagnostic purposes then later for security purposes. As security professionals realised the attacker's techniques could be recognised by monitoring patterns of network activity. Such devices later became the Intrusion Detection Systems of today. A practical example of this can be witnessed when installing and testing Snort[1] where the first test after installation is to "sniff" the traffic on the wire.

As companies ventured into the world of eBusiness one of the first challenges was to protect their systems from the very medium they had just connected to, namely the Internet. This issue was addressed with a network traffic filter called a firewall, the single most important component in perimeter security. Firewalls provided a blocking function to allow the passing of packets to defined addresses and ports, for both ingressing and egressing network traffic.

Security professionals such as Stephen Northcutt[2] have since recommended egress filtering as way of stopping the propagation of invasive behaviour on the Internet, such as worms and Trojans. These programs invade and corrupt unprotected systems and then use those damaged systems as a host much like a disease infecting the Internet. And like a disease there are a number of preventative measures to diagnose, quarantine and heal these viruses. A number of organisations have since risen to the challenge by producing innovative tools that detect, isolate and even repair these infected files.

Hackers or more accurately crackers[3] use a number of tools to qualify, seek out and attack targets. Tools used in the process of footprinting, scanning and enumerating[4] are often readily available either in the form of standard utility programs, such as the humble "ping", through to sophisticated tools from Internet repositories eg "Nessus"[5] from www.nessus.org is a vulnerability assessment tool and commercial sources such as "L0phtcrack (LC3)[6]" from @Stake Inc for a password security assesment.

As well as the information derived from the internal environment there is a stream (flood) of information available from vendors and the Internet security community regarding new vulnerabilities and solutions.

This paper deals with the management of information coming from the environment you control, or at least the configurations of, typically including such devices as border routers, firewalls, intrusion detection system's, virus scanners, vulnerability assessment tools, operating systems, databases, applications etc.

Before we go into the methodologies let us first examine the data protocols in detail used to transfer event information from the security devices to a security information management system.

Data formats:

Simple Network Management Protocol[7] (SNMP)

SNMP is a popular protocol used to manage devices on a TCP/IP network. It has three basic instructions for polling information GET, GETNEXT, and SET and a single TRAP command. Polling uses port 161 and is initiated by the management host to gather statical information from the networking devices.

Traps use port 162 and are asynchronously initiated by the networking device for exception reporting, some common examples are line circuits changing state or power supply failure. SNMP Information is stored in Management Information Bases or MIB's; MIB trees provide a hierarchical structure for the MIB parameters to be stored. These parameters can be fixed or variable, write enabled or read only.

SNMP caters for the storage and transmission of simple to complex data sets, typically describing the parameters of a network device. The MIB variables are identified with an Object Identifier, OID which is uses a dotted decimal notation and contain both standard and optional proprietary information. An example is 1.3.6.1.2.1.1.2 which is the parameter "sysObjectID".

When we examine the SNMP protocol data unit or PDU as defined in RFC1157, we see that an SNMP message must contain at least one variable binding or varbind, which is a variable's length, OID, type and value. The exception is the TRAP PDU type, which can contain zero varbinds. If they use SNMP, security devices typically alert via TRAPs.

This means that a SNMP message can typically contain from zero to a large number of varbinds per message. Now as the varbind contains both the variable's type and value, then the management system needs to know in advance what to expect in the way of messages, if it is to do anything useful with the data as all of the varbind OID's are in the proprietary fields.

This way more information can be contained in a SNMP packet but that it takes more effort to extract this at the management station, with preloaded decoders for every different sysObjectID. SysObjectID's are allocated per version per model per vendor so keeping up with the versions is an effort. Also as SNMP is UDP then the connectionless method of the protocol does not guarantee delivery.

SNMP Version 1 and 2 do not include any forms of encryption, SNMP V3 includes encryption but has not been as popular as V1 and V2 due to the complexity of implementation and lack of vendor support, but it is the opinion of this author that after the CERT Advisory CA-2002-03 "Multiple Vulnerabilities in Many Implementations of the Simple Network Protocol"[8] and the level of awareness by both the vendors and users that a renewed interest in SNMP V3 will occur.

SNMP trapping is available in CheckPoint FW1, CiscoPIX, NetScreen and Sunscreen firewalls also BlackIce, IceCap, ISS RealSecure, Dragon, NFR, Sessionwall and Raptor IDS's, and also McAfee antivirus devices.

System Log Files (SYSLOG)

Next we discuss syslogs, see "The BSD Syslog Protocol" [9] for a more complete description. Syslogs have been is use for the transmission of notification messages on Unix operating systems for a significant period of time. Simplicity in design has lead to wide acceptance of the Syslog protocol.

The architecture is described with three building blocks, the **device** which sends the Syslog message, the **relay**, which provides the forwarding function and the **collector**, which is the end point of the message. Relays are optional and can filter or add to the Syslog messages to the data flow. Devices can send to one or many relays or collectors. Relays and collectors can receive from one or many senders.

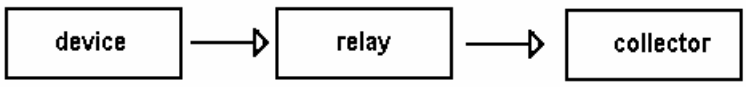

Figure 1: Syslog Message Flow

Syslog message are UDP and directed to port 514 by default and have a maximum packet size of 1024 bytes. The recommended message format consists of 3 main parts, the priority (PRI), header (HEADER) and message (MSG).

The PRI is a mathematical combination of a Facility and Severity code. The Facility has 24 numerical codes from 0 to 23 which map to various subsystems such as kernel, user, security, FTP daemon etc. Severity has 8 levels from 0 (emergency) through 7 (debug). The PRI equals the Facility value multiplied by eight added to the Severity value. It is transmitted as a three digit ASCII decimal number enclosed in angle brackets "<…>". The digits need to be visible (printable characters) i.e. in the range of %d48 "0" through %d57 "9", of 7 bit ASCII, with a requirement of no leading zero's. There is an exception of a single zero for the Kernel "0" Emergency "0" combination.
An example for a security (Facility=4) alert (Severity=1) would be 4*8+1=33, that is a PRI of "**<33>**".

The HEADER is a combination of other components namely the TIMESTAMP and HOSTNAME. The TIMESTAMP is the local time in format of "Mmm dd hh:mm::ss" with fixed values of "Mmm" being the standard month abbreviations, [Jan, Feb,Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec] and the "dd" being field a fixed two digit using leading space padding. The time is local system time in standard 24-hour format padded with leading zeros.

Please note no "year" parameter is included in the timestamp. The HOSTNAME is the what the local system is named, if there is no hostname then the IP address is used. Also the HEADER immediately follows the trailing ">" of the PRI, and the HOSTNAME is separated by a single space from the timestamp. To extend our example the Syslog message is now,

```
<33>Apr 14 11:30:10 localhost
```

The MSG is also in visible characters, and this typically means 7 bit ASCII characters %d33 – 126 with %d32 for space are used. It has two parts the TAG and the CONTENT. The TAG is used to for the name of the program or process used to create the message. The TAG field is limited to 32 characters and is usually terminated with a square bracket, colon or space character.

The CONTENT gives us the details of the message, the structure is not defined so the messages range in format and style, but the aim is to pass on a clear and concise message to the operator.
Our example could now be

```
<33>Apr 14 11:30:10 localhost xinetd[840]: START: sgi_fam pid=1243
```

The advantages of the Syslog protocol are, small fixed packets, easily relayed through a hierarchy to Syslog-server, simple to present to the operator due to clear text, escalation is reasonably easy to implement based on the PRI. Packets that conform to the recommendations allow interrogation programs to use search criteria that includes, limited date ranges, time frames, hostname, severity, facility and process name.

The disadvantages include a lack of a "year" field necessary for long term archiving, limited information available for decoding as typically a single Syslog has only the date, time, facility, severity, hostname, process and message. Data like who is the user, target and source addresses, etc can be included in the CONTENT field but it is extremely difficult to decode automatically. As the Syslog protocol does not include any security components and is defined using the visible characters it is easily intercepted and spoofed.

Another point is that even though any packet sent to port 514 is considered a valid Syslog message, relays are tasked with sending Syslog packets untouched that conform to valid PRI and TIMESTAMP values. If they do not confirm the relay attempts to make the packet conform by prefixing a valid PRI and/or TIMESTAMP to it. As the packet is limited to 1024 bytes, important datain the CONTENT field may be truncated when the relay adds the prefix.

Syslog's are popular with Cisco routers, Cyberguard, Netscreen, Snort, Watchguard and applications on Unix operating systems.

Windows EventLogs

Microsoft developed a different application to collect and administer the events from the Windows Operating System and Window's applications. These are stored in three database structures named, System, Security and Application. The Event viewer is provided for analysis of these records and also offers access to other windows systems. There are a number of tools available to manage or convert the NT event log into either a SNMP Trap or a Syslog file.

Other protocols

Some products also offer secure API's between the console/management station and the device. Check Point Software Technologies[10] have gained a lot of support with the OPSEC[11] program, with over 300 partners. This offers vendors an API that communicates with OPSEC compliant products. This channel is typically available and used for both provisioning and monitoring of the remote devices.

Intellitactics[12] offer a NetReader format that is more general and supports transmission of include source/target, ip/mac/hostname, native event code, category code, business grouping, priority, operating system, system version, CVE, risk levels, and more security related parameters in a proprietary protocol.

Tier-3's Huntsman[13] uses a Common Data Format CDF after collection before forwarding to the decider, the exact nature is not disclosed but it has both security and compression involved.

The vendor's who use proprietary protocols tend to use agents on the devices to collect and process the information into a standard format ready for transmission.

Others particularly in the vulnerability assessment products, produce reports ranging from plain text through to more detailed graphics in html. An excellent example of a vulnerability assessment tool is Nessus[4].

Typical event messages

This is very dependent on the environment and the device logging configuration but typically firewalls produce thousands to millions of events per day, Network IDS's in the range of hundreds to tens of thousands, Windows NT events are typically hundreds to ten of thousands per server and virus scanners can produce hundreds to millions of events. Vulnerability assessment tools can produces pages of reports per system. These values are extremely dependent on the environment and configuration of the auditing settings of the various devices.

One of the challenges is the information from each of these sources is in it's own format, from router hex codes through verbose vulnerability analysis reports.

Typically a high degree of product expertise is required to decode the various messages to understand the importance of the individual message. So just as a Checkpoint trained and experienced technician is required to read and understand the Checkpoint logs, another skill set is required for each separate vendor's product.

Processing Methodologies

In this section I would like to go through a methodology to add context to the incoming information. This is based on the Spectrum Security Manager by Aprisma Management Technologies[14]. In Figure 2 we can see the overall architecture that collects information from the sources, runs it through an intelligent rules system and provides notification and storage of the events.

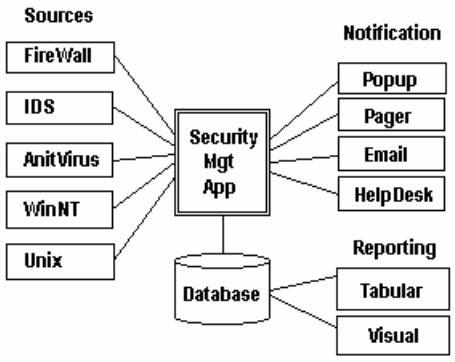

Figure 2: Security Information Management Architecture.

Data Collection:

Syslog servers have been popular for redirecting Syslogs into a central, file-structured repository, mainly for archiving. This is a reasonably efficient method of saving the data but it does suffer from a lack of indexing and retrieval is often cumbersome.

SNMP traps are typically directed at enterprise management systems, such as HP's Openview Network Node Manager, Tivoli's Enterprise Console and Aprisma's Spectrum Enterprise Manager etc. As the network administrators tend to manage these systems, the quality of notification can vary dramatically. Coupled with the limited bandwidth of the event consoles this tends to be a suboptimal solution. Also in the network space SNMP traps are usually sent as a notification of a critical event, such as a link down or power supply failure, hence there is a close correspondence between event and required action. With security events the relationship is much more loose, one or many events of a certain combination will now be required before a response is appropriate.

SMTP is usually directed at the administrators in the form of email notifications which are now stored in another location, the email repository.

NT event logs are stored on the servers as indexed files and typically have log storage configured to roll over existing logs when space or time criteria is met.

A better approach is to centralise the information into a database hence providing a unified repository and the search capabilities of a data warehouse.

Event Categorizing

Event categorizing is understanding what a particular message is and then producing a human readable "type" to describe that type of event. The schema for implementing such a categorization could use the source of the information such as "firewall" or "ids" or "antivirus", or a class of event such as "Authorisation" or "Configuration".

Categorizing substantially increases the effectiveness of the forensic search queries and subsequent notification and reports. It makes the message user friendly and easily understood. A practical example would be "ids.detect.dos.synflood" for the detection of the synflood attack from any of the vendor's intrusion detection systems.

Correlation

Correlation can be achieved in a number of ways. Categorisation allows correlation between different security devices to exist, for example if a password break-in attempt was recorded from all the networks and systems with the same essential details say "username, password and response" the security staff could easily detect a stolen users identification. Also at a macro level a sequence of events from various devices may when combined mean a certain attack vector can be determined and reacted to appropriately.

Prioritisation

The next step is to understand the importance of this event to your business's security and therefore react accordingly. Using the Model of Detection[15] that utilizes the concept of forward deployed sensors "indications and warnings" can be utilized to be proactive and align your defences against the impending attack vector. To be effective, prioritisation needs to be a combination of the attack and your vulnerability to that attack. An example would be detecting CodeRed. If it is detected in the internet facing systems, then it is just "noise". If it is detected in the trusted regions of the network, it is potentially dangerous! However, if you use Apache web servers and have removed the IIS components from your systems then a low priority could be assigned.

Analysis

Due to the complexity and variety of the asynchronous event streams, analysis is at best demanding. Currently the need for tools to cut through the noise without losing the detail is the challenge. This falls into two main categories tabular and visual. Tabular analysis typically uses keyword searches or time ranges to zoom in on the information of interest. Visual analysis uses maps to graphically display relationships in the data.

Tabular Tools:

ACID[16] provides a console to the IDS collectors to provide reporting and packet level investigation. It is available from CERT and has a number of integrations into IDS's such as SNORT. Some examples of the search and display capabilities are presented in the following figures. This is very useful for detailed tracking and analysis but does address the business impact.

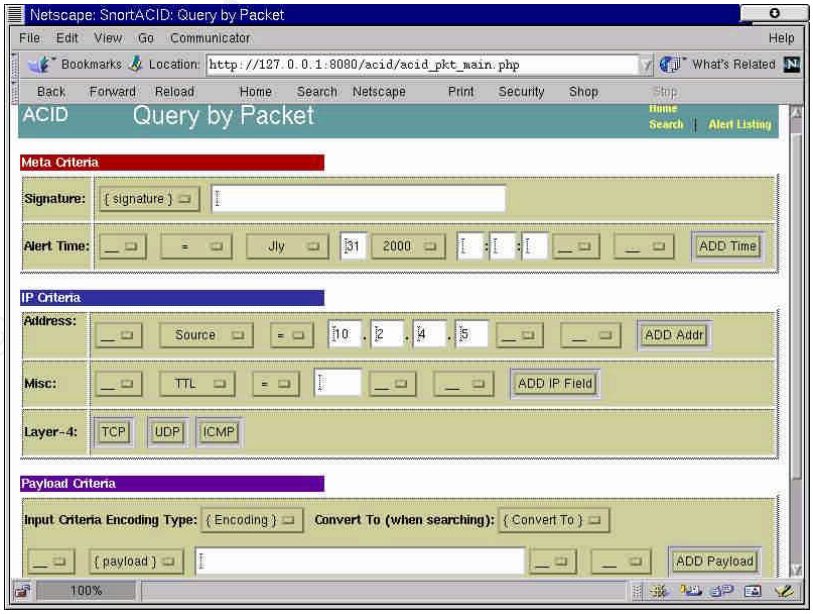

Figure 3. ACID: Query to find packets matching on detailed criteria.

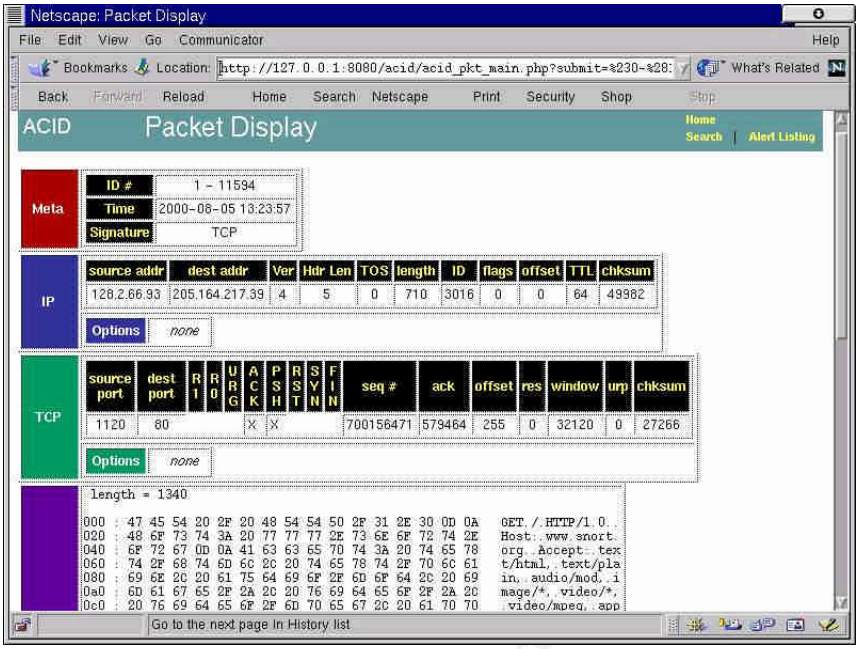

Figure 4. ACID: Packet level decoding of selected packet.

For managing data that has been categorized, Spectrum Security Manager[14] provides over 80 canned reports to provide higher level search criteria and some of these are displayed in the following figures. Although there is a slight overlap, both products compliment each other, providing easy access into different layers of the security management regime.

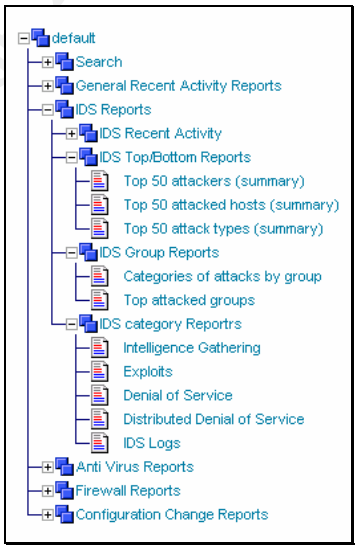

Figure 5. SSM: Report selection.

Report Name: ids-top-50-attackers
Description: Top 50 attackers (summary)

Total Rows: 50
Result Page: 1 2 Next
Total Pages: 2

num	s_ip	s_hostname	type	zone
150	10.0.0.4	NULL	ids.detect.recon.probe	acme.r&d.unknown
145	10.0.0.15	NULL	ids.detect.recon.probe	acme.r&d.unknown
107	10.0.0.4	NULL	ids.detect.recon.probe	acme.p&s.wks
106	10.0.0.15	NULL	ids.detect.recon.probe	any.unknown
105	10.0.0.15	NULL	ids.detect.recon.probe	acme.p&s.wks
92	10.0.0.4	NULL	ids.detect.recon.probe	acme.r&d.lab
85	10.0.3.227	NULL	ids.detect.recon.probe	any.unknown
73	10.0.0.4	NULL	ids.detect.recon.probe	any.unknown
72	10.0.0.15	NULL	ids.detect.recon.probe	acme.r&d.lab
44	10.0.0.114	NULL	ids.detect.recon.probe	any.unknown
44	10.0.0.4	NULL	ids.detect.recon.probe	acme.r&d.svr
43	10.0.0.15	NULL	ids.detect.recon.probe	acme.r&d.svr
36	10.0.0.202	NULL	ids.detect.recon.probe	acme.r&d.unknown
34	10.0.10.50	NULL	ids.detect.recon.portscan	any.unknown
28	10.0.0.202	NULL	ids.detect.recon.probe	acme.p&s.wks
21	10.0.0.114	NULL	ids.detect.recon.probe	acme.r&d.unknown
21	10.0.0.202	NULL	ids.detect.recon.probe	acme.r&d.lab
18	10.0.0.113	NULL	ids.detect.recon.probe	any.unknown
15	10.0.0.4	NULL	ids.detect.unsecure.service.snmp	any.unknown
14	10.0.0.46	NULL	ids.detect.recon.probe	any.unknown

Figure 6. Report of the Top 50 Attacking hosts.

Visualisation Techniques

Trending and relationship mapping form the main visualisation techniques. Gary Geisler[17] cited Ben Shneiderman's[18] seven data types;

- One-Dimensional
- Two-Dimensional
- Three-Dimensional
- Multi-Dimensional
- Temporal
- Hierarchical
- Network

Typically security management has been reported in the one-dimensional sense, e.g. how many CodeRed attacks were perpetrated against the farm of web servers last month etc. Whereas the correlation of Target and Source addresses is done implicitly by security staff when determining who is attacking what in their environment. This can be visualised using two-dimensional reports, whereas correlating the attack type and amount across a global organisation or the Internet requires multi-dimensional techniques.

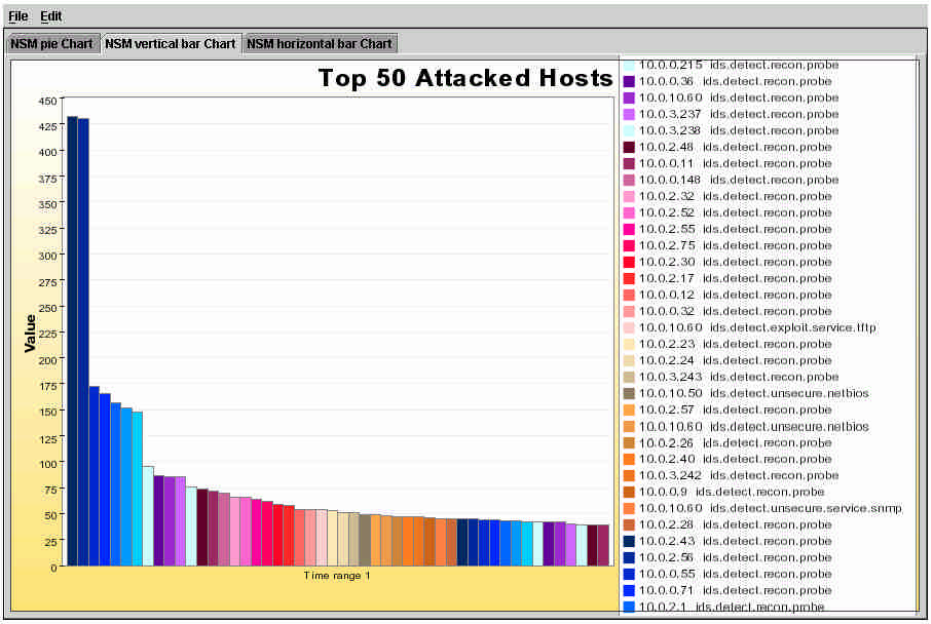

Figure 7. SSM: One–dimension display of attack types.

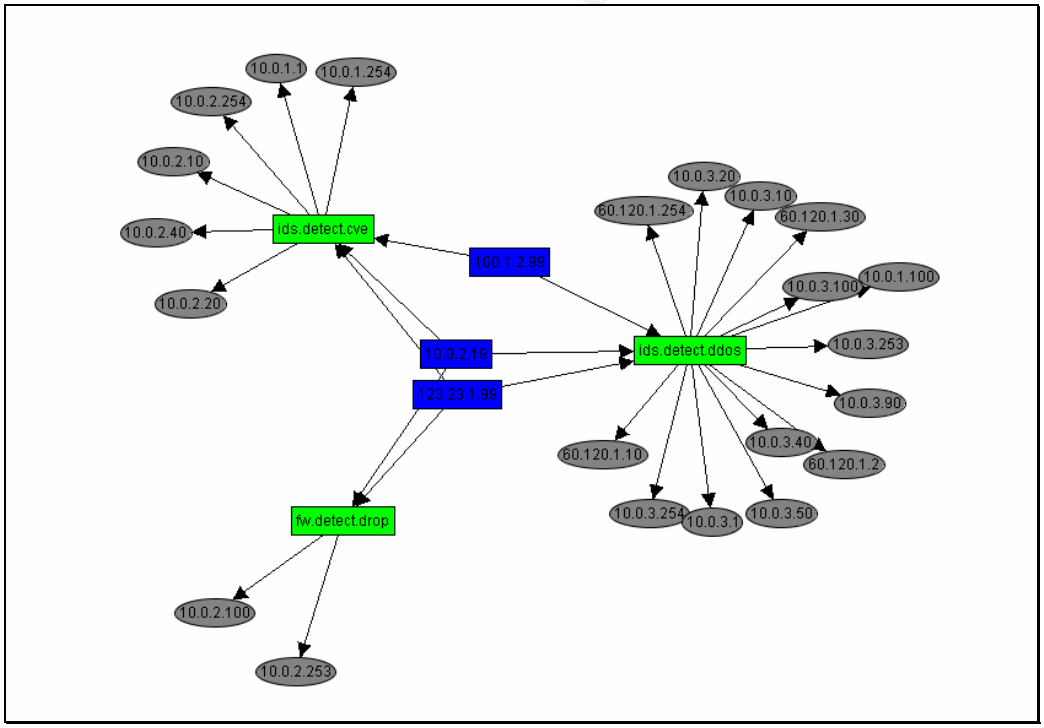

Figure 8. SSM Two-dimension display of relationships between source and target and attack types.

Notification

An essential component of any solution is appropriate notification. Typically notification mechanisms range from high priority pager or SMS, through to email and then long term reporting. Some of the key features are quick response, concise and meaningful messages. This enables the Trouble Ticketing system or On-call support person receiving the page to make a reasonable and timely judgement for the appropriate action.

Conclusion:

Companies are expecting more for less, security professionals be nature of their role place restrictions and controls on the IT infrastructure and staff and are considered to be a overhead.

The amount of information requiring analysis is beyond the ability of humans to read and react to in a timely manner. The quantity of output from the security devices and operating systems is increasing with time and new product development.

By categorizing, zoning and prioritising the incoming data security systems can add context to the information and be used in the real time and historical reporting.

Techniques that allow data presentation in a variety of visual formats can reveal attack patterns and trends that are hidden in the tabular data, revealing relationships rather than just quantifying the data.

The main protocols in use today provide differing levels of security and information capacity. There is no universal protocol that addresses the needs of all the devices in the IT infrastructure.

It is the opinion of the author that in the same way Common Vulnerabilities and Exposures[19], (CVE) has brought together the Internet community; efforts for a Universal Logging Protocol[20] (ULP) would improve the responsiveness and effectiveness of security professionals.

Security professionals now have a choice, whether to use some of the recently products to manage their security information or be left behind in the sea of meaningless log information.

References:

1. Marty Roesch, Author of Snort, March 2002, http://www.snort.org/docs/writing_rules/
2. Steven Northcutt, Principal SANS Instructor and Author. Jan 2002
3. crackers, http://www.tuxedo.org/~esr/jargon/html/entry/cracker.html
4. S.McClure et. al. Hackers Exposed 2nd Edition http://www.hackingexposed.com/ (2001)
5. Renaud Deraison. (www.nessus.org)
6. @Stake, LC-3 Password auditing and recovery application. http://www.l0pht.com/research/lc3/index.html
7. J.Case, M.Fedor, M.Schoffstall, J.Davin SNMP. May 1990, http://www.faqs.org/rfcs/rfc1157.html
8. CERT Advisory CA-2002-03 http://www.cert.org/advisories/CA-2002-03.html Multiple Vulnerabilities in Many Implementations of the Simple Network Protocol Feb - Apr 2002
9. C.Lonvick The BSD Syslog Protocol http://www.faqs.org/rfcs/rfc3164.html Aug 2001
10. Check Point Software Technologies http://www.checkpoint.com/ Apr 2002
11. OPSEC Partner Program and Architecture http://www.opsec.com/ Apr 2002
12. Intellitactics "NSM Feature Sheet" http://www.itactics.com/html/nsm_feature_sheet.html#8 Apr 2002
13. Tier-3 Huntsman Architecture White Paper http://www.tier-3.com/whitepapers.asp Oct 2001
14. Aprisma Management Technologies www.aprisma.com SSM White Paper http://www.aprisma.com/products/SSM/SSM-WP.pdf 2002
15. S.Northcutt (2000) et. al. "SANS Security Essentials Vol1.1" Measures of Detection: p1-24
16. Roman Danyliw, ACID (Analysis Console for Intrusion Databases) http://www.cert.org/kb/acid/ roman@danyliw.com
17. Gary Geisler (1988) Making Information More Accessible: A Survey of Information Visualization Applications and Techniques. http://www.ils.unc.edu/~geisg/info/infovis/paper.html#network
18. Shneiderman, B. (1998). "Information Visualization." Designing the User Interface: Strategies for Effective Human-Computer Interaction (Third ed., pp. 522-541). Reading, MA: Addison Wesley Longman, Inc.
19. CVE Common Vulnerabilities and Exposures http://cve.mitre.org/cve/ Apr 2002
20. ULP Universal Logging Protocol http://www.hsc.fr/gulp/charter.html Nov 1997