Edith Cowan University

Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study.

The University does not authorize you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following:

- Copyright owners are entitled to take legal action against persons who infringe their copyright.
- A reproduction of material that is protected by copyright may be a copyright infringement.
- A court may impose penalties and award damages in relation to offences and infringements relating to copyright material. Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.
A study on denial of service –resistance of some IPsec-implementations

Mika Müller\textsuperscript{1,2}, Jari Arkko\textsuperscript{2}, and Teemupekka Virtanen\textsuperscript{1}

\textsuperscript{1}Telecommunication software and Multimedia Laboratory
Helsinki University of Technology
Email: mmuller@mail.niksula.cs.hut.fi, teemupekka.virtanen@hut.fi

\textsuperscript{2}Ericsson Finland
Email: \{mika.muller, jari.arkko\}@ericsson.fi

Abstract

IP security (IPsec) is in global use for example in corporate Virtual Private Networks. It is also intended for the protection of nodes in the third generation (3G) mobile networks. Denial of Service (DoS) is a threat especially in 3G networks where availability requirements are very strict. This thesis is about identifying those threats and presenting methods for analyzing IPsec implementations and their vulnerabilities to certain Denial of Service attacks.

The objective of this study is to review IPsec DoS vulnerabilities, and to produce and analyze tools for this. The best entry points for DoS attacks are in IKE protocol, so the scope of the study is limited to attacks against IKE.

The results show that implementations differ very much from each other in robustness against chosen attacks. In some attacks the best implementations do not suffer from DoS at all, but poor implementations may even crash. Simple protections, such as hard-coded limits for memory consumption, works well against the tested DoS attacks.

Keywords: DoS, Ipsec, IKE

Attacks against IKE

All the following attacks with "Cookie"-prefix were originally presented by William Simpson in Simpson (1999). They are mostly considered flaws in the IKE-protocol, so IKE-implementations should not be able to resist them without new product development.

IKE uses cookies that are specified by the ISAKMP-standard (NWG, 1998). The cookies are randomly generated data which are used to identify Initiator-Responder pairs. Every ISAKMP header contains I-cookie and R-cookie (Initiator and Responder respectively), except for initial proposal. This is depicted in Table 1.

<table>
<thead>
<tr>
<th>Message #</th>
<th>Operation</th>
<th>I-Cookie</th>
<th>R-Cookie</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>IKE starts ISAKMP SA negotiation</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>(2)</td>
<td>IKE responds ISAKMP SA negotiation</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>(3)</td>
<td>IKE initiates Ipsec SA negotiation</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>(4)</td>
<td>IKE responds to Ipsec SA negotiation</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 1: Cookie status for ISAKMP-messages

\url{http://ro.ecu.edu.au/ecuworks/6759}
Cookie Crumb Attack

Storing the state about cookie-pairs leads to a dilemma of leaving all state-information in memory as half open connections, while waiting for Initiators to continue with key-exchange requests. If a malicious Initiator makes a large amount of proposals with a different randomly generated IcookiP, all the stored states combined will result in a lack of memory in the Responder. These pieces of state-information that finally overwhelm the memory were called 'Cookie crumbs' in Simpson (1999).

The Cookie Crumb Attack can be executed by flooding the target IKE with Main-mode ISAKMP-proposal packets which have different Icookies, SPI:s and nonces.

Cookie Jar Attack

Since IKE doesn't limit the number of ISAKMP-proposals done by a single Initiator, it is possible for the Initiator to collect a huge amount of responses with valid cookies, and then send all the key-exchange requests at once, so the Responder will run out of resources while simultaneously calculating the shared secrets (Diffie-Hellman) and decrypting the nonces.

The attacker doesn't need to make expensive computations since payloads for key-exchange and authentication can be properly formatted garbage.

Aggressive mode attacks

IKE has two operation modes, Main mode and Aggressive mode. In the Main mode, IKE executes cookie exchange before it will do any public key operations.

Using aggressive mode we should be able to cause a significant CPU load with any standard IKE-implementation, by just trying to send multiple key-exchange proposals with garbage contents. It will be interesting to see if there will be differences between implementation about how can they resist this.

Christmas Tree Attack

Christmas Tree packet is defined in Raymond et al. (1996) as "A packet with every single option set for whatever protocol is in use" The term derives from a fanciful image of each option bit being represented by a different-colored light bulb, all turned on.

This could lead to ie. buffer overflows in some implementation and thus crash that particular IKE. Some implementations may limit the size of the proposal-packet and thus limit the effects of this attack.

The test environment

We set out to find suitable security scanner software that could perform the attacks automatically with minimum coding effort. We selected the following software for closer examination (Table 2).

<table>
<thead>
<tr>
<th>Security scanners</th>
<th>Packet generators</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satan</td>
<td>Spak</td>
<td>Ericsson IPsec</td>
</tr>
<tr>
<td>Saint</td>
<td>sendIP</td>
<td></td>
</tr>
<tr>
<td>Nessus</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Selected software
We chose to implement the tool as plugins for Nessus and as stand-alone C-modules, when power of NASL was not sufficient. All IKE-related code were written from scratch in any case.

![Network configuration diagram](image)

**Figure 1: Network configuration**

The test-environment consisted of a lab-network which uses private IP-addressing. 100M ethernet were mostly used. Figure 1 shows the network configuration.

We used RedHat linux 7.2 machines with PIII/500mhz CPU and 192M memory. In addition we had to use one FreeBSD machine in order to run KAME IPsec. These machines had equal hardware resources, but the FreeBSD-machine had only 10M ethernet while the others had 100M.

We also included a Cisco 2514 router, which has only 10M ethernet and a Motorola 68030 processor, which is over ten times slower compared to other target machines. The actual Ipsec-module is running over Cisco IOS12.0, and it has no hardware acceleration on it.

The two Linux machines were used to run the DoS attacks. We could have used FreeBSD to run the attacks too, but found no specific need to do that, because we already had two machines to cross-test each other when needed.

All the machines in Figure 2 were used as targets for DoS attacks. Table 3 shows which software each machine had. It only contains the relevant information for our DoS-attacks. List of important applications are the most frequently used applications. Ping was needed to test and adjust interoperability of IKE-implementations, and for tests to use a ping flood for exhausting the CPU.

<table>
<thead>
<tr>
<th>Machine</th>
<th>Ipsec</th>
<th>IKE</th>
<th>Important applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux 1</td>
<td>Ericsson IPsec</td>
<td>Ericsson/SSH</td>
<td>Nessus, Ethereal, Ping</td>
</tr>
<tr>
<td>Linux 2</td>
<td>FreeS/WAN IPsec</td>
<td>Pluto</td>
<td>Nessus, Ping</td>
</tr>
<tr>
<td>FreeBSD</td>
<td>KAME IPsec</td>
<td>Racoon</td>
<td>Ping</td>
</tr>
<tr>
<td>Cisco 2514</td>
<td>Cisco IPsec</td>
<td>Cisco</td>
<td>none</td>
</tr>
</tbody>
</table>

**Table 3: Targets of DoS**

**Software overview**

The software solution we implemented consists of a number of Nessus-plugins, basically one plugin for each attack, but variations of attacks had to be written because of some differences between target IPsec implementations.

Choosing Nessus as the platform for our attacks was a good initial solution, in order to write the modules quickly from the scratch. This was easily accomplished with NASL and with the help of Ethereal-sniffer.
There were problems in implementing some of the more complex attacks though, mainly due to lack of power in NASL. For more a complete set of attacks, NASL is not sufficient, but Nessus-modules in C are.

The process of writing the plugins was straightforward. After the initial NASL-code for forging UDP-packets was written, the other modules could re-use the same code, and modifications were mostly made on raw ISAKMP-data, which we captured with a sniffer. We then had to modify some parts of the ISAKMP-packet for each attack by hand or by generating dynamic values for the Initiator-cookies, the SPI:s and the nonces.

**Attacks implemented**

We implemented attacks that are summarized in Table 4. Since FreeSWAN did not accept as many configurations as others, we have to fit the parameters according to its requirements.

<table>
<thead>
<tr>
<th>Attack name</th>
<th>Affected resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cookie crumb</td>
<td>Memory + CPU</td>
</tr>
<tr>
<td>Christmas tree</td>
<td>CPU</td>
</tr>
</tbody>
</table>

Table 4: Attacks implemented

<table>
<thead>
<tr>
<th>IKE mode</th>
<th>Main mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>IKE authentication</td>
<td>Pre-shared keys</td>
</tr>
<tr>
<td>IKE encryption algorithm</td>
<td>3DES (DES for Cisco)</td>
</tr>
<tr>
<td>IKE hash algorithm</td>
<td>md5</td>
</tr>
<tr>
<td>IKE diffie-hellman group</td>
<td>modp1024</td>
</tr>
</tbody>
</table>

Table 5: IKE-parameters used in implemented attacks

In Cookie Crumb Attack we divided the captured real ISAKMP-packet into pieces, and separated the cookie part from rest of the data. Then we ran this cookie part in a big loop which increments the cookie value by one every round. We sent the packet to the target every round also by forging a new UDP-packet. That was somewhat heavy operation and using UDP-sockets for sending the updated data there should have been faster. Unfortunately NASL did not let us specify the source port on UDP-socket, so the socket-version would not had worked on FreeSWAN, which requires that the port is 500.

In the Christmas tree attack we needed to build a large UDP-packet. This is not possible in NASL without using UDP-sockets, and we did not use those for above reasons. That is why we were able to create only as big UDP-packet that can go unfragmented in ethernet. It was less than 1.5KB long, containing only 32 different transforms. That packet was suitable for initial testing and debugging of the packet-data and length fields. Then we decided to implement the attack in C, which was quite straightforward, just basic socket programming and the data was specified in similar way as hexadecimals as in NASL. We then started to experiment with packets of different lengths. We used only the C-version of the attack when we did the actual testing against different IKEs.

**THE RESULTS**

All the results here have been achieved mostly with default settings in each IKE implementation. These default settings correspond to IKE-parameters presented at the previous chapter. The process priorities on operating systems were also default for these tests.
We assume that the default settings are typically in use, when people utilize these IPsec implementations around the world. Thus we choose to test with the defaults, when we need to choose the settings to analyze for each implementation. This will give us knowledge for better analysis of real world attack scenarios.

For each attack we present a summary of effects on the system resources. Then we present graphs of CPU/memory usage of IKE-processes where appropriate. Then the implications of the attack to the legitimate IPsec traffic are presented. Those are the actual DoS effects and thus the most essential part of the results. Finally the DoS-vulnerability in the real world is discussed, based on these implications.

**COOKIE CRUMB RESULTS**

The results analyzed here we first gathered with our Nessus-module. Then we made comparisons with the C-language version from Simpson (1999), but against our expectations, the over 10 times faster version was not superior in exhausting memory, since it caused FreeBSD to freeze even though IKE was not running in it.

The Cisco did not freeze under this attack, but it experienced serious slowdown for apparently the reason that its processor is very slow. The reason for the complete freezing was left unknown, but it did not depend on ethernet bandwidth, since the Cisco and the FreeBSD both had a slower 10M ethernet. The reason might be in network interface buffers, which are filled up and since X-windows uses the network, it can't allocate buffers and freezes.

The C-language version is later referred as *crumb.c*. It will generally result to more efficient Denial of Service due to a massive packet flood, and especially against FreeBSD platforms, because it jams it in any case. The NASL-module is later referred as *crumb.nasl*.

<table>
<thead>
<tr>
<th>Effects on the system resources</th>
<th>KAME</th>
<th>Cisco</th>
<th>FreeSWAN</th>
<th>Ericsson</th>
</tr>
</thead>
<tbody>
<tr>
<td>IKE crashed</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Consumed memory</td>
<td>192M+</td>
<td>12.5M</td>
<td>4M</td>
<td>2M</td>
</tr>
<tr>
<td>Memory left</td>
<td>0M</td>
<td>0.5M</td>
<td>180M</td>
<td>190M</td>
</tr>
<tr>
<td>Time to exhaust memory</td>
<td>Hours</td>
<td>10 mins</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Slowdown of UI (crumb.nasl)</td>
<td>Serious</td>
<td>Minor</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Slowdown of UI (crumb.c)</td>
<td>Frozen</td>
<td>Serious</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>CPU load (crumb.nasl)</td>
<td>95%</td>
<td>96%</td>
<td>3%</td>
<td>10%</td>
</tr>
<tr>
<td>CPU load (crumb.c)</td>
<td>NA</td>
<td>99%</td>
<td>72%</td>
<td>79%</td>
</tr>
</tbody>
</table>

Table 6: Cookie Crumb Results

The effects on system resources of each implementation are described below. Table 6 summarizes the results. "NA" in CPU load means that we could not measure it, since the UI was frozen in FreeBSD/KAME.

Note that 'time to exhaust memory' can not be compared directly between Cisco and the others, because Cisco has only 13M of physical memory, while others have 192M. Same rule applies to CPU consumption, because Cisco has over 10 times slower CPU. The CPU load means an average cpu load percentage during the first 10 minutes of the attack.
KAME

KAME was the worst performer under this attack. Only its IKE crashed with this attack. The \textit{crumb.c} version of the cookie crumb attack completely froze the UI of the FreeBSD while it had no effects of Linux machines with the same hardware and only minor effect to Cisco, which has slow hardware.

KAME does not have any garbage collection for old state-information. We proved that by stopping our attack and measuring memory consumption. It stood flat until we continued the attack. This might be fixed in later versions though.

Figure 2 shows the memory consumption of the IKE-process during 3 hours. The consumption speed steadily decreases, since the performance of our Nessus attack module decreases steadily over time. This can be confirmed by Figure 3, which shows the CPU consumption over time. There is no need to fix that bug now, because KAME was the only one that needed a longer testing period, and testing for DoS efficiency lasts only about 15 minutes.

![Figure 2: Mem usage for KAME](image)

![Figure 3: CPU usage for KAME](image)

We could speed up memory consumption by re-starting the attack script periodically, and running the script on two machines simultaneously. With this kind of limited DDoS-attack we were able to crash the IKE-process of KAME in less than 3 hours. First it used up all the virtual memory in the system. Then the IKE-process reported 'failed to get buffer' a dozen times, before it was killed either by itself or by the operating system.

Cisco

Cisco performed better than KAME, but it used up most of it's memory quickly, before it found some kind of balance, where old state-information expire at the same pace as new one is created. Thus the attack lost its power after consuming about 95 \% of total memory. After three hours we stopped the attack, and Cisco was able to recover lost memory very quickly. This can be seen from Figure 4.
The CPU could be kept busy constantly though, and there was noticeable slowdown of the UI during the attack. The CPU utilization was kept at 90-99% during the attack.

![Graph showing mem usage for Cisco](image)

**FreeSWAN and Ericsson**

Both Ericsson and FreeSWAN were unaffected memorywise by this attack, but for different reasons. Ericsson has a hardcoded memory-limit which prevents the IKE-process from consuming too much memory.

FreeSWAN on the other hand uses similar method than Cisco to reach balance, but the expiry time is set very short, so the balance will be reached early with very little memory consumption. FreeSWAN will have different results in different environments, while Ericsson should not exceed the hardcoded limit at any circumstances.

**Christmas tree results**

We used 2 different sized proposals in the testing process. The first one contained only 32 transforms. That proposal fits inside an unfragmented UDP-packet. All implementations processed that packet, but the DoS-effects were different. This version of the attack is later referred to 32 transforms.

Our second attempt was made with a proposal that contained 254 transforms. They were inside a fragmented UDP-packet. Only Cisco replied to this packet. This version of the attack is later referred to 254 transforms.

Sending either of the packets alone did not produce significantly higher resource consumption or Denial of Service than a normal IKE-proposal. That is why we chose to analyze the effects when flooding either of the packet as fast as possible.

**Effects on the system resources**

<table>
<thead>
<tr>
<th></th>
<th>KAME</th>
<th>Cisco</th>
<th>FreeSWAN</th>
<th>Ericsson</th>
</tr>
</thead>
<tbody>
<tr>
<td>IKE crashed (32 transforms)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>IKE crashed (254 transforms)</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Memory consumption</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>CPU load (32 transforms)</td>
<td>97%</td>
<td>95%</td>
<td>72%</td>
<td>74%</td>
</tr>
<tr>
<td>CPU load (254 transforms)</td>
<td>Crashed</td>
<td>96%</td>
<td>20%</td>
<td>0%</td>
</tr>
</tbody>
</table>

Table 7: Christmas Tree Results
The effects on system resources for each IKE are summarized in the Table 7. The memory consumption applies to both packets, although we could only measure it with 32 transforms, because the bigger packet crashed KAME.

**KAME**

KAME did not reply to either packet, reporting of "invalid payload length". It still suffered from resource consumption and Denial of Service.

32 transforms caused the highest CPU load on KAME, even though the hardware is multiple times faster than the Cisco. That suggests that KAME uses more processing time per packet than others. That was confirmed by checking the CPU load for IKE process in normal use. It was significant, compared to FreeSWAN and Ericsson, where the CPU-load was barely noticeable.

32 transforms also had a steady 20 Kbytes/s memory consumption effect. KAME consumed memory when we sent the same ISAKMP-packet to it multiple times when testing the Christmas Tree attack.

We tried to do the same by replaying the first packet of Cookie Crumb attack. Interestingly there was no memory consumption then.

The proposal with 254 transforms affects as delay for traffic for the first 10-20 seconds. Then the whole FreeBSD machine crashes and must be restarted completely to recover.

**Cisco**

The Cisco experienced heavy CPU consumption with both 32 transforms and 254 transforms. This reflected also as slowdown for the User Interface. We also found out that the Cisco has no reasonable hard-coded upper limit for the packet size, since it replied normally to 254 transforms, while other implementations reported an error in the payload size.

We also accidentally found out that the Cisco crashes to a single packet of 254 transforms while it is running on debug mode. That might be because of too much debug information causes a buffer overflow. It is not a problem in production environment, because debug mode should not be used there.

**FreeSWAN and Ericsson**

We tried to adapt our C-program to FreeSWAN, which required source port to be 500, and found out that it consumes less CPU than attacking with non-legitimate port number. Debug information showed that FreeSWAN answered the proposal, but we did not receive any packets at the other end though.

Something in FreeSWAN prevents it from answering to either number of transforms. It complained something about the structure of the packet, so it must have more strict requirements also in this matter.

Also Ericsson complained about the payload type with 32 transforms, but did not report anything with 254 transforms. We could still inflict significant CPU load, by attacking with the unmodified versions of our Christmas Tree Attack. As we can see from the Table 7, both implementations performed quite even, but only FreeSWAN suffered CPU load with 254 transforms. This indicates that Ericsson can filter out these big packets before they are processed in IKE, but FreeSWAN rejects the packet after it has been examined by IKE.

**Correlation to real world**

Our test results can be divided into following three categories.
Total Denial of Service happens when IKE process crashes, or the consumption of network/resources cause infinite delay. In this case a user does not usually know what went wrong. The user might try to establish connection multiple times, because he/she does not know when it is going to work again.

Partial Denial of Service happens when some packets get through, and SAs can be established with some waiting period. The user might not easily tell a difference between this and the Total Denial of Service if the waiting time is very long. In the case of network congestion is the main reason, then it will not help much to finish SA negotiations, since the usability is still not good enough in any internet application. In the case of IKE vulnerability, the user has to wait a particular waiting period before he/she can start to work with a secure connection.

Continuous Random Packet Drop: Random delays independent of IKE phases are caused only by network congestion in these results. Even a delay of 5 seconds is too much for working without frustration. If the average delay is 10 seconds or more, then many protocols like SSH does not function at all.

These results should change significantly when tested between real routers at the Internet. There asymmetric attacks, like crumb.nasl, should be able to crash KAME with memory consumption. High bandwidth attack should not be able to inflict the same CPU load, since most of the packets would be dropped by intermediate routers. Some Denial of Service effects would probably still persist, since the relative number of attack packets to legitimate packets is overwhelming.

<table>
<thead>
<tr>
<th>Overall robustness</th>
<th>KAME</th>
<th>Cisco</th>
<th>FreeSWAN</th>
<th>Ericsson</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource management</td>
<td>Poor</td>
<td>Average</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>DoS resistance (crumb.nasl)</td>
<td>Weak</td>
<td>Poor</td>
<td>Average</td>
<td>Excellent</td>
</tr>
<tr>
<td>DoS resistance (crumb.c)</td>
<td>Poor</td>
<td>Poor</td>
<td>Poor</td>
<td>Good</td>
</tr>
<tr>
<td>DoS resistance (Christmas tree)</td>
<td>Poor</td>
<td>Poor</td>
<td>Average</td>
<td>Weak</td>
</tr>
</tbody>
</table>

Table 8: Evaluation of differences

Conclusions

Denial of Service against IPsec is relatively easy to accomplish by abusing the weaknesses in IKE. Our task was to develop a testbench for DoS-attacks against IKE, in order to improve the robustness of IPsec.

The task turned into also developing a process for testing and analyzing differences between IPsec implementations. We used a publicly available tool called Nessus which had its own scripting language called NASL. The limitations of NASL discouraged the implementation of all the planned attacks.

Nessus was not as superior for automatic testing of IPsec implementations as we first expected. It is more efficient to implement the attacks as C-modules. It should also be noticed that the effects of our attacks can not be seen from the Nessus-client, but only from the target machine, or from a peer IPsec machine which communicates with the target.

The protection against DoS attacks is becomes the responsibility of the programmer. That shows in the results since the FreeBSD's KAME IPsec performs very poorly, although it otherwise obeys the standard, and interoperates with for example Ericsson IPsec which performs well against DoS.
We do not know what kind of protection mechanism the Cisco router had, but our tests showed that memory consumption almost stopped after a certain limit, which implies that there is either a hard-coded limit or some other form of protection in place.

Both the Ericsson and FreeSWAN implementations contained protection against memory consumption. They can not avoid CPU consumption attacks totally, but at least they recover quickly after the attack.

References


Intrusion detection using a pre-IDS system

Marcin Dobrucki¹ ², Jonna Sär², Teemupekka Virtanen¹

¹Telecommunication software and Multimedia Laboratory
Helsinki University of Technology
E-mail: {marcin.dobrucki, teemupekka.virtanen}@hut.fi

²Nixu Oy
E-mail: {marcin.dobrucki, jonna.sars}@nixu.com

Abstract

Despite apparent benefits of automated systems which could detect intrusions and possibly stop them in time, IDS in computer networks are not very widely deployed (Rintamäki, 1999).

We identify some of the common security breaches in corporate LANs based on reports and surveys. By isolating specific problems which account for significant part of the incidents, we look for ways to use small tools to combat them without the need to deploy full-scale NIDS. The goal is to evaluate benefits of such approach when working with highly limited resources.

Prioritization of deployment should focus on allowing correlation to minimize false positives at the detection phase and automation of human-performed tasks at the incident handling phase.

Keywords: IDS, network security tools

NETWORK SECURITY PHASES

Each area of network infrastructure implementation is assigned a budget, and unfortunately in most cases, the budget too small to cover the design wish-list. We must set priorities as to what to implement first. We must also remember that the system must be maintainable and provide some kind of return on investment, otherwise the resources which we spent on it are lost. Our prioritization should be based on risk analysis and assessment, our countermeasures, and ability to react.

We must have a clear view of what we expect from our security measures. Figure 1 suggests a very simplified view of stages in building of network security. Going from left to right, we expect the overall security level of the network to increase. The IDS is just one plausible step in this building process, and not necessarily a required one. Its implementation depends on security goals and resources.

![Figure 1: Security phases in small corporate network](image-url)
The stages in Figure 1 are hard to define rigidly, but in general we can think of them as follows: the **Bare** stage represents a point where a company leases a line and connects a network to it without implementing any security measures. We define the **Basic** phase when the company implements a FW, and possibly divides the network into separate zones such as DMZ or extranets. Beyond the **basic** stage, companies begin to look for active defenses. This typically involves active monitoring of network traffic, or HIDS. Whereas the gap between the **bare** and **basic** functionalities is quite simple to close, the deployment of an intrusion detection system is a large and difficult project. Beyond the IDS phase we have defined an **Advanced** stage, where anything from large data-mining IDS systems to verified platforms can be found. Small companies rarely have the resources (or needs) to pursue such designs.

The difficulties, costs, and maintenance of IDS systems lead to a question if there is some more cost-efficient ways to improve network security? Is it possible to implement any of the IDS functionality without an actual intrusion detection system. We introduce an idea of a **pre-IDS** phase, a set of improvements which are designed to mitigate most of the common problems related to network intrusions.

**Figure 2**: Pre-IDS-phase

Our goal is to transform the situation depicted in Figure 1, and make it more as the one shown in Figure 2. Although assigning values to security is hard (or impossible), the idea should be clear.

**PRE-IDS**

We make the following basic assumptions about the current security deployment: The firewall has been designed, configured, tested, and deployed successfully, and a policy exists which defines adjustments to rules. The network has been structured to isolate functionality, and create zones which could be considered as separate networks. The zones have been created based on functionality, host visibility, and security considerations. The currently deployed security mechanisms are maintained and supervised. This includes system log monitoring, and system updates upon patch releases from vendors.

Our capabilities are going to be limited. The strengths of the **pre-IDS** lie in isolating the events of interest with a low false-positive rate and reporting them in clear fashion. This allows a human to interpret the results quickly, and react accordingly.

Recent studies have shown that up to about 70-80% of corporate security problems originate within the companies (Seppänen, 2002). Although the detailed list depends on our security policies, some of the typical events of interest which we would like to detect are:

1. Address spoofing. Are there any packets on our network with interesting looking source address?
2. Address spoofing #2. Are there any new bindings between IP addresses and MAC addresses?
3. Has any of our computers put any of their interfaces in promiscuous mode?
4. Has any of our computers started sending or receiving more traffic than usual?
5. Has any of our computers started to listen on a port which was previously closed?
Spoofing can be used twofold for malicious purposes in LANs; either claiming an IP other than own, or the MAC address. Attackers will spoof for a variety of reasons, including masking their identity or switch trickery. The false positive rate for detected spoofs is very low, because under normal circumstances, no legitimate action should require spoofing of any kind.

Because of its relatively low false-positive ratio, and simplicity, we recommend that anti-spoofing mechanisms be present continuously, and regardless of whether a proper IDS is implemented or not.

Typically, Ethernet systems listen to traffic, and pick up frames addressed to them only. However, it is possible to put the system in promiscuous mode (as is done by the IDS sensors for instance), where all traffic is passed on to the TCP/IP stack. If on a given network, a machine goes into promiscuous mode, it is a strong indication of abnormal activity (though not necessarily malicious).

There are twofold problems with promiscuous mode detection: it is somewhat unreliable, and it is post-fact. If a hacker has put a system into promiscuous mode to sniff network traffic, an unknown time has passed since the system was actually compromised.

We recommend that anti-sniffing checks be run at least once in a while, perhaps in correlation with a vulnerability scan, and not as permanent deployment.

On typical networks, it should be possible to establish long-term statistical models for traffic (Focus, 2002a). Unexplainable variations to either netflows, load, or latency might indicate that some system is experiencing some abnormal activity.

Typically, in networked environments, systems will have open ports, with some services listening on them. It is unusual, however, to have service ports open up without interference from the system administrator, and may indicate that some backdoor has been set up by an intruder. By periodic scanning for open service ports, it might be possible to determine if any machine is running extra services, possibly malicious.

As with the sniffer issue above, scanning for ports is a post-event activity. If we find open ports on systems which should not have them, the intruder is already in control (possibly partial) of the system. Because port scanning does not provide an audit trail of activity, we can only guess as to the period during which the intrusion occurred if our scans are done at periodic intervals.

EVALUATING PRE-IDS

Whenever possible, evaluations have been based on real network installations run by experienced system administrators. We believe this approach gives better results than simulated tests in laboratory environments. Feasibility of several tools have been tested in simulated conditions in a small laboratory network.

ARPWATCH

ARPWatch is an Ethernet monitoring program for keeping track of Ethernet/IPaddress pairings. Its main limitation as a security tool is that it must be run locally within the Ethernet segment.

RUNNING ARPWATCH

ARPWatch has very low requirements in terms of processing power, storage, and configuration. It is also fairly easy to install and use. The following examples have been taken from a network where ARPWatch is run on a Netra X1 under Solaris 8. The server is connected to a switched network, and
the switch port runs in normal mode meaning that ARPWatch is only listening to traffic directed to the server, and ARP broadcasts. Whenever an ARP event occurs, a mail message is generated and sent to the system administrators. There are twofold events:

1. An unknown hardware address has been detected: new station joined the network
2. The IP/Ethernet address pair has changed: DHCP assigned new address, network card has changed, etc.

On the network in question ARPWatch generated 162 event alerts during about a 7 month period. Of course, initially, when the address database is empty, the number of mails will be quite high, but as all the machines get recorded, we should only receive notification of real events.

**EVALUATION**

Overall, the administrators reported ARPWatch as a very useful tool for monitoring hardware-address changes in the local network. However, use was more oriented towards network debugging than security.

No formal policy or procedures were instantiated for this tool. Overall effort of deployment was about one day which included creation of the switch query-scripts and some minor testing. Administrators also commented that since the presence of attacker machine on the network probably meant that they have somehow forced the physical defenses, ARPWatch does not really qualify in any way as a simple pre-IDS solution.

Possible applications in other environments include misuse detection. Malicious insiders may try to masquerade their identity by spoofing their IP address. Based on evidence, ARPWatch would provide clean audit information of such event, and in the above setup, even pinpoint the physical location from which the event occurred.

**MRTG & CRICKET**

The Multi Router Traffic Grapher (MRTG) 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. It works by communicating, with SNMP, with network switches, and extracts traffic information from them. MRTG works on a multitude of platforms including popular UNIXES and WindowsNT. It has been widely deployed to monitor traffic loads on networks, and system administrators are well familiar with it. Cricket is a tool that lets users visualize a set of measurements over time (Allen, 1999). It was designed to help solve problems in using MRTG in large infrastructures, but essentially, it does very much the same thing.

**REQUIREMENTS AND INSTALLATION**

MRTG and Cricket are rather low in their requirements. For beginner users, Cricket recommends only using Cisco routers, series 2500, 3600, 7200 and 7500. Guided installation tours are available for both, so deployment should not cause problems.

**RUNNING STATISTICAL TOOLS**

We have evaluated installation of statistical tools in two environments. Our first environment is a university laboratory where MRTG and Cricket are used side by side with Snort. The installation was part of the network design phase, and allows tracking traffic from the network and individual hosts.
The laboratory network runs 100Mbps Ethernet, with backbone running at 1Gbps. Network traffic monitoring has shown that typical network load is around 10% despite NFS mounts. Administrators have reported that clearly visible traffic profiles of daily usage and weekends can be observed.

Both MRTG and Cricket are used as a correlating help for detection from other sources. A number of incidents in the laboratory network has been investigated by monitoring traffic profiles from suspected hosts. Results have been positive in terms of accuracy, but required a certain delay in order to observe changes in traffic patterns.

A second scenario includes a much broader infrastructure, a network which connects Italian universities and public libraries. MRTG is used along with a number of custom wrappers which analyze traffic flows in network routers. The unique-ness of this installation is that it is the primary method of intrusion detection, and it is limited to processing traffic from a C-network block. Upon the detection of some anomaly, the administrators start a closer investigation.

The system is run by three persons: a system administrator who maintains the network, and two intrusion detection analysts. A rough estimate of 25-30% of daily time is spent browsing through traffic profiles looking for signs of anomalies.

A simple incident response procedure is installed, with pre-written report templates. Typically, the forensic, analysis, and reporting stage takes about 3-4 days for one of the analysts.

EVALUATION

In both cases, using statistical reporting tools have proved easy, beneficial, and applicable to intrusion detection. However, in both cases the systems are used in a different manner. We have hence deducted that statistical analysis is useful not only as a pre-IDS deployment, but also plays a part in advanced IDS. The trend seems to be that as network size and complexity grows, administrators abandon signature-based IDS systems and focus on statistical profiles, too.

We should note however that statistical tools such as MRTG or Cricket are only useful in tracking 'loud' incidents. This is typical of script kiddies. The resolution and accuracy of the tools is not good enough to help detect low and slow attacks which an experienced attacker might carry out.

SNIFTER DETECTION

A number of tools have been developed for determining if a network interface is running in promiscuous mode. We have selected antisniff and sentinel which are publicly available free of charge.

Requirements and installation

Installation of both software on the test machine was quite trivial, and involved running the installation commands for an OpenBSD port. The hardware requirements seemed equally low, however for future use, the number of bpf in the kernel could be increased for antisniff.

As per documentation the software is unable to detect special-purpose probes or hardware sniffers, and will most likely only work against typical computers which have interfaces in promiscuous mode. We consider this sufficient for our purposes, as our goal is to detect subversion of intranet systems which have been made to work as intelligence gathering stations.
For these tests, we built a simplistic test network as portrayed in Figure 3. The network comprised of four computers running various operating systems. They were connected through a 10Mbps hub, and no outside connections were provided.

All of the machines run several services such as web or mail servers, however we purposefully did not include any traffic, as these kinds of tests would normally be run outside office hours in production networks.

![Sniffer detection test network](image)

**Figure 3** Sniffer detection test network

**Running sniffer detection tools**

*antisniff* *Antisniff* has six different modes it uses to test for presence of promiscuous interfaces. Surprisingly, the most accurate results in this network were obtained with the *NT EtherPing* option, however some *ICMP time delta* test has also given promising results. One of the big cons of *antisniff* was that it only takes a single IP address as argument for running the tests, and hence is not really usable for sweeping networks.

Since L0pth (authors of *antisniff*) has become Security Software Technologies (Softwaretech), *antisniff* has been ported to Microsoft Windows and received a nice graphical user interface to aid in the detection of sniffers. The Windows version of *antisniff* has not been tested in this study.

*Sentinel* Sentinel offered three different modes for detection, two of which we used for testing: ARP test, and ICMP echo test. The options for providing a range of addresses to be checked are also very flexible ranging from a single address, to a list of addresses listed in a file, to scanning an entire C-class network.

The ARP -test detected two promiscuous mode machines 0.20 and 0.40 (see Figure 4). The ICMP echo reply test narrowed this to 0.40 only. Altogether 11 IP addresses were used in the test, four of which actually existed on the network.

Sentinel worked quite fast, using about five to ten seconds per IP. This allows scanning of even larger networks in a reasonable amount of time.

**EVALUATION**

*Sentinel* could well be used to detect sniffing systems on the network. However, some kind of tool to correlate the results from different tests would probably greatly improve the accuracy of results. *Antisniff* version tested during these experiments is rather a curiosity tool that administrators might want to run against a single host which is under a suspicion of sniffing.

Based on these short test runs, we could definitely add promiscuous mode detectors to a pre-IDS toolbox. Although the detection of compromised systems would likely occur well after the actual intrusion, the ease of running and interpreting the results make this a viable method.
Evaluating NIDS – Snort

In this section we look at what full-scale NIDS have to offer. We have selected Snort is highly popular, and available for a multitude of platforms with one of the largest, and most enthusiastically maintained rules databases. A good indication of how much Snort influences current signature-based NIDS development is that Snort and Dragon rules are used by other vendors to update their systems (Focus, 2002b).

Requirements

Snort runs on a variety of platforms, including most UNIXES, Windows, and others. The requirements are mostly dictated by the following three factors:

1. The speed of the network, typically 100Mbps
2. The number of signatures
3. Event history

The first two dictate requirements for the network interfaces and processor speed. The third one dictates speed and size of the storage the system must have available. In single-sensor environments, or even in dual-sensor scenarios, it is typically sufficient to utilize a modern PC with sufficient disk space to store several days worth of logs.

Our tests were run on two systems: a Pentium MMX laptop, and a Pentium III desktop machine. The operating system used in both cases was OpenBSD (3.0 and 3.1).

Installation

Installation in itself is not difficult. In most instances it amounts to either installing a precompiled package, or running the make installation. The difficult part is tuning the system for a particular network and purpose.

A rough estimate for installation and configuration of the underlying operating platform and Snort (in single-sensor/console configuration) is two working days. In our analysis we used two Snort installations. The first is a trial-run of an intrusion detection system in a small company. Snort is installed on an OpenBSD system and combines the console. SnortSnarf is used to parse logs which are saved and maintained on the same system. The second installation is in a larger, university network where Snort runs in proxy mode between the network and the firewall.

Running Snort

The first step after installation was to determine what typically happens on the network. This helps eliminate small glitches which might have previously gone unnoticed, and which might be causing some of the rules to trigger alerts.

In scenario number one, Snort logs are inspected about once a month by an experienced security expert. Since the network is thought to have strong perimeter defenses and an established usage policy, the threat of penetration is seen as very low. Snort is used mostly to keep a record in case an intrusion is noticed by other means.

No automation has been built into the system. There is also only low degree of help expected from the system in case an intrusion is noticed. The logging facility records IPs and port numbers, but little more information about events. This allows the system to run unattended for long periods of time between inspections.

This trial run has been running for about six months at the time of this writing, and no intrusions were noticed or inspected using it. No reports were presented to management, and no incident handling plan has been deployed.
In the second scenario, Snort is used more actively to monitor activities on the network. It has been configured as a proxy, with all traffic passing through it to allow the administrators to see intrusions from the outside, and misuse by legitimate users on the network.

Very little trimming has been done to the Snort rule set as the university network allows a good degree of freedom for people to install software, services, and machines.

Because in this configuration, the system is rather large, a statistical tool, Cricket, is used to double-check suspicious activity which triggers alarms.

Two part-time administrators monitor the system on daily basis. The intention is to find problems and patch holes. There is no intent to pursue the guilty parties beyond a notice to their service providers or equivalent.

EVALUATION

If used actively, and possibly by correlating data with other systems, a full-blown NIDS provides most of the information that all the other tools we presented do and some that the other systems can't provide. The information is also much broader, noting intrusions 'as they happen', not after-the-fact detection. However, as pointed by the contrast in required supervision work, an IDS requires substantial human resources to provide benefits.

IDS might also be considerably more useful in environments where the perimeter defenses either are not very strong, or cannot be made tight enough to provide a high level of security.

CONCLUSIONS

We have identified some common causes of problems in corporate networks, and suggested a pre-IDS security measures which could be used to combat a number of those issues.

The results from the pre-IDS stage evaluation are mixed. Some small tools are well suited to perform a certain amount of intrusion detection, while others should be seen more as providing network maturity which allows efficient deployment of NIDS. Tools such as ARPWatch and sentinel are useful in monitoring behavior of computers behind the security perimeter.

Finally, intrusion detection systems are not an answer to all network security problems. They require a certain level of maturity, and are only effective if monitored and maintained. Too often, existing facilities are not used to full potential, and the faults are being covered by IDS deployment.

REFERENCES

Allen J.R. (1999), Driving by the rear window: Managing a network with cricket, In 1st Conference on Network Administration, WebTV Networks, April 1999


Seppänen R. (2002), Tietoturvaan syydetään rahaa heikoin tuloksin, Itviikko, Research by the Economic Intelligence Unit, Finland, 2002 (A lot of money is spent for information security without results, in Finnish)

Focus (2002b), Various. Focus-ids mailing list, May 2002

Softwaretech, http://www.securitysoftwaretech.com
The Real Face Of War

Mark Hill

School of Media and Communications
University of New South Wales, Australia
Email: mshill@rpi.net.au

ABSTRACT

Terrorism has again resurfaced into the spotlight, and while terrorism may proliferate into different technological forms, such as chemical, biological, nuclear and cyber-terrorism, an emphasis on the technological basis of terrorism and on technological solutions, may in fact obscure the real threat of terrorism, that of its ability to adapt and transform the everyday into objects of fear. This is hampered by a lack of definition, and threatened by the media's visual and dramatic orientation of events. Warfare is characterised by both technology and information, and a particular bias towards one in preference to the other may in fact be a weakness rather than a strength.

Keywords: Terrorism, Information Warfare, Media

Terrorism is now a ubiquitous term, but its sheer revival and mass dissemination since the events of September 11 2001, doesn't necessarily equate with a global understanding or a clear definition. Writers such as Walter Lacqueur, Bruce Hoffman and Noam Chomsky have all stressed the need to address the fundamental issue of defining what Terrorism is and who Terrorists are? But as Robert Keohane points out this is as much about political power as it is pragmatic, where much of the 'activity in world politics becomes a competition for specifying what the norm means' (Keohane, R.2002.p142). Yet legislation is being pushed and passed in a number of countries, in response to September 11, and more obviously, a 'War on Terrorism' has been declared by the United States. Though this is not to say that some definition nor understanding of these events and their implications does not exist, as Caleb Carr notes 'the first year of the twenty first century produced images that will likely identify the decade, if not the generation, to come' (Carr, C.2002. p3). Images that, one imagines, will recur with frequency given U.S. President George Bush's declaration that the war, 'will not end until every terrorist group of global reach has been found, stopped, and defeated' (US Dept of State. 2002). Such rhetoric seems problematic given not only the apparent poverty of the term Terrorism itself, but more so because such an ill defined term seems to be promiscuously disseminating itself technologically, seeding it's own sub classifiers, so that we now have not only terrorism but biological, chemical, nuclear and cyber-terrorism. The last leading to Dick Clarke (Special Advisor to the U.S. President for Cyber Security) to not only declare that "Our economy, our national defence, increasingly our very way of life, depends upon the operation, secure and safe operation of critical infrastructures, that in turn depend on cyberspace" but more ominously that 'America has built cyberspace, and America must now defend its cyberspace' (Clarke, D.2001). Given then this extension of Terrorism and its implications, not only across time, but also across space and place, it seems not only necessary but prudent, to discover what its broad features may be in these foundational stages of a long war on terrorism.

War has always involved two key elements, technology and information, but the blend is neither fixed nor universal. Technologically 'the apex of industrial age warfare would typically pit two or more nation states against one another, each equipped with roughly equivalent technology, at least in the strategic sense.....from 1800 through 1945, it was virtually impossible to disadvantage the enemy through means other than physical dislocation of his ground forces through manoeuvre' (Leonhard, R.
but as Professor of military theory James J Schneider notes 'warfare constantly evolves and reconstructs itself' (Leonhard, R. 1998. pix), thus while war may have until recently, had a sense of symmetry to it, it must be remembered that this is an apparent symmetry, one based upon a particular cultural interpretation of war, namely that of a western one. Given that war can be rather diverse, it is necessary to keep in mind Jeremy Black's observation that 'if wars are different, and military tasking, therefore, very varied, then it is unhelpful to think in terms of a single hierarchy of military capability, however that hierarchy is arranged, or to argue in terms of a narrow range of military characteristics'. (Black, J.2001. p6). This seems of more importance once we move into the modern period, where after this apex, came the global cold war of potentials never realized but always feared. A period that also saw the emergence of the Revolution in Military Affairs and its concomitant swing towards high tech war. But things are changing, as Virilio noted in 1993, after the first bombing of the World Trade Centre, 'after the age of the balance of terror, which lasted some forty years, the age of imbalance is upon us' (Virilio, P. 2000. p19). The Symmetry of war begins to become replaced by an asymmetric advantage. An advantage that the Revolution in Military Affairs believes lies in continuing to push towards a technological edge. But as Virilio further noted 'it has reached the point where soon, if we don't look out, a single man may well be able to bring about disasters that were once, not long ago, the province of a naval or air force squadron' leading to 'a previously unimaginable equation: one man = total war'.

Given then the constant images of war in the media, this then seems one of the most dominate faces of the war on terrorism, a technological interpretation. One made explicit in America by the passing of the USA Patriot Act 2001, whose full name, Uniting and Strengthening America by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism, highlights the bias. There seems thus a problem with understanding not only the act of war but also the field of play, a problem in recognising that the shift from symmetry to asymmetry suggests that control over the field of play and the rules of engagement cannot be total, no matter how hi-tech the angle. Wars can use technology but war is never purely technological.

Terrorism has a long history of interest in technology, Terrorism was, as Laqueur notes 'the application of modern science to the revolutionary struggle' (Laqueur, W.2002. p36). In London in the 1880s Anarchists were already advocating that attention should be paid, to chemistry and technology. One of the first successful uses of a letter Bomb (long before the Unabomber made it a signature calling card) was the killing of a Hungarian vicar general by a group of Romanian terrorists, prior to WWI (Laqueur, W.2002. p94). Also the first recorded hijacking of a plane was done in 1931 in Peru during a Military coup (Laqueur, W. 2002. p108). Unlike the American urge for a technological war of dominance, terrorist use of technology tends to utilises its most salient characteristic of dispersion. It is then thus not surprising that terrorism adapts to technological change, moving from biological to chemical to cyber-terrorism, before an endgame of nuclear-terrorism. What then tends to be the advantage is this, militaristically 'the goal of obtaining an advantage over the enemy can be served by various means. The means will change from war to war, or even from day to day, but the underlying concept of advantage remains' (Leonhard, R. 1998. p59). For terrorism that advantage is in change itself, in the ability to transform, in the ability to adapt the everyday into a weapon through imbuing it with fear.

This is most clearly evident in the events of September 11 and its after-effects, take for instance the transformation of the individual in society. Before September 11, Charles Jonscher wrote 'if you could tour the twin towers of the world trade centre when they were deserted of human life, in the early hours of the morning, you would be able to count a million dollars worth – more or less – of digital electrical equipment on each floor. But if you chanced upon a security guard you would confront, in that single person, more processing power than is present in the electronics wired into all those 110 storeys' (Jonscher, C. 1999. p29). But it is this potential we now fear, veneration turns to vilification, Virilio's one man=total war, and becomes personified in the form of Osama Bin Laden. Such simplicity while politically pragmatic ends up disseminating and disrupting one of the fundamental networks of society, the network of trust. And nowhere is that more evident then in the transformation
of the internet and global telecommunications structures as the backbone of globalisation and liberalism capitalism, into the backbone of global disruption, into a haven for terrorism.

Earlier it was mention that there are two facets to any war, technology and information, and in looking at terrorism, what becomes apparent is that while western society, in particular the USA has opted towards a techno centric stance, terrorism on the other hand, tends towards an info centric view. The attacks of September 11 seem to work so well because of a dominance of dispersive technologies in the west, and a dominance of its media towards the visual and the dramatic. It is why the images of that day will mark a generation and mark the outlines of the newly unfolding century. This is especially evident in the transformation by terrorism of the internet, already a highly mistrusted technology. Mistrusted precisely because it lacks boundaries and borders, mistrusted because it lacks a recognisable face.

Fear thrives on the invisible, the launch of Sputnik by the Russians in 1957, helped solidify the Cold War, which later provoked the American Military to develop the foundation of the internet. But it must be remembered that back then, few people had ever seen, let alone touched, a computer (Levy, S. 1994. p18), in 1971 the Arpanet linked just 23 host computers in only a dozen or so cities in the US and yet by December 1995 about 16 million people were using the internet (jonscher, C. 1999. p160), by February 2002, 544.2 million, approximately 8.96% of the world's population (Thornton , S. 2002).” An uptake helped along significantly in 1993 by Marc Andreeson’s releasing of the internet browser (which again highlights the preponderance and thus the susceptibility of Western culture towards the visual). While John Perry Barlow may have described the new age pre September 11 as the digital age, the age of a universal network with no one in charge, no president, no chief (Virilio, P. 2000. p8) the initial celebration of such unchallenged openness now seems naïve, and forever compromised. While the promise is one of use, the problem is one of the user. Osama Bin Laden and his terrorist network, may have been inhabiting a series of caves in Afghanistan, but it should be remembered that they were ‘complexes equipped with an extensive range of modern communications systems, including HF and VHF radios, telephone and fax connections, and computers for emails and links to websites and electronic bulletin boards’ (Ball, D. 2002. p63).

Americas response to such use has been to open the DARPA Information Awareness Office, with program titles such as the Total Information Awareness program, again seem to stress full spectrum dominance through technology, and yet in contrast, even though Osama Bin Laden used a satellite phone from 1996 onwards, in 1999, he ceased using it, some speculating due to his awareness of its insecurity (Ball, D. 2002. p63). It could thus be argued that the ‘war on terrorism’ is similar then to submarine warfare, where ‘the actual destruction of the enemy is in some ways anticlimactic. The real battle is about detection’ (Leonhard, R. 1998. p71). The real battle, as such is invisible, a battle more dependent upon organisation and information about how terrorism is organised and organises.

It thus continues a trend already evident last century, for while the Twentieth Century was called by Eric Hobsbawn an age of Total War (Hobsbawn, E.1994), it was also according to Whitaker the Century of Intelligence, ‘The systematic and purposeful acquisition, sorting, retrieval, analysis, interpretation, and protection of information’ where spying had ‘become a systematically organized bureaucratic activity with its own specialized institutional structure, its own technologies, its own scientific knowledge base, and its own semi autonomous role in global politics’ (Whitaker, R.2001. p5). Such a recognition of the role of information gathering and its importance not only to the private but also the public, should be evident, in not only the fact that the search engine Google was so overwhelmed by the amount of people using it on September 11 and in the days afterwards that it had to restructure its software and presentation format (Wiggins, R. 2001), but also by the public feeling that there was a breakdown in the Intelligence community, a failure to inform. A problem not helped by increased surveillance, and an increased awareness of the failure of surveillance technology to live up to its promise, something already evident in the publicized failures in the intelligence community’s surveillance and search engine DCS1000, which was until recently named Carnivore (Olsen, S. 2002). Compounding such technological problems are also interpretational problems, where such mistakes could be labeled evidence of hacking, but could just as easily be software failures, on the internet this is often impossible to distinguish between (Schwartau, W.2000. p19).
All of this though seems more problematic for the defense against terrorism than for terrorism itself. As has been noted, terrorism seems to adapt to technologies, rather than remain dependent upon them. Even though Osama Bin Laden used the internet, he also used human couriers and hand delivered instructions (Ball, D. 2002 p63). It should also be remembered that, while the internet seems revolutionary, the amount of users still number less than 10% of the world's population, and that 'presently three billion people – nearly half the world's population – live on less than $2 a day' (Thornton, S. 2002). For most people the closest they get to technology is the television, which for most is if not the real, than the only face of war, and one that terrorism seems all to aware of.

The media as Leonhard notes ‘Although they can and should influence the course of a war, they cannot and should not influence the course of a battle’ (Leonhard, R. 1998. p24) because ‘when viewed conceptually, the media no longer simply report military activity, but rather participate in it’. As an example, he cites the gulf war, still a source of debate. Yet ‘during the actual conduct of the war, the media’s coverage of the Scud attacks and the Patriot defences gave the appearance that the Patriot could effectively nullify scud attacks. This appearance arguably was more important than the actual performance of the Patriot, because the media’s reports left impressions on the leaders of Iraq, Israel, and the coalition states’ (Leonhard, R. 1998. p31). Such sensitivity to the impact of information, perhaps helps explain why Governments have started to white-wash the internet, for fear not only of terrorists accessing technical documents, but also uncertainty over what may potentially be used against them, what impressions may be construed or misconstrued. As Michael Ignatieff pointed out in regards to the Kosovo Crisis, ‘war does not become illegitimate simply because citizens see carnage on their screens. It becomes illegitimate when the political reasons for it no longer convince’ (Ignatieff, M.2000. p187). The targets of terrorism are often symbolic, and while September 11 2001, presently holds the world fascinated, we should not forget September 1972, that thus while some writers like Carr and Laqueur may indicate the pointlessness of terrorism politically, most will see it sensationally, noting an event thirty years ago ‘The Olympiad arouses the people's interest and attention more than anything else in the world. The choice of the Olympics, from the purely propagandistic viewpoint, was 100 percent successful. It was like painting the name of Palestine on a mountain that can be seen from the four corners of the earth.’ (Reeve, S. 2000. p207)

War has always been about advantage rather than destruction. Technology and information while components of war, by their evolving nature will change the face of war, advancement in one does not necessarily guaranteed success over the other. There is a dynamic relationship at work here, whether symmetric or asymmetric, it should be noted ‘“better” technologies are not always immediately self-evident’ and ‘“better” technologies do not always win wars’ (Neiberg, M. 2001. p187). Terrorism is a war that may appear technological, whether it uses chemical, biological, or nuclear weapons, or even cyberspace, but this is not to confuse it with techno war. Thus while the face of war apparent to most, is the one that our television sets will show us, and which the media will report, which initially will seem technological, this is not without its dangers. The visible weapons of war; fighter jets, bombers, battleships, missiles, bombs and explosions are examples of a wish to technologically totalize war but they are not the totality of war, nor are they economically feasible if a war is to be a long war. As the US Military acknowledges ‘It is difficult to counter the threat that terrorists pose. Currently, terrorists are able to move freely throughout the world, to hide when necessary, to find unpunished sponsorship and support, to operate in small, independent cells, and to strike infrequently” (DARPA, 2002). The most powerful weapon they use then, is the same as their most powerful form of defence, that of a community. What differentiates communities one from the one is information and trust, which if manipulated or abused, can easily shift to mis-information and fear. Thus while this is a war begun with vague definitions, we must be careful how it is redefined, it should be remembered that the face of war alters with time, ‘In World War I only 5% of all casualties were civilian; in World War II that number was 50%; and in conflicts through the 1990s, civilians constituted up to 90% or more of those killed, with a high proportion being women and children’(Chesterman, S. 2001. p2), the face of war is thus becoming a face increasingly hard to forget.
References:


Carr, Caleb. 2002. The lessons of terror; a history of warfare against civilians: why it has always failed and why it will fail again. Little Brown. London. UK


Will New Laws be effective in reducing Web sponsorship of Terrorist groups?

M Warren¹ and W Hutchinson²

¹School of Information Technology, Deakin University, Geelong, Australia. Email: mwarren@deakin.edu.au

²School of Computer and Information Science Edith Cowan University, Mount Lawley, Western Australia Email: w.hutchinson@ecu.edu.au

ABSTRACT

The paper investigates the use of the Internet by terrorist and dissident group for publicity, propaganda, and fund raising. It examines the new anti-terrorism legislation passed in the last few years (especially the UK Terrorism Act), and its impact in the Internet presence of proscribed groups.

Keywords: Terrorism, Law, UK Terrorism Act, propaganda.

INTRODUCTION

In the past, terrorist and dissident groups had a lot of trouble conveying political messages across to their supporters without being censored in some way. The medium was by word of mouth, clandestine posters, violent 'events' and so on. Fund raising usually had to be done locally on a personal basis. Now the Internet can be used to 'spread the word' and collect funds globally.

The term 'terrorist' is normally used to denote revolutionaries who seek to use terror systematically to further their views or to govern a particular area (Wilkinson, 1976), although with the new terrorism legislation the term has been broadened (see below).

THE CURRENT SITUATION

We are now facing a situation where terrorist groups are now developing Internet sites and using Internet technologies. The areas where terrorist/resistance groups are using the Internet are:

PROPAGANDA/PUBLICITY

Terrorist groups have difficulty in relaying their political messages to the general public without being censored, they can now use the Internet for this purpose. Different terrorist groups/political parties are now using the Internet for a variety of different purposes. Some examples are (Coggle & Warren, 2002):
**Tupac Amaru Revolutionary Movement (MRTA)**

In 1997 a Peruvian terrorist group known as MRTA took over the Japanese embassy in Peru, taking a number of hostages. During this time, the Web Site of the MRTA contained messages from MRTA members inside the embassy as well as updates and pictures of the drama as it happened (Regan, 1999).

**Chechen Rebels**

Chechen Rebels have been using the Internet to fight the Russians in a propaganda war. The Rebels claimed to have shot down a Russian fighter jet, a claim refuted by the Russians until a picture of the downed jet was shown on (Kavkaz.org), the official Web site of the Chechen Rebels. The Russians were forced to admit their jet had in fact been shot down (Krushelnycky, 2000).

**FUNDRAISING**

Azzam Publications, based in London and named after Sheikh Abdullah Azzam, a mentor of Osama bin Laden; (VPC, 2001) is a site dedicated to Jihad around the world.

It is alleged that the Azzam Publications site, which sells Jihad related material from books to videos, was raising funds for the Taliban in Afghanistan and for allied guerrillas fighting the Russians in Chechnya (Miami Herald, 2001).

![Figure 1: Example of Azzam Multi Language Sites](image)
After September 11th, Azzaro Publications came under increased pressure to the point where its products could no longer be purchased through their site. In a farewell message published on their site, they provide alternatives to ensure that funds can still be raised and sent around the world to fight the 'struggle'. One such way suggested on the site is the Hawala system. This system involves no trail, no bank accounts, no way for money to be traced. It is based on a handshake and a code word (Frantz, 2001).

In 2002 the main Azzaro site went back on-line, offering the same fundraising options. The new site also mirrored itself around the world and provides its content in a number of languages including: Arabic, English, German, Spanish, Indonesian, Bosnian, Turkish, Malay, Albanian, Ukrainian, French, Swedish, Dutch, Italian, Urdu and Somali (see figure 1). The reason for doing according to the Azzaro site “is to protect against Western Censorship Laws”. It will probably prove to be difficult to close the Azzaro site in the future, when the information is mirrored around the Internet in a variety of languages.

NEW LAWS AND THE CONTROL OF TERRORIST USE OF THE INTERNET

Over the last two years a number of countries, such as the US, UK, Australia, and Canada, have passed and implemented a number of anti-terrorist laws. These laws have an impact in most Western countries and have the potential to severely restrict activities such as funds collection, propaganda, and command and control.

Taking the UK Terrorism Act 2000 (HMSO, 2000) as an example, Section 1 of the Act is quite specific about what constitutes 'terrorism'. Basically, it includes any use or threat of action, which involves violence, damage to property, threatens life or health, or is designed to seriously interfere with or disrupt an electronic system (Section 1(2)(c)).

Many powers contained within the Act are also related to the definition of ‘terrorism’. The investigation powers contained in Part IV (sections 32-39 inclusive) require, in some form, an act of terrorism. Under the ‘investigation powers’ the police are empowered to search, freeze assets, and prevent fund raising.

The Act refers to action, people, and governments inside and outside the United Kingdom. Therefore the Act makes any intended and serious disruption to a computer system an act of terrorism unless it had criminal intent. Strangely, it makes hacking a terrorist organisation's site anywhere in the world from anywhere in the world a 'terrorist act'. So if an individual with a grudge against the Klu Klux Klan, seriously disrupted that organisation's Web site from India, it would be an act of terrorism. Presumably, the government of the United Kingdom could then apply for extradition of the perpetrator, if it so desired. Sections 59 and 60 go into more detail in this area and state that inciting acts of terrorism in overseas states is illegal, even if the incitement is done outside the United Kingdom. Thus a Kurd in the USA inciting actions against Turkey is committing an offence. There seems to be a distinct division between states and 'other' groups. For instance, it is questionable whether the law would be equally applied to both sides in the present the Palestine/Israel situation. The Act exempts perpetrators of 'terrorist acts' if they work for the United Kingdom government. It is also interesting to ponder if a state (such as Iraq) could be classified as a 'rogue' or 'terrorist organisation', and therefore liable under the Act.

The effect of being a proscribed organisation under this Act is profound indeed in that, without wrong doing having been proved, the members, organisers and supporters of the organisation are in effect subject to criminal sanctions by the actions of the Home Secretary. Such things as membership support, convening meetings, addressing meetings of such organisations lead to the potential of 10 years imprisonment. The disturbing aspect of the above is that bodies who are at the other end of the
spectrum from armed and aggressive organisations could be the subject of draconian legislative provisions. Therefore anything aiding these organisations is a 'terrorist' act.

Section 3 of the Act expands those 'concerned' with terrorism to those that 'promote' or 'encourage' terrorism. So a Web page that supports the Irish Republican Army (IRA) is committing an offence. The Act can also be interpreted to include Web addresses, so the domain name of 'IRA' could be seen as encouraging and supporting terrorism. This might come as a surprise to the International Reading Association and The Internet Retirement Alliance who actually do use this name.

Sections 12 and 13 of the Act indicate that inviting, or indicating any support for proscribed organisations is illegal. Any Web site that uses a slogan for support of these organisations is committing an offence. So the casual phrase "Up the IRA" might have legal implications (in fact, just writing this slogan in this paper might also have implications under this legislation). Section 15 bars the collection or the provision for the collection of funds for a proscribed organisation. Section 23(5)(b) indicates that any monies collected for this purpose must be paid back. Of course, these latter points have implications for Web designers, but the Act goes further and could have ramifications for ISPs. Section 17 reinforces the point that ISPs should not allow the collection of monies and must inform the authorities if there is any reasonable cause to suspect a site is being used for this purpose. Section 19 adds to this, and states that ISPs and their employees must disclose any suspicions as they come to their attention.

In fact, any information that might assist a terrorist act is covered (Section 58). This principle is extended to communication and e-mail. Using e-mail to control a terrorist group is illegal. As the definition of terrorist is so broad, many activist groups could be caught by this Act. It is also of concern to any organisation. Staff using e-mail for these types of deeds could well catch their employers into aiding and abetting terrorism under the Act.

Similar legislation is mirrored in many Western countries. Of course, the draconian UK Act has yet to be tested, and its ability to enforce it over various jurisdictions is problematic, but it does show a trend toward a restriction of terrorist activities in cyberspace. It is yet to be seen whether this trend outweighs that of expanding Internet use by terrorist groups. However, the Act is indicative of two trends. The internationalisation of national laws, and the intrusion of government into cyberspace.

**RECENT DEVELOPMENTS**

Since September 11th, there has been a shift towards global co-operation and the internationalization of the law in terms of its application. An indication of this can be found in the following examples:

Post September 11th cooperation between normally hostile states has resulted in law enforcements officers operating in a global role. For example in August 2002, Pakistani and FBI security officials arrested 12 suspected Muslim militants in the North-West Frontier Province of Pakistan suspected of planning future terrorist attacks (BBC, 2002a).

- Non USA citizens fighting for the Taliban in Afghanistan were arrested, deported to Cuba and held under US Military Law. As these included three UK citizens, Labour backbenchers have called on the UK government to pressure the US to treat these men as prisoners of war (PoW) and observe the Geneva Convention (BBC, 2002b). Under the Geneva Convention, PoWs must be tried by the same courts and under the same procedures as US soldiers. That would mean war crimes trials through courts-martial or civilian courts, not in closed military tribunals (BBC, 2002c).
The Australian Federal Court (September, 2002) ordered Frederick Toben to remove material that describes the Nazi Holocaust as a hoax. The federal court found that the website contravened Australian Federal anti-racial vilification laws (ABC, 2002). Self-confessed racist Van Tongerean was the leader of the Australian Nationalist movement (ANM). The ANM was behind the fire-bombing of five Chinese restaurants in the late 1980s (The West Australian, 2002a). Within 24 hours of his release from prison (September, 2002), the ANM website was rehosted in the USA in order to be immune to Australian federal laws. The group's website is hosted by a Utah-based company and is contained within the White Pride Coalition Australia site (The West Australian, 2002b).

This international cooperation could see legislative national jurisdictions disappear in relation to 'terrorists' groups. September 11th has spawned some unlikely allies in this area: Pakistan and the US, China and the US, and so on. The longevity of this mutual desire to rid the world of dissident and terrorist groups (often those who opposed an entrenched government) is yet to be tested. One wonders the support that would be given to a nation desiring to suppress a Christian group from the Western allies.

CONCLUSION

With the introduction of new laws and attitudes are we seeing the erosion of civil liberties in Western societies by terrorism legislation. The fear of ISPs and employers of prosecution causing strict Internet usage, the filtering of certain sites (even at a national level (e.g. Singapore), means that censorship and control can easily be enforced. Are we seeing a situation that it will be much more difficult for terrorist groups to get their message to a global audience and stop their ability to fund their activities through online collection of funds?

REFERENCES

URL: http://www.abc.net.au/rn/talks/8.30/lawrpt/stories/s683070.htm
Accessed 5/10/02

BBC (2002a) FBI helps Pakistan arrest 'Militants',
URL: http://news.bbc.co.uk/1/hi/world/south_asia/2221229.stm
Accessed 5/10/02

BBC (2002b) UK presses to see Cuba captives,
URL: http://news.bbc.co.uk/1/hi/uk_politics/1763056.stm
Accessed 5/10/02

BBC (2002c) Cuba captives 'should be PoWs',
URL: http://news.bbc.co.uk/1/hi/uk_politics/1776961.stm
Accessed 5/10/02


URL: http://www.rferl.org/nca/features/2000/05/F.RU.000511130424.html,


The West Australian (2002a). I won't back off, says defiant van Tongeren,

The West Australian (2002b). Extremist uses Net to bypass anti-racism laws,  

Regan, T. (1999) How Terrorists use the Internet to spread their messages,
VPC (Violence Policy Center) (2001) *Firearms Training for Jihad in America*, URL:  


Baseline Security Standards Evaluation

W. J. Brooks and M. J. Warren

School of Information Technology
Deakin University, Geelong, Australia,
Email: wjbrooks@deakin.edu.au

ABSTRACT

Information security is now recognised as an important consideration in modern business, with a variety of guidelines and standards currently available to enable different business environments to be properly protected. However, financial and operational constraints often exist which influence the practicality of these recommendations. New baseline security methods such as Australian and New Zealand Standard (AS/NZS) ISO/IEC 17799:2001 and British Standard (BS) 7799 represent minimal standards which organisations can use to improve their security. The aim of the paper is to look at the effectiveness of baseline security standards through the use of evaluation criteria, which assesses their effectiveness.

Keywords: Security, Evaluation Criteria, Risk Analysis, and Baseline Security.

INTRODUCTION

During the last few decades the use of Information Technology has become more widespread in all areas of society, and the types of activities that it performs or supports have become increasingly more important. As a result, information systems are now heavily utilised by all organisations and relied upon to the extent that it would be impossible to manage without them. The growth of organisational business internetworking via e-commerce applications present further opportunities for unauthorised access to information systems (Lyre 1999) and this is now a major worry. Can risk analysis help to protect organisations?

The aim of risk analysis is to eliminate or reduce risks and vulnerabilities that affect the overall operation of organizational computer systems. Risk analysis not only looks at hardware and software, but also covers other areas such as physical security, human security, and business and disaster protection. In practice, there are major problems with the use of risk analysis; the time taken to carry out a review, the cost of hiring consultants and/or training staff. To overcome these negative aspects, baseline security standards were developed. Baseline security standards offer an alternative to conventional risk analysis methods as they represent the minimally acceptable security countermeasures that an organisation should have implemented. These countermeasures are applied in a generic manner, e.g. every organisation should have the same baseline security countermeasures.

The advantages of using baseline methods include (Brooks 2002):

- it is cheap to use;
- it is simple to use;
- no training is required to use the method; and,
- It is quicker then undertaking a full security review.
The disadvantages of using baseline methods include (Brooks 2002):

- the generic nature of baseline security methods mean they may not solve all of the organisational security requirements;
- the fact that they have been designed for use within a general environment mean that they may not be suited for all environments, i.e. healthcare or small businesses;
- there is no suggestion about how the security countermeasures may be implemented; and,
- they do not contain cost benefit details.

A number of baseline security methods have been developed in recent years. But do these baseline security standards actually offer a minimal level of protection; can organisations expect the correct to be offered by these baseline security standards? This was the rationale behind developing evaluation criteria that could be used to demonstrate how effective baseline security methods are.

**RESEARCH METHODS**

Since the research described in this paper is primarily qualitative in nature, the focus will be on observation, description, and interpretation of data. Qualitative research is defined as, ‘any kind of research that produces findings not arrived at by means of statistical procedures or other means of qualification’ (Strauss et al. 1990). Qualitative methods are used to gain a better understanding of a situation about which little is yet known, or new perspectives where much is already known. The methods used are data collection, data analysis, and data evaluation.

**Data Collection**

This stage involves the collection from various sources to gather authoritative and empirical data (Graziano et al. 2000) relating to information security (IS) that has relevance to IS frameworks and malicious software (MS). For the purpose of this research, the data collection stage consisted of the qualitative elements of observation and examination (Hill et al. 1967). The observation and examination of data involves data being gathered from existing literature relating to IS and will utilise various sources of data including:

- Theses;
- Standards;
- Journals;
- Conference papers;
- Magazines;
- Books; and,
- Internet.

The examination of the definitions, concepts, and theories relating to MS is particularly suited to a qualitative approach.

**Data Analysis**

Qualitative data analysis is defined as, ‘working with data, organising it, breaking it into manageable units, synthesising it, searching for patterns, discovering what is important and what is to be learned, and deciding what you will tell others’ (Bogdan et al. 1982). A content analysis was used to place the raw data collected into logical, meaningful categories. These categories, described later in the report, were identified by using a process called open coding (Strauss et al. 1990).
The goal of open coding is to create conceptual categories into which observed data will be placed, thus creating the initial model to be developed. The identified categories will be re-examined for the purpose of grouping similar definitions, concepts, and theories into the same category. Subsequent modifications and amendments of the categories will occur throughout the analysis process.

For the purpose of re-examination, a process called axial coding (Strauss et al. 1990) was used to determine how the identified categories are linked. During this process determination of sufficient data supporting the interpretation was conducted and, if necessary, the initial categories were revised, which lead to a re-examination of the raw data.

**Data Evaluation**

This stage of the qualitative research involves the verification, validation, and reliability of the data analysed.

Verification of this research entailed re-examination of the data collected to crosscheck and verify emergent conclusions. This process of examining and re-examining the collected data is similar to a process called tactics for generating meaning (Miles et al. 1994).

Validity is concerned with how well the data is gathered, analysed and fits into the concept (Graziano et al. 2000). To validate the results from the evaluation of the IS baseline standards a process called external validity (Graziano & Raulin, 2000) was used. This allowed for the testing of the evaluation criteria’s soundness and appropriateness to MS and baseline security.

Within qualitative research the issue of reliability is concerned with the level of objectivity with the research, and ensuring that data generation and analysis are appropriate, thorough, and accurate (Silverman 1997). Reliability is provided through the feedback from external parties.

**SUMMARY OF EVALUATED BASELINE SECURITY STANDARDS**

**Australian Communications-Electronic Security Instruction 33 (ACSI 33)**

The ACSI 33 is a government standard developed by the Defence Signals Directorate (DSD) to provide guidance to Australian Government agencies on how they can protect their information systems (DSD 2000). These information systems may hold classified or non-classified information but all should have some degree of protection if a reliable or accurate service is to be maintained. The ACSI 33 is intended for IT specialists only, such as, for example, IT security administrators. The standard is made up of a series of handbooks each covering a specific topic (i.e. Risk Management) with relevant security issues and, where appropriate, endeavours to categorise the security countermeasures into identified risk levels, defined as 'grades'(DSD 2000). These grades have been formulated to provide a minimum policy level that should be implemented. The ACSI 33 has been written to be consistent with the Commonwealth Protective Security Manual (PSCC 2001), and with two Australian Standards, AS/NZS 17799:2001 and AS/NZS 4360:1999 (ACSI, 2000).

**Australian / New Zealand Standard for Information Security Management (AS/NZS 17799)**

The AS/NZS 17799 is intended for use by managers and employees who are responsible for initiating, implementing, and maintaining information security within their organisation and is considered a basis for developing organisational security standards (Standards Australia 2001). It is an internationally recognised ISO standard (ISO/IEC 17799) that comprises of two parts: AS/NZS 17799:2001 (Standards Australia 2001) and AS/NZS 7799:2:2000 (Standards Australia 2000).
Part 1, ‘Code of Practice for Information Security Management’, is identical to the BS 7799.1:2000 (British Standards 2000a). It takes into account recent developments in information processing technology, and changes the emphasis towards information security to encompass information and IT projects, resources, networks, and projects (Standards Australia 2001). A comprehensive set of controls comprising the best information security practices currently in use are provided in this standard, which means that organisations are able to pick the guidelines most appropriate to them. The guidance and recommendations provided throughout this standard can be used as a basis from which, for example, a corporate policy can be developed, and should not be quoted as if they were specifications (Standards Australia 2001).

Part 2, ‘Specification for Information Security Management Systems’, is identical to the BS 7799.2:2000 (British Standards 2000b) and forms the basis for an assessment of the information security management system (ISMS) of an organisation, and can be used as a basis for a formal certification scheme (Standards Australia 2000). It specifies requirements for security controls to be implemented according to the needs of individual organisations.

German IT Baseline Protection Manual

The IT Baseline Protection Manual is a nationally recognised standard in Europe developed by the German Federal Agency for Security in Information Technology (BSI) and presents a detailed set of standard security measures that apply to virtually every IT system (BSI 2000). The aim of these security recommendations is to achieve a security level for IT systems that will satisfy normal protection requirements and can also serve as a basis for other systems and applications requiring a higher degree of protection. This is achieved through appropriate application of organisational, personnel, infrastructural and technical standard security safeguards (Henze 2000).

The IT Baseline Protection Manual includes (Henze 2000):

- Standard security measures for typical IT systems with ‘normal’ protection requirements;
- A description of the threat scenario that is globally assumed;
- Detailed descriptions of safeguards to assist with their implementation;
- A description of the process involved in attaining and maintaining an appropriate level of IT security; and,
- A simple procedure for ascertaining the level of IT security is attained in the form of a target versus actual comparison.

RESEARCH BACKGROUND

This section describes the steps involved in the development process of the evaluation criteria and the baseline security evaluation framework.

Analysis of Information Security Threats, Controls, and Baseline Standards

A content analysis was performed on existing information security threats (i.e. viruses), controls (i.e. anti-virus software), and information security baseline standards to determine a set of generic components that related to MS.

Development of the Evaluation Criteria for Malicious Software

Stage 2 was concerned with transforming the research generated from stage 1 into evaluation criteria. The evaluation criteria contains four categories each listing the definitions of each component based on the content analysis of information security. These are:
• The Information Security category relates to the ISMS, which is concerned with policies, regulations and rules that IT management would establish and implement;

• The Origin of Threats category relates to the types of threats, vulnerabilities, and attacks that the system could be at risk to;

• The Security Controls category relates to the generic security controls, countermeasures and safeguards that an organisation can implement to reduce the risks to the system; and,

• The Recover and Containment category relates to procedures, policies, and regulations that should be in place to help an organisation recover from a disaster or be ready for the unexpected. This category also covers procedures on information backup.

Development of the Evaluation Framework

Once the evaluation criteria was complete in stage 2, a framework needed to be developed that would allow for a quantitative method of assessing each baseline security standard against the evaluation criteria. The development of the new evaluation framework was based on the research undertaken by Von Solms et al. (1993).

Based on previous work (Von Solms et al. 1990), they defined an IS management model that is based on two-dimensional evaluation; the horizontal axis is for a number of defined IS categories, and on the vertical axis are various levels of IS numbered 0 (no security) to 5 (absolute security). The effectiveness of the IS countermeasures in place throughout the system are evaluated on a scale 0 to 5 and plotted on the two-dimensional graph. Figure 1 shows a graphical representation of this method.

![Figure 1. Information Security Evaluation Model (Von Solms et al. 1993)](image)

From this model they were able to define five operational security environments (OSEs) (Von Solms et al. 1992). These five OSEs are as follows:

• The Ideal OSE: an environment as described in the information security policy. This level is not always obtainable;

• The Prescribed OSE: an environment as dictated by external parties;

• The Baseline OSE: an environment that has been set as a goal by top management and is reachable by the organisation;
The Current OSE: the current state of security within the organisation; and,
The Survival OSE: the minimum set of information security controls and counter-measures that are needed to stay operational.

Using these five OSEs, Von Solms et al. (1993) developed an evaluation method (CRAMMEX) based on security risk analysis (CRAMM) and other management tools. CRAMMEX allows for the effectiveness of security measures to be evaluated on a scale of 0 to 5, and, with the development of the five OSEs, helps to provide IT management with a more concise, clear picture of the current IS status of the organisation.

Within our research we were looking at baseline security risk analysis and determined that the baseline security evaluation framework would be based on three of the OSEs: These are:

- Ideal OSE (High rating) and is the highest attainable level of security that can be implemented by an organisation;
- Baseline OSE (Medium rating) and is the minimum level of security that should be implemented by an organisation for their system to be secure; and,
- Survival OSE (Low rating) and is the minimum level of security measures needed to for an organisation to stay operational, but the system is classified as insecure.

This aspect of our research is particularly important because it describes the initial development of the new evaluation framework for baseline security risk analysis. Replacing the concept that baseline security is just a single level of security, but showing that baseline security could actually be represented by a number of different levels. As we stated before the limitations of our approach, is that we are just looking at the malicious aspects of the security baseline standards.

Table 1, shows the structure of the criteria evaluation, and the rating that would be used based upon the content analysis, which was described earlier.

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Individual Results</th>
<th>Subcategory Total</th>
<th>Category Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Information Security</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a. ISMS Level</td>
<td></td>
<td>4 = L, 8 = M, 12 = H</td>
<td></td>
</tr>
<tr>
<td>o Assets to be protected</td>
<td>1 = L, 2 = M, 3 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Origin of Threats</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a. Organisational</td>
<td>3 = L, 6 = M, 9 = H</td>
<td>8 = L, 16 = M, 24 = H</td>
<td></td>
</tr>
<tr>
<td>b. Human Failure</td>
<td>1 = L, 2 = M, 3 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c. Deliberate Acts</td>
<td>4 = L, 8 = M, 12 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Security Controls</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a. Security Awareness</td>
<td>5 = L, 10 = M, 15 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. Authorisation Security</td>
<td>3 = L, 6 = M, 9 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c. Virus Protection</td>
<td>3 = L, 6 = M, 9 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d. Data Integrity</td>
<td>6 = L, 12 = M, 18 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>e. Web Security</td>
<td>3 = L, 6 = M, 9 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f. Encryption Technologies</td>
<td>4 = L, 8 = M, 12 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>g. Organisational Security</td>
<td>4 = L, 8 = M, 12 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>h. Hardware and Software Security</td>
<td>3 = L, 6 = M, 9 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Recovery and containment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a. Disaster recovery</td>
<td>1 = L, 2 = M, 3 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. Contingency planning</td>
<td>1 = L, 2 = M, 3 = H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c. Information Backup</td>
<td>1 = L, 2 = M, 3 = H</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Number to Alpha System for each Individual Evaluation
Table 1 shows the structure of the breakdown of evaluation criteria results in the following way:

- Individual Results represents the value for each component (i.e. assets to be protected);
- Subcategory Total represents the value that is the sum of the values of each component within that subcategory; and,
- Category Total represents the value that is the sum of the values of each subcategory within that category.

**BASELINE EVALUATION RESULTS**

**Results Explained**

Each standard was evaluated using the criteria described previously, with each of the four key components being assessed. The criteria evaluation generated twelve sets of results (based upon the structure shown by table 1). We have summarised the results in Figure 2 in order to show a comparative representation of the results of each standard.

From the analysis of the results obtained by the criteria evaluation on MS, the following conclusions were made:

1. The ACSI 33 is a standard for the management of security controls and its main focus is providing appropriate security controls for web security. But it lacks detail on threats and recovery and containment procedures, which is a major issue for government agencies that require a high level of security due to the sensitive nature of data that is stored in their systems;
2. The AS/NZS 17799 is a standard for the management of security controls and its main focus is security awareness and authorisation security. Even though it provides more information on threats and recovery and containment procedures than the ACSI 33, it still does not provide sufficient technical details on security controls to allow organisations to maintain a secure system (i.e. it says what should be there to protect the organisation but does not mention how to implement it); and,
The BSI is a standard for the technical implementation of security controls and it focuses on deliberate acts, web security, hardware and software security, and recovery and containment procedures. However, the standard lacks in detail on the steps involved in developing IT management security policies, which can be a problem for organisations that require guidance on writing security policies because they will not be aware of what is required in a security policy, thus incorrect procedures could be implemented resulting in a system being insecure.

DECOMPOSITION OF EVALUATION RESULTS

Similar to how most risk analysis procedures provide organisations with a high-level analytical assessment of their IT systems, the evaluation criteria and framework provide a high-level analysis of the assessed baseline security standards. But how this differs from traditional risk analysis procedures is that the results of each standard can be broken down to represent the individual components within each category. To clearly demonstrate this concept a breakdown of the results for one of the baseline standards will be discussed. For the purpose of this discussion, the results of the Hardware and Software Security subcategory, within the Security Controls category, of the ACSI 33 standard will be explained.

![ACSI 33 Category Evaluation](image)

**Figure 3.** ACSI 33 Category Evaluation

The result for the Security Controls category in Figure 3 concludes the following:

*Baseline level of information for the Security Controls category: this is because the standard is directed at employees that have a high technical background and require little assistance to implement security controls.*

The problem with this analysis is that it does not state which components (i.e. Virus Protection, Data Integrity, etc) of the category are lacking, or provide sufficient detail. To overcome this problem the evaluation criteria and framework allow for each category to be broken down into specific subcategories so that the results of the analysis become much clearer to the individual. Figure 4 below depicts the result of each subcategory within the Security Control category.
Figure 4. ACSI 33 Subcategory Evaluation

The results in Figure 4 show that the (high-level) category evaluation of the baseline standard can be quite deceiving. Even though the Security Controls category was assessed and rated the ‘baseline’ level, here (Figure 4) it becomes clear that this level of quality is not consistent with every aspect of the Security Controls category. These subcategories can also be broken down into their individual components so that we can obtain a more precise result. For example, the result for the Hardware and Software Security subcategory in Figure 4 concludes the following:

Survival level of information for Hardware and Software Security: this is a problem because government agencies store and process a large amount of classified data and the need for appropriate physical security measures is increased. If the security in place is inadequate then the possibility of unauthorised access is increased.

But if we look at the individual components of this subcategory in Figure 5 we can see that the ACSI 33 standard provides no information on laptop policies, low-level information on periodic runs of anti-virus software, and a medium level of information on physical security.
ACSI 33 Hardware / Software Security

Figure 5. ACSI 33 Component Evaluation

Therefore, by using the new evaluation framework not only were we able to assess the baseline standards from a high-level perspective but it was also possible to individually assess each component of the standards that related to MS. This method also provided a means for pin-pointing the discrepancies in each standard, which is not provided by traditional risk analysis methods.

IMPACT UPON BASELINE SECURITY STANDARDS

The evaluation of the baseline risk analysis security standards has shown that there is a distinct difference between each standard. Even though it would be assumed that all baseline standards would be of the same standing. In terms of what the evaluation means for each of the standards, in reality very little. Each standard was designed for a particular target audience and as such fulfils the target audience requirements. But as stated before, it is no longer the case that baseline security standards should be assumed to be of an equal level of standing in terms of protection.

CONCLUSION

Information security baseline standards have been developed to provide security safeguards, implementation advice, and aids for numerous IT systems in use today. The three standards looked at in this study, the AS/NZS 17799 standard (Standards Australia 2000, 2001), the BSI standard (BSI 2000), and the ACSI 33 standard (DSD 2000) each suggests a ‘safe’ minimal level of security that organisations can implement. The problem is that many IT professionals view baseline security differently and this can cause confusion for organisations, as they cannot always make a well-informed decision on which standard best suits their needs.

Assessment of the three baseline standards by use of the security evaluation criteria proved that not all baseline standards provide the same level of information on security controls and countermeasures. The new baseline security evaluation framework shows that baseline security is not just one level but three individual levels (ideal, baseline, and survival). The new criteria based upon MS was assessed against three baseline standards. The results prove that the three standards provide different levels of information for ISMS, threats, security controls, and recovery and containment procedures, even though all three are considered to be baseline standards and of the same level.

We have shown within this paper that baseline security risk analysis methods are not all equal.
REFERENCES


Available from: http://www.bsi.de/english/
Accessed: 15th March 2001


Available from: http://www.ag.gov.au
Accessed: 4 November 2001


A Formalisation of an Information Infrastructure Security Risk Analysis Approach

T. B. Busuttil, and M. J. Warren

School of Information Technology
Deakin University, Australia
E-mail: tbb@deakin.edu.au, mwarren@deakin.edu.au

ABSTRACT

With the protection of Information Infrastructure (II) becoming a priority for all level of management, the need for a new security methodology to deal with the new and unique attack threats and vulnerabilities associated with the new information technology security paradigm. The formalisation of the concept of a fourth generation security risk analysis method which copes with the shift from computer security to information warfare is the next step toward handling security risk at all levels. The paper will present the formalised procedure of fourth generation models and their application to infrastructure protection and the associated advantages of this methodology.

Keywords: Information Infrastructure, Security Risk analysis, Information warfare.

INTRODUCTION

Understanding and managing Information Infrastructure (II) security risks is a priority to most entities dealing with Information Technology and IW scenarios today (Libiciki 2000). Traditional Risk Analysis (RA) was well suited to these tasks within the paradigm of computer security where the focus was on securing tangible items such as computing and communications equipment (NCS 1996; Cramer 1998). With the growth of information interchange and reliance on Information Infrastructure, the ability to understand where threats lie within an organisation, regardless of size, has become extremely difficult (NIPC 1996). To place a value on the information that is owned and used by an organisation is virtually an impossible task (Busuttil and Warren 2001a). The suitability of Risk Analysis to assist in managing IW related security risks is unqualified, however studies have been undertaken to build frameworks and methodologies for modelling Information Attacks (Beer 1984; Molander et al. 1996; Johnson 1997; Busuttil and Warren 2001b; Hutchinson and Warren 2001) which will assist greatly in applying Risk Analysis concepts and methodologies to the burgeoning information technology security paradigm, Information Warfare. The concept of fourth generation security risk analysis, which takes the form of the conceptual model of layered logical transformation models (Busuttil and Warren 2002). These models allow stakeholders to apply risk analysis to traditional IW scenarios so as to deal with the problems of scalability and inaccurate cost analysis as well as being dynamic enough to keep up with the constant changes occurring in information infrastructure and information attacks.

REVIEW AND DEVELOPMENT OF SECURITY RISK ANALYSIS

The field of security risk analysis (SRA) has, so far, evolved over three generations (Table 1) (Baskerville 1993). Early (1st generation) SRA models focused on fulfilment of checklists. The logical move was then made toward computer-based mechanistic engineering models, which still incorporated
checklists but also featured some qualitative measures of risks, threats and vulnerability. With the shift from computer security to information security, third generation, logical transformation models (LTMs) were looked at as the solution to quantifying information system requirements. A quantitative method was required due to the fact that putting a time and/or money value on information assets was virtually impossible when compared to cost benefit analysis of computing technology assets.

<table>
<thead>
<tr>
<th>Generation</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>First</td>
<td>Checklisting Methods</td>
</tr>
<tr>
<td>Second</td>
<td>Computer-Based Mechanistic Engineering Methods</td>
</tr>
<tr>
<td>Third</td>
<td>Logical Transformation Model Methods</td>
</tr>
</tbody>
</table>

Table 1: The generations of information technology security risk analysis (Baskerville 1993)

**Logical Transformation Models (LTM)**

The aim of these models is to show the security problems and the solutions. The ‘logical’ aspects of these models relate to the security process, whilst the ‘transformation’ aspect relates to how the security process can be transformed in work management (Bass 1985), organizational theory (Leifer 1989) and social understanding overall (Leifer 1989). Earlier methods assume a much smaller array of potential information system achievements and for this reason tend to fail to service management in the true understanding of the scope of information systems (Avison and Wood-Harper 1991). The basic approach taken with LTMs is to firstly, find a system function that has a perceived flaw. When this problem has been found it is then important for a solution to this problem to be found. Once the solution to the problem has been found it should then be tested. If test requirements are met then the update should be incorporated into the system. The perceived advantages are that a more flexible approach to security design is offered based on a much lower level of conflict between security and system usability. Disadvantages lie in the fact that because these methods are still relatively new, there tends to be a lack of user experience. Because of the ‘Transformation’ aspect of the models, cost benefit evaluation is more difficult to carry out due mainly to issues of timeliness and accuracy (Baskerville 1993). The three distinguishing characteristics of logical transformation models are:

- Emphasis on producing the right type of security for a particular system, rather than just implementing the security correctly;
- Security design will be within the scope of a logical and/or transformational model;
- A move away from cost-benefit risk analysis accompanying an increased usage of abstraction models.

An overview of the characteristics of logical transformation systems is shown in table 2 (Baskerville 1993). This table defines a logical transformation modelling system. This is done by first making assumption about the system then completing activities to finally derive strengths and weaknesses.
Characteristics of Logical Transformation Methods

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Activities</th>
<th>Strengths</th>
<th>Weaknesses</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Ideal security solutions will only evolve from an understanding of the broad problem situation</td>
<td>• Model Building</td>
<td>• Flexibility in controls</td>
<td>• Lack of experience</td>
</tr>
<tr>
<td>• Abstract models clarify the organisational problems, and more effective controls will result</td>
<td>• Stakeholder Analysis</td>
<td>• Excellent Documentation</td>
<td>• Abstract controls are difficult</td>
</tr>
<tr>
<td>• Designs founded on such models will prove flexible, adaptable as well as, consequently longer-lived</td>
<td>• Translation of abstract models into reality</td>
<td>• Low maintenance costs</td>
<td>• Physical security details are postponed</td>
</tr>
<tr>
<td>• There are few universal solutions</td>
<td>• Implementing physical models</td>
<td>• Closer connection</td>
<td>• Difficult cost evaluation</td>
</tr>
<tr>
<td>• Controls place constraints on information systems</td>
<td>• Maintaining</td>
<td>• Lessen the relevance of risk analysis</td>
<td>• Not as useful for existing systems</td>
</tr>
</tbody>
</table>

Table 2: Characteristics of Logical Transformation Methods (Baskerville 1993)

4TH GENERATION SRA AND INFORMATION INFRASTRUCTURE PROTECTION

The previous advancement in SRA was a logically de-evolutionary step. The adoption of baseline standards as a method of SRA is basically a formalised approach at building a system, using 1st generation model checklists, for security assurance. The uptake of baseline standards amongst organisations validates a belief that standardisation of security is accepted amongst IT security policy makers (Brooks 2001). Due to the scope and scalability of IW threats and vulnerabilities, baseline security is not a realistic option for IW SRA.

Rationale for new LTM-based SRA methodology for IW

Third generation SRA methodologies, LTM, were designed to work well when built into information system SRA scenarios from the beginning (Baskerville 1993). The major characteristics of IW which set it apart from information security (IS) are the need to take (1) scalability, (2) flexibility and (3) difficulty in cost evaluating of threats, vulnerabilities and attacks into account when considering IW. So to adapt LTM SRA technology from IS to IW these issues of scalability and adaptability must first be dealt with.

One of the major advantages of LTM is the ability to build security into information systems in an adaptable manner (Baskerville 1993). The flexibility of control that is possible when designing security using LTM is a definite strength when dealing with IW concerns as threats, vulnerabilities and targets are constantly changing. The problem of scalability can also be dealt with by bringing forward the concept of layering the LTM so as each level of information infrastructure can have 1 to
many LTMs that each depicts a security solution. Any number or level of information infrastructures can be included in the overall model.

The problem of cost evaluation is solved, as LTMs do not focus on cost evaluation. IW-based cost evaluation is virtually impossible so factoring it out at the model level is a way of making sure security is built well over the breadth and depth of the system. Focussing on one major area to secure can often be a downfall of organisations (Cramer 1997). The only minor difficulty is the need to classify which Information Infrastructure level contains particular entities, problems etc. A proposed solution to this problem is to also include scope in the modelling methodology to handle infrastructure interfaces. This would be where security issues regarding physical and/or logical links between two infrastructure levels would be discussed. The notation of different levels of infrastructure is shown in table 3.

<table>
<thead>
<tr>
<th>Infrastructure Level</th>
<th>Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global Information Infrastructure</td>
<td>GII</td>
</tr>
<tr>
<td>National Information Infrastructure</td>
<td>NII</td>
</tr>
<tr>
<td>Organisational Information Infrastructure</td>
<td>OII</td>
</tr>
<tr>
<td>Personal Information Infrastructure</td>
<td>PII</td>
</tr>
</tbody>
</table>

Table 3: Infrastructure-level Notations

4th Generation SRA – Layered Logical Transformation Models

The proposal of the idea of a fourth generation of SRA model comes about as a result of the lack of suitability of the aforementioned SRA methodologies to Information Warfare (Busuttil and Warren 2002). This next generation will involve the application of logical transformation methods across the layers of information infrastructure as shown in figure 1.

CASE STUDY – APPLICATION OF CONCEPTUAL MODEL TO REAL-LIFE SCENARIO

This case study shows the application of Layered Logical Transformation Models to an organisation. The organisation that will be focussed on is the online retailer, Amazon.com. The reasons for the selection of Amazon.com is that it relies on the NII for business continuity whilst also providing connection infrastructure for its customers’ PII. Amazon.com runs its business within an organisational infrastructure and relies on the national and global II to provide the underlying communications and computing support necessary. Amazon.com should not do anything to actively compromise its connections to the NII/GII or the actual NII/GII. Customers of Amazon.com, rely on a dependable method of connection to Amazon.com’s OII whilst they should also be able to ensure they do not actively attempt to compromise the connection to Amazon.com’s OII or the actual OII. The following list shows the 5 major principles that need to be upheld (Busuttil and Warren 2002):
1. Amazon.com can expect a certain level of service from level of infrastructure above.

2. Amazon.com should do all it can to ensure that the links between itself and any higher level infrastructure entities are secure.

3. Amazon.com should focus on securing itself to the best of its abilities in four major categories. Defending against:
   - High-level infrastructure attacks;
   - Internal attacks;
   - Low-level infrastructure attacks and;
   - Partnership attacks.

4. Amazon.com must ensure that the connection to the lower infrastructure level is not compromised and should also expect a degree of care to be exercised by the user.

5. Amazon.com should ensure that the integrity of lower level infrastructure components is upheld during any interaction with customers and should also expect users to maintain II entities.

The conceptual diagram of this case study application including references to the implied steps and the level of hierarchy they must take place within is shown in figure 2.

![Diagram](image_url)

**Figure 2**: Amazon.com’s application of LLTM concept (Busuttil and Warren 2002)

**LOGICAL TRANSFORMATION MODELS IN II SECURITY RISK ANALYSIS**

When building an information security system using logical transformation models there are a number of steps that need to be followed. Firstly, a committee representing a large cross-section of the involved system users should undertake the approach as this will assist in the exposition of infrastructure definitions, threats, vulnerabilities and countermeasures. For each defined piece of the information infrastructure the following information needs to be stored:

- Infrastructure definitions;
- Infrastructure threats and vulnerability assessments on each infrastructure level.

Once threat and vulnerability assessments have been completed the organisation can then attempt to map the vulnerabilities to likely threats so as to get an overall understanding of the problems that face
the organisation undertaking this risk analysis approach. The following formal steps are required for completing 4th generation security risk analysis:

1. Form risk analysis committee;
2. Define Infrastructure;
3. Complete threat and vulnerability assessment on each infrastructure level;
4. Derive countermeasures based on findings from steps 2 and 3.

Step 1 should be completed once at the beginning of the lifecycle of the risk analysis process. Step 2 should be completed once for each piece of infrastructure that is introduced to the overall system. Steps 3 and 4 should be completed once at the beginning of the analysis to cover all the parts that exist at this time within the infrastructure system and should be updated regularly for both new and previously integrated infrastructure entities. A formal description of each of the aforementioned steps follows.

FORM RISK ANALYSIS COMMITTEE

The first step in the 4th generation Security Risk analysis methodology, layered logical transformation modelling is to construct a committee with a wide cross-section of understanding regarding the current computing environment within the organisation in which the risk analysis is being undertaken. This committee should encompass people from all levels of the organisation e.g. management to clerks, and also different areas of expertise e.g. computing to accounting. The reason for this diversity to be inherent within the panel undertaking the analysis is that the organisation are looking for all infrastructure security risks and the wider the net is cast the more likely each ensuing step will be completed to an efficient level. The members of a small (10-50 employees) IT company’s committee would possibly consist of the following staff:

- Chief Executive Officer (1);
- Chief Information Officer (1);
- System Administrator (1);
- Systems Analyst (1)
- Secretary (1);
- Receptionist (1);
- Programmers representatives (2);
- Accounts representative (1);
- Webmaster (1).

Other types of organisations may have different committee member positions, however, the most important facet to the formation of this committee is to ensure a diverse range of backgrounds and areas of expertise.

DEFINE INFRASTRUCTURE

This step requires the committee to classify what sort of II it is dependent on. In the case of the case study, Amazon.com makes use of an organisational II that administers personal IIIs whilst being reliant on a national II. At this stage the system boundaries (Vidalis and Blyth 2002) should be mapped so as to understand where different LTM’s are required for different layers of II. The total OII should be broken down into sections that can be defined, classified and analysed separately. This definition of infrastructure entities may include a mapping to the infrastructure, including it’s interfaces to other infrastructure within and outside of the organisation as well as the current security measures currently in place. Previous security incidents (if any) and the relevant countermeasures taken (if any) would also assist in the further steps in the model. A basic example of this step is shown in table 4.
The fourth step requires the completion of a threat assessment. The threat assessment should include a thorough rundown of likely threats to the organisation as a whole and also any known threats to particular entities within the OII. A method of threat assessment for electronic payment systems (EPS) named ‘Threat Assessment Model for EPS’ (TAME) (O’Mahony et al. 1997) shows a loosely coupled decision loop that allows for on-the-fly adjustment to system threats and inputs and outputs (Vidalis and Blyth 2002). Despite being developed for use with EPS this methodology, due both to it’s flexibility and non-reliance on cost analysis, should prove suitable for IW and infrastructure protection security risk analysis. The conceptual model takes into account the following interchangeable and also interoperable stages:

- Assessment Scope;
- Scenario Construction and Modelling;
- Threat agent and Vulnerability Analysis;
- Evaluation.

These stages consist of a number of steps which are all ongoing and simultaneous allowing for a thorough and ongoing approach to threat and vulnerability assessment (Vidalis and Blyth 2002). The conceptual flow of information within the TAME system is presented in figure 3.

![Figure 3: TAME system](image-url)
Table 5 lists the steps involved at each stage of the TAME methodology.

<table>
<thead>
<tr>
<th>Stages</th>
<th>Sub-Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assessment Scope</td>
<td>Business Analysis</td>
</tr>
<tr>
<td></td>
<td>Stakeholder Identification</td>
</tr>
<tr>
<td></td>
<td>System Boundaries Identification</td>
</tr>
<tr>
<td></td>
<td>Threat Agent Identification &amp; Selection</td>
</tr>
<tr>
<td>Scenario Construction &amp; Modelling</td>
<td>Scenario Generation</td>
</tr>
<tr>
<td></td>
<td>System Modelling</td>
</tr>
<tr>
<td></td>
<td>Asset Identification</td>
</tr>
<tr>
<td>Threat Agent &amp; Vulnerability Analysis</td>
<td>Threat Agent Preference Structuring</td>
</tr>
<tr>
<td></td>
<td>Threat Agent Capabilities</td>
</tr>
<tr>
<td></td>
<td>Vulnerability Type Identification &amp; Selection</td>
</tr>
<tr>
<td></td>
<td>Vulnerability Complexity Analysis</td>
</tr>
<tr>
<td>Evaluation</td>
<td>Stakeholder Evaluation</td>
</tr>
<tr>
<td></td>
<td>Scenario Selection &amp; Conflict Resolution</td>
</tr>
<tr>
<td></td>
<td>Threat Impact Analysis</td>
</tr>
<tr>
<td></td>
<td>Threat Statement Generation &amp; Transfer</td>
</tr>
</tbody>
</table>

Table 5: Steps in TAME methodology (Vidalis and Blyth 2002)

DERIVE COUNTERMEASURES

The final step in this security risk analysis is to derive countermeasures for the threats and vulnerabilities that were identified in the vulnerability and threat assessments. These countermeasures should attempt to solve the security problem being faced whilst also attempting to maintain a reasonable degree of subjective cost benefit. The derivation of countermeasures can be done in many ways:

- Applying bug fixes;
- Updating security policy;
- Informing and training staff;
- Installation of new software and hardware solutions;
- Deleting unrequired associations;

The formal presentation of these countermeasures should be delivered as shown in table 6 for each vulnerability/threat pair:

<table>
<thead>
<tr>
<th>Vulnerability</th>
<th>Threat</th>
<th>Countermeasure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Server</td>
<td>DOS</td>
<td>Install firewall and configure correctly to assist in the stopping of DOS attacks</td>
</tr>
<tr>
<td>security hole</td>
<td>attacker</td>
<td></td>
</tr>
</tbody>
</table>

Table 6: Basic example of a Countermeasures Table

CONCLUSION

The fourth generation NII security risk analysis methodology is a move toward dealing with the scalability issues that have in the past meant that RA was not immediately adaptable to information warfare and other information infrastructure protection requirements. This methodology would prove to be helpful to organisations with mid-level infrastructure such as an organisational information infrastructure if undertaken in solitude however the true benefits of this methodology would be seen if it was put into practice by higher level infrastructure stakeholders. This up take by higher-level infrastructure would lead to higher dependability and reliability being built into infrastructure system from the outset. IW needs a unique security methodology that is useful at dealing with all the previous concerns that CS and IS dealt with along with the ability to be adaptable and scalable also. When
researching existing methodologies, logical transformation models proved to be a suitable method for coping with adaptability issues. The scalability issues are dealt with through the application of multiple layers of LTM's. Cost evaluation has been found to be an outdated function when analyzing IW risks, LTM's have the added feature of being solution-oriented and independent of any cost evaluation procedures. This methodology is fairly flexible but has a format that will be easily transferable from a formal rule set to a computer-based methodology.

REFERENCES


Evaluating IS Security Policy Development

S.B. Maynard & A.B. Ruighaver
Department of Information Systems
University of Melbourne
Australia
1Email: seanbm@unimelb.edu.au
2Email: anthonie@unimelb.edu.au

ABSTRACT

Rapidly increasing threats to the security of information systems is forcing organizations to put more effort into improving security policy quality. An initial approach to improving the security policy development process may be to enforce similar standards to those used in information systems development. This will focus those developing the security policy on the content of the policy, and also on the documentation of why content is there and for what reasons. This will enable a proper evaluation of the quality of the resulting security policy, similar to the state-of-the-art evaluation standards used in information technology security evaluation.

Keywords: Security Policy, Policy Development

INTRODUCTION

In many organizations, especially in small and medium sized ones, securing the organizations’ information is not considered to be a core business objective. Small to medium businesses frequently underrate the risks to their valuable information assets; underestimating how costly it would be if it fell into competitors’ hands or was misused.

Surveys conducted over the past 5-10 years have shown organizational interest in security is on the increase and that an upsurge in the implementation of security measures, including the development of organizational security policies, is taking place (James and Coldwell 1993, Ernst and Young 1995, Kearvell-White 1996, Davis 1996, Ernst and Young 1998). What these surveys do not show, however, is whether the quality of the organizations response to the acutely increased threats over this period has been adequate or not.

For an organization to have an adequate response, it needs a properly documented Strategic Information Security Policy. In this research the term Strategic Information Security Policy (SISP) is defined as a set of rules and procedures designed at the strategic level of an organization aimed to protect the information assets of the organization. The quality of this security policy has considerable impact on its capacity to implement adequate measures to prevent, or react to, security attacks as well as its capacity to limit the damage these attacks may cause.

A security policy forces a company to plan for the possibility that their information system may be a viable point of attack, either internally or externally by people, or through a natural disaster. By planning for the possibility of an attack and identifying where an attack may occur the security policy is enforcing some protection of the organization’s information. Possible problem areas are identified and are acted upon when the policy is implemented.
Our research, in particular, concentrates on how we can improve the quality of the strategic information security policy, which an organization produces. Several issues that need to be addressed are evident:

- How do organizations develop state-of-the-art strategic information security policies?
- Is the development of these policies conducive to their latter evaluation?
- What are the factors organizations use to test whether the policy is a success?

Research on each of these areas tends to be very normative, stating only what should be done, without any obvious evidence of practical application within organizations. Published research on what is actually happening within organizations is rather scarce. Warman (1995) observes, 'It is interesting therefore to note the contrast between the ideas and theory of security policy that appear to be recognised and accepted, and the actual practice of their implementation within organizations [which does not follow the theory].'

There is an extensive body of research on security audits; the evaluation of a particular company's systems to determine if they are secure, sometimes with and other times without reference to the security policy that the company has developed. Such audits focus on the evaluation of whether hardware, software and personnel can be considered secure and whether the policy implementation is still satisfactory. The premise used here is that because the company has a security policy and because we are testing the security of the company's systems, then that security policy by inference is also evaluated. This does not take into account the problems involved with the development and maintenance of the security policy, which in turn may signify that aspects of the policy are not appropriate in some situations. In no research, to the author's knowledge, has the security policy itself been the target of evaluation techniques from either the development or use perspective.

In this paper we attempt to initiate the development of security policy evaluation principles. We first discuss the development of standards for information system security evaluation in general. Then we argue that a similar approach to evaluation should be used for Strategic Information Security Policies. Following this we focus on the policy development process and discuss the need to further improve the re-use of security policies. Finally, we compare policy development with information systems development and emphasize the importance of adequate documentation of the policy development process.

INFORMATION SYSTEM SECURITY EVALUATION

The concept of security evaluation originates in the US Department of Defence with the Trusted Computer System Evaluation Criteria (TCSEC) or the Orange Book published in 1985 (US Department of Defence 1995). For several years this was the baseline for security evaluation particularly in 'high risk' government institutions, but also in some commercial situations. Since the publication of these criteria, significant changes to the computing industry have taken place causing evolution from this baseline.

Since 1995 there have been many attempts to develop a standard form for security evaluation. In 1991 the European standard for evaluation: Information Technology Security Evaluation Criteria (ITSEC) was developed by France, the UK, the Netherlands and Germany (Nash, Brewer et al. 1991). Also in 1991, ISO/SC27 WG3 began work on evaluation criteria to be used in quality assurance of products. The Canadian evaluation effort began in 1993 with the Canadian Trusted Computer Product Evaluation Criteria (CTCPEC) (CSSC 1993) as did the new US standard, aimed at updating the TCSEC standard (NIST/NSA 1993). This effort was shelved, as researchers started a cooperative effort between the USA, Canada, France, Germany, the Netherlands and the UK (Overbeek 1995) to develop a set of Common Criteria for Information Technology Security Evaluation (CC). These security evaluation standards do not just focus on the evaluation of finished products but also on the development process. The CC approach attempts to combine the best aspects of both TCSEC and ITSEC to try to ease the mutual recognition of evaluation results between nations. The problem with
each of these methods is their narrow focus on the product and its development process, rather than on the whole environment in which that product will be implemented. So even if the product has a high security standard, it may be implemented in an organization with a security policy that is substandard, incorrectly implemented, or even missing.

From these initial efforts a number of standards have now been produced that focus on the organisational implementation of security rather than on products. These standards include BS 7799, AS/NZS 17799:2001, ISO 17799. Unfortunately, the adherence and uptake of these standards in industry is questionable. For instance in the UK where the BS 7799 standard has been in use in its revised form since 1999, many organisations, whilst being aware of the standard, are unable to state what it covers. “Whilst BS 7799 has become the international standard of security, only 15% of people responsible for IT security in the UK are aware of its contents” (DTI 2002). Whilst there is no information regarding similar experiences with the ISO 17799 or AS/NZS 17799:2001 standards anecdotal evidence seems to suggest that the findings of the UK survey would be comparable internationally.

The question to be asked, therefore, is how one should evaluate the security of an organization. Is it sufficient only evaluating the end product, or should one also evaluate the process used to produce the required level of security. As we believe that the Strategic Information Systems Security policy is one of the major corner stones of an organizations security, the same question should be asked on how to assess the quality of this policy. In our view, the evaluation of the policy itself will have no merit without an evaluation of the policy development process and, if relevant, an evaluation of the policy’s maintenance.

SECURITY POLICY DEVELOPMENT

A review of literature reveals several different process models, which organizations can use for the development of custom security policies. Many of them make no distinction between the different kind of security policies for which they may be used and almost all concentrate more on the full life cycle of a security policy and omit any serious discussion on the major issues encountered in the actual policy formulation.

The development process that is probably the most focussed on policy formulation is the one proposed by Bayuk (1996). Essentially the process starts after the identification of the assets to be secured, and goes through a series of prototype documents, and results in the production of a draft policy. After this policy is approved and published, part of the process repeats itself to ensure that the policy is continuously updated. Still most of the model accentuates the need for the actual management of the security policy development process and the actual advice on how the teams responsible for drafting the documents should approach their task is minimal.

Another example of a process of security policy development is presented by Control Data (2000). They use a three-phase process: Policy, Enforcement, and Assurance, which describes the life cycle of the security policy. With this method, the development of the policy, including identifying the threats and risks, is done in the policy stage and enforcement focuses on the action of a security policy. This is where the policy is in use and is constantly monitored and tested by the organization’s day-to-day activities. The Assurance phase is where the implementation of the policy and its strategy effectiveness are tested. Additional factors about the success or failure of the policy arise and are fed back to the policy stage. With this particular process some documentation takes place that will allow some indication of why things were done in a particular manner, and may enable an audit trail.

There are a number of other less well-described development processes available in the literature. Each of these policy development processes has many common factors, which we summarised in Table 1.

Table 1 A summary of other process oriented development methods
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Steps in planning a security policy</td>
<td>How to develop a security policy</td>
<td>Developing a security policy</td>
<td>Guideline for developing an IS security policy</td>
</tr>
<tr>
<td>Steps</td>
<td>1. Study Risk</td>
<td>1. Research policy content</td>
<td>1. Determine what assets need protection</td>
<td>1. Develop a work group</td>
</tr>
<tr>
<td></td>
<td>2. Formulate Policy</td>
<td>2. Draft policy</td>
<td>2. Determine the level of protection for each asset</td>
<td>2. Brainstorm and develop policy points for review</td>
</tr>
<tr>
<td></td>
<td>3. Develop standards about why something should happen and the issues involved</td>
<td>3. Obtain management approval</td>
<td>3. Determine internet usage</td>
<td>3. Once reviewed, develop a draft policy</td>
</tr>
<tr>
<td></td>
<td>4. Issue policy to staff</td>
<td>4. Determine the threats that exist</td>
<td>4. Once finalised, get top management endorsement</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5. Get Co-operation from Management</td>
<td>5. Monitor and maintain</td>
<td>5. Explore how to address the treats identified</td>
<td>5. Brief employees and gain signatures from them</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8. Develop an implementation plan</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9. Add a recovery section in the policy</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10. User training</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>11. Respond to incidents.</td>
<td></td>
</tr>
</tbody>
</table>

As can be seen from the table, each of the authors offers similar basic steps for the development of a security policy document. Individual authors also put in specific details dependant on the researcher’s audience and purpose. Only the methods proposed by Woodward (2000) and DTI (1999) suggest a monitoring and feedback loop. DTI (1999) does not formally include this as part of the policy method, but as part of developing security for an organization. In each of the methods there is some sort of assessment of what needs to be protected, then the development of policy statements takes place.

In practice, there is not much evidence that any of the above processes are rigorously followed in industry. There is anecdotal evidence that most small to medium sized organizations find these development processes much too expensive and just borrow security policies from other organizations, deleting and adapting policy statements not relevant for their situation.

Even many larger companies do not develop security policies from scratch. They use a method, which uses a set of pre-written authoritative policy statements that are used to produce a workable policy. This involves determining the area of risk within the organization and then selecting from a number of pre-written statements about that particular area. This is done in a similar manner to piecing together a jigsaw. Often a sample policy is shown to give some sort of idea what a completed policy should look like.

This method is a fairly inexpensive way an organization can go about producing a policy. All an organization needs to do is to purchase a document outlining the policy statements with directions on how to use them. Alternatively a company representative can be hired to tailor the policy statements for the organization, producing a tailored security policy.
Pentasafer Security Technologies, a company specialising in security policy development uses this approach and sells a document that has 1000+ precise security policy statements that you can select from to make a policy. Other companies offer similar approaches. A report compiled for the Canadian Government suggests that this approach would be appropriate, with some changes, for adoption within government departments (Canadian Government Report 1996). All major areas of security policy are covered and the scope, according to the report, is similar to current Government security policy. Further, the report suggests that the benefits of this method lie in its ease of use, and comprehensiveness.

The approach adopted by Pentasafer categorises the audience for each policy statement, so that the particular policy statement selected is tailored for the particular audience. Policy statements are further categorised based on the environment risk in which the policy will be implemented. So a high-risk situation has a set of policy statements specifically worded for 'high risk' environments. The approach is complete for a high level set of system security policy statements. However, according to the Canadian Government report, it is not valid for lower level policies. Also, this approach does not address the maintainability or enforcement of the policy.

While re-use of security policy statements can be a valid approach to produce high quality policies, there are inherent problems in this method, depending on how the re-use was conducted. If the development process was purely a 'cut and paste' approach without reference to the risks apparent for a particular organization then there is no guarantee that the policy produced will be effective for the organization. If however, the approach used was to identify the risks facing the organization and then use the 'cut and paste' method, a more organizational focused policy will be produced. The assumption made here about this approach is that the sources of the policy statements are effective policies, and that the statements themselves are of a high quality. To ensure that the resulting policy is of high quality as well, new development processes need to be worked out; processes that emphasise the major issues in the re-use of security policies. While re-use in software engineering is a mature area of research, we have been unable to find any major references reporting on research in the re-use of security policies.

ENABLING THE EVALUATION OF SECURITY POLICIES

In the previous section we started to expose the gap between theory and general practice in the development of security policies. When we compare this general practice with the current state-of-the-art in information systems development other deficiencies will become evident.

The development of an information system progresses through several distinct phases from analysing the problem through to the implementation of the system. Throughout this process each step is documented through a series of deliverables that range from a feasibility study, through to training manuals and system documentation. In the development of a security policy this self-documentation process generally does not occur.

McMillan (1998) suggests that security policies should only contain principles. Many policies developed currently attempt to fit everything into the security policy: the justification of importance and specific system instructions and descriptions. Certainly, the security policy itself is already long-winded enough, without having details on how the document was created, who was consulted in its production or how the policy formulation was achieved. Nor will there be any documentation of political problems that may have occurred during the implementation of the policy, or of how to train people about the policy. With the use of documentation techniques during the development of the policy, however, a security policy could become a principles document again. Other issues not dealing with the principles of security policy would be documented elsewhere along with the justification for the policy.
The most important information that is often missing or just inadequate when we attempt to evaluate a security policy in an organization is the requirements documentation. Without a clear understanding of the requirements, evaluation of the quality of a security policy is almost impossible. An analysis of the risks/threats faced by the organization is only part of the requirements needed for the development of a security policy. The organization's objectives and other political issues that influence the development, implementation and acceptance of the security policy are just as important.

The availability of extensive documentation including the outcomes of an extensive requirements analysis will allow the evaluation of the security policy to reach a greater depth, instead of just superficially evaluating the end product. For instance, rather than concentrating only on whether the policy has been implemented in a particular area, the evaluation can now also consider how that area was developed within the policy. Documentary evidence could be evaluated to determine if the policy adequately covers all issues identified within development without watering any of them down. However, security policies vary greatly depending on the context of the organization and one would expect that their development would also vary. Some commonality between policies would exist, even as target areas digress. Unlike product evaluation however, many criteria in security policy evaluation will be of a subjective nature. This is because of the subjective nature of developing a policy and of the environment in which the policy is implemented.

CONCLUSION

Information System security evaluation research in many instances focuses on the evaluation of how well information systems are secured in relation to some sort of policy statement or security plan. Little research however, focuses on the manner in which security plans (or policies) are developed, and none, to the authors knowledge, attempt to evaluate the process of generating a security plan, or to evaluate the security plans themselves.

This paper provides a preliminary focus on several issues that need to be addressed in the development of security policy in organizations to enable a proper evaluation of these policies. We have found several identifiable areas where similarities exist with the current system security evaluation processes. These methods of security product evaluation do not merely evaluate the finished product, but attempt to evaluate the complete development process. This not only makes the evaluation process more comprehensive, but also aids in the quality assurance of the product.

In comparing the security policy development process occurring in many organizations to the current practice in information systems development, it becomes evident that the documentation coming out of the policy development is currently negligible. In fact, in real terms, documentation of security policy development is in the 1970's when compared to software development efforts. From the information systems development perspective, the documentation in software development is quite evolved and, as a result, the failure of many projects has been avoided through the use of the prior documentation. Similarly, providing the security policy evaluation with the required documentation may enable the evaluation to identify possible improvements in the policy development process.

REFERENCES


NIST/NSA (1993). Federal Criteria for Information Technology Security (FC), Draft 1, NIST/NSA.


Firewall or FireFolly – An initial investigation into the effectiveness of Personal Firewalls in securing personal computers from attack.

Jeshua Yee

School of Computer and Information Science
Edith Cowan University
E-mail: jyee@student.ecu.edu.au

ABSTRACT

With the increase in the use of Personal Computer (PC) firewall, this study looks at the security features provided by eight PC firewall; they are BlackICE, Deerfield, Kirio, MacAfee, Outpost, Sygate, Tiny and Zonealarm. The Nessus network scanner was used to probe these systems in a variety of configurations. The paper discusses the results and their implications for users of this new range of software.

Keywords: Personal Computer firewall, Nessus network scanner, firewall security features

INTRODUCTION

Today, with the advent of Personal Computer (PC), a PC no longer acts as a mere word processor or for calculating simple spreadsheets. It has now become an important tool for communications, program organisation and business processes in traditional places of commerce and enterprise. In addition, many of the company’s personnel work regularly by logging on the company’s network remotely (Yasin, 2000). Therefore, a malicious attack on a home PC could have a detrimental effect on the user’s daily routine, business and financial well-being (Hulme, 2002). This problem is further compounded with the increasing deployment of affordable broadband in the form of cable and ADSL services. These services provide reliable 24-hour high bandwidth connectivity to the Internet (Fleishman, 2001; Goldsborough, 2002; Janss, 2000; Radcliff, 2001).

With the increase in hacking incidences (Ravendran, 2002), small and large organisations alike are moving to protect their PC’s and the valuable data they contain (Clark, 2001; Goldsborough, 2000a; Schwartz, et. al., 2001). This has created a lucrative market niche for personalised PC firewall products with a resultant surge in the production of PC firewall software as a result (Clark, 2001; Gani, 2002; Goldman, 2002; Harrison, 2000). These products may consist of just a simple system to filter any incoming traffic, to bundled tools that include a firewall, virus scanning, parental content control, privacy control, intrusion detection and encryption (Andress, 2001; Hummel, 2000; Yasin, 2000). Some differences between an enterprise firewall and a PC firewall are shown in Table 1.

While PC firewalls may not have the sophisticated configurations that an enterprise firewall product has, they are purported to provide strong security features that a commercial product provides (Beach, 2001). However, many PC firewall users would debate this claim (Beckman, 2001; Crouch and Captain, 2001; Dalton, 2001; Goldsborough, 2000b; Radcliff, 2001).
Table 1: The differences between an enterprise firewall and a PC firewall (adapted from Whitmore (2002) and Framingham (2001))

<table>
<thead>
<tr>
<th></th>
<th>Enterprise Firewall</th>
<th>PC Firewall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Maintenance cost</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Skill required to operate</td>
<td>High to very high</td>
<td>Low</td>
</tr>
<tr>
<td>Size</td>
<td>Medium to large</td>
<td>Small</td>
</tr>
<tr>
<td>Host software</td>
<td>Minimalist Configuration</td>
<td>Often contains other third parties software and services</td>
</tr>
</tbody>
</table>

This project aims at examining and testing the security features of a selection of PC firewall software.

**METHOD**

The firewalls are downloaded from the Internet and are listed in Table 2. Each is installed on a PC with Microsoft Windows 98 Second Edition Operating system. It is tested using a network security scanner, Nessus. Nessus was installed on a Redhat 7.2 Linux machine.

<table>
<thead>
<tr>
<th>Product Name</th>
<th>Company name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlackIce Defender 2.5 (BlackIce)</td>
<td>Internet Security System</td>
</tr>
<tr>
<td>Deerfield Personal Firewall 1.0.1.0 (Deerfield)</td>
<td>Deerfield.Com</td>
</tr>
<tr>
<td>Kerio Personal Firewall 2.1 (Kerio)</td>
<td>Kerio Technologies</td>
</tr>
<tr>
<td>McAfee Firewall 3.0 (Mcafee)</td>
<td>McAfee Security</td>
</tr>
<tr>
<td>Outpost Firewall 1.0 (Outpost)</td>
<td>Aginitum</td>
</tr>
<tr>
<td>Sygate Personal Firewall Pro 5 (Sygate)</td>
<td>Sygate Technology</td>
</tr>
<tr>
<td>Tiny Personal Firewall v 2 (Tiny)</td>
<td>Kerio Technologies</td>
</tr>
<tr>
<td>ZoneAlarm 2.6 (ZoneAlarm)</td>
<td>Zone Labs</td>
</tr>
</tbody>
</table>

Table 2: Firewalls tested in this study

Nessus is a free software written by Renaud Deraison and Jordan Hrycaj of Nessus Consulting SARL in France (Danielyan, 2001). This software is chosen for its robustness and it is a true client and server application. It has a neat graphical interface that provides an easy-to-use and convenient front-end to the system. Its modular design allows Nessus to be updated easily on a daily basis. In addition, Nessus has a special scripting language, called the Nessus Attack Scripting Language (NASL), which is used to describe vulnerabilities and effects of attacks. (Danielyan, 2001, Daraison, 2000). The program also works in conjunction with NMAP (Fyodor, 1998) using operating system TCP/IP fingerprints to identify systems and whilst providing ping and scan support.

These firewalls were tested on their default settings, minimum settings and maximum settings. Finally, in their default settings, these firewalls are then tested with a popular instant communication software, ICQ, and point-to-point software Bearshare.
RESULTS

Nessus ranks each security weaknesses according to the following Table 3 (Note: Nessus has another risk category also called Security hole with “Serious” risk factor. It is more critical than the “high” risk factor. However, this risk category is not included in Table 3 because none of firewall products reached this level):

<table>
<thead>
<tr>
<th>Risk Category</th>
<th>Risk Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security hole</td>
<td>High: there is security weakness and a hacker is able to breach the security barrier</td>
</tr>
<tr>
<td>Security warning</td>
<td>Medium: there is security weakness, however, a hacker would need to use more sophisticated skills to breach the security barrier</td>
</tr>
<tr>
<td>Security notes</td>
<td>Low: A difficult attack to implement or low real threat</td>
</tr>
</tbody>
</table>

After each complete scan, Nessus would list out the number of risk category as well as itemise each risk or vulnerability found. Where possible it provides suggestions or directs the user to URLs that contain counter measures and fixes. With only the Windows 98 Second Edition Operating System installed, the number of risk category for each firewall is listed in Figure 1. Figure 2 shows the number of risk category with ICQ or Bearshare installed.

Figure 1 reveals that at minimum settings, all firewalls have security flaws that are equivalent to a PC that has no firewall installed. When the security level is set to maximum, all firewalls, except Kirio, successfully protects the PC from an intruder’s probe. At default settings, Deerfield does not show any difference from the baseline. Outpost, Sygate and Zonealarm does not have any detected security flaws. McAfee Personal Firewall receives a security note. Kirio and Tiny does not fix the security hole. In addition, when compared with the baseline, Tiny scores high for security warnings. At default settings, Kirio produces the same security weakness as when its security level is set to maximum (Note Outpost has only one setting).

Figure 2 shows that, apart from the Outpost, Sygate and Zonealarm systems, installing ICQ or Bearshare does not affect the security performances of the firewall when their security levels are set at default. With ICQ installed, Outpost and Sygate receive a security note. Installing Bearshare affects some of the firewall security features; Sygate produces a security note; Zonealarm produces the same number of security holes and warnings as a PC not installed with any firewalls.

DISCUSSION

With the exception of Kirio, none of the firewalls revealed any security weaknesses when it is installed, at maximum settings, along with the operating system. This suggests that with the operating systems alone, the firewall is providing adequate protection from an intruder trying to hack in through the Internet. At default level however, Deerfield, Kirio and Tiny does not eliminate the security holes, but did cause a reduction in the security warnings. Predictably, these security weaknesses are present even after installing ICQ or Bearshare. The detected security holes and security warnings expose the PC to possible attack from the Internet.

Installing ICQ or Bearshare have measurable negative impacts on McAfee, Outpost, Sygate and Zonealarm firewall. This suggests that the installation of third-party softwares can compromise the security integrity provided by the PC firewalls. Furthermore, the adverse impact Bearshare has on Zonealarm firewall suggests that installing the wrong software can in fact, nullify the security provided by the firewall.
Figure 1: Result of scanning with only base operating system installed

Figure 2: Result of scanning with ICQ or Bearshare installed
The findings in this report indicate that the securities of the PC firewalls need further research. Some questions that needed answers include:

1. Would installing a third party software compromise the security integrity provided by the PC firewall, even when its security level is set to maximum?
2. Would installing two or more third party packages have a compounding and negative impact on the security of the firewall?

These questions warrant further investigations because this current study indicates that, at default settings, third party softwares have negative impacts on the security integrity of the PC firewall. A PC may often contain many third party packages. In addition, setting the firewall to maximum may imply to the user that they are fully protected. However, this study has gone some way to suggesting that this may in fact be misleading and may give the PC-user a false sense of security and safety while they are using the Internet when in fact they are open and vulnerable to attack.

REFERENCES

Andress, M., (2001). Building a better virus trap. *InfoWorld*, 23(47), 54


Hulme, G., (2002). Help to combat the next big blended threat. *Informationweek*, 882, 18


If you go down to the Internet today – Deceptive Honeypots

Suen Yek and Craig Valli

School of Computer and Information Science
Edith Cowan University
Email: s.yuek@student.ecu.edu.au

School of Computer and Information Science
Edith Cowan University
Email: c.valli@ecu.edu.au

ABSTRACT

This is preliminary research into the effectiveness of deceptive defensive measures in particular honeypots that use deceit as a primary defensive and offensive mechanism. Initial research has been conducted using the Deception Tool Kit and its ability to fool commonly available network scanning tools such as Nessus and Nmap. The preliminary research indicates that these deceptive tools have a place in a modern network defence architecture.

Keywords: deception, network, security

INTRODUCTION

Attacking trends over the last 5 years have shown Internet connections to be the increasingly cited point of attack (Power, 2002). This challenges the prior conception that most attacks are internal. While inside attacks still show significant numbers, the growth in reported out-sourced attacks up to 60% on WWW/Company sites (Power, 2002).

Statistics show 90% of respondents detected computer security breaches with financial losses within the last 12 months. Furthermore up to 40% detected Denial of Service (DoS) attacks (Power, 2002). These attacks indicate the mounting concern of defacing company reputation combined with theft of proprietary information, and financial fraud.

The goal of this project was to determine the ability for trapping and analysing the results of potentially dangerous attacks on a server when using a honeypot as the prime forensic gathering tool. For this purpose, a honeypot will then be defined as a “resource whose value is in being attacked or compromised” (Spitzner, 2002).

These experiments were carried out in a private and secluded network of eight workstations and a server within the University. The victim machine was running Linux Redhat 7.2 operating system (OS) and had the DTK installed. The attack PC was primarily a Linux Redhat 7.2 machine however Windows2000 was cross platform tool Nmap to confirm results. The DTK itself used several deceptive operating systems with decoy port addresses and outputs to re-direct probes to the DTK. Information regarding the attacks was recorded via the logging features of the DTK and the conventional syslogd daemon. These results were then analysed to deduce the level of effectiveness of the honeypot in a real life situation.
WHY USE A DECEPTIVE HONEYPOT?

The DTK has the ability to deceptively mimic the following operating systems: Windows NT, Linux, HP-UX, SCO Unix, SGI, IBM AIX, Sun Solaris, SunOS and Ultrix. The deception toolkit has been used as the primary architecture of the honeypot. Firstly, it was designed to be used as a defensive tool that systems administrators could use to defend systems. The DTK mimics industry standard servers and the services they provide by listening for inputs and redirecting traffic to customisable Perl script files. These script files then respond as an installed server or daemon for that particular operating system should do when it is fed commands that are legitimate or otherwise.

By deploying bogus services, the machine will appear to contain seemingly numerous useful ports, and it will output responses that are intended to appear typical of a functioning server. While doing this, the DTK will record the actions of the intruder through log files that can then be analysed to determine the modus operandi of the attacker. The deceptive honeypot is intended to extend the time detection window as the attacker is drawn into probing services that are digital chameleons that have no real payload or substance.

The data collected from honeypot testing is “normally of high value” (Spitzner, 2002). This is because information extracted from the analysis can be easily collected, organised, and documented. The high value information includes network activity and movements of the attacker, once in the system. Additionally, honeypots only capture information that is targeted to it. Therefore there is no overwhelming bandwidth or activity to overlook network progress by dropping packets, and potential attacks (Spitzner, 2002). Consequently, there is a more efficient use of resources to provide manageable amounts of useful data for providing attack intelligence to the defender.

The honeypot was designed to act as a fully functional mail server installed within a typical online business running SSH, SMTP and POP3 services. For optimal operation in a real-life situation, the honeypot would be required to be set up in an independent location away from the legitimate servers or within a tightly controlled and protected DMZ (Demilitarised Zone).

Design of the honeypot involves allocating inactive ports to a potentially viable host. This is intended to deceive the attacker into thinking they can receive valuable information from scanning port traffic and determining where that port connects to and identifying flaws in the movement of traffic (McClure, 2002).

The intended function of the honeypot is to confuse and disorientate the attacker by falsely directing them through bogus host lines that may or may not provide information that appears to be informative or even important. As there will seem to be numerous available ports, scanning will take considerably more time, depending on the configuration of the honeypot, and consequential disorientation may result in one of two ways. Firstly, the attacker will be bored, confused or angered and will cease hacking attempts. Secondly, the attacker will believe they have hacked into the system server, and will believe they have received valuable information.

In either case, the honeypot will have achieved its desired purpose in keeping record of the actions taken by the attacker unknowingly, and their consequent attacking strategies.

The diagram above depicts a situation where a potential intruder enters through an Internet connection to the company router. The router has a connection to all network stations, including the honeypot. It then automatically directs the intruder to the honeypot, and away from genuine company assets. Similarly, an internal router directs traffic from within the company network without interference to the honeypot.
METHODOLOGY

There are various types of system testing and penetrating tools freely available to download from the World Wide Web. Two popular attacking softwares chosen were Nessus and Nmap. Both of these tools have won various industry accolades for software innovation and best of breed.

Nessus is a popular network security scanning and auditing tool (Insecure, 2000). Nessus checks for vulnerable systems by detecting all the ports running any given service and then probes and tests their security against known vulnerabilities. Nessus uses a server/daemon: nessusd, and a client nessus. Nessusd monitors the attacks and locates the security holes, then reports them to the client Nessus. The client then interfaces with the user and displays the results.

The results show information on which ports were scanned and corresponding responses. The client output references possible security holes and exposure. More importantly, for the scope of this research, buffer overflows on ports suggest points of entry and manipulation to malicious users to initiate a DoS attack as a subsequent vulnerability.

Nmap (Network Mapper) is a freely available utility used for network exploration or security auditing. By using a technique known as OS fingerprinting (Fyodor, 1998) which examines returned IP packets received from the host Nmap is able to determine hosts available on the network, ports used, any packet filters or firewalls in use and what operating systems and versions are in use.

The attacks were performed through specifying ranges of port to scan. These may be upon the assumption that the potential outside intruder has already performed some form of systematic fingerprinting, which is a tactic used to obtain company profiles of domain names, network blocks, and individual IP addresses connected to the Internet (McClure, 2001). Alternatively, the assumption also can be that the intruder may be internal, where the IP addresses are already known, or easily accessible. Though not the highest, this is also a common source for attacks (Power, 2002).

Once IP address ranges are known, the intruder will perform port scanning in order to determine live hosts. This is often time consuming on the attackers part and is not entirely conclusive or accurate (McClure et al, 2002). The information that a potential intruder will receive can range from complete disclosure of the system’s makeup including operating system (OS), network configuration and loaded services. This then allows the attacker to identifying related OS and service vulnerabilities. As many security vulnerabilities and exploits are dependent on the OS version an attacker can easily adjust their code to attack those weaknesses (Fyodor, 1998).

The DTK was implemented and through with the chosen attacking software brute force attacks and methods were used. The thoroughness of the results of the attacks can then be compared to the actual logged information taken from the DTK log files and the standard syslogd facilities on the Linux system. Thus the honeypot DTK will show its level of effectiveness in distracting the attacker from real port addresses and its potential to prevent hazardous damage.

Limitations on the research are that the seclusion of the experimental network does not connect to the Internet and World Wide Web. This honeypot was designed to act as a fully functional mail server installed and would operate with SSH, SMTP and POP3 services.

TESTING AND EVALUATION PLAN

Nessus was used to brute force the DTK in each of its deceptive OS’s. The probed ports were 1 - 1024, 12345, 1246, 2049, 5999 - 8000, 10000 - 28000. The Maximum number of threads was set to 8 and TCP connect scans were used to probe the ports on the victim host.
When the scan is complete a report is generated by Nessus with any detected security warnings and associated notes a sample follows for the SMTP service when using SGI deception.

**Warning found on port smtp (25/tcp)**

The remote STMP server seems to allow remote users to send mail anonymously by providing a too long argument to the HELO command (more than 1024 chars). This problem may allow bad guys to send hate mail, or threatening mail using your server and keep their anonymity.

Risk factor: Low.

Solution: If you are using sendmail, upgrade to version 8.9.x. If you do not run sendmail, contact your vendor.

CVE: CAN-1999-0098

**Information found on port smtp (25/tcp)**

Remote SMTP server banner:
netsec.ecu SGI ESMTP Sendmail 8.1.2/8.1.3

**ANALYSIS OF RESULTS**

When all scans are complete on each deceptive OS, the log files that DTK generated for each OS is then imported into Excel spreadsheets for further viewing and analysis. Filtered data was retrieved and sorted into the spreadsheet. It was then evaluated by simply counting and recording buffer overflows on each of the probed OS’s.

<table>
<thead>
<tr>
<th>Port</th>
<th>AIX</th>
<th>SGI</th>
<th>SUN</th>
<th>ULTRIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>5</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>25</td>
<td>94</td>
<td>31</td>
<td>2</td>
<td>32</td>
</tr>
<tr>
<td>110</td>
<td>75</td>
<td>24</td>
<td>5</td>
<td>26</td>
</tr>
<tr>
<td>365</td>
<td>5</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>893</td>
<td>5</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>2049</td>
<td>6</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>5999</td>
<td>6</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>6001</td>
<td>6</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>8000</td>
<td>6</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>10000</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12345</td>
<td>4</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>12346</td>
<td>4</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
</tbody>
</table>

Table 1 – Deceptive Buffer Overflows by Port/Service

Many security holes publicised are due to buffer overflows as a form of attack on company servers (Graham, 2000). Therefore it is a common problem faced and is a notable response from the Nessus reports. The number of buffer overflow indicates the number of times the DTK was able to output a red herring to intruders. Where an overflow of data is normally generated, there is a high likelihood that the program will crash or give an intruder root or high level access or privilege to a system.

Nessus believed it detected the following problems with the various deceptive operating systems
A naïve hacker or script kiddie would typically rely heavily on tools such as Nessus to provide them with potential targets that they could compromise (Conry-Murray, 2001). This reliance on pre-compiled tools where they do not have to understand or be able to manually execute the attack is a hole in their offensive strategy. This enables the defenders to extend the detection window for an attacker as they are literally shadow boxing in a deceptive honeypot while leaving forensic trails of their activities.

Although the detected problems were rated as low they still leave an opportunity for a hacker to attempt to attack the system.

The SMTP service was the service that provided the most deceptive information and demonstrated the highest level of buffer overflows to a would-be attacker. All of the deceptive OS implementation provided faked remote banners that took the form of netsec.ecu (Deceptive OS) ESMTP Sendmail 8.1.2/8.1.3. In hacking guides (Anonymous, 2002; Fadia, 2002) commonly available on the Internet that target the SMTP service and in particular the Sendmail program this information is used to determine what sort of attack the attacker should deploy to penetrate or dupe the system. This is some of the initial intelligence gathering that an attacker would undertake. Based on that knowledge they would then attempt various exploits on the system. Nessus believed it perpetrated buffer overflows that resulted in denial of service or allowed the successful anonymous relay of mail. It also further

### Table 2 - Nessus Results No Dangerous Plugins Used

<table>
<thead>
<tr>
<th>Security</th>
<th>Rating of Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Holes</td>
</tr>
<tr>
<td>LINUX</td>
<td>0</td>
</tr>
<tr>
<td>NT</td>
<td>0</td>
</tr>
<tr>
<td>SOLARIS</td>
<td>0</td>
</tr>
<tr>
<td>HPUX</td>
<td>0</td>
</tr>
<tr>
<td>SUNOS</td>
<td>0</td>
</tr>
<tr>
<td>AIX</td>
<td>0</td>
</tr>
<tr>
<td>SGI</td>
<td>0</td>
</tr>
<tr>
<td>Ultrix</td>
<td>0</td>
</tr>
<tr>
<td>SCO</td>
<td>0</td>
</tr>
</tbody>
</table>

### Table 3 - Nessus Results Dangerous Plugins Used

<table>
<thead>
<tr>
<th>Security</th>
<th>Rating of Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Holes</td>
</tr>
<tr>
<td>LINUX</td>
<td>0</td>
</tr>
<tr>
<td>NT</td>
<td>0</td>
</tr>
<tr>
<td>SOLARIS</td>
<td>0</td>
</tr>
<tr>
<td>HPUX</td>
<td>0</td>
</tr>
<tr>
<td>SUNOS</td>
<td>0</td>
</tr>
<tr>
<td>AIX</td>
<td>0</td>
</tr>
<tr>
<td>SGI</td>
<td>0</td>
</tr>
<tr>
<td>Ultrix</td>
<td>0</td>
</tr>
<tr>
<td>SCO</td>
<td>0</td>
</tr>
</tbody>
</table>
believed that the mail server was an open mail relay (Finlay, King, & Herman, 2001; Rosenthal, 2002) which allows malicious attackers to pass mail through the server to other persons.

The SSH service in deception mode mimicked buffer overflows and core dumps where by the attacker would once again thought they would have performed a successful denial of service on the SSH daemon. The POP3 service also gave away banner information which a hacker could use in the same manner as the SMTP banner to search for vulnerabilities.

All of the deceptive OS’s demonstrated to Nessus vulnerabilities that simply did not exist in any real form on the victim system. This is further confirms the veracity of claims (Cohen, 1998) “The net effect is that attack tools that automatically scan for known vulnerabilities find what appear to be large volumes of vulnerabilities. When the attacker tries to interpret the results of automated scans, there is not enough information to tell which of the detected vulnerabilities are real, and the number of detected vulnerabilities is very high and dominated by deceptions.”

One of the problems encountered was the DTK’s inability to counteract Nmap OS fingerprinting techniques. The DTK returned consistently inconclusive results guessing that the OS was Standard: Solaris 2.x, Linux 2.1.???, Linux 2.2, MacOS. This gives the attacker a choice of four OS’s to choose or potentially the fingerprint of a DTK. This is would pose a potential weakness when multi-homing sites on the one system using network aliasing.

CONCLUSION

The use of deception for defence has been around since the dawn of time and will be here from some time to come. Its place in a network defensive strategy is still relatively unclear. The use of deception based systems such as DTK has the ability to fool many of the common scanners used by naïve or inexperienced attackers. This then leaves the naïve hacker at the mercy of their ignorance in successfully attacking the real system that in turn provides the defender time to instigate countermeasures to prevent further attack or redirect further attempted incursions.

The DTK as tested provided extensive forensic data in its log files and to the syslogd functions on the attacked system. This forensic data would aid greatly in the investigation of an attempted break in.

The use of deceptive honeypots has weaknesses that need examination and further resolution if they are to be effective as a defensive mechanism. With the advent of multi-homing on one system/interface the ability to deceptively portray that interface as multiple systems/interfaces is an important extension of a deceptive honeypot to cope with modern networking technologies. Whether a more intricate deception that is more detailed and descriptive will aid in increasing the deception needs further investigating.
REFERENCES


Finlay, I., King, B., & Hernan, S. (2001). CERT® Incident Note IN-2001-02 - Open mail relays used to deliver "Hybris Worm": CERT.


With Speed The Hacker Cometh...

Craig Valli

School of Computer and Information Science
Edith Cowan University, Australia
Email: c.valli@ecu.edu.au

ABSTRACT

This paper is an examination of six months of Intrusion Detection System (IDS) reports and firewall log entries for a small enterprise that has a new broadband ADSL connection. The paper examines the information contained in the logfiles and the implications of detected activities by would be attackers. An examination of the issues that the deployment of broadband has for home and small business users is also undertaken.

Keywords: broadband, intrusion detection, firewalls, home users, ADSL

INTRODUCTION

This paper is an examination of six months of Intrusion Detection System (IDS) reports and firewall log entries for a small enterprise that has a new broadband 1.5Mbit ADSL connection. Over 60,000 log file entries from machines within the enterprise were examined for this case study. The author used simple statistical measures to examine the information contained in the log files.

The paper examines what patterns of scanning or attacks were targeted at the machines in the enterprise. Timing of attacks on various ports or services will be checked against releases of known vulnerabilities from security vulnerability databases such as CERT and CVE. The paper then discusses the implications of the findings of the case study and looks specifically at issues for the deployment of broadband connections for the home and small business user.

THE NETWORK LAYOUT

The IDS and firewall on the two servers in the DMZ detected and logged over 60,000 suspicious events from May 2001 to November 2001. The enterprise upgraded from an existing 128K ISDN connection to a new 1.5MB ADSL connection for the primary reason of increased bandwidth in May 2001. The other main benefit was that of cost savings in physical provision of a connection to the Internet for the enterprise down from $4400 per annum to just $700 per annum. The 2 servers were both hardened RedHat 6.2 Linux boxes that existed within a DMZ configuration. see figure 1 below

![Figure 1 - network layout](image-url)
Hera is a dual homed bastion host that acts as the secondary router in the DMZ design. Both Hera and Neptune run a basic firewalling rule set that performs mandatory blocks of access from bogus or undelegated IP address ranges. In addition it denies access to all server ports other than what is meant to be open and available for access these ports are outlined in Table 1.

<table>
<thead>
<tr>
<th>Hera</th>
<th>Port 22</th>
<th>SSH</th>
<th>Port 25</th>
<th>SMTP</th>
<th>Port 53</th>
<th>DNS</th>
<th>Port 80</th>
<th>HTTP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neptune</td>
<td>Port 22</td>
<td>SSH</td>
<td>Port 53</td>
<td>DNS</td>
<td>Port 80</td>
<td>HTTP</td>
<td>Port 443</td>
<td>HTTPS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1 – Open Ports on Servers

The primary DNS for the organization is located on Hera with Neptune providing secondary backup services. The intrusion detection system (IDS) used was Portsentry (Psionic, 2001) set with a high trigger level to lower false positives in the detection of suspicious network activity. Upon detecting suspicious activity the IDS triggered a firewall rule that then blocked all traffic from the attacking IP.

Full logging via syslogd was done on both these machines and logs were secured via a script that extracted relevant information from the logfiles daily and forwarded them via e-mail to a machine (Apollo) located beyond the DMZ.

The script extracted two main forms of data that was all Portsentry IDS alerts and messages plus the entire kernel based firewall alerts and messages. These messages showed the initiation of dynamic blocking of sites that triggered the IDS. A typical firewall denial entry looks like this:

```
Nov 30 15:04:36 neptune kernel: Packet log: input DENY eth0 PROTO=17 ATTACK_IP:32800 NEPTUNE:53 L=60 S=0x000F O=8941 F=0x0000 T=8 (#64)
```

The logfiles yielded 64073 unique lines of firewall-based information, Hera generated 61565 of these and the remaining 2489 were generated by Neptune. This was obtained by concatenating the logfiles together in raw format initially. Then using a customized Perl script to extract information from these logfiles in the DSHIELD format (Consulting, 2002). This is a tab-delimited file that contains for each line:

1. Date
2. Host (UserID)
3. Count (number, used to summarize identical records, default=1)
4. Source IP address (in 1.2.3.4 format)
5. Source port
6. Target IP address
7. Target port
8. Protocol
9. TCP flags – SYN, ACK, FIN, URG, RES

These extracted log entries were then imported into an Access Database and SPSS for basic statistical analysis and examination of any trends or patterns in the data set.

**ANALYSIS OF LOG FILE ENTRIES**

There were 1201 unique attacking IP numbers recorded by the log files and identified by the analysis process. Of these attacking IP numbers the top 15 displayed in Table 2 below were responsible for 46095 (71.9%) of all logged attack entries to the machines in the DMZ. The top 15 attackers activities can really only be considered as brute forced automated attacks upon the systems. Upon further
analysis of the log files most of the top 15 attackers did so in the period 12th – 15th August 2001 where 49320 (80.1%) of all logged attack lines were recorded. During this period Code Red (CERT/CC, 2001b) was emergent and some of these log entries for this period showed Code Red signatures. The problem was that the attack that caused this traffic was not in fact a Code Red attack but was an attempted distributed denial of service attack upon the DNS services on port 53 directed at Hera which is the primary DNS server for the network.

<table>
<thead>
<tr>
<th>Attacker</th>
<th>Frequency</th>
<th>Percent</th>
<th>Cumulative Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 attacker</td>
<td>13376</td>
<td>20.9</td>
<td>20.9</td>
</tr>
<tr>
<td>02 attacker</td>
<td>7370</td>
<td>11.5</td>
<td>32.4</td>
</tr>
<tr>
<td>03 attacker</td>
<td>3855</td>
<td>6.0</td>
<td>38.4</td>
</tr>
<tr>
<td>04 attacker</td>
<td>3832</td>
<td>6.0</td>
<td>44.4</td>
</tr>
<tr>
<td>05 attacker</td>
<td>2562</td>
<td>4.0</td>
<td>48.4</td>
</tr>
<tr>
<td>06 attacker</td>
<td>2281</td>
<td>3.6</td>
<td>51.9</td>
</tr>
<tr>
<td>07 attacker</td>
<td>1771</td>
<td>2.8</td>
<td>54.7</td>
</tr>
<tr>
<td>08 attacker</td>
<td>1548</td>
<td>2.4</td>
<td>57.1</td>
</tr>
<tr>
<td>09 attacker</td>
<td>1545</td>
<td>2.4</td>
<td>59.5</td>
</tr>
<tr>
<td>10 attacker</td>
<td>1544</td>
<td>2.4</td>
<td>61.9</td>
</tr>
<tr>
<td>11 attacker</td>
<td>1476</td>
<td>2.3</td>
<td>64.2</td>
</tr>
<tr>
<td>12 attacker</td>
<td>1380</td>
<td>2.2</td>
<td>66.4</td>
</tr>
<tr>
<td>13 attacker</td>
<td>1199</td>
<td>1.9</td>
<td>88.3</td>
</tr>
<tr>
<td>14 attacker</td>
<td>1181</td>
<td>1.8</td>
<td>70.1</td>
</tr>
<tr>
<td>15 attacker</td>
<td>1171</td>
<td>1.8</td>
<td>71.9</td>
</tr>
<tr>
<td>All Others</td>
<td>17978</td>
<td>28.1</td>
<td>100.0</td>
</tr>
<tr>
<td>Total</td>
<td>64069</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 2 – Number of Scans per top attacker

The logfiles on Hera 10th August showed attacker01 attempting what appears to have been an operating system fingerprinting scan. Then on 11th August a selective scan on ports 21, 22, 25, 53, 80, 111, 443, 520, 995, 3128, 3130, 8080 would have indicated that the attacker at least knew they were dealing with a UNIX based system. The evidence for this was that they went after port 111 (AUTH), 3128 and 3130 that are Squid Proxy default ports that are only found on UNIX/POSIX variants. To add further weight to the argument there was no scan done of the SMB and NetBIOS ports 137, 138 and 139, which an inexperienced hacker or hacker targeting Windows based machines would have attempted.

The Portsentry IDS placed blocking routes for the IP address of the attacker during the scan on port 53 effectively dropping all of their traffic to this service. Then on the 12th August an attack was directed at the Domain Name Services (DNS) daemon on Hera, which resulted in the 43116 log entries relating to attacks on the DNS services over a 74 hour period, or an average of just less than 600 probes an hour.

The controller in this case attacker01 appeared to control or compromise the systems in groups of three whether this was limitation of bandwidth or the tool that was used is unsure and cannot be determined from the evidence. The majority of the attack patterns were cycled consistently e.g. controller, host1, host2, host3, controller, host1, host2, host3, controller, ..... which further indicates an automated or script based attack in nature.
The time to live (TTL) between each host also varied across compromised host groupings with one group capable of a less than 1 second delay to almost 5-10 second TTL delay between hosts in another group. The compromised hosts used static ports from which to commission the attacks. But the selection of port number used to attack was not consistent across the compromised attack devices. The other interesting point was that the length of packet sent also varied in each of the attack packets from a Length of 44 bytes to 324 bytes. The packet patterns indicated fragmented attacks and attempted buffer overflows by the attacker. Which due to the speed and the complexity of launching these packets manually would further indicate automation of the attacks.

The effect of the attack was that the primary domain server performance was severely compromised during this attack. The owners of the system emailed the administrator of the domain for attacker01. The administrator immediately responded and shutdown the compromised server which halted the distributed denial of service.

<table>
<thead>
<tr>
<th>Port</th>
<th>Protocol</th>
<th>Frequency</th>
<th>Percent</th>
<th>Cumulative Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>DNS</td>
<td>47997</td>
<td>74.91</td>
<td>74.91</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
<td>8459</td>
<td>13.20</td>
<td>88.11</td>
</tr>
<tr>
<td>25</td>
<td>SMTP</td>
<td>2743</td>
<td>4.28</td>
<td>92.39</td>
</tr>
<tr>
<td>520</td>
<td>EFS</td>
<td>1548</td>
<td>2.42</td>
<td>94.81</td>
</tr>
<tr>
<td>111</td>
<td>SUNRPC</td>
<td>731</td>
<td>1.14</td>
<td>95.95</td>
</tr>
<tr>
<td>21</td>
<td>FTP</td>
<td>334</td>
<td>0.52</td>
<td>96.47</td>
</tr>
<tr>
<td>137</td>
<td>NETBIOS-NS</td>
<td>74</td>
<td>0.12</td>
<td>96.59</td>
</tr>
<tr>
<td>17300</td>
<td>KUANG2THEVIRUS</td>
<td>33</td>
<td>0.05</td>
<td>96.64</td>
</tr>
<tr>
<td>4417</td>
<td>?????</td>
<td>27</td>
<td>0.04</td>
<td>96.68</td>
</tr>
<tr>
<td>3053</td>
<td>DSOM-SERVER</td>
<td>20</td>
<td>0.03</td>
<td>96.71</td>
</tr>
<tr>
<td>22</td>
<td>SSH</td>
<td>19</td>
<td>0.03</td>
<td>96.74</td>
</tr>
<tr>
<td>443</td>
<td>HTTPS</td>
<td>17</td>
<td>0.03</td>
<td>96.77</td>
</tr>
</tbody>
</table>

Table 3 - Port Scan Frequency

The high percentage of domain name service based attacks concurs with new vulnerabilities (CERT/CC, 2001a, CERT/CC, 2001d) that were found in the BIND daemon which is used to run DNS services in 2001. Many machines if unpatched were susceptible to these attacks some of which allowed for root/administrator compromise of the system as well as buffer overflow based denials of service. Some of the logged entries attempted to send large packets of over 900 bytes to the BIND servers that would have caused buffer overflow based attacks if the servers were in fact vulnerable to that particular attack.
Over 60% of all of the malicious traffic to port 80 were SYN packets sent by a wide range of attacking IP numbers. The packets were either SYN scans or maliciously formed denial of service packets. Of the remaining log file entries in the http daemons log for the corresponding timestamps the probes were against known CGI vulnerabilities and known buffer overflow exploits. The speed at which these probes took place would indicate the use of automated vulnerability scanners or they were pre-programmed by the attackers. There was a high proportion of NIMDA (CERT/CC, 2001c) based attacks in this traffic as would be expected for the latter section of the log file entries relating to the http services.

SMTP (Simple Mail Transport Protocol) was the next most highly probed service. SMTP is one of the most vulnerable services that is run on servers due to its complexity of configuration and overall size. Many of the corresponding entries again indicated activity of automated tools for the majority of traffic simply brute probing for known vulnerabilities. SUNRPC is also a highly vulnerable port on a Linux based PC.

FTP services were scanned 334 times which is interesting considering there was no services or inetd configuration for this service. The ftp binaries had in fact been removed from the system. This would indicate that many of the attackers were naïve or careless in their probing of the systems. FTP is also one of the most vulnerable services that is deployed on systems and hence automated vulnerability scanners will probe for FTP many vulnerabilities.

**IMPLICATIONS OF THE LOG FILE ANALYSIS**

These are logfiles that are gathered from systems where the administrator is a competent network security professional. The problem is that many home and small business users are now adopting ADSL and other broadband technologies as replacements for conventional modems and are largely unaware of the increased security implications and are typically not computer or network security literate (Glass, 2001, Goldberg, 2000, Hoffman, 2000). Modems are typically restricted and hampered in their ability to sustain a high intensity scan due to bandwidth and technological barriers. ADSL on the other hand does not have such impediments and can see home users with a 1.5 Megabit per second connection to the Internet awaiting an attack by those with malicious intent.

This faster connection leaves the broadband user with a decreased attack detection window in which to detect and respond to probes by the attacker. Conversely, it allows the attacker to spend a smaller amount of time in a position where they may be detected scanning or probing a host. With a high bandwidth connection they are also able to scan a wider range of ports due to this increased speed. Based on the assumption that ADSL is 30-50 times faster than a conventional 56K modem this means an attack that took 5 minutes to commission previously could now take place in as little as 6-10 seconds with ADSL.

The logfiles showed patterns of attack that are consistent with the widespread use of automated attack tools and vulnerability scanners. This is consistent with data from other authors where there is an increase in scanning activity on the Internet brought about by script kiddies activity (Conry-Murray, 2001a, Conry-Murray, 2001b, Harrison, 1999). Much of this activity is often done without knowledge of the tools used or their true destructive capabilities.

Many of the attacks that were recorded in the logfiles were also perpetrated as a result of vulnerabilities becoming "known" to the wider public through release of these details on legitimate sites such as CERT or CVE. These activities would indicate again bulk scanning and probing for the new vulnerabilities by attackers with little understanding or knowledge of the attack as many vulnerability scanners will quickly incorporate any new vulnerabilities into their scanning database for use by legitimate users.
CONCLUSION

The introduction of broadband technology has many benefits all of which can work against the user from a security perspective. This case has demonstrated that users are potentially susceptible to a wide range of attacks due to the increase in available bandwidth to their home desktops through the adoption of broadband. There are several questions that arise for instance does the ISP have an obligation to provide more secure services to broadband customers? Are the risks of successful attack raised by the installation of broadband services, which in turn provides reduced attack and detect windows? To what extent does the home user or small business have the ability to recognize or even defend against an attempted automated attack?

Many of the users of new mass marketed broadband services it could be argued would lack basic network security skills. To further exacerbate the problem experiences of cases like (Love, 2001) would indicate that software vendors are not aiding novice users in the effective deployment of personal and home use firewalls. Further research is needed in this area to identify the risk that broadband potentially raises as it becomes more ubiquitous in deployment leaving an increasingly larger collection of viable victim systems for hackers to compromise.

REFERENCES


Swarming Attacks and Agents

M.J. Warren, M. McDougall and K. Pascoe

School of Information Technology, Deakin University,
Geelong, Victoria, Australia.
E-mail: mwarren@deakin.edu.au

ABSTRACT

Swarming is a new security application of a very old idea. The idea of swarming is to attack a host or organization with multiple attacks. These types of attacks can be hard to defend against as illustrated by past events (discussed within the paper). The paper will also explore how new technologies such as software agents could be used in order to develop the ultimate swarming attack method.

Keywords: Swarming, Security, Software Agents

INTRODUCTION

The concept of swarming originated from nature. ‘Hive’ or ‘nesting’ organizations, such as ants or bees, use ‘blanketing’ tactics, which is striking the target from all directions. The methods of swarming differ between ants and bees. Bees can only swarm once as stinging results in the stinger’s own death. Ants use a behavior called ‘swarm raiding’. That is, they move in linear formations, but can shift into swarming mode when it is time to attack. Ants use swarming in pursuit of food, defense of the hive, and also in territorial wars against other ants (Ronfeldt and Arquilla, 2000).

One definition of swarming is:

‘Swarming is a seemingly amorphous, but deliberately structured, coordinated, strategic way to strike from all directions at a particular point or points, by means of a sustainable pulsing of force and or fire, close in as well as from stand-off positions. This notion of “force and/or fire” may be literal in the case of military or police operations, but metaphorical in the case of NGO activists, who may, for example, be blocking city intersections or emitting volleys of email and faxes. Swarming will work best – perhaps it will only work - if it is designed mainly around the deployment of myriad, small, dispersed networked maneuver units. Swarming occurs when the dispersed units of a network of small (and perhaps some large) forces converge on a target from multiple directions. The overall aim is sustainable pulsing – swarm networks must be able to coalesce rapidly and stealthily on a target, then dissever and redisperse, immediately ready to recombine for a new pulse. The capacity for a “stealthy approach” suggests that, in netwar, attacks are more likely to occur in “swarms” than in more traditional “waves.”’ (Ronfeldt and Arquilla, 2001)

The attacking computers do not necessarily have to surround the target, like military forces or ants in nature. There just needs to be a sufficient number of computers participating in the coordinated attack. The number of attacking computers could total in the hundreds or possibly thousands. The attack could be repeated requests for a web page, repeatedly pinging the target, or anything that will cause the target computer to be swamped with unnecessary processing or information and thus result in severe debilitation.
CURRENT RESEARCH INTO SWARMING

Most research into swarming technologies has little to do with swarming attacks in the context of this paper. Much research in swarming has to do more with flocking, rather than what we consider to be swarming. It is important to differentiate between flocking and what is considered swarming in this paper. We are using the concept of swarming to imply attacking from all directions simultaneously. Flocking, on the other hand, is synonymous to the way birds flock together when they travel.

Some other swarming research areas are SWARM (see http://www.swarm.org), which is a simulation system. It is a toolkit for building multi-agent simulations, one goal being to aid researchers to create specific computer-related research tools. Another is ISAAC/EINSTEIN (see http://www.can.org/Isaac), which is a simulation of a swarming attack in a military sense.

A relatively new research area is swarming intelligence, which uses the observations of swarms in nature to imitate these phenomena in programs to perform useful tasks. An example is in the way ants can always find the shortest path, even when their environment has been altered. This phenomenon is used to create routing algorithms using swarming intelligence and mobile agents. The possibility of using swarming intelligence to develop the technology for swarming attacks is uncertain. Observing the behavior of ‘swarm raiding’ by ants may be used, but there are not enough parallels between swarming by ants and swarming over networks, as physically surrounding the target is not necessary in cyberspace. Analyzing ‘swarm raiding’ to create military simulations of swarming attacks may be more feasible, but again, there is some uncertainty as to the extent of parallels between military swarming and swarm-raiding ants.

DDOS (Distributed Denial of Service)

A Denial of Service (DOS) is an attack on a computer that is used to deny the user use of the system. An example of this is a Smurf attack where the attacker sends a large amount of IMCP echo (ping) traffic to broadcast addresses. If the added traffic becomes too great, then the computer will have to expend too many resources processing the ICMP echo traffic, thus rendering the computer unusable to legitimate users.

DDOS, is an advancement on DOS. DDOS still performs a Denial of Service attack on a target, but the attack comes from many computers attacking simultaneously. How this is achieved is shown in Figure 1 and explained below the figure.

![Figure 1 DDOS Structure](image)

There are four main types of systems involved in a DDOS attack. They are the Target, Agent, Handler and Client. Different sources may have different names for the Handlers and Agents, but their functionality is the same. The Target is the computer to attack or victimize. The Agent and Handlers
are the compromised computers. That is, they are computers that have been infiltrated due to weak security and have DDOS tools installed on them. The Agent is where the disabling network traffic is generated from. The Handler controls the Agents, maintains a list of all responding Agents and signals the Agents when to begin an attack as well as the method of attack. The Client controls the Handlers by telling them when to inform the agents to attack, the method of attack, etc. It is clear that there is a hierarchy of control in a DDOS attack (Criscuolo, 2000).

Performing a DDOS attack is a two-phase step (Criscuolo, 2000):

Phase 1: Compromising as many systems as possible. This involves infiltrating systems with weak security to install the DDOS tools as well as programs such as rootkit to hide the DDOS tools' presence;

Phase 2: Once the systems have been compromised, the attack can begin. The compromised systems will generate the network traffic to bring down the targeted site.

Difficulties in dealing with DDOS attacks

DDOS attacks do not capture or modify data, but as their purpose is solely in swamping the target to the extent that the target becomes unusable for legitimate users. The client controls the attack remotely. Encryption is used to hide communication and forged source addresses are used to conceal their location. It is difficult to trace the source due to stealth capabilities built into the DDOS tools. If tracing the agent is possible, then it is necessary to trace the handler, followed by the client (Criscuolo, 2000).

There are many problems that a business faces when attempting to deal with an attack. Although it is possible to detect the attack, there is very little that can be done to stop it (Lemos, 1999). One possibility is shutting down the system, which of course effectively denies the legitimate users use of the Internet, which is what the attack was designed to do in the first place.

Since the attack uses so many servers, which are controlled by a single hacker, determining the source of the attack is impossible. Each connection to the victim's site can be made to appear legitimate as they are coming from different sources. Generally, when under a DOS attack, you can deal with this kind of attack by blocking the address that is attacking. But, since the attack is coming from such a large number of addresses, you cannot block every address of a computer that is trying to connect to the site because many of these connections may be legitimate. It may be possible to block some addresses that are clearly a part of the attack, but not all. Blocking all of the addresses is effectively the same as shutting down the system (Lemos, 1999).

The biggest task in setting up for one of these attacks is finding enough systems to infiltrate to install the necessary attack tools. Unfortunately, this is not all that difficult to do. The security measures taken by too many people are inadequate, simply due to the fact that they know very little about security. Also, hackers are automating this process by using programs to scan the Internet and networks connected to the Internet looking for known security holes (Lemos, 1999). Mobile agents are ideally suited for this purpose. Mobile agents will be addressed later in this paper.

Distributed coordinated attack

Another term used to describe a concerted attack on a target computer is a distributed coordinated attack (DCA). The major concept behind a distributed coordinated attack is to utilize a large number, potentially hundreds, of computers, i.e. workstations or servers, to coordinate an attack on a single computer/server. The attack is intended to disable the server by requesting information, e-mails or bombarding it with e-mails to the extent that the server is totally inundated with information and can
no longer function (Cohen, 1996). Clearly a DCA is another or similar term for a swarm attack or a DDOS attack.

An example of a DCA occurred in March 1996. An attack in which the server contained code directing a browser to attack other sites was used in conjunction with the automatic loading of background patterns provided by many web browsers. Users at thousands of sites across the Internet automatically, without their knowledge or consent, and in the background, attempted to telnet into all.net (Cohen, 1996).

The following is a list of characteristics of Distributed Coordinated Attacks that make them difficult to prevent and identify the source, causing them to represent a serious threat.

**Characteristics of DCA’s**

The source of the attack is only indirectly related to the site that appears to be launching the attack. This means that it may be impossible to track down the real source of the attack without the cooperation of people at two or more sites.

A DCA will likely involve a high overall rate of attacks even though the contribution to this rate by each vector site may be low or even singular. Although DCA attackers may keep rates below the detection threshold of the victim, it is likely that most perpetrators will select higher rates of attack believing that they are adequately shielded by indirection.

Vectors in DCA’s are likely to be completely unaware that their system is being exploited. It may be difficult to explain this to systems administrators who have innocent users that are unwittingly made to participate in an attack which doesn’t lead to any outward indication on their system. (Cohen, 1996)

**SWARMING IN CYBERSPACE**

Electronic Civil Disobedience (ECD) works on the theory of protest actions, but rather than setting up physical blockades, their protest medium is cyberspace. This enables protesters to take part in a sit-in or demonstration from anywhere in the world (Wray, 2002). The Electronic Disturbance Theatre (EDT) is a group of activists who engage in ECD as the form of protest. Much of this section is focused on the EDT, as these people have actively pursued and participated in mass swarming demonstrations over the Internet. Their use of swarming techniques demonstrates a possible future direction of DOS attacks. FloodNet is a web based software device developed by EDT, and is designed to cripple a target’s website (Wray, 2002).

**FloodNet**

FloodNet was originally designed as a tool in support of the Zapatistas. The original intention was for the activists, at a pre specified time, to continuously hit the reload button on their browsers, which was pointed to certain Mexican sites. The theory was that if enough people took part in this sit-in, the sites would become flooded with requests to view the page, and therefore the site would be blocked. FloodNet is a java applet that automated this process. Its first use was in a protest against Mexican President Zedillo’s website, where every 7 seconds FloodNet would request to view the page. As a result, the site was effectively blocked, with over 8000 protestors participating in the attack (Wray, 2002).

FloodNet was also used in other protests, the targets being other Mexican government sites and US corporate sites. Countermeasures have been devised against the FloodNet attacks. In Mexico, JavaScript code repeatedly opens a window in the attacking browser, eventually disabling the browser. From the Pentagon, a java applet called ‘hostileapplet’ generated a series of rapidly appearing Java coffee cups across the bottom of the browser screen coupled with the phrase ‘ACK’ (Parkins, 2000).
The EDT intends to continue to develop a variety of tools such as FloodNet, which are accessible and easy to use by cyber activists. A project called SWARM is one of these (see http://www.nyu.edu/projects/wray). The EDT consider SWARM as follows:

‘Think of a swarm as an array of FloodNet-like devices, arising, acting, and dispersing simultaneously against an array of cyberspacial political targets. If the electronic pulses generated by our FloodNet actions are represented by a small mountain stream, the electronic pulses generated by a swarm of convergent ECD actions are a raging torrent.’ (Wray, 2002)

Tribal FloodNet

The problem with (or possibly the purpose behind, depending on your viewpoint) FloodNet, is that its success is, among other things, dependent on having a large number of participants in the attack. A new tool, called Tribal FloodNet, programmed by a German hacker named Mixter, enables one person to launch a massive DDOS attack. Tribal FloodNet affected Yahoo and other US sites in 2000 (Ronfeldt and Arquilla, 2001).

These tools have led to controversy with ‘hacktivists’ including EDT, Electrohippies and Cult of the Dead Cow, to name a few. These ‘hacktivists’ claim they are undemocratic and secretive. However, this shows that more efficient means of bombarding a target with a coordinated attack are in development, and many tools are available at present (Ronfeldt and Arquilla, 2001).

Assessment of Swarming

The idea put forward here is whether swarming is an issue that needs to be considered in the future. The concept of performing a distributed attack on a target is clearly a goal for many ‘hacktivists’, as supported by their FloodNet system, and SWARM proposal.

Several swarming attack methods to cause DDOS have been outlined in this paper. The FloodNet system was designed to swamp web pages with the attacks coming from many sources attacking simultaneously. The sources in this case are a mass of individuals worldwide, initializing their FloodNet system at a given time.

The introduction of Tribal FloodNet has enabled the same objectives to be achieved by one person, making the tool much more automatic. Also, since one person can generate the attack, it is clear that the idea of swarming has more appeal than declaring protests by hacktivists since it can be used as a genuine malicious attack.

Another aspect that makes swarming a serious issue is that since the attack comes from so many places, it is difficult to trace the source. Also once under an attack, little can be done to stop it. The only viable way to prevent these attacks is to ensure that every network connected to the Internet is secure.

In conclusion, swarming attacks, that is, performing distributed attacks on a target simultaneously, is a current issue. Also, it is clear that more automatic ways of issuing swarming attacks are being considered, developed and implemented. The use of agents is being implemented to assist in the setting up, i.e., using programs to scan the Internet and looking for security limitations to invade a system to install DDOS attack tools.

FUTURE DEVELOPMENT OF SWARMING TECHNOLOGY

This section of the paper introduces Mobile Agents and discusses the possibility of using this technology in the development of swarming attacks. Much of this section is speculation and theory in order to hypothesize where swarming technology could be heading in the future. It will cover a definition of mobile agents, the properties of mobile agents, the potential use of mobile agents in
swarming attacks and associated problems or difficulties that need to be overcome if mobile agents are to be used in swarming attacks.

Mobile Agents

There are many different definitions of agents in existence, so providing a single absolute definition is almost impossible. Franklin and Graesser (1996) define agents as:

'An autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future.'

Mobile agents can be used to search for information that is distributed across web servers and other systems. The agents can transport themselves over large networks, or from one network to the next in order to complete their tasks. Allowing the agents to communicate with each other and share information can lead to the agents being able to carry out more complex tasks than can be achieved autonomously (Busuttil and Warren, 2001).

Table 1 below displays a list of nine properties for Mobile Agents (refer online to http://www.msci.memphis.edu/~franklin/AgentProg.html). Not all agents have every property listed, but most agents have at least the first four properties.

<table>
<thead>
<tr>
<th>Property</th>
<th>Other Names</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reactive</td>
<td>Sensing and acting</td>
<td>Responds in a timely fashion to changes in the environment</td>
</tr>
<tr>
<td>Autonomous</td>
<td></td>
<td>Exercises control over its own actions</td>
</tr>
<tr>
<td>Goal-oriented</td>
<td>Pro-active purposeful</td>
<td>Does not simply act in response to the environment</td>
</tr>
<tr>
<td>Temporally continuous</td>
<td></td>
<td>Is a continuously running process</td>
</tr>
<tr>
<td>Communicative</td>
<td>Socially able</td>
<td>Communicates with other agents, perhaps including people</td>
</tr>
<tr>
<td>Learning</td>
<td>Adaptive</td>
<td>Changes its behavior based on its previous experience</td>
</tr>
<tr>
<td>Mobile</td>
<td></td>
<td>Able to transport itself from one machine to another</td>
</tr>
<tr>
<td>Flexible</td>
<td></td>
<td>Actions are not scripted</td>
</tr>
<tr>
<td>Character</td>
<td></td>
<td>Believable ‘personality’ and emotional state</td>
</tr>
</tbody>
</table>

Table 1: Properties of Agents

Mobile agents have some very useful properties. They can be used to scan the Internet for information or to find the best price on a product, for example. However, as with any technology, mobile agents can also be used in a malicious manner. They can be programmed to cause damage to systems and other agents (Busuttil and Warren, 2001).

Using mobile agents to perform swarming attacks is a very realistic possibility. Consider the Code Red worm, which was programmed to hit the US government site with DOS attacks. The Code Red worm uses a security hole in the Microsoft server software. Every time the worm finds a Microsoft IIS4.0 or 5.0 web server running Windows NT 4.0 or 2000 operating system, it scans the Internet and replicates itself on other servers matching that description. It defaces web pages with the message ‘Hello! Welcome to http://www.worm.com!/ Hacked by Chinese!’. The cycle of the worm per month is as follows:

1st - 19th: replicates and spreads, defacing web pages
20th - 27th: launches Denial of Service attacks on the White House
28th - 1st of following month: it goes dormant.
As a note, the attack on the White House was prevented and a patch made available to repair the security hole in an attempt to clean up the worm (Korea Times, 2001).

The purpose of the worm reflects what would be required of the mobile agent. This worm was programmed to replicate and initiate DOS attacks on the White House. It was very much a swarming attack, as potentially thousands of worms were preprogrammed to strike the White House simultaneously. Admittedly, the attack was unsuccessful, but it shows that swarming attacks with mobile agents are possible.

As stated previously, it is not unrealistic to assume that an agent can be programmed to replicate and position itself in hundreds of locations over the net. This is a much more efficient method than individually infiltrating computers with weak security to install DDOS tools. Agents could also be used to scan the Internet to find security weaknesses. As mentioned earlier, there are many security holes in networks and programs have been written to scan the Internet looking for known security holes. Agents are ideally suited to this as they can communicate with each other, reporting the weaknesses to hosts.

Each replica of the agent could also be programmed to repeatedly ping the target, for example, causing a swarming DDOS attack. The agent, as many viruses/worms do, could disguise itself, possibly as a searching agent or any number of other camouflages, with the attack code lying dormant until the appropriate time arrives.

The agents need a place to replicate and distribute themselves over the networks. The Code Red worm used a security hole and there are many other security weaknesses for agents to infiltrate. To maintain communication, many hosts will be needed to communicate with each agent. Of course the agent could be created to cause havoc, then cease to function, requiring no communication with the source. As with DDOS tools, source addresses could be forged and communication encrypted, making it harder to trace the source.

An agent does not have to possess each of the properties of agents given in Table 1. However, there are several properties that would make mobile agents a good option for automating swarming attacks. The agents would have to be reactive, able to adjust to changes in environment. The attacks would take place over the Internet, which is highly dynamic, so reaction is necessary. Autonomous, the agent would be written then left alone to infiltrate systems and replicate. Goal-oriented, the major goal would be to cause a DDOS attack, but another goal would be to find weak security. Temporally continuous, the agent will be a continuously running program. Communicative, the agents will have to communicate with each other, or with hosts, to determine the attack time. Mobile, clearly, the agents will be traversing the Internet.

Also, if the agent was learning and flexible, it could recognize what led to the failed attacks, and ‘learn’ from the experience and change the attack process for the next time. They may also be able to determine exactly how much force is required to take down a system. Possibly, when determining the extent of force required, the agent could be able to replicate itself sufficiently to create that kind of force.

Problems with the use of Mobile Agents in Swarming Attacks

One of the things that led to the failed attack with the Code Red worm was the knowledge that the worm was going to attack at a specified date, which was the same every month. To overcome this, the attack could be made random. Then the attack time needs to be communicated to all agents, which would be difficult to maintain. A huge number of agents would be required to make the attack successful. Dealing with such a large list of agents, potentially in the hundreds or thousands, would require communication with each other or via hosts, to start the attack. In addition, a list of agents would need to be kept.
Also, another issue that needs to be considered is how difficult it would be to identify the source. Since the agent is roaming the Internet, it is probable that the chances are slim. However, to maintain control over the agent, the attacker would have to be able to communicate with the agent, probably via an agent host.

The worm was written and sent to cause damage and then left alone, which is a more simplistic version of what is being hypothesized here. Using mobile agents in a swarm attack would require much greater communication. Exactly how difficult this would be to achieve is unknown.

CONCLUSION

In the first section of the paper, it was shown that attacking a target from many locations could be an efficient way to incapacitate a system. It has been done by hacktivists in terms of digital Zapatismo. They have been shown to want to expand their product line to make for more efficient protests, but they are unlikely to want to automate their protests to such a scale, since part of what they stand for is the group protest. However, there are always those out there who want to utilize more efficient ways to take down systems, as shown by the development of Tribal FloodNet and many other DDOS tools in existence.

So, the question is, where does development go from here? More automatic/autonomous ways of developing and using attack tools are constantly being developed. Utilizing agents appears to be the next logical step and this has the potential to totally automate the process of a swarming attack, allowing little or no interaction with the source. There are always security weaknesses to be exploited that would allow the agent to replicate and distribute itself over the net.

REFERENCES


Franklin S. and Graesser A. (1996) Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents, Institute for Intelligent Systems, University of Memphis, Online; http://www.msci.memphis.edu/~franklin/AgentProg.html Accessed 9/10/02


Accessed 9/10/02

Accessed 9/10/02

Accessed 9/10/02

Accessed 9/10/02
FIRESTORM: Exploring the Need for a Forensic Tool for Pattern Correlation in Windows NT Audit Logs

A. Ahmad¹, and A.B. Ruighaver²

¹Department of Information Systems
University of Melbourne, Australia
Email: a.ahmad@dis.unimelb.edu.au

²Department of Information Systems
University of Melbourne, Australia
Email: tobias@dis.unimelb.edu.au

ABSTRACT

Computer Forensic investigators have traditionally been concentrating on the extraction of evidence from confiscated computer systems used by suspected offenders. Relatively less emphasis has been placed on the analysis of systems that have experienced a security violation. This paper discusses the need for new forensic tools capable of assisting forensic investigators in analyzing computer security incidents. In an effort to assist the forensic investigator in this process, we explore the need for a new forensic tool, FIRESTORM. We discuss how FIRESTORM would allow an investigator to compress some of the non-relevant details in those patterns that have been identified as possibly relevant, to improve the visualizing of any correlations between these patterns. We have called this technique semantic compression.

Keywords: Computer Forensics, Computer Security, Semantic Compression.

INTRODUCTION

Computer Forensics is a relatively new field in computer science and is still undergoing a process of evolution and definition. Whereas the area of computer security is primarily concerned with preserving the integrity of a system, computer forensics is typically applied after a system’s security has failed and the reason behind the system’s abnormal behavior is being investigated. When this happens, a forensic investigator may be employed to determine what events lead up to the particular security incident.

The primary sources of evidence that investigators have at hand on a computer system are audit logs and the state of the file system. Following the seizure of computer systems utilized by the offender, forensic investigators employed by law enforcement agencies traditionally concentrate on data extraction from the offender’s machine with the aim of discovering incriminating evidence that can be used in court. Most forensic tools, therefore, aid investigators in their search for such evidence and few if any, attempt to analyze the characteristics of the ‘offended’ machine to determine what transpired at the ‘scene of the crime’.

There is a need for better forensic tools, offering improved assistance to forensic investigators in identifying anomalous system and user behavior from the audit logs of a system under investigation. Audit logs usually contain information about the user’s actions on the system including the times they logged in and out, what commands they executed, and the behavior of the processes that ran as a result of the user’s actions (Hosmer, 1993). Traditionally, the system administrator tailors the audit policy to
improve detection of anomalous system events. Statistical tools, or other state-of-the-art Intrusion Detection Systems, can then be used to help the system administrator identify possible security violations. The forensic investigator’s job, however, is more complex than Intrusion Detection. A forensic investigator will need to thoroughly examine the audit logs generated by the conventional auditing of the system and, if necessary, compare them with the audit logs generated by re-enactments of the incident, to formulate a hypothesis of what occurred on the system and then prove that hypothesis.

This paper will first introduce the role of a forensic investigator. We discuss the problems encountered in the configuration of audit strategies and the influence these strategies will have on the investigation of the resulting audit logs. In the second part of the paper we describe FIRESTORM, a tool we developed for the forensic investigation of NT audit logs.

FIRESTORM aims to support the forensic investigator in the identification and visualization of different patterns in the audit log that may be relevant to the investigation at hand. We discuss how semantic compression, the method used by FIRESTORM, allows us to reduce the inevitable information overload experienced by the forensic investigator when asked to uncover possible correlations between any of the patterns that have been identified.

FORENSIC INVESTIGATIONS

Forensic investigations typically consist of two phases. The first phase, known as the exploratory phase, is an attempt by the investigator to identify the nature of the problem at hand and to define what s/he thinks transpired at the scene of the incident. For example, in a hacker case the investigator may need to pinpoint the source of the break in. In a corporation with hundreds of computers and thousands of entry points this may well be a daunting task. Another possible scenario may involve a network server crash. Such incidents involve numerous unknown factors that need to be examined by the investigator before a plausible explanation can be formulated.

Once the investigator has determined what s/he thinks took place the induction ends and the deduction, i.e. the evidence phase, begins. The evidence phase revolves around the accumulation of proof admissible in court that deductively proves the conclusion of the forensic investigator.

The exploratory phase of the investigation tests the investigator’s ability to detect patterns in what appears to be a chaotic scenario. Each scenario consists of recurring patterns that define a commonly occurring “normal” sequence of events, eg. users following their usual patterns of computer/network usage, backups taking place according to their pre-determined schedule etc. The patterns that form this “normal” sequence of events when identified by the investigator allow him/her to visualize any disruptions or anomalous sequences of events that may have taken place. The solution to most cases lie in these anomalous occurrences or patterns.

Anomaly detection is not a new field, it has been researched for a considerable length of time. This has led to the construction of various tools employing rule-based, neural network based, statistical and other techniques in detecting anomalies. The use of these tools are often not suitable for the purposes of forensic investigation. A forensic investigator is not necessarily interested in the most anomalous event. He is looking for a specific pattern of events that helps explain what happened. What is required is a tool that assists investigators to sift through the complex patterns in an audit log and allow the investigator to hide uninteresting details in previously identified patterns in the expectation that the more subtle patterns, or any correlations between patterns, will reveal themselves.
EVENT LOGGING

The audit trail and the state of the file system are the primary sources of evidence for the investigator but since the file system is difficult to preserve, especially immediately after an incident is occurred, we are initially concentrating on forensic analysis of the audit log.

Logging is a process by which a system generates a profile of notable events that have taken place on a system and identifies the participants, such as users or system processes, in those events.

The audit log will, in the first instance, be configured to help the system administrator understand the circumstances that lead to important events. In most cases, the audit policy will be tailored to anticipated situations. System administrators set up audit policies in order to capture details that they think are critical in indicating the ongoing progress of a system. Typically only those events are audited that indicate dangerous behavior and alarm bells are attached to them to expedite a response. Only rarely will the audit policy take into account a possible forensic investigation after a security incident has been detected.

Existing audit trails collect large amounts of data that are difficult to analyse within reasonable periods of time and frequently leave out much information that is relevant to security incidents (Lunt, 1993). Audit trails must increase coverage of operating system events to allow administrative personnel to gather more relevant information.

Previously administrators have audited single operating system events to monitor the behavior of users. Single events have not been sufficient to provide administrators with enough information about the overall behavior of users. Instead administrators must audit real world events rather than operating system events or more specifically behavioral monitoring modes to track usage patterns. Behavioral monitoring involves the tracking of a collection of real world actions termed “behavior” that interest forensic investigators.

A primary obstacle in improving the concentration useful data in event logs is coarse granularity in the control of most auditing systems provided by the audit interface to administrators by which they may specify the type and amount of data they would like to record relating to a specific event. Logs of such operating systems tend to record too much data little of which is relevant.

Full logging is not an option as it produces a torrent of data that can bring the system down to an unacceptable crawl. Some selection is necessary. The audit policy will often be limited by the available storage space as well (Schaen, 1991).

From a forensic point of view, you may never have enough details. Any unnecessary filtering of event logs may result in the possibility that an alternative cause for the incident cannot be eliminated. This is especially important in criminal cases where proof beyond reasonable doubt is a requirement.

The only viable solution to logging with an intention to conduct forensic analysis is to log as many details as possible while employing sophisticated audit reduction and compression techniques to reduce the volume of the logs to a manageable size.

FIRESTORM

We identified a definite need for a tool for a forensic investigator that is similar to a computer engineer’s logic analyzer. Such a tool would set an audit policy and subsequently analyze the resulting event logs and provide a graphical monitor that displays either system or user behavior with respect to network accesses, file usage and login/logout behavior. Such a tool can be used to analyze the
behavior of a single user or to compare the behavior of several users in relation to specific system events.

Compared to other operating systems, such as Unix, the event logs that NT can produce are significantly more detailed. Each single task a user executes may end up generating several related entries in the event log. If a reasonably complete auditing policy is enabled, the opening of a single file would already generate more than 4 log entries. Each user logging in to the computer system may generate 40 or so log entries. Hence using the logs to determine the sequence of events becomes a tedious process.

The singular most striking characteristic of the event logs in NT is the difficulty faced by the forensic investigator in deciding how these entries correspond to the real set of events. The high level of details needed to present a complete picture of what happened on the system turns the search for relevant data in these event logs to a process similar to finding a needle in a haystack.

FIRESTORM has the ability to vary the run time speed of the events displayed on screen with the real time scale in which the events were logged. Hence the flurry of events that lead up to an intrusion can be slowed down to enable investigators to scrutinize the relations with other concurrent incidents and to pinpoint events of interest. In the case where events have scrolled off the screen, FIRESTORM is able to rewind and playback repeatedly to allow the investigator to examine events more carefully.

Semantic Compression

In order to balance the need for detailed audit policies with the needs of the forensic investigator for less complexity, we have developed a new auditing technology called semantic compression. Semantic compression analyzes the generated audit logs and, using available knowledge on how the operating system generates audit logs, will attempt to reduce redundancy in the event log by combining several related events into a single new event.

Semantic compression is aimed at generating a higher level of abstraction in the event logs, based upon a model of the discrete behavior of the users of a system. For example when a user logs into the NT environment without the occurrence of any abnormal system behavior, semantic compression may result in a single semantically compressed log entry, instead of the initial large number of individual events. This entry will highlight such standard information as the user's name, the login time, and the host computer the user logged in to, along with all of the other information originally produced by the event log. More importantly, through its existence, this single log entry also indicates that all other audit events have been generated, examined and found to be in order. If any irregularity has been found, such as the omission of an expected event, the audit-log-compression tool can either report this in the newly created audit log entry or it can abort the creation of the new entry, leaving all the original entries in the audit log.

A semantically compressed audit log yields a sequence of events that readily correspond to real-world actions hence assisting forensic investigators in their correlation of recurring patterns. FIRESTORM has been designed to enable investigators to "hide" details in patterns that have been identified, thereby removing from view the clutter of repeated uninteresting data in the expectation that this might reveal the more subtle underlying patterns for further analysis. Following an iterative cycle of pattern identification and subsequent removal of decidedly uninteresting details about each pattern, the investigator can now focus his/her attention on details that require more scrutiny and may well offer clues to his/her investigation.
The use of semantic compression is, however, not as straightforward as expected. The main problem is the variability of the logs generated by NT given a set of predefined actions. For example, logging on at different times of the day on the same machine or at the same time of the day at different machines may produce a different set of event log entries. This has made it considerably more difficult to identify a single algorithm capable of providing semantic compression for a large number of different event sequences. In the current version of FIRESTORM we are still relying on the forensic investigator to identify patterns suited for semantic compression.

Further Work

Semantic compression is more than just a new methodology for a forensic investigator. Semantic compression has potential as an auditing technology when it can be employed on a real-time basis. Real time use is more difficult because a sequence of events must be completed before a substitution with a semantically compressed log entry can occur. The timing characteristics of a specific sequence of log entries targeted for compression may not be deterministic, which affects the ability of an auditing system to semantically compress such a sequence of events in real time.

At this point in time there are a number of other factors that further reduce the effectiveness of FIRESTORM as a forensic tool. To begin with, we are currently relying on the integrity of the original audit logs in the sense that they have not been falsified or modified in any way (Hashii, 1997). Additional security mechanisms, such as encryption, will need to be used in future versions to safeguard the chain of evidence and may complicate the implementation of semantic compression algorithms.

Although semantic compression has been discussed in the context of Windows NT based logs, it is a valid approach for UNIX based syslog as well. Semantic compression can be used to generate higher levels of abstraction in firewall logs, access control logs, and so forth.

There is no doubt that FIRESTORM is a useful tool in the explorative phase of a forensic investigation. However, to expand its use to the evidence collection phase of a forensic investigation, the issue of its integrity has to be addressed as well. The handling of the audit data, especially its storage and semantic compression along with the techniques used to draw conclusions from such audit data will have to be evaluated by trusted forensic experts before such a tool can be used in the court system (Schaen, 1991).
CONCLUSION

Computer forensics is typically applied after a system’s security has failed and the reason behind the system’s abnormal behavior is being investigated. The primary sources of evidence used in such investigations are audit logs and the state of the file system. Unfortunately, investigators are often hampered by audit policies which are typically tailored to anticipated situations primarily due to performance constraints.

The only viable solution to logging with an intention to conduct forensic analysis is to record as many details as possible while employing sophisticated audit reduction and compression techniques to reduce the logs to a manageable size. FIRESTORM would be a simple yet powerful tool that employs a new audit technology termed semantic compression that helps investigators to analyze such audit logs.

Semantically compressed logs yield a sequence of events that readily correspond to real world actions hence assisting forensic investigators in their identification of recurring patterns. FIRESTORM would be designed to enable investigators to identify repeated patterns of uninteresting data and “hide” them from view. This technique helps the investigator to visualize the more subtle underlying patterns for further analysis. Following an iterative cycle of identification and subsequent removal of decidedly uninteresting data allows the investigator to focus on details that require more scrutiny and may well offer clues to the investigation.

REFERENCES


DRAGGING THE LEGAL PROFESSION INTO THE INFORMATION WORLD: AN ANALYSIS OF INFORMATION WARFARE ISSUES AND STRATEGIES ASSOCIATED WITH THE LEGAL PROFESSION

Kristen Perry1  Mark C. Williams2

1University of Western Australia
Crawley, Western Australia, 6008

2School of Management Information Systems
Edith Cowan University
Churchlands, Western Australia
Email: m.williams@ecu.edu.au

ABSTRACT

We examine the concept and practicalities of information warfare (IW) as applied to the legal profession, the various strategies available to law firms to combat the effects of IW. In general, we recommend a defensive strategy to protect sensitive information and maintain market leadership position. Naturally, this strategy will vary depending on the firm under consideration and the market conditions. Other aspects we consider to make the IW strategy effective, such as the need for ongoing risk assessment, effective human resources management and ongoing education. If implemented properly an IW strategy will assist law firms to advance their market position and may even provide a platform for organisational change and transformation.

INTRODUCTION

According to Arthur C. Clarke, within a short period of time, technology will advance to levels indistinguishable from magic (http://krellian.tripod.com). There is no doubt that technological discoveries have already surpassed the incredible, and the consequences of these discoveries have been both dramatic and unpredictable and led to unprecedented opportunities (Wilson, 2000).

One of the perhaps unintended consequences of the development in information systems is the emergence of information warfare ('IW'). IW has emerged as a result of the IT revolution, and is transforming the nature of future warfare (Barker, 2000). While IW encompasses such a broad range of issues, in the context of business, IW is a metaphor which refers to achieving and maintaining an information advantage over business competitors and adversaries (Cramer, 1997). It is both a weapon and a strategy to protect, defend and advance an organisation's success at the expense of its competitors.

Another of the unintended consequences of the development in information systems is the rise of computer crime. The potential cost of IW is immense in both social and financial terms (Walker, 2000). According to a study conducted by CSI with the participation of the Federal Bureau of Investigation (FBI) International Computer Crime Squad, in 1998, organizations incurred $136 million in known losses due to computer crime (Rapalus, 1998, published on http://www.gocsi.com/prelea11.htm). Approximately 70% of information systems professionals reported their organisations had experienced major computer crimes resulting in direct or indirect losses of hundreds of millions of dollars (Bottom, 2000). This is bound to increase as the result of the 'I Love You' virus manifest themselves, a virus which spread so quickly due to the social engineering factor (Stucker, 2000). Appendix 1 outlines the most important findings CSI identified in its study. Appendix 2 identifies the most common methods of computer attack methods.
The future will see a substantial increase in the number of potential targets and potential perpetrators of IW, and the perpetrators will be harder to catch due to the anonymity offered through the use of digital communication systems (Grabosky et al 1998). And the fact is, IW is very close to home: there is no Fortune 500 company which hasn’t been hacked (Radcliffe, 1997), and estimated damage to these companies is about $250 billion per year (Benesh, 2000).

In knowledge based industries, such as the legal sector, information creation and dissemination is a primary business function and good management of this process will lead to a competitive advantage. Protection and defence of this core business function will be one of the determining factors of the success of these organisations. Perhaps more than in any other industry, integrity, quality, accuracy, reliability and timeliness of information is paramount in today’s global environment (Saarelainen, 1996), as legal decisions made on the information usually have drastic effects on people’s lives.

In many organisations information technology is now the single largest capital expense (Weill et al, 1998). As e-commerce capabilities become a standard feature of many businesses, and as more reliance is placed on information systems, it is only a matter of time before legal services are provided over the internet. According to Ernst & Young’s 1999 survey of over 4,300 organisation in 35 countries, 45% of respondents intended to use e-commerce within the next two years (Figg, 1999). As the world turns toward e-commerce, the legal profession must consider the viability of such a move and assess the security risks that are peculiar to the legal profession.

The legal sector often drags its heels in adopting changes to business practices, including its information systems. Traditional paper methods of information creation and storage of intellectual property are common. However, the majority of larger practices have adopted information systems in their practices, and are placing increasing reliance on these systems. For example, the use of intranets, e-mail, on-line legal information such as CCH, Butterworths, and Internet access is fairly standard throughout larger firms.

Hand in hand with the adoption of information systems and placing increasing reliance on those systems in the legal profession as in any other industry is the risk of an IW attack from both internal and external sources. Not only do leaders of organisations have to consider the risk of competitors’ attacks, but also of their own employees tampering with information, which is more often the case (Haag et al, 2000). Accordingly, this paper will make a detailed examination of IW from both military and business perspectives, consider the elements which comprise IW, consider issues of IW endemic to the legal profession and outline strategies to deal with IW.

INFORMATION WARFARE

Definition issues

IW is still imprecisely defined. However, the general consensus in the masses of literature on the topic tend to define IW by reference to two generic categories: either as a physical attack on information networks, or as a metaphor for advancing an organisation’s strategy for success.

IW as a military strategy

To provide a background to the paper, we consider it relevant to discuss military strategy. IW in the military sense aims to gain superiority over the enemy by attacking information systems and networks, centred on resources, infrastructures controlling these resources, such as power, electricity, money, air and traffic (Libicki, 1998; Kruczek, 1998). Morthe (1998:568) defines IW as ‘a state activity which have an incapacitating effect on the availability of the owners of any information network to use or manage that network’. As a military strategy, however, IW is not limited to state governed action, as it does not require state sponsorship or vast resources. IW techniques are available for any person or organisation to use (Molander et al,1997), and the tools of IW are easily acquired (Morth, 1998).
Indeed, the effect of emerging information and communications technologies internationally ensures there are more empowered groups and individuals regardless of geographic location (Studemeister, 1998).

IW is more disturbing than traditional military strategies which required physical presence. IW attacks on networks can cripple society by destroying information networks providing power, transport, defence and medical services (Morth, 1998) – any country or individual can wage IW (Libicki, 2000), and the threat is immeasurable as enemies cannot be seen or assessed. IW is the next generation of warfare methodology, which is exemplified through advanced technology which allows for controlled effects on targets, and higher precision weaponry (Alexander, 1999). This is the new warscape, which aims to disable the enemy’s crucial systems, not kill its population.

Military IW may be harder than it looks however – we are yet to see a city crippled through IW other than in Hollywood scenarios. There are several reasons for this, some of the major reasons being cost; the ‘boomerang’ effect of IW strategies; and law enforcement (Libicki, 2000).

**Strategic IW**

If IW is extrapolated to the business arena and assimilated into an organisation’s business plan, IW can then be defined as a metaphor (Libicki, 1995) for achieving and maintaining an information advantage over competitors or adversaries. This type of IW is emerging as the threat of physical military confrontation decreases and as business competition increases (Kruczek, 1998). Furthermore, as there has been a proliferation of commercial off the shelf technology on a global scale, more and more individuals and organisations have (McHale, 1999). Accordingly, nations losing their monopoly on the ability to wage warfare – criminals, disgruntled employees, commercial competitors and terrorists are now potential threats (Barker, 2000).

The purpose of strategic IW is to induce your opponent to make decisions which benefit your organisation’s goals (Libicki, 1998) and to gain your organisation a competitive advantage. IW is becoming ‘an obvious way to win a war’ (Berkowitz, 2000:9) by striking your opponents’ information networks, whether they be military or business opponents. As competitive advantages can impact an organization’s success or failure, it is important to understand factors which can affect the organisation, and the framework created by the new technologies and paradigms (Cramer, 1997) which are capable of winning or taking away competitive advantages.

The object of IW in this secondary sense is to penetrate competitors’ processing systems and influence key decision-making processes in ways that are beneficial for the attacking organisation (Saarelainen, 1996). Many commentators consider that IW is an integral part of a business strategy for defending and advancing an organisation’s goals and market domination. Furthermore, as in war, surprise can be a powerful strategic weapon – and the use of IW can provide a competitive edge in achieving the preempting surprise in the commercial marketplace (Cronin et al 1999).

While these general definitions of IW are a starting point, it is important to consider the various elements which constitute IW in order to be able to integrate them into a business strategy. An analysis of the elements of IW is provided in Appendix 5.

**INFORMATION WARFARE ISSUES ENDEMIC TO THE LEGAL PROFESSION**

Is IW relevant to the legal profession? So what does IW mean to the legal profession?

As clients (of law firms) adopt current information systems, they are increasingly demanding technologically savvy services. In a market of fierce competition for clients, the client’s specifications and requirements for the transmission of legal services are treated as gospel and all efforts are made to accommodate them. Many law firms are in constant communication with their clients via e-mail, and
Some firms have adopted electronic data exchange, usually via ELF Technology (usually insurer clients) which uses Lotus Notes as an operating platform. Major firms have to adopt these technologies in order to offer timely services to their clients, or indeed, to retain their clients. Australian legislation, such as the Electronic Transactions Act and the equivalent state laws, is similar to some jurisdictions in the US which have adopted legislation that ensures electronic contracts sufficient to create a binding contract (Fiori, 2000).

While electronic communications facilitate and increase the timeliness of the lawyer client relationship, there are serious risks for the lawyer involved if confidential material falls into the wrong hands or is tampered with to the detriment of the client (Jones et all, 1999). These risks include the possibility of legal action being taken against the lawyer, professional negligence claims and actions being taken by the Legal Practitioners' Committee. In other words, if a firm is subjected to an IW attack, and sensitive legal material is exposed, such as prospectus materials, financial information, litigation strategies or opinions, there are repercussions involved which are unique to the legal profession.

SECURITY MEASURES

The main security issues pertaining to the legal profession lie in using the Internet as a means of communication. As the Internet shares networks, there is a risk that sensitive legal information such as prospectus materials, financial information, litigation strategies and opinions, adverse evidence which are increasingly being transmitted via electronic communications and stored on the firm's networks can be hacked or cracked; spoofed (read and interfered with), sniffed (read). Accordingly the main security issues which concern law firms are:

Integrity, that is, whether the electronic communication reaches the client without being read or tampered with; and Authentication, that is, whether the electronic communication sent to the client was actually sent by the named author. (Jones et al 1999).

To some extent, digital signatures and encryption go some way to protect electronic communications. Digital signatures contain algorithms within the e-mail which are unique to each communication. Digital signatures also allow the recipient to verify the authenticity of the document and any tampering will alter the digital signature — there is only one public key to unlike the message hash (Libicki, 1995). While not a complete solution, digital signatures are useful in preventing spoofing. Encryption scrambles entire messages and can only be unscrambled by a matching 'key' (a digital signature). This is the preferred method of sending sensitive data (Jones et al 1999).

However, the talent of black hatters cannot be underestimated. Further, given the dollar value involved in many legal transactions which are determined over the Internet via electronic mail, the determination and resources of competitors and spoofers may mean that digital signatures and encryption technologies are insufficient to prevent e-mails and other electronic communications being a sufficient precaution to an IW attack. Furthermore, as noted, legal practitioners are notoriously slow at implementing technological changes into their practices. The following section considers the implications of an IW attack for legal practitioners.

Legal Professional Privilege ('LPP'): lose it and weep...

'LPP' is a legal term which means that communications between lawyer and client which have been created to obtain legal advice or to prepare for actual or contemplated litigation are confidential and cannot be disclosed or used in evidence. The client actually owns the privilege, and only the client can waive the privilege (Heydon, 1996).

LPP exists to protects the relationship between a lawyer and the client, to ensure that the lawyer can act in the best interests of the client and to encourage frankness between the solicitor and client.
law holds privilege in a position of paramountcy, and it overrides all other duties a lawyer owes because it is an essential part of the perfect administration of justice (Baker v Campbell (1983) 153 CLR 52).

If privileged documents and communications are disclosed to anyone other than the client, then privilege is lost and that communication can be used in legal proceedings. An old common law rule states that even if privileged documents are stolen or wrongfully obtained, the privilege is lost (Calcraft v Guest [1898] QB 759). While more recent cases have attempted to ameliorate this rule by bringing notions of fairness and equity to the decision as to whether privilege is lost, the fact remains that if someone has inspected the privileged communication, it is very hard indeed to maintain that privilege ought not be waived (Ross, 1998:307).

The duties the lawyer owes the client as a result of the lawyer client relationship include duties: of confidentiality; to ensure that the client’s claim for LPP is not lost (Commissioner of Taxation v Citibank Ltd, (1989) 85 ALJR 588 at 596, 20 FCR 403 at 414); and to refrain from disclosing confidential communication, and the law imposes a very onerous standard on lawyers to meet this duty.

Electronic communications between lawyer and client will no doubt often be privileged communications, providing these communications satisfy the tests outlined above. Lawyers who fail to implement security measures expose their communications to possible security breaches, and the clients may lose LPP over those communications as a result. Clients then have the ability to sue their lawyer for breaching the duties the lawyer owed to them (Jones et al, 1999). The lawyer may also be professionally sanctioned under the Legal Practitioners Act 1893 (WA).

In Australia, it is uncertain whether electronic communications which are intercepted or privileged communications hacked into via the law firm’s network remain privileged communications. The old rule remains that if a party to litigation obtains a copy of a document whether by accident, trickery or theft, that party may tender the document in evidence (Bell v David Jones (1948) 49SR (NSW) 223:227). However, it should be remembered that the intrusion or interception has to be discovered before the defending law firm can take any action to defend the privilege of the communications. In many instances where there are huge volumes of documents involved (which are increasingly being scanned and stored on the firm’s networks to save physical space), an attacker having a ‘sniff’ is unlikely to be detected. Where core documents are involved, sniffing may give the opponent key information or strategy to win a case by using IW.

**How to win from information warfare**

Given the importance of confidentiality of communications between solicitor and client, and the likelihood that IW attacks are unlikely to be detected by law firms, due to the ‘stop gap’ security measures most firms utilise, how can law firms adopt IW as an integral element to their strategy? Any IW strategy a law firm adopts will affect its competitive posture (Cramer, 1997), and will influence the market place, competitors, customers, general public and suppliers to achieve an organisation’s business objectives (Saarelainen, 1996). There are many postures a law firm can adopt, although the appropriate response will depend on the nature of the threat (Grabosky et al, 1998) and on the appropriateness of such a response for a law firm. The following section briefly considers the literature on IW strategies, and assesses their viability for implementation in a law firm.

**Corporate Strategies**

As legal recourse to IW are considered to be insufficient by many organisations, many corporate strategies have emerged in response to attacks. According to Cramer (1997), there are several corporate IW strategies available to organisations. Defensive: this posture emphasises information protection, such as access-control and limited external systems interconnections. Law firms are likely to adopt this posture because it is not aggressive, and
appears to be ethical. As this strategy is best suited to market leaders or organisations which benefit from maintaining the status quo, and provides for defence from attackers, this is likely to suit the major two or three law firms in Australia. The likely applications in a law firm are encryption and digital signatures to prevent unauthorised access to electronic communications and tight access-control to networks, however, in major law firms all employees usually have and require internet access.

Some firms may adopt a more aggressive defence position, of counterattack. An example of counterattacking is ‘Blitzkrieg’, designed by Future Vision Group which purported to counterattack by sending viruses to an intruder’s computer (Dequendre, 1999). It is doubtful whether counterattacking is an appropriate inclusion in a law firm’s IW strategy due to the legal and ethical implications inherent in this response.

Offensive: this strategy emphasises information denial including attacks on the market leader, such as sniffing and spoofing. This posture is usually adopted by organizations dissatisfied with their current standing, and are trying to take down their stronger adversaries, or gain an unfair advantage over other law firms in, for example, an aggressive takeover bid, or litigated matter.

Quantity: this strategy emphasises supreme information transport capability, by transporting massive amounts of data over a large well-established infrastructure. As this posture makes attacks impractical due to a wide disbursement of low sensitivity information making an IW attack time consuming and probably frustrating, it is unlikely that this posture would be adopted by law firm as communications are usually highly sensitive and at this point, are not high volume.

Quality: This strategy emphasises on information management, making this a core capability. The advantages of this strategy is that the organisation can manage its information needs better than its competitors, and makes better use of less information. This posture may have advantages in a highly competitive, cost-sensitive market. This strategy is likely to be adopted by smaller law firms trying to break into the market dominated by the national and international firms.

Sponge: This strategy emphasises huge volumes of information collection within their own organisation and from other organisations. Sponge strategies are essentially ‘followers’, as they monitor, collect and copy innovative changes in their competitors. The competitive advantage of this method is the savings of R&D. To some extent, all major law firms adopt this technique to keep ‘in the race’.

**Defend yourself!**

The defensive strategy is likely to be the most important to major Australian law firms. Saarelainen (1996) outlines a defensive strategy, which suggests that organisational leaders ought to implement management systems and techniques which prevent any harmful impacts from hostile IW attacks.

A defensive strategy should:

- adopt a strategic risk management system which can detect and minimise the effects of IW;
- ensure the accuracy, reliability and quality of information; and
- integrate into the business strategy to ensure continued information integrity.

The strategic risk management systems supporting a defensive strategy can include:

- information security systems;
- business intelligence systems;
- quality and environmental management systems;
- specific internal and external communication systems;
- legal management systems;
- information protection services, (digital signatures and encryption); or
- a matrix of elements from each system combined to produce one strategic management system capable of obtaining, analysing, disseminating and protecting information. (Saarelainen, 1996)
Toughen up, buster!

Bolstering the law firm’s security will offer some protection against IW. However, absolute security is impossible if the attacker has the motivation, resources and determination to invade the firm’s systems (anon(b)).

The best way for law firms to maintain tight security would be to operate a closed system, where the system receives no input from the outside world (Libicki, 1995). This would eliminate users’ access to the Internet, a strategy adopted by US based company Blue Cross/Blue Shield, a $1.2b (US) operation with 2,000 in-house users (Radcliffe, 1997). However, as lawyers are increasingly requiring on-line access to legal information services such as CCH and Butterworths, this strategy is not feasible for larger law firms.

The most important issue in implementing any information security system is the nature of the industry being dealt with as there is no one size fits all solution to information security. Accordingly, assessing the security of the organisation’s security needs is vital. The firm’s security program should include:

- identification of information systems requiring protection;
- determination of the value of the information assets;
- identifying potential threats associated with the information asset;
- identify the vulnerability of the computer/EDP system to each of these threats;
- assessing the risk exposure for each asset;
- select and implement security measures;
- audit and refine the security program on an ongoing basis.

(Anon, b)

Once the assessment of the information systems is complete, the law firm would probably secure those systems through:

- password and verification procedures;
- digital signatures to establish strong links of authenticity;
- AI and neural networking to detect suspicious transactions;
- Monitoring the use of remote access to networks; and
- conducting integrity tests of systems.

(Grabosky, 1998)

In addition, the law firm, which operates on a national and international basis, would need to install firewall protection, and these firms would require high-end type firewalls. These firewalls are fit for high security end users and protect large amounts of sensitive information (Lodin, 1999).

Should security of information become exceptionally important to the firm, then other cutting edge security measures can be implemented such as new technologies such as:

- biometric security devices/anomaly detection software identifying unusual computer usage patterns;
- and recording keystrokes of users and comparing them to hackers’ keystrokes.

Ongoing risk assessment & education

Ongoing risk assessment is a necessary part of any strategy, and essential to its success. Risk assessment is a process of continuous improvement and should consider the effectiveness of the current procedures and protocols. Additional improvements to the IW strategy should be initiated when necessary to improve the effectiveness and performance of the system (Saarelainen, 1996).

A further aspect of ongoing risk assessment is the continual need to train and retrain for technological competence. Intel for example teaches employees that security is everyone’s job, though employee orientation, newsletters and spot checks (Radcliffe, 1999).
The human side of computer security is easily exploited (Wilson, 2000), and most information losses are caused by accidents, ignorance or disregard for information security policies and practices rather than the programming genius of a few hackers (Bottom, 2000). For example, the IT department of a law firm using intranet e-mail communications system named GroupWise (which many Australian law firms use), lost the manual and accidentally created proxies with read/write privileges. Accordingly the files, calendars and e-mails of the proxied lawyers became an open book for all staff to read (Bottom, 2000). An excellent education course would prevent these types of losses.

My experience has been that the majority of senior lawyers and partners do not consider technology issues serious enough to warrant their concern, being the reason ‘why we hire IT people’ – some are even unable to operate a computer. Furthermore, training is unbillable time, and there never seems to be enough time for it (Tie, 2000). However, given global trends toward e-commerce, the difficulties of ensuring all members of a law firm are trained does not lessen the important need for it, regardless of the individual lawyer’s hourly rate (Tie, 2000). Such training should focus on the methods and procedures dealing with intruders, issues associated with remote access, security of virtual private networks, e-mail security.

**Human Resources Management**

Effective human resources management is another strategy which law firms can employ in an IW strategy. In particular, management of disgruntled present and former employees with information technology expertise who are in a position to inflict significant damage calls for heightened attentiveness to issues of security. Over 50% of institutions, which experienced intrusions or attempted probes of their internal systems, traced them to current employees. (Grabosky, 1998), and between 70 – 85% of all serious hacker attacks involve inside employees (Libicki, 1995).

At present, the majority of lawyers are unlikely to have the capability to inflict this type of damage. However, the number of IT specialists employed by law firms is increasing, and it is only a question of time before lawyers on the whole become more technologically savvy. Simple procedures such as monitoring e-mail attachments to prevent loss of intellectual property and deactivating passwords after employees leave the organisation may minimise loss information.

**Just in case ...**

Just in case the firm’s IW strategy fails to prevent an IW attack, then the organisation needs to have a contingency plan in place to minimise the effects of the attack. This may entail inhouse procedures, or it may be outsourced, such as IBM’s 24 hour emergency responses services which delivers incident management assistance and damage control to clients who have experienced intrusions (Grabosky, 1998). Considerations pertinent to law firms when devising a contingency plan include:

- identifying crucial information systems;
- daily back ups and off-site storage of those systems; and
- ensuring business interruption insurance policies cover the possibility of financial loss flowing from an IW attack.

**CONCLUSION**

If indeed technology is advancing toward levels indistinguishable from magic, then it is a reasonable extrapolation that IW will also develop in its intensity and effectiveness. As law firms increasingly rely on information systems for communication with clients and for internal purposes, the threat of an IW attack will become ever more present. This increased dependency on computers means that firms are vulnerable unless properly protected: one should not rely on capabilities on which one cannot depend or defend (Libicki, 1995).

For lawyers, IW may result in the loss of LPP over confidential communications, which may result in legal action against the lawyer involved and professional sanctions. This paper examined the various
strategies available to law firms to combat the effects of IW and a defensive strategy was recommended to protect sensitive information and maintain market leadership position. Naturally, this strategy will vary depending on the firm under consideration and the market conditions. Other aspects were considered to make the IW strategy effective, such as the need for ongoing risk assessment, effective human resources management and ongoing education. If implemented properly an IW strategy will assist law firms to advance their market position and may even provide a platform for organisational change and transformation.

Regarding further research, an anonymous reviewer’s comments brilliantly lead the way:

A well crafted paper except for the fixation with digital signatures as a security measure. There are other measures that are applicable to tracing/protecting documents such as audit and transaction logs, file integrity checkers using MD5 hashing to mention just and possibly need some exploration. In terms of network protection the deployment of external and internal Intrusion detection systems would help greatly protect systems or the use of VPN technologies to secure end to end communications. Encryption is not the safeguard that many people think it is many of the techniques to transmission are weak see any article relating to current wireless security. The dilemma the legal profession has with disclosure and communication of confidential data across an insecure possibly hostile network such as the Internet is an interesting one and one I would like to see it further explored. For instance what about the position of being a lawyer who has to defend a terrorist - the ECHelon and CARNIVORE programs would certainly be deployed against you and what would result?

REFERENCES


Baker v Campbell (1983) 153 CLR 52; 57 ALJR 198


Commissioner of Taxation v Citibank Ltd, (1989) 85 ALJR 588 at 596, 20 FCR 403 at 414, per Bowen CJ and Fisher J


Stucker, H., 2000: 'E-mail viruses can create havoc for insurers', National Underwriter, volume 104(23):3, 47


www.kurellian.tripod.com – a very interesting futurist site. I highly recommend visiting this site – I've whiled away many hours here!

www.infowar.com
Protection of New Zealand in the age of Information Warfare

M.J.Warren and J.McIntyre

School of Information Technology,
Deakin University,
Geelong, Victoria, Australia, 3217.
Email: mwarren@deakin.edu.au

ABSTRACT

NII (National Information Infrastructure) has a big impact on all developed countries. A countries NII is concerned with infrastructure in terms of gas and electricity supplies, banking and finance sectors, airports and transport, defense forces, telecommunications and government departmental information systems. The aim of this paper is to evaluate how New Zealand is protecting its NII in terms of developing strategy and implementing national policies for critical infrastructure protection.

Keywords: New Zealand, Critical Infrastructure Protection, National Information Infrastructure.

INTRODUCTION

Global, National, Critical, and Defensive Information Infrastructures have become necessary and vital elements for nations worldwide (Stagg and Warren, 2002). These infrastructures are a mix of open and closed networks, private and public systems, the Internet and government, military, and civilian organizations (Anderson, 1998). They are important vehicles for the generation of national wealth throughout developed societies; they increasingly influence the power and capability of nations (Westwood, 1996), cause radical changes in international finance and commerce (Cordesman, 2000), and are powerful stimulants to the creation of content, ubiquitous communication, and access to new markets (NRC, 1996).

NEW ZEALAND PROFILE

New Zealand is a modern, small country located close to Australia. A summary of New Zealand details can be found in Table 1 (CIA, 2001).

| Total area total of New Zealand | 268,680 sq km |
| Government type                | Parliamentary Democracy |
| GDP                            | $67.6 billion (2000 est.) |
| Population                     | 3,864,129 (July 2001 est.) |
| Spread of Population           | 80% of the population lives in cities; |
| Internet Service Providers (ISPs) | 36 (2000) |
| Internet users                 | 1.34 million (2000) |

Table1: New Zealand summary details

The New Zealand government is very aware of the importance of new technology and it is actively promoting e-commerce at a national level. The New Zealand government had determined the importance of National Information Infrastructure Protection (NIIP). They have deemed that NIIP is focused on improving the protection of New Zealand's critical infrastructure from cyber attacks. The New Zealand government has deemed "New Zealand people and business depend on the continuing supply of various services such as power, telecommunications and health care. Critical infrastructure
includes the wires, machines, and software needed to make this happen, such as power lines and telephone exchanges". (New Zealand Government, 2000a). It should be remembered that New Zealand has been one of the few developed countries in the world that have been affected by Critical Infrastructure failure. In 1998, power to Auckland Central Business District failed for a five week period which affected 50,000 inner city workers and 6,000 residents and resulted in major disruption to businesses and residents (The Times, 1998).

NEW ZEALAND CRITICAL INFRASTRUCTURE PROTECTION

The NIIP Project was initiated in October 2000 with an aim to improve the protection of New Zealand’s critical infrastructure from information-borne threats. The E-Government Unit, State Services Commission released an initial consultation report on the 8th, December, 2000 entitled Protecting New Zealand’s Infrastructure From Cyber-Threats (New Zealand Government, 2000b).

The outcome of the report was to define the government perception of current protection of the New Zealand NII (National Information Infrastructure) and analysis of what are the critical sectors of the New Zealand NII were. The key sectors that were identified were

- Banking Facilities;
- Power Distribution;
- Offshore and Inter-Island Telecommunications Links;
- Denial of Service Attacks on the Internet.

The report highlighted a number of major steps that were required, which were:

- Building New Zealand capability and knowledge in respect of security and incident handling;
- Deterring attacks through legislation and enforcement;
- Promoting standards for risk management and security in infrastructure;
- Requiring government agencies to adopt appropriate information technology security measures.

This initial report led to further report being released on the 11th June 2001 which detailed the development of critical infrastructure protection within New Zealand (New Zealand Government, 2001a).

This report recommended that a Centre for Critical Infrastructure Protection (CCIP) be set up to provide a free service to New Zealand infrastructure owners, government agencies and to some extent the New Zealand public. The report detailed the modus operand of the CCIP centre as well as detailing budgetary and time framework for its development.

The report also highlighted that the staff of the unit will consist of a unit manager, an administration officer, three Watch and Warn Centre analysts and supervisor, and two incident and infrastructure analysts. The center will initially be based within GCSB (Government Information Security Bureau) Head Office in Wellington, New Zealand.

The minister concerned with developing the CCIP released a press released detailing its importance for New Zealand protection, he deemed the importance of the CCIP as being (Mallard, 2001):

- That New Zealand should have its own unit to monitor IT security and risks on a day to day basis, and to provide advice and training on securing infrastructure from cyber-attacks;
- Able to offer advice to the Government on potential improvements to New Zealand law, and New Zealand’s role in international treaties, particularly to outlaw ‘denial of service’ attacks over the Internet.

The CCIP report identified the main risks that face New Zealand critical infrastructure are (New Zealand Government, 2001a):
Banking Facilities
Some of New Zealand Banks were planning to move their retail processing away from New Zealand. As well the New Zealand Reserve Bank was planning to move its computers for its real time gross settlement system to Sydney Australia.

Power Distribution
With the greater reliance of information technology to manage the power distribution, the security and vulnerabilities of such systems has become more of an issue. Also there is a flow effect to the banking facilities and the telecommunication sector if there were disruption to the New Zealand power distribution.

Offshore and Inter-Island Telecommunication Links
There is a risk to the Offshore and Inter-Island Telecommunication Links as they are primarily submarine cables. They are very vulnerable to many types of physical attacks.

The inter-island cables lay on the ocean floor and are not buried and the two cables lay close together. It is obvious any attacks on any of these cables could have serious effect on the Internet as well as business operations within New Zealand.

Denial of Service Attack
A very serious threat for New Zealand E-commerce organizations are on-line attacks. Denial of Service Attack have recently become a more common type of attack, which is hard to trace to the original source and could cause huge disruption.

DESIGN AND OPERATION OF THE NEW ZEALAND CENTRE FOR CRITICAL INFRASTRUCTURE PROTECTION

It is intended that the New Zealand Centre for Critical Infrastructure Protection be modeled on the following approach (New Zealand Government, 2001a):

- Watch and Warn;
- Investigation and Analysis;
- Outreach, Strategy and Training.

The hours of operation for the CCIP is 24 hours a day 7 days a week in order to offer a real time service. The CCIP officially started operation in April, 2002. Figure 1 gives an illustration of how the New Zealand CCIP work and how it interacts with other partners.

Figure 1: The New Zealand CCIP Model
The key aspects of CCIP are (Garden, 2002):

- The 24/7 Watch and Warn Centre operates a day watch from 7am to 7pm, and a reduced 'emergency' night watch from 7pm to 7am. The Watch and Warn Centre uses the Web, newsgroups, e-mail, IRC, telephone, fax and conventional news media to maintain an overview of Internet vulnerabilities, threats and incidents. It does not actively monitor Internet traffic. It is the national incident reporting and response co-ordination centre. The Watch and Warn Centre will also produce weekly, quarterly and annual reports on vulnerabilities, incidents, and issues relevant to the security of the critical infrastructure and the Internet.

- The Analysis and Investigation function has two staff. The Senior Infrastructure Analyst is the CCIP's lead in incident handling and investigation. This section is developing expertise in analysing critical software vulnerabilities and exploits.

- The Outreach function is arguably one of the most crucial outputs of the CCIP, since protection of the critical infrastructure will require awareness and co-ordination amongst many people in a number of professions and in a multitude of government and non-government organisations. The objective of the Outreach function will be to develop relationships and networks with and between many of those individuals, and to facilitate appropriate training to them as appropriate. Through the Outreach function, the CCIP will play a part in a mixture of formal and informal partnerships between the Government, critical infrastructure organisations and private industry in many areas of critical infrastructure protection.

ISSUES AND PROBLEMS AND LIMITATIONS OF NEW ZEALAND APPROACH

A major problem that New Zealand faces is that some of New Zealand banks were planning to move their retail processing IT systems away from New Zealand (New Zealand Government, 2001a). This could cause major problems since it means aspects of the New Zealand NII will be remotely located and the CCIP center would be unable to protect that aspect of New Zealand's NII. A solution to the problem would be the countries hosting the New Zealand banks IT systems would agree to protect them as part of their own NII, but this approach is very unlikely to work. There are then legal issues, new laws past to protect New Zealand NII would have no legal status in other countries where remote aspects of New Zealand NII may be located.

There is a lack of general information about IT Security for New Zealand organizations in how to protect themselves. The New Zealand Security of Information Technology (NZSIT) publications are produced as guidelines for New Zealand government organisations by GCSB. They produce the following IT Security advisory documents (GCSB, 2002):

- NZSIT 100 Security of Computer Systems (Apr 92)
- NZSIT 101 IT Security Policy Handbook (Jun 00)
- NZSIT 102 Certification of Government Computer Systems (Apr 97)
- NZSIT 103 IT Security Evaluation Criteria
- NZSIT 104 Risk Analysis of Government Computer Systems (Sep 94)
- NZSIT 105 Configuration Management (Jun 94)
- NZSIT 107 INFOSEC Standards (available as NZ Standard MP6653)
- NZSIT 200 Security of Personal Computers (May 96)
- NZSIT 202 Security of Local Area Networks (Nov 96)
- NZSIT 203 Secure Use of Public Data Networks (Aug 97)
- NZSIT 204 Authentication Services & Mechanisms (Apr 94)
- NZSIT 207 Declassification of Storage Media (Jan 96)

Table 2: Available NZSIT Documents
These documents can only be described as being old and in some cases antique e.g. 1992 in terms of IT security protection. The New Zealand government is dealing with the issues in terms of promoting security within E-Commerce (New Zealand Government 2001b, 2002). The problem is that up-to-date IT security information within New Zealand is only being offered in a limited number of security areas, which reduces the overall effectiveness of such initiatives to nationally promote IT security within New Zealand.

New Zealand Standards is also promoting AS/NZS 4444 (now known as AS/NZS 17799) as a way for commercial organizations to implement Information Security Management. The AS/NZS 4444 is a baseline security standard. Baseline security standards offer an alternative to conventional risk methods. Baseline represent the minimally acceptable security countermeasures that an organisation should have implemented. These countermeasures are applied in a generic manner, e.g. every organisation should have the same baseline security countermeasures implemented (Warren & Hutchinson, 1999).

The advantages of using baseline methods include (Warren & Hutchinson, 1999):
• cheap to use;
• simple to use;
• training is not required to use the method;
• it is quicker then undertaking a full security review.

The disadvantages of using baseline methods include (Warren & Hutchinson, 1999):
• generic nature of baseline security methods mean they may not solve all of the organisational security requirements;
• the fact that they have been designed for use within a general environment mean that they may not be suited for all environments, i.e. healthcare or small businesses;
• they do not suggest how the security countermeasures may be implemented;
• they do not contain cost benefit details.

The problem is that a minimal level of security is being promoted for all New Zealand organizations. There is also an additional problem that Standards New Zealand do not have a full time committee in place to advise on IT security issues.

The other issue is size, the CCIP center has a budget of NZ$1 million per year and eight full time staff (New Zealand Government, 2001a). Would this center be able to handle a prolong attack against the New Zealand NII or an attack against New Zealand NII at a number of different levels, being carried out simultaneous?

Another problem that the CCIP face is their reliance on non New Zealand organisations for information. For example the CERT (Computer Emergency Response Team) information provided to New Zealand is actually provided by AusCert (the Australian CERT group). The issue here is time, there will be a time lag in CERT information provide to Australia who then provide it to New Zealand. This means that CCIP will not be able to operate in real time since the information they rely on is not provided in a real time environment. This could also be a possible attack method against New Zealand, you could have a situation that non New Zealand organizations are providing CCIP with false information, who then make decisions on the basis of that false information.

CONCLUSIONS

By New Zealand establishing the CCIP, it has shown that small countries can be proactive in protecting their NII and CII. The model that the CCIP is based upon, is a workable model that will help to protect New Zealand critical infrastructure protection. The issue is whether the security threats and weaknesses put forward by this paper will disrupt functionality of the New Zealand CCIP in the longer term.
REFERENCES


The Times (1998) Auckland’s Blackout is likely to last for weeks, February 28th, London, UK.


The U.S. National Infrastructure Protection Center: 1999-2001 - A Research In Progress Study

M.K. Lavine

Visiting Assistant Professor
Department of Computer and Information Sciences
College of Science and Mathematics
Towson University, Towson, Maryland, U.S.A.
E-mail: mlavine@towson.edu

ABSTRACT

In 1996, the increasing threats of cyber-terrorism and information warfare lead U.S. President Clinton to form the President's Commission on Critical Infrastructure Protection (PCCIP), which resulted in a number of landmark government initiatives to combat various computer security issues. This paper investigates the role of the National Infrastructure Protection Center (NIPC) and the information provided by the organization. As a federal government agency, the NIPC helps improve national security by building an information partnership that shares information with computer security professionals and researchers. Preliminary results from stage one of an ongoing research project based on the Ph.D. dissertation project of the author are presented.

Keywords: National Infrastructure Protection Center, Computer Security, Information Assurance, National Security

INTRODUCTION

The U.S. like other countries around the world has taken new actions to fight cyber-terrorism and information warfare. Perhaps one of the most important early initiatives in this area was the formation of the Presidential Commission on Critical Infrastructure Protection (PCCIP) in 1996. This commission was charged with defining what areas are covered by the concept of critical infrastructure. Critical Infrastructure (CI) is a sub-set of the general National Information Infrastructure (NII).

A precise definition of infrastructure is 'the basic, facilities, services, and installations needed for the functioning of a community or society, such as transportation, and communication systems, water and power lines and public institutions' (Lukasik, Greenberg and Goodman 1998, p. 11). A general overview of each of these areas was announced in Critical Infrastructure Protection, Presidential Decision Directive 63, which was issued by executive order in May 1998.

A total of eight functional areas were determined to be critical in order to sustain normal government operations related to civil obedience, order and critical services (Alexander and Swetnam 1999). These are briefly described in the following table below:
<table>
<thead>
<tr>
<th>Functional Area</th>
<th>Brief Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Oil and Gas</td>
<td>Encompasses various elements of the energy sector including: electricity, oil,</td>
</tr>
<tr>
<td></td>
<td>natural gas and all related sub-networks and delivery systems.</td>
</tr>
<tr>
<td>2. Financial Services</td>
<td>Deals with those areas necessary to support the national economy such as bank</td>
</tr>
<tr>
<td></td>
<td>transfers, stock and bond markets as well as supporting transaction processing</td>
</tr>
<tr>
<td></td>
<td>database systems.</td>
</tr>
<tr>
<td>3. Water Supply</td>
<td>Includes drinking water, firefighting, public safety, heating/cooling and</td>
</tr>
<tr>
<td></td>
<td>business uses as well.</td>
</tr>
<tr>
<td>4. Emergency Services</td>
<td>Covers such areas as police, fire, ambulance and emergency management services</td>
</tr>
<tr>
<td></td>
<td>in the event of disasters (ordinary and catastrophic).</td>
</tr>
<tr>
<td>5. Government Services</td>
<td>Includes federal, state and local agencies dealing with the general public</td>
</tr>
<tr>
<td></td>
<td>welfare.</td>
</tr>
<tr>
<td>6. Electrical Power Services</td>
<td>Addresses all major areas of electric power generation, storage and</td>
</tr>
<tr>
<td></td>
<td>transportation necessary to the provision of vital services.</td>
</tr>
<tr>
<td>7. Transportation Services</td>
<td>Covers all areas of people and consumer goods movement, thereby protecting</td>
</tr>
<tr>
<td></td>
<td>our national role and impact in the global economy.</td>
</tr>
<tr>
<td>8. Telecommunication Services</td>
<td>Includes all components that maintain the public telephone network, Internet</td>
</tr>
<tr>
<td></td>
<td>viability and computer systems used on a daily basis.</td>
</tr>
</tbody>
</table>

Table 1: U.S. Critical Infrastructure Functional Areas

(Adaptered from Alexander and Swetnam 1999 and Rathmell 2000)

One of the PCCIP’s recommendations was to develop a single organisation, which would have overall responsibility for securing and protecting U.S. critical infrastructures. With this, the National Infrastructure Protection Center was formed, and today it serves as the U.S. government’s central command for various initiatives related to computer security.

The NIPC has a number of important responsibilities. These are best summarized in the organizational mission statement, as follows:

1. Deter, detect, assess, warn, respond, and investigate unlawful acts involving computer and information technologies and unlawful acts, both physical and cyber that threaten or target our critical infrastructures;
2. Manage computer intrusion investigations; support law enforcement, counter-terrorism and foreign counterintelligence missions, related to cyber crimes and intrusion; support national security authorities when unlawful acts go beyond crime and are foreign-sponsored acts on United States interests; and
3. Coordinate training for cyber investigators and infrastructure protectors in government and the private sector. (Available at http://www.nipc.gov/)

The NIPC’s work is on an inter-agency basis and is coordinated by the Federal Bureau of Investigation (FBI), the primary federal law enforcement service in the United States. Recent criticism by the U.S. General Accounting Office has highlighted the need for increased resources, additional training and reduced bureaucracy (U.S. General Accounting Office 2001). Furthermore, the agency is relying on contributions from the private sector to provide information about security vulnerabilities; this too may be problematic and controversial.
The organisation has developed a diverse group of Information Sharing and Analysis Centers, web information systems and inter-organisational systems all of which are used to communicate information internally and externally. The major types of NIPC communications are summarised in the following table for 1999 through June 2002.

<table>
<thead>
<tr>
<th>Communication Type</th>
<th>1999</th>
<th>2000</th>
<th>2001</th>
<th>2002 (YTD June)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assessments</td>
<td>0</td>
<td>10</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Advisories</td>
<td>7</td>
<td>13</td>
<td>21</td>
<td>7</td>
</tr>
<tr>
<td>Alerts</td>
<td>3</td>
<td>10</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Information Bulletins</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>Total</td>
<td>10</td>
<td>33</td>
<td>29</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 2: Major Types of NIPC Communications for 1999-2002

Assessments address broad, general incident or issue awareness information and analysis that is both significant and current but does not necessarily suggest immediate action. Advisories focus on significant threat or incident information that suggests a change in readiness posture, protective options and/or response. Alerts detail address major threat or incident information addressing imminent or in-progress attacks targeting specific national networks or critical infrastructures. Information bulletins represent general-purpose information about the nation's critical infrastructure.

Additionally, the NIPC publishes CyberNotes, a bi-weekly newsletter. These documents provide detailed information about five main categories of security vulnerabilities: (1) Bugs, Holes and Patches (2) Exploit Scripts and Techniques (3) Viruses, (4) Trends and (5) Trojans. These newsletters are being used as the primary source of information for this research project.

**CURRENT RESEARCH DESIGN**

Current research into this specific programme appears to be very limited. Previous studies in the United States have focused on other organizations or secondary data sources such as: the Computer Emergency Response Team/Coordination Center (CERT/CC), the FBI/Computer Security Institute's Annual Computer Crime Survey and traditional survey instruments.

In Section One bugs, holes and patches address known weaknesses in operating systems and software applications. Preliminary data has been generated to investigate the frequencies of these items based upon the assigned risk ranking. This information is summarised in following table.

<table>
<thead>
<tr>
<th>Risk Ranking</th>
<th>Item Quantity</th>
<th>Percent of Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>1,156</td>
<td>41.64 %</td>
</tr>
<tr>
<td>Medium/High</td>
<td>90</td>
<td>3.24</td>
</tr>
<tr>
<td>Medium</td>
<td>848</td>
<td>30.55</td>
</tr>
<tr>
<td>Low/Medium</td>
<td>60</td>
<td>2.16</td>
</tr>
<tr>
<td>Low</td>
<td>524</td>
<td>18.88</td>
</tr>
<tr>
<td>Other</td>
<td>98</td>
<td>3.53</td>
</tr>
<tr>
<td>Total</td>
<td>2,776</td>
<td></td>
</tr>
</tbody>
</table>

Table Three: Risk Ranking of Bugs, Holes and Patches

The second section of the newsletter deals with recent exploit scripts and techniques. This type of attack creates a risk to all organisations through the execution of known malicious files. The types of exploit files have been summarised by file type in Table Four below:
Table Four: Script File Type Summary

<table>
<thead>
<tr>
<th>Reported File Extension</th>
<th>Item Quantity</th>
<th>Percent of Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>.c</td>
<td>878</td>
<td>33.47%</td>
</tr>
<tr>
<td>.sir</td>
<td>521</td>
<td>19.86%</td>
</tr>
<tr>
<td>.gz</td>
<td>17</td>
<td>.65</td>
</tr>
<tr>
<td>.gx</td>
<td>50</td>
<td>1.91</td>
</tr>
<tr>
<td>.1zip</td>
<td>302</td>
<td>11.51</td>
</tr>
<tr>
<td>.4 beta 3</td>
<td>232</td>
<td>8.84</td>
</tr>
<tr>
<td>.03d</td>
<td>81</td>
<td>3.09</td>
</tr>
<tr>
<td>All Other</td>
<td>542</td>
<td>20.66</td>
</tr>
<tr>
<td>Total</td>
<td>2,623</td>
<td></td>
</tr>
</tbody>
</table>

At this time it further analysis is being pursued to increase the researcher's understanding of specific reporting conditions and scenarios.

In section three, broad IT security trends that the NIPC has become aware of during the bi-monthly newsletter period. These are divided into two major groups: probes and scans; which are then discussed in a narrative, non-tabular format. Accordingly, limited quantitative analysis could be performed since the information is exclusively textual. Nevertheless, some research insights are expected during the second stage of the project.

Viruses are detailed in the fourth major section of the newsletters. Specific in information about known common viruses include: the primary virus name, alias name(s) if any, type of virus and brief textual summary of the virus. Table Five below provides a summary of the various types of virus reported:

Table Five: Types of Viruses Reported

<table>
<thead>
<tr>
<th>Virus Type</th>
<th>Item Quantity</th>
<th>Percent of Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word 97 Macro Virus</td>
<td>385</td>
<td>36.60%</td>
</tr>
<tr>
<td>VB Script Worm</td>
<td>149</td>
<td>14.16</td>
</tr>
<tr>
<td>Excel 97 Macro Virus</td>
<td>66</td>
<td>6.27</td>
</tr>
<tr>
<td>Win 32 Worm</td>
<td>59</td>
<td>5.61</td>
</tr>
<tr>
<td>General Macro Virus</td>
<td>53</td>
<td>5.04</td>
</tr>
<tr>
<td>File Infector</td>
<td>28</td>
<td>2.66</td>
</tr>
<tr>
<td>Boot Virus</td>
<td>26</td>
<td>2.47</td>
</tr>
<tr>
<td>Not Classified by NIPC</td>
<td>113</td>
<td>10.74</td>
</tr>
<tr>
<td>All Other</td>
<td>173</td>
<td>16.44</td>
</tr>
<tr>
<td>Total</td>
<td>1,052</td>
<td></td>
</tr>
</tbody>
</table>

Section five discusses Trojans. This section includes the following key information about Trojans: the Trojan name, version number, alias name(s) if any, brief description and reference information. Statistical analysis of this information is underdevelopment for these items.

While these research results are put forth they remain preliminary and are subject to further review and analysis. In the terms of the qualitative portion of the study a detailed content analysis has been formulated. This component of the study will be conducted after the quantitative portion is completed.
CONCLUSION

The NIPC offers a unique service to all organisations and the general community through its information-sharing model. This information is gathered from a variety of sources and is disseminated to the public at large, discussing such issues as denial of service attacks, computer viruses, exploit scripts and other issues. Although some civil liberty and privacy groups feel that the government is taking too great a role in managing the U.S. computer security environment (Electronic Privacy Protection Center 1998), the researcher hopes that this project will provide useful feedback, suggestions and recommendations to the NIPC itself, computer security researchers and other professionals.

REFERENCES


Benefits of Recognition as a Centre of Academic Excellence in Information Assurance by the U.S. National Security Agency

M.K. Lavine¹ and S. Azadegan²

¹Visiting Assistant Professor
Department of Computer and Information Sciences
College of Science and Mathematics
Towson University, Towson, Maryland, U.S.A.
²Email: mlavine@towson.edu
²Email: sazadegan@towson.edu

ABSTRACT

This paper examines the process undertaken by a large university to achieve recognition as a Centre of Academic Excellence by the National Security Agency (NSA) in the United States. This unique program builds on the critical and growing demands for IT security personnel in the public and private sectors. Beginning in 1999 with just seven certified institutions of higher education, this program has grown to include thirty-five designated colleges and universities in 2002. As a work-in-progress, this paper investigates the objectives and benefits of this programme and details the approach taken by a large university that was successful in receiving this recognition. Also, discussed is a brief history of the programme and its major requirements. The researchers hope to suggest that since IA is a global issue similar programmes could be developed and offered across the world.

Keywords: IT Security Education, Information Assurance, National Security, Academic Excellence, Training

INTRODUCTION

The Centres of Academic Excellence in Information Assurance program is sponsored by the U.S. National Security Agency (NSA) and is believed to be the only programme of its kind. It represents a unique approach to information assurance education at the national government level. Building on the recommendations of the landmark Presidential Commission on Critical Infrastructure Protection (PPCIP), new venues for collaboration between academia, government and industry have emerged (Alexander and Swetnam 1999). These different stakeholder groups have also been developing new technologies, national and industry standards, and academic programs. Previous research by Reynolds (1998) suggests educational initiatives related to information warfare. Additionally, Yang (2002) offers an analysis of the impact of computer security on computer science programs and offers excellent suggestions on how various individual computer science courses can be enhanced by adding security assignments, laboratory experiments and exercises; the researchers are most interested in addressing the related issue from a macro-level.

In 1999, the Centres of Academic Excellence in Information Assurance programme began accepting applications from universities and colleges. Over the past few years, this program has grown significantly and creates a number of advantages for institutions of higher education, the research and industry communities. Towson University achieved this recognition in March 2002.

BACKGROUND INFORMATION ABOUT TOWSON UNIVERSITY
Towson University is the second largest university in the state of Maryland and is located on the east coast of the United States. It is a comprehensive metropolitan university located in the northern suburbs of the city of Baltimore with approximately 14,500 undergraduate and 2,500 graduate students enrolled. There are nine colleges in the university and various research and teaching centres.

In terms of computer information systems, the Department of Computer and Information Sciences (CIS) resides within the College of Science and Mathematics it offers a Bachelor of Science in Computer Science, Bachelor of Science in Computer Information Systems and a Masters of Science in Computer Science. During the most recent academic year, the number of enrolled students was 300, 400, and 200 respectively. Further academic programmes are offered by the Centre for Applied Information Technology (CAIT), an independent unit of the university that is affiliated with the CIS Department. In addition to six graduate certificates, this unit also offers a Masters of Science in Applied Information Technology. In recent years, all programmes have experienced noticeable increases in enrolment which is consistent with national trends.

The thirty-two full-time faculty members in the CIS Department have diverse research and teaching interests. Additional part-time staff, numbered thirty as of the Spring 2002 semester. Members of the faculty have a long and successful series of research projects including significant funding by the National Science Foundation (NSF). In the CAIT, faculty members are predominantly full-time industry practitioners who are teaching part-time.

Building on its faculty expertise and strong academic programmes, Towson University decided to make a submission in early-Autumn 2001. A small working group of five faculty members gathered all necessary information and a project manager developed the actual submission materials. In March 2002, the university received official confirmation from NSA of their designation as a Centre of Academic Excellence in Information Assurance.

PROGRAMME ACCREDITATION AND EVALUATION CRITERIA

The current programme provides an initial three-year accreditation period after which organisations must re-apply for re-certification. Various standards which have been developed Committee on National Security Standards which is affiliated with the U.S. Department of Defence and was formerly known as the National Security Telecommunications and Information Systems Security Committee.

The following table summarises the standards used to evaluate each organisation’s submitted materials, as follows:

<table>
<thead>
<tr>
<th>National Security Telecommunications and Information Systems Security (NSTISS) Standard</th>
<th>Reference Number</th>
<th>Effective Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>National Training Standard for Designated Approving Authority (DAA)</td>
<td>4012</td>
<td>8/97</td>
</tr>
<tr>
<td>National Training Standard for System Administration in Information Systems Security</td>
<td>4013</td>
<td>8/97</td>
</tr>
<tr>
<td>National Training Standard for Information Security Officers (ISSO)</td>
<td>4014</td>
<td>8/97</td>
</tr>
<tr>
<td>National Training Standard for System Certifiers</td>
<td>4015</td>
<td>12/00</td>
</tr>
</tbody>
</table>

Table 1: Primary Current Standards for Program Evaluation
Using these standards as a basis for evaluation, the formal submission is evaluated based on criteria set forth in ten separate sections. The two blind reviewers assigned by NSA are required to follow this scale and each section contains multiple sub-criteria that must be considered carefully by the author/submitter. With minimum values for each category it is critical that the submission meet these basic requirements. Table Two below summarises the main criteria, which are then described in more detail.

<table>
<thead>
<tr>
<th>Section</th>
<th>Brief Description</th>
<th>Minimum Points</th>
<th>Maximum Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>One</td>
<td>Academic program information</td>
<td>25</td>
<td>40</td>
</tr>
<tr>
<td>Two</td>
<td>Information assurance knowledge</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Three</td>
<td>Information assurance practices</td>
<td>10</td>
<td>25</td>
</tr>
<tr>
<td>Four</td>
<td>Information assurance research</td>
<td>15</td>
<td>35</td>
</tr>
<tr>
<td>Five</td>
<td>Information sharing</td>
<td>10</td>
<td>35</td>
</tr>
<tr>
<td>Six</td>
<td>Faculty participation in IA activities</td>
<td>15</td>
<td>30</td>
</tr>
<tr>
<td>Seven</td>
<td>University library and reference resources</td>
<td>15</td>
<td>30</td>
</tr>
<tr>
<td>Eight</td>
<td>IA degree information</td>
<td>5</td>
<td>65</td>
</tr>
<tr>
<td>Nine</td>
<td>Dedicated IA education or research centre</td>
<td>25</td>
<td>50</td>
</tr>
<tr>
<td>Ten</td>
<td>IA faculty background and expertise</td>
<td>10</td>
<td>35</td>
</tr>
</tbody>
</table>

Table 2: High-Level Program Evaluation Criteria

Section One requires universities to map the academic programmes they offer to the various NSTISSL standards summarized in Table One above. This is perhaps the most difficult section of the submission to prepare since it is based on all IA related courses taught at the university; which are not limited to traditional computer security courses. This requires the submitting organisation to gather syllabi and other information about courses in such areas as: business, management, engineering and law.

Section Two examines the knowledge and content of IA topics in different courses offered through the university. Building upon the information gathered for Section One, individual learning objectives and assessment techniques need to be detailed. For example, a course in Cyber Law that is offered in the College of Business and Economics will likely discuss issues related to intellectual property, evidence and privacy.

Section Three requires information about the information assurance practices utilised by the university. Normally, this will include details about the security plans, awareness programmes and IA personnel at the University. Much of this information can be obtained from the Chief Information Officer and his/her deputies.

Section Four deals with information about IA research. In particular, it addresses how IA research is incorporated into student projects both in IA specific and other courses. For instance, an introductory course in IA may require a research paper in encryption or security standards, similar examples should be highlighted in this component.

The focus of Section Five involves how IA information and technologies are shared. Examples of supporting information may include: web-based teaching technologies and resources, on-line library
databases and information about local, national and international conferences where students and faculty members participate.

Section Six deals with IA faculty participation in terms of practice and research. For this section, it is very useful to include specific information about faculty conference presentations, publications and research grants. Similar information about graduate students is also very effective.

Section Seven specifically deals with the availability of library and reference materials. Noteworthy examples should include: historical and contemporary textbooks, research reports, academic journals and practitioner magazines in IA and other related areas such as audit and control.

Section Eight covers specific information about degree programs at the bachelors, masters and doctoral levels. While a university may not offer a degree in IA, instances where IA may be offered as a concentration or area of focus should be highlighted. For instance, at Towson University the B.S. in Computer Science offers a track in Computer Security.

Section Nine examines what type of particular research centre(s) at the university focus on IA. These centres can be at the department, school, college or university levels.

Finally, Section Ten deals with the background, qualifications, and utilisation of IA faculty. In this final section it is very important to include related information about all full-time, part-time, visiting and adjunct faculty members and research associates from all related academic departments and research centres.

Recognition is normally given to a university or college as a whole and not to a specific school, department or academic programme. In the past, organisations could make an application for accreditation either electronically or manually. However, new applicants are required to submit their materials on-line via a secured NSA server.

MAJOR ISSUES AND CHALLENGES ENCOUNTERED

During this project many issues and challenges were encountered. Reflecting on these issues can provide other organisations with some potential best practice information. The project team was entirely composed of four faculty members from the Computer and Information Sciences Department, one faculty member from the Mathematics Department and two adjunct faculty personnel from CAIT. The most difficult issue faced was how to gather information from other academic departments about their courses offered. To address this issue, each member of the project team was tasked with gathering syllabi from other academic departments (Law, Accounting, Management etc.). However, more information was needed regarding learning objectives, teaching methods and assessment techniques. To ensure that all necessary information was gathered and understood individual meetings with faculty colleagues were held.

Another issue dealt with how to gather information from the University’s Information Systems Department about their personnel, policies and procedures. In this situation, the project manager met with the university’s Chief Information Officer (CIO) who made available other personnel in his department, as well as, copies of all necessary supporting documentation. This collaboration was found to be very useful on this project and is likely to result in future benefits on upcoming projects.

Lastly, a colleague from a previously accredited COE functioned as an external consultant and peer reviewer. This function was very useful since it provided for overall quality assurance, subject matter expertise and coaching of project team members. The experiences of this individual in related academic and professional activities proved to be very useful.
BENEFITS OF ACCREDITATION AND RECOGNITION

This programme offers a number of important benefits to participating organizations and members of the faculty and student body. Firstly, the number and type research opportunities are expanded. This is always valuable to a university and its faculty, and is even more important because a number of new research grants require recognition by NSA. Secondly, it assists universities in attracting students to join its degree programs. Thirdly, faculty recruitment is aided because information assurance is in scarce supply and this recognition demonstrates the institutions commitment to information systems security. Lastly, the practitioner community benefits as well by knowing that recruitment candidates were enrolled in a degree program at a NSA recognized college or university.

Even if a university is not successful on its first attempt, a number of important lessons can still be learned, as this process proved to be an excellent overall academic assessment exercise. Firstly, a broader understanding of its IT related academic programmes will be understood. Course content crossover issues and gaps can be identified which can then be used to improve the quality of individual courses. Secondly, a greater awareness about IT security will likely be achieved by the university’s information systems and academic leadership; which should strengthen the organisation’s overall understanding. Thirdly, opportunities for new courses, training programmes perhaps even certificates and degrees that might be considered by academic and administrative departments.

NSA as a government agency also recognizes numerous benefits from this programme. The agency has been able to create an environment whereby institutions of higher education share information about their academic programmes and what students are being taught related to IA. This creates a sound basis for a true working relationship with academia and industry. Specifically, the agency helps foster a sense of participation and cooperation with the academic community. Primarily this involves development of future training standards and funding of basic research projects. Additionally, the agency also is able to participate on a proactive basis with the development of course materials and instructional exercises in the IA field. Lastly, NSA has been successful in involving leading IT companies like Microsoft and CISCO in this programme. This serves to promote IA as an issue at the industry and business level. Bringing the industry, academic and government groups together also has some intangible benefits.

UNIVERSITIES RECOGNISED AS COAE/IA

Beginning in its first year, 1999, seven universities were accredited as Centres of Academic Excellence in Information Assurance by NSA. These included: James Madison University, George Mason University, Idaho State University, Iowa State University, Purdue University, University of California at Davis, and University of Idaho.

Then in 2000, another group of seven universities were similarly recognized by NSA. These included: Carnegie Mellon University, Florida State University, Information Resources Management College of the National Defense University, Naval Postgraduate School, Stanford University, University of Illinois at Urbana – Champaign, and University of Tulsa.

Then in 2001, a further group of nine universities were accredited. These included: Drexel University, United States Military Academy, Georgia Institute of Technology, University of Maryland – Baltimore County, Mississippi State University, University of North Carolina – Charlotte, Norwich University, West Virginia University, and Syracuse University.

Most recently in 2002, another thirteen universities were awarded COE status by NSA. This group included: Air Force Institute of Technology, George Washington University, Indiana University of Pennsylvania, New Mexico Tech, North Carolina State University, Northeastern University, Polytechnic University, State University of New York – Buffalo, State University of New York –
During 2002, all of the original seven institutions recognized in 1999 were all re-certified. It is noteworthy that the re-accreditation process requires a complete and total re-submission of current materials to NSA in the same format as first time candidates. The following table provides a summary of currently accredited institutions of higher education by type:

<table>
<thead>
<tr>
<th>Type of Higher Education Institution</th>
<th>Total Number Accredited As of June 2002</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public College or University</td>
<td>23</td>
</tr>
<tr>
<td>Private College or University</td>
<td>9</td>
</tr>
<tr>
<td>Military/Government Affiliated College or University</td>
<td>4</td>
</tr>
<tr>
<td>Grand Total</td>
<td>36</td>
</tr>
</tbody>
</table>

Table 3: Summary Data for the Current Centres of Academic Excellence in Information Assurance by Type

Please note that some private colleges and universities may receive some federal and/or state funding and that public colleges and universities receive various types and levels of federal and/or state funding.

CURRENT AND FUTURE RESEARCH OPPORTUNITIES

Current research into this specific programme appears to be extremely sparse. The authors of this paper hope to begin a research project in this area in the near future. In terms of specific research opportunities, the following specific topics are currently under consideration by the researchers:

1. Assessing the effectiveness of the Centres of Academic Excellence program,
2. Comparisons to other similar national government initiatives in other countries, and
3. Methods of enhancing the programmatic standards and criteria to reflect current industry practices.

The researchers welcome the opportunity for collaboration with colleagues at other organisations.

CONCLUSION

The U.S. National Security Agency’s Centres of Academic Excellence in Information Assurance programme is an exciting initiative in the area of information assurance education. The successful efforts undertaken by Towson University in 2001 may serve as a useful case study for other organisations embarking on similar efforts. The benefits to NSA, Towson University and its faculty and students are plentiful. Furthermore, the existing framework of standards and criteria established for Centres of Academic Excellence in Information Assurance in the U.S. could be considered as a prototype for other national governments looking to develop programmes in IA education and research.

REFERENCES


From Information Security to Information Warfare: a paradigm shift

W. Hutchinson

School of Computer and Information Science
Edith Cowan University
Mount Lawley, Western Australia
Email: w.hutchinson@ecu.edu.au

ABSTRACT

As the Western nations move further into the Information Age, the strategic nature and value of information becomes more apparent. The conventional approach to the security of that (corporate) information and its associated systems has conventionally been a protective and a reactive function. The contemporary concept of Information Warfare has meant that the security function must expand to incorporate both the concept of information as a ‘target’ and that of a ‘weapon’. This paper is the (ongoing) story of changing conventional university information security courses (and thus research) into this new paradigm.

Keywords: Information warfare, information security, intelligence, education.

BACKGROUND

Edith Cowan University (ECU) in Western Australia has been involved in security education for a number of years. Primarily run by the School of Engineering and Mathematics (Engineering) and the School of Computer and Information Science (Computing). These Schools are both in the Faculty of Communication, Health, and Science. Engineering concentrates on physical and electronic security. It has a number of awards such as the Bachelor of Science (Security Science), and post-graduate awards in Security ranging from Graduate Certificate to doctoral studies. Computing has awards as minors in undergraduate awards, and postgraduate awards ranging for Graduate Certificates in Computer Security to doctoral studies. Examples of course units are illustrated in Table 1.
Graduate Certificate in Security Management  
(Engineering)  
Any 4 units from:  
- Computer Security  
- Information Security  
- Computer Facilities Security  
- Security and Risk Management  
- Physical Security  
- Electronic Security I  

Graduate Certificate in Computer Security  
(Computing)  
- Computer Security  
- Information Security  
- Computer Facilities Security  
- Database Security  

Table 1: Examples of units in existing courses

It can be seen that courses had a lot of overlap. However, recently this situation has changed with Engineering dropping computer-based units for more physical and electronic security units.

Apart from these course based post-graduate courses, there were (and still are) research based courses in Computer Security and Security Science as well a doctoral programme in each. However, there was no course that covered the information aspects of security and intelligence.

A CHANGE IN OUTLOOK

The developing area of Information Warfare was beginning to interest a number of academics around the university coming from a diverse number of areas in the university. The process of recognising this common interest came slowly at first. An interested individual started the process by ‘trawling’ various Schools and found academics working in related areas from Justice Studies (cybercrime), Security Science (intelligence), Information Systems (information warfare), and Computer Science (computer/network security).

Strangely, the first course in Information Warfare did not come from the two 'big' players of security education with ECU (Computing and Engineering) but a unit within the Doctor of Business Administration (DBA) offered by Information Systems (in the business faculty) called 'Information Warfare'. This doctoral unit attracted a lot of attention from doctoral students. It consisted of four modules:

- **Information Warfare** based around offensive information warfare (Denning, 1999; Waltz, 1998), and deception
- **The Intelligence function** based around the intelligence cycle (Boni & Kovacich, 2000; Herman, 1996) and the concept of a 'surprise attack' (Kam, 1988)
- **Security** based on the concepts of computer security (Schneier, 2000)
- **Future trends** based initially on the concepts of cyborgs and 'swarming' (Arquilla & Rondfeldt, 2001)

After this unit was finished, a number of events started to consolidate the ideas, which had been developing. Firstly, an inter-faculty group was formed interested in the areas of information warfare and cybercrime. This had members from the areas mentioned above. A book was written for the requirements of the Information Warfare unit (Hutchinson & Warren, 2001a) with a colleague from another University, the first issue of Journal of Information Warfare was published, and, at the end of 2001, a successful international conference was held in Information Warfare and Security.

At the end of 2001, the author was put in charge of the computer security courses in Computing (having moved from Information Systems). The team involved in these courses then had an audit of the security offerings within Computing. It became evident that there was a lot of overlap. The most
startling duplication was that between computer and information security. After a detailed search on
the Web, it became evident that the terms 'computer security' and 'information security' were being
used synonymously.

INFORMATION WARFARE AS THE PARADIGM

As a beginning to the process of 'renewing' information security to the 21st century, it seemed sensible
to establish the meaning of 'information'. The conventional linear definitions of data, information,
knowledge, and wisdom with each stage having a greater degree of collation and involvement with
context and learning did not appear to be very useful. In fact, this definition had appeared to delineate
many functions such as knowledge management, information management, and so on. (The concept of
'wisdom management' was beginning to be touted!) What was needed was a definition that reflected
the integrated nature of the 'information' security function. Thus, the development of the courses began
to be based around the definitions of data, information, and knowledge developed from Boisot (1998)
- see Hutchinson and Warren (2001a, 2001b). In his model, data is associated with a thing, and
discriminates between different states of the thing it describes. It consists of attributes of the events or
objects it describes. On the other hand, knowledge is an attribute of an agent. Knowledge is a set of
interacting mindsets about data activated by an event. Hence, in most circumstances the word 'agent'
means a human being or a group of people. Information is the set of data filtered by the agent within
the bounds of the knowledge held by the agent. It establishes a link between the agent and the data.
Figure 1 illustrates the concept.

Data

Using the model developed above, the basic concepts of information security can be shown. Figure 1
illustrates the main attack strategies. Most successful attack strategies are integrated affairs and hence
would be a coordinated effort on one or more elements. It would also be ongoing and flexible, and not
static. However, for the purpose of demonstration each element will be separated.

If the target is the data, a number of things can be done:

- **Deny access to data**: this can be achieved by attacks on hardware or systems containing the
data or its collection, or deletion of data. As much data has a temporal dimension, it could also
involve the delaying of access to data to the point at which it becomes useless. These attacks
can range from denial of service to the deliberate withholding of data.

- **Disrupt or Destroy data**: this is similar to the above, but disruption can be caused to the
system collecting and storing the data, or to that part of the system, which disseminates it.
Destruction of the data can occur by physical destruction of the storage medium, or the data
itself, so it becomes irrecoverable in the time needed to make it useful. Of course, it can be
argued that data is never destroyed, just the medium on which it is stored.

- **Steal data**: much corporate data is confidential and can also give competitive advantage.
Theft of this data (and remember, theft of data can go unnoticed as the victim could still have it)
might give insights into the workings of the attacked thereby giving the attacker a possible
business, negotiation, or criminal advantage.

- **Manipulation of data**: data can be added, deleted, or amended to give the attacker advantage.
A person committing fraud would often use this method.

Context

The objective in alerting the context of a situation is so the target will misinterpret the data being
presented. This can be achieved by affecting environmental or sensory signals received by he target in
any particular situation. It is similar to an attack on data but is more ephemeral. When attacking
'context', one is trying to alter the situation in which the data is viewed; this can include such things as
place, sensory surroundings, and political climate. However, attacking the data is really concerned with manipulating the data to be interpreted.

**Knowledge**

The strategies to deal with knowledge tend to be more long term. As mental models are developed by a person's experiences, they are created by education, social interaction, emotions, and so on. Changing perceptions is directed more toward the people themselves, and their thought processes. This can include public relations, advertising, and incentives. The assumption is that the attacker will **exploit** any situation created by the attack. It is also assumed the victim will attempt some defence.

![Diagram showing the relationships between data, context, knowledge, information; and the methods by which each element can be attacked (from Hutchinson & Warren, 2001b)]

It became obvious that information security was more than just data security. In fact, the whole area is akin to the conventional definition of intelligence and counter-intelligence (Richelson, 1995). This made information security a much broader concept than just a factor of computer security. It became an integral part of an organisation's structure. It was a continuum from the extreme of physical destruction of equipment and data storage media to very subtle perception management (psychological operations) - see Campen & Dearth (2000).

**THE NEW COURSE**

An initiative was now taken by Computer Science to develop a multi-school, interdisciplinary course involving all those mentioned above plus the School of Communications and Multimedia who would write the 'softer' units on Perception Management. The normally difficult task of coordinating different organisational entities was made easier by overt support from senior management. The design of the course was sent to various people industry, government, and academia for comment.
The course had to include all the elements included in the Boisot based model (figure 1): data, knowledge/context, and information. In order, these mapped thus:

- **Data**: conventional computer/network, information security units
- **Knowledge/context**: perception management
- **Information**: intelligence

These elements had to be held together with a 'general' unit on information warfare that brought them all together. The Boisot model was emphasised in each of the units especially the first introductory subjects. The initial Information Warfare unit introduced the unity of all these elements in an organisational sense. The advanced Information Security unit also integrated the subjects. (This unit should really have been named Advanced Information Warfare). Treating 'information security' as a proactive and dynamic activity was thought to make it more relevant to organisations and bring it more into the mainstream business, rather than a marginalised function regarded as a cost. Information Security became a value added business role.

The range of topics required for a comprehensive information security course is illustrated in Figure 2.

The course developed is a postgraduate offering in Information Security and Intelligence and is available in on-campus mode or fully on-line. It consists of three stages: Graduate Certificate, Graduate Diploma, and Professional Masters. Entry requirements are an undergraduate degree or five years appropriate work experience. Table 2 shows the composition of the units for each stage.

![Figure 2: Examples of the range of topics involved in the concept of information warfare/security](image)

The compulsory content gives the student the range of technical skills required in a technological environment as the softer management and psychological aspects of information security. The elective content gives the participant the ability to either take a range of units in the area, or concentrate on a specialist area such as computer/network security. As it was originally designed as a course work Masters, Stage 3 originally consisted of three, advanced specialist units. However, a number of prospective students expressed a desire to either complete work-based projects or research. The option was then added to allow students to complete a work or research related project instead of the final three advanced units.

In stage 1 (see Table 2), the compulsory units cover both the defensive ('Information Security') and offensive ('Introduction to Information Warfare') aspects of the area. The two elective units allow the student to complete two specialist units from the hard and soft spectrum of topics. Stage 2 covers the range of topics within the intelligence function. The units cover advance Information Security (from both defensive and offensive perspectives), the psychological impacts of information usage ('Perception Management'), and the principles of intelligence and counter-intelligence...
(‘Contemporary Intelligence’). Stage 3 then allows a student to further research a topic of interest, or take advanced subject units.

<table>
<thead>
<tr>
<th>Compulsory units</th>
<th>Elective units</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Stage 1: Graduate Certificate</strong></td>
<td><strong>Take any 2 units from:</strong></td>
</tr>
</tbody>
</table>
| Information Security *(a conventional information security unit covering the defensive function of security)* | *Technology based units*  
| Introduction to Information Warfare *(a general unit integrating all the elements in the Boisot model)* | Database Security  
|  | Computer Security  
|  | Introduction to Knowledge Management  
|  | The Information Society  
|  | Fundamentals of Cyber-crime  
|  | Physical Security  
|  | **Social/human related units**  
|  | *(Media and Advertising)*  
|  | Media and Nation  
|  | Global Communications  
|  | Introduction to Media and Communication  
|  |

| **Stage 2 - Graduate Diploma** |  |
| Contemporary Intelligence *(covering the information/intelligence aspects of the Boisot model)* |  |
| Perception Management *(covering the knowledge/context aspects of the Boisot model)* |  |
| Information Security *(covering a continuation of the both the lower level Information Security and Information Warfare units, integrating them both)* |  |

| **Stage 3 - Masters** Research project (3 units) OR |  |
| any three Advanced from: |  |
| *Technically oriented units* |  |
| Computer Security |  |
| Network Security |  |
| Database Security |  |
| *Social/human oriented units* |  |
| Media and Social Issues Ethics, Values and Moral Decision Making |  |
| Current Issues in Security |  |
| Advanced Security Risk Management |  |
| Advances in Security Technology |  |

Table 2: The break up of units for Information Security and Intelligence awards
The main, desired graduate attributes from this course (apart from content knowledge and computing skills) are based in the cognitive realm. Each unit has its own stated objectives but overall the student should develop skills in observation, analytic and forensic skills, inductive and deductive reasoning, and lateral thinking.

Also, there is a requirement to set information security in a social, ethical, political, and organisational context. These are achieved by a number of exercises such as:

- computer forensics (for example: analysis of log files)
- case studies (analysis of situations)
- 'thinking like the enemy' exercises (students will have to justify the position of the enemy, etc.)
- debates (students will debate contentious subjects such as staff surveillance, biometrics, privacy, etc.)
- conclusion exercises (students given scant information and asked to come up with scenario to expose hidden assumptions)
- scenario exercises (role playing in situations)
- lateral thinking exercises
- observation exercises

Basically, the student will have to have the skills to think both like an attacker and a defender, and make decisions whilst considering the social, ethical, organisational, and legal context of the problem.

CONCLUSION

This paper describes how using the concept of information warfare has expanded the idea of the range of information security education programmes. Originally, a subset of computer security it now encompasses the whole organisational context of information as a 'weapon' (the information warfare influence) and information as a 'target' (the security perspective). In modern, networked organisations, the role of information security is more than just protecting the information base (which necessarily involves both people and data). The course described above attempts to include all the elements required in an integrated information security strategy. It acknowledges that security is not a passive function but one that is actively involved in both defensive and aggressive uses of information.

The content of the course was developed with the aid of people from the Australian Institute of Professional Intelligence Officers, the Department of Defence, the West Australian Police Service, and industry. Each of these contributors was giving opinions and advice as individuals rather than official representatives, but the range of experience and expertise present gave the developers some confidence that the development was proceeding along the correct track. The unit contents were sent out for comment, as was the final course structure.

The overall success of this venture has yet to be determined as there have been no graduates; the course starting in mid-2002. There has been much interest from industry, intelligence related government departments, the military, law enforcement, and even local government. The present student cohort comes from the finance industry, law enforcement, military related private industry, and local government. The course's effectiveness will be monitored and be fine-tuned to accommodate any shortcomings.

REFERENCES


