Incident Handling Assessment of Division W of Ajax Corporation

By Michael McNally   June 23, 2001

Advanced Incident Handling and Hacker Exploits
GCIH Practical Assignment Version 1.5c

Table of Contents

Executive Summary

On August 17, 2000 Division W of Ajax Corporation had been made aware that someone had compromised one of their systems. At first it seemed as though the problem was caused by a miss-configuration, but as the investigation went on, it was quickly discovered that the problem was the result of an intrusion.

A review pointed out that an unauthorized user accessed the payroll system that serves one of our regions, luckily, the account had low-level privileges and the hacker did not elevate those privileges. This kept the hacker from possibly from reading or modifying our payroll data. The system was not compromised in any other way. Users lost availability of service for three machines for one week, two machines for three days and 5 machines for about three weeks. The computer support staff spent over 200 hours recovering.

In order to prevent these types of intrusions into our systems the following report makes some very specific recommendations in the follow-up section of this document.

1 Introduction

This document has been written in order to fulfill my requirements for the SANS Practical assignment for completing the Advanced Incident Handling and Hacker Exploits version 1.5c training. The information in this document will provide the reader with an understanding of what is being done at Division W of Ajax Corporation, not the real name of a very real organization. With every organization some things are being done well, others not so well and some not good at all.

The document covers the six main phases of incident response: Preparation, Identification, Containment, Eradication, Recovery and Follow-Up. In each section are discussed issues of security in that phase and how Ajax Corporation is addressing them. The discussion uses a real incident that involved several systems in order to highlight the issues with regard to security incidents. All information has been sanitized. The reader should understand that this is paper about a real incident and how it was handled, this handling should not necessarily be considered the “correct” way to do it.

2 Preparation

The following issues are discussed to assist the reader in understanding the readiness of Ajax Corporation to deal with Security Incidents.

2.1 Policy

The more I work in security the more I realize how important the role of policy formation is when going about the business of protecting your assets. Policy however, does not contain information about how to conduct operational tasks in order to effect security, for instance, how to set up a firewall. This is necessary if security is to really be implemented. The necessary level of detail needed to do this is contained within the guidelines. Following is text that will explain the roles of three different levels of detail of documentation that make up the necessary documentation to implement good policy practices.

Garfinkel, Simson, Gene Spafford discuss the three basic building blocks of policy, standards and guidelines in developing the security program:

> Policy plays three major roles. First, it makes clear what is being protected and why. Second, it clearly states the responsibility for that protection. Third, it provides a ground on which to interpret and resolve any later conflicts that might arise [. . .]What the policy should not do is list specific threats, machines, or individuals by name—the policy should be general and change little over time. [. . .] Standards are intended to codify successful practice of security in an organization. They are generally phrased in terms of "shall". Standards are generally platform independent, and at least imply a metric to determine if they have been met. Standards are developed in support of policy [. . .]. Guidelines are the "should" statements in policies. The intent of guidelines is to interpret standards for a particular environment—whether that is a software environment, or a physical environment. Unlinke standards, guidelines may be violated, if necessary. As the name suggests, guidelines are not usually used as standards of performance, but as ways to help guide behavior. (35-37)

Ajax Corporation has about 35 pages of policy and recommended best practices, which covers most aspects of use and management of computer resources. The information is very good. There is a section, which includes the phrase that all users should have no expectation of privacy when using corporate networks. This sentence gives security people the needed authority to conduct an unrestricted investigation. The reader should note this is one of the most important authorities those involved with security can have. Without this, personnel may not be able to view vital information that could uncover a security breach.

As pointed out previously, procedural guidelines are necessary in order to have an effective Security Program. Ajax does not have an effective security program due to the following policy related issues:

1. There is no development of procedural guidance that reflects the developed policy. Only one Division out of four has such procedural guidance. This has frustrated; leaving Systems Administrators left to their own methods for enacting procedures that may or may not reflect policy.
2. There is no program of auditing at the organizational level. This has resulted in the organization not knowing who is really in compliance.
3. There are no statements of consequences due to not complying with the policy in the policy statements. This results in providing little motivation for those that do not understand the wisdom of the policy that they are to follow.

These oversights have led to a patchwork of successes and terrible failures in complying with policy.

2.2 Prevention

Historically much of the program of prevention has focused on the host. In Division W, the corporate platforms have what are called Technical Advisory Committees (TAC) that serves to provide guidance on an operating system for use by the Systems Administrators. Guidance is provided that is very specific and complete for instructing the Systems Administrators in installing, configuring and updating of operating systems and applications that run on them. These TAC groups keep up to date on security releases, making any necessary changes to their guidance. End users are made aware of security releases by email. Phone support is also available on a limited basis. In addition to configuration management, the TAC groups have provided additional prevention in the way of TCP Wrappers on the Unix platform. TCP wrappers essentially make decisions about which source IP can use a given application (e.g., telnet, ftp), however their use is limited but very useful. For the NT platform some administrators have deployed an application called Nuke Nabber that provides some security controls in the way that TCP wrappers does.

In the last year there has been a major effort to deploy firewalls in Division W. Previously there was no filtering of network traffic at the perimeters, except for blocking hostile IP addresses. Division W has about 165 sites that need to have firewall capability at them.

Early in the year 2000, a pilot test had been conducted testing two firewall solutions. The pilot took about 8 months. During the pilot, the following was determined: effect on throughput rate of network traffic, improvement in security, the ability for those in the protected area behind the firewall to continue to do their work and the needed resources to deploy and manage the firewalls. The result of the pilot proved that the throughput on the wire was not significantly reduced, that the deployment of a firewall will reduce the number of vulnerabilities at a location (i.e., the measure of security), and that users can continue to do their work without disruption after the deployment of the firewall. The results of the pilot also gave a good estimate of the amount of resources needed to deploy and manage the firewalls.

After presenting the results to management, a proper funding level was authorized. The work to protect 165 sites was started, as of today 150 sites are protected with firewalls. All sites should be protected by the end of calendar year 2001.

2.3 Management Support

There has been some opportunity to present the case of a need for increased funding for security. Presentations have been made which included specific instances of incidents describing their impact to the organization. To date there has been insufficient funding to hire a sufficiently sized security team. This has left the organization with an incomplete security program.

2.4 Incident Preparation

The following has been done in preparation for an incident:

1. Every potential user is presented with a logon banner which includes the warning that they may be prosecuted if they are not authorized to use the system and that there should be no expectation of privacy.
2. An incident response team had been set up for responding to events.
3. An email group was set up that could be used by Systems Administrators that could be used to alert the response team.

2.5 User Education

Users are required to take a security-training course when they start on the job. I took the training when I started 13 years ago. It consisted of a couple of type written pages of things you should know. During my entire time while working as a Systems Administrator was I ever made aware of a Security Policy, who to contact in the event of a security event, what constituted a security event or what to do when some event happened. Training in the last year has improved. There is still not any information on what constitutes a security event and written procedures for Systems Administrators.

Information on who to contact if a security event or question arises is very well known. There is an email alias that contains the addresses of all security team members. This ensures that everyone that should be notified gets notified if email is used as the means of contact.

There has been a lot of work in making presentations at organization conferences, meetings and by using email on the status of security.

2.6 Incident Response Team

Three years ago a charter was drafted that laid out the purpose and make-up of a team of individuals that would respond to events. All of the members but four had full-time jobs doing work other than security. Four members of the group have had formal training at SANS conferences and on the web. The others have learned by doing Systems Administration work

The team has yet to establish information on what tools and techniques that is to be used in the event of an incident. Thus team members are as a group left to their own ideas about what tools are needed and how to use them. This is a terrible oversight that needs to be corrected.

Members usually act on their own as to how to proceed. Usually the team member deals with a Systems Administrator using email or the phone to resolve the problem. The current practice is to contain, clean and re-establish function. Some events are not even properly analyzed to determine the real cause. Systems have been rebuilt not ever discovering the cause or extent of the intrusion.

Incidents that require involvement of the authorities are usually handled well. All collected evidence is turned over to the legal authorities and any cooperation is made available to them in pursuing the case. These types of incidents are recognized by the team and systems administrators for what they are and treated as beyond the scope of work by the security team. However, there is no information and training about how to properly deal with these types of incidents. Information on how to provide a well-documented trail of handling of evidence would be helpful in these situations.

2.7 Disaster Recovery Planning

Disaster recovery plans have not been updated to include a computer security incident. This weakness has been made evident in some incidents that demanded a hard look at removing a critically important server during a compromise. Fortunately the evidence discovered was not strong enough to justify this step during one incident that involved a mission critical system. The risk was taken to keep services up while at the same time deeply investigating what had happened.

2.8 Incident Response Procedural Guidance

There is no procedural guidance for Systems Administrators that could assist them in investigating an event. That guidance could provide them in understanding what to expect from the Response Team and how to proceed through the process. As for the checklist there is guidance on the CERT web site, which we may adopt for future use as the procedural guidance is developed. The reader is referred to the following links as reference:

http://www.cert.org/tech_tips/intruder_detection_checklist.html http://www.cert.org/tech_tips/win_intruder_detection_checklist.html

2.9 Incident Response Contact Information

One of the most important pieces of information needed when responding to an incident is knowing who to call that has responsibility over a system. Often the security personnel are notified about a system, which is identified only by the IP address. The Division has over 5000 systems at about 165 locations. Having information about who has administrative control over that IP is invaluable in quickly dealing with that system.

When registering systems to be connected to the network, the current procedure is for the Systems Administrators to provide information to one of three DNS registrars. Information on the machine IP, hardware address, name, the name of the Systems Administrator, their phone number and street address is provided to the registrars by a fax machine.

However, all of this information is on paper copy in the registrar's office. Performing a name lookup on the IP address will provide only the name of the registrar. For several years the security team has been asking to have that information made electronically available so that a contact person can be determined during an incident. So far that request has been unfulfilled despite many requests and meetings.

2.10 Incident Reporting

The reporting of incidents is very critical to the Security Program. This information can assist in providing management with just cause for outlaying funds for the Security Program. The information can also serve to understand the bigger picture of incidents, which could serve to improve defenses or understand the scope of an attack on the systems. It is important that the reporting is as easy as possible; otherwise people will not fill them out. Currently, there is an electronic form to fill out for an incident. The process is to download the PDF, fill it out by hand and then fax it to the Security Manager who puts it into a cabinet. The process has caused many to not fulfill the request to file an incident. Also, the information as it is in written form is not very useful.

Currently the team is building a web-based form that will populate an electronic database. This information will assist in developing reports that can be used to apprise management on incidents. This information can also be used to develop an understanding of trends in incidents.

2.10 System Access by Security Team

The security team does not have access to passwords. Only the sites Systems Administrators have them. Without their assistance the team cannot gain access to these systems. Last year only Division W within this organization got the authority from senior management to remove a system from the network that had been compromised. The rest of the organization has yet to get that authority.

3 Identification

The first known date and time of my awareness of the incident was on the morning of August 17, 2000. A Systems Administrator walked into my office and stated that users were complaining that their systems were unusable. That an error message would pop up on the screen of computers with the message that there was a duplicate IP or in some cases it would report a duplicate hardware address. The event seemed unusual but at this point we were still investigating a potential mis-configuration and not a security event.

In order to determine which two hardware addresses are using the same IP address we used the ARP (Address Resolution Protocol) command with the “a” option to list the arp table information on systems. The author Douba, Salim explains the role of arp in networking:“The IP address you assign to a host is independent of the MAC address that is hardwired on the network interface card in that host. As such, every host (network interface) ends up maintaining two addresses: the IP address, which is significant to the TCP/IP protocols only, and the MAC address, which is significant to the network access layer only. [. . .] This is where ARP comes in to handle the IP address, which is useless from Ethernet’s point of view (assuming Ethernet at the MAC layer), to a MAC address that Ethernet understands. Put differently, ARP translates the symbolic address that the host uses to identify itself at the physical and data link levels.
ARP handles address resolution by sending out of the MAC interface (for example, Ethernet) a broadcast messages known as an ARP request, which simply says, “I, host 147.27.2.5, physically addressable 0x0000c015ad18, want to know the physical address of host 147.27.34.1.” Of all the hosts that receive the broadcast only jade responds, using a directed ARP response packet, which says, “I am 147.27.34.1, and my physical address is 0x00001b3b21b2. “. (82-83)

The arp table is a list of cached ARP information as explained by Douba, Salim “When an IP address is resolved to its equivalent MAC address, ARP maintains the mapping in its own special ARP cache memory, improving transmission efficiency and the response time to user requests.”. (84)

By using the arp command with the “a”option

```
arp –a
```

It was discovered that output similar to the following was being seen where the same IP address was being used by more than one physical address on the network:

```
Net to Media Table
Device IP Address                    Mask   Flags           Phys Addr
------ --------------------         ---------------         ----- ---------------
hme0   ajax.com                      255.255.255.255         00:00:00:00:00:00
hme0   ajax.com                      255.255.255.255         00:00:00:00:00:01
```

An hour afterward I connected up with the network team in the building. They had a hardware address of the machine that was causing the problem. However, they were having trouble locating the machine. There were no records available to them to locate the machine in the building with only the hardware address. The next day it was discovered that a person in the section that supports computers had that information for most systems under their management. The person responsible for registering systems (DNS registrars), who also work under the same management as the network team does not electronically record that information. So if he is not working that day which was the case during this event, than that information is not available to the network team. There are over 2600 systems in this building. Anyway, using sniffers the network team was able to determine which machine was causing the problem.

The system that was causing the problem was responding to ARP requests with its own hardware address and IP address. It was discovered that a PC was running Red Hat Linux 6.1 and that it was using many Virtual IP addresses on one Network Interface Card (NIC). Using the command last which shows all login and logout information the following was found; the output that has been sanitized showed an IP address from a foreign country. We checked with the real user, he did not make this connection.

```
        Joe pts/4        192.168.1.1  Wed Aug 23 20:14 - 20:25  (00:10)
```

The system was not checked for anything other than this at the time. It was late in the day; we removed the system from the network and placed it into a secured area for further analysis.

On Monday August 21, 2000 it was found that the problem had resurfaced. That is, the IP addresses in the ARP cache of systems were found to have the same hardware address for the same IP address. The machine was quickly located. We also were working on the assumption we were dealing with a security incident. This machine was also a PC running Red Hat Linux 6.1.

We concentrated most of our efforts in reviewing information from the following sources: the system log files, output from the netstat command, output from the last command, output from the who command and output from the ps command. The system log files would give us information on connection attempts to system services, reboots and other items of interest. The netstat command gave us information on listening ports, as we were interested to see if the hacker planted any servers. The last command gave us which userids were accessed from where and at what date and time. The who command gave us an idea of who may still be logged in on which terminal lines. The ps command gave us an idea of which processes were running. Having a good idea, which services should be running we could determine which processes should not be. This information could possibly give us an idea if any hacker tools were running. This information could also give us information on what binaries to search for and also possibly help us to locate a place where the hacker dropped their toolkits. We found no unusual events except that the there was an account named games and this had been used to gain access from the same foreign address that we found to access the first system.

We also reviewed the process table using ps and found nothing unusual. Upon analyzing the directory under the account for joe, code was found on the system to gain root easily. First logging into the system with low access privileges, I then ran the code left by the hacker and was instantly granted root access to the system. There was also code left on the system that caused the system to act as a chat relay.

Hacker toolkits were found under two user accounts in their home directories and in /tmp on two systems under the name "…", this name was chosen as an attempt to obscure the presence of these tools. We also searched for other names that may hide the name by using the command:

```
ls –bl
```

This command will print out a detail listing of files (one per line) and include any non-printable characters in the file names, again, this is an attempt to obscure the presence of files left by the hacker. The following listing was found under the joe user account:

```
70 joe (ksh): ls -altr
total 1974
drwxr-xr-x   4 joe dis         512 Aug 14  1998 MH-Mail/
-rwxr-xr-x   1 joe  dis       2509 Aug 14  1998 .xsession*
-rw-r--r--   1 joe dis         968 Aug 14  1998 .mh_profile
-rw-r--r--   1 joe  dis       1088 Aug 14  1998 .profile
-rwxr-xr-x   1 joe dis         464 Aug 14  1998 .logout*
-rw-r--r--   1 joe dis        2019 Aug 14  1998 .exrc
-rw-r--r--   1 joe dis        1094 Aug 14  1998 .exmhsedit
-rwxr-xr-x   1 joe dis       16675 Aug 14  1998 .Xdefaults*
-rw-r--r--   1 joe dis         306 Aug 14  1998 .login
drwxr-xr-x   3 joe dis         512 Aug 24  2000 .../
-rw-r--r--   1 joe wrd      957952 Sep  4  2000 rbwinst.tar
drwxr-xr-x   9 joe wrd         512 Sep  6 16:39 ../
drwxr-xr-x   4 joe wrd         512 Sep  6 16:40 ./
71 joe (ksh):
```

From the README file of the code left behind, the program description is as follows:

```
psyBNC 2.1
----------

This program is useful for people who cannot be on irc all the time.
Its used to keep a connection to irc and your irc client connected.

Being installed on a shell with a permanently connected machine you stay
connected as long you want or until the program crashes *g*
```

On August 23, 2000 it was discovered that two Sun systems had logins from the same compromised account discovered on the first two Red Hat systems. In addition, another account was also compromised. These two systems had no other evidence of tampering other than unexplained logins on these two accounts using telnet.

This was found on one of the Sun machines; this shows connections from foreign addresses for this account The real user did not make this connection.

```
---------------------------------------------------------------------
Ksh: last | egrep [0-9]+.[0-9]+.[0-9].[0-9]
sam   pts/2       192.168.1.1        Thu Aug 24 20:12 - 20:19  (00:07)
sam   pts/6       192.168.1.1        Thu Aug 24 09:17 - 09:18  (00:00)
joe   pts/1       192.168.1.1        Sat Aug 19 11:53 - 12:50  (00:57)
joe   pts/1       192.168.1.1        Sat Aug 19 11:39 - 11:51  (00:12)
joe   pts/2       192.168.1.1        Fri Aug 18 17:03 - 17:10  (00:06)
joe   pts/8       192.168.1.1        Fri Aug 18 12:37 - 13:35  (00:57)
```

Reviewing systems methodically was problematic for me as there were no checklists provided by the security team. I was only on the job about 4 months when this happened. There was a lot of pressure to leave a system operational from the system owners on one side and the security personnel to take the system down and scrub it on the other side. I took a risk-based approach based on the following known evidence:

1. Only Red Hat Linux systems had been severely compromised, most likely using the program discovered on one system, which gave an unprivileged user root access. No such tools that were usable on the Solaris platform were discovered on the Sun.
2. The date of the logins from the remote IP addresses found on the Sun platforms were later than the dates found on the Red Hat Systems we reviewed. This fact led me to think that the accounts were already compromised.
3. There was no odd activity reported by the Systems Administrators and no other irregular evidence that could be found. We used information from the system logs, netstat, ps, who and last.
4. We knew that these systems were tightly managed. That is they followed all of the TAC instructions as to configuring the system, so I had confidence in its security.

Based on the above information I made the determination that the hacker simply used the compromised account to gain access and could not elevate their privileges and then gave up. One year later it appears that it was the right decision to leave this highly critical system running.

Upon review of other systems on the network we found that the SGI systems were problematic as no one had any real experience with this operating system.

Using previously gained knowledge of the previously compromised systems assisted in our review. It was important for us to proceed with administrators without blame and purely providing our assistance in resolving the matter.

We scheduled a complete review of all Sun and SGI systems with two Unix experts. The review found unexplained log ins on the SGI systems that predated all other dates/times found so far. It was also discovered that the SGI systems were being managed improperly. After meeting with the System Administrators and System owners of these systems we discovered that there was a misunderstanding between the two groups and that no one was really managing the systems. One system was several OS releases behind. Several people had root access; several other accounts were for people were no longer working for the organization and the trust relationships between them and other systems were unnecessary. On the SGI systems after scanning the systems using the Internet Scanner product from Internet Security Systems we found there was a vulnerability in the rpc program Tool Talk on the SGI programs, see CERT announcement at

http://www.cert.org/advisories/CA-1998-11.html

Upon running the exploit from the scanner tool against the rpc two of the systems shutdown, the third left us staring at a root shell on the system. The system reported no evidence of any login or any other indication that this had happened. The date and time was critical in our search to establish a beginning entrance point into the network. At first we thought that the Red Hat systems were the entrance point into the network. Our conclusiong is that the Tool Talk exploit was most likely the entrance point into the network. From there the two accounts were used to gain access to other systems.

4 Containment

The on-site team was a part-time person and myself. I had the complete cooperation from the team leader that administrated the systems. We knew each other and had a good working relationship.

All users were notified of what we discovered. All users and Systems Administrators were told to change their passwords. We were concerned that due to the fact that the network was accessible to someone that either userids and passwords were compromised on the wire or that the password information could have been pulled and discovered using a password cracker utility. The two compromised accounts were locked immediately and the users notified that they were having their passwords changed. They were told what the new password was and asked to change them at their earliest convenience.

The first Red Hat Linux machine was removed from the network permanently as it was to be retired soon anyway. The second Red Hat Linux System was removed from the network and had its entire system backed up using a product called Arkeia; see http://www.arkeia.com for information, on the second system. This type of backup is a file-based backup as opposed to an image-based backup. The tape was an AIT-2 format. The tape device is housed in a 6 tape library from Qualstar model 46120. The Sun systems we further tightened trust relationships between them and other systems. All SGI systems were removed from the network as these were a very real threat to the rest of the network. Many users access this system and there are trust relationships between these systems and other Unix systems. For instance, file systems are mounted from two Sun systems to these systems.

5 Eradication

Preparation really failed us; at the time no files were being signatured using MD5 hashes. We kicked the idea around of signaturing file systems against known good systems or even binaries from the vendor sites. This was quickly dismissed. As there would inevitably be many files that would not be the same, as these systems were not at the same patch levels and different applications were loaded on them.

Our defense strategy focused on the host.

The Red Hat Systems that were brought back up had their drives reformatted and the most recent Red Hat version installed on them. Also, we used a newly developed checklist to check against for improving security on the system, for instance, a list of only those services that should be enabled in the inetd.conf file.

We painstakingly went through the Suns and SGI systems to ensure that both systems were in their own NIS domain and that trust relationships would not allow remotely executing programs or shells between these two platforms. This was a good approach as the SGI machines belonged to users who are involved in a specific project of limited use to others outside the group using the SGI systems.

On the SGI systems, one system was retired, as its function was not really needed. Remaining systems had their disks erased and the operating systems brought up to the current level. Secure shell was compiled for the platform and instructions developed for the server and client by the Unix TAC team.

After the compromised system was authorized for use again we conducted a vulnerability scan on the systems to make us and the owners have more trust in the security of a system. Systems Administrators also consulted with the security team to make any other recommendations to ensure that the new system was now as secure as possible. The SGI site provides a good guidance on securing their systems and this guidance was used to tighten security.

6 Recovery

The identification phase was crucial to this phase, as what we suspected as having happened would dictate what was necessary to restore a system to a pristine state. For instance, a first strike date/time would dictate what would be the last clean backup that could be used to restore files.

All of the Red Hat systems were completely reloaded with data being restored from the date/time before any known incident using the first strike date found on the SGI systems.

The Sun systems had their operating systems rebuilt. The data was left intact except for the two accounts that were compromised. The data from these areas were removed and data restored from a date before the first strike date on the Sun systems. The SGI systems had all their data removed and then restored from a date before the first strike date.

The systems before releasing to the system owners were checked to see if key applications were functioning as expected. Once this was checked the owners were notified and then the systems put back into production.

The systems were monitored for some time using vulnerability scanners and intrusion detection software for any irregular vulnerabilities or activity.

7 Follow-Up

A formal incident report form was filled out that included an estimate of the amount of time spent cleaning up the incident after a consensus was reached by all involved parties.

We never had a formal lessons learned meeting face to face. There were several meetings to discuss where improvements could be made to improve prevention and response. There were also a lot of email that was sent between members of the security response team and management. As a result, the following improvements have been made:

1. File systems on Sun systems are now being signatured using MD5 hashes
2. The Linux operating systems is now officially supported by a TAC group. This has resulted in very good guidance for configuring a system securely.
3. The security team now has the authority to remove a system from the network without prior approval from system owners.
4. There is now a fully funded program to deploy firewalls at all perimeters within the organization. There are about 165 such perimeters. After some discussion we have settled on placing an enhanced IOS called the Cisco Secure Integrated Software (CSIS) from Cisco on each of our routers. See the URL http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t5/iosfw2/index.htm. The software provides for very good protection including packet filtering, application filtering and Intrusion Detection (about 59 signatures). The software is also very low cost. For most of the sites this works well, however, there a couple of sites that will need a stand-alone solution.
5. Secure shell is now being widely used. Openssh which can be found at http://www.openssh.com/ has been adopted as the choice for the server and client on the Unix systems. Tera Term which can be found at http://hp.vector.co.jp/authors/VA002416/teraterm.html is being used with the ttsh extension which can be found at http://www.zip.com.au/~roca/ttssh.html. The ttsh extension is necessary to provide the encryption for telnet for the Windows platform. This is a freeware client. We also use a freeware ftp client, for use on the Windows platform, that can be found at http://www.i-tree.org/. All applications supports version 1.5 of the ssh protocol. This is certainly better than using ftp and telnet.
6. Procedures for handling an incident are now in draft. The target audience is the security handler.

7.1 Resources and/or Activities Needed To Improve Security

Quarterly Vulnerability Assessments – In some cases tcp wrappers was not properly installed, an assessment may have pointed out a problem before the compromise.

Logical Isolation – Public services are not isolated from our internal systems, having this in place at the time of the incidents may have kept people from jumping from system to system unencumbered.

Clear text passwords – Users are sending clear text passwords on the LAN and also across the wire of an ISP. When someone gets access to the network where a clear text userid/password crosses it is vulnerable to discovery.

Manpower – When responding to this problem, there was not adequate manpower to be dispatched to assist in the original look at the problem. One person was available (very helpful); he was pulled off other responsibilities to assist. Having the available manpower can help in solving the following problems and answering important questions:

1. Discovering the problem.
2. Containing the problem.
3. Determining how they got in. Not an easy question to answer as this requires an expertise and available time to spend. This is paramount in importance, as we want to make adjustments to ensure that this does not happen again.
4. Analyzing the event(s) and documenting it sufficiently.
5. Were there any contributing factors to the break-in? Was security policy being followed?
6. What is required to re-establish systems to a pristine state? That is, we want to ensure that no system files were modified so that the hacker can come back or provide information about us for future use or perhaps leave system functions in an unknown state. The current approach is to rebuild a system even if a single telnet session is seen that demonstrates that a userid is compromised.

System Logs – In one case, the system logs indicated that the system was being attacked and this fact went unnoticed. There are several issues with logs:

1. Are we logging everything we need to be logging in order do a proper analysis? Logs can be used to detect an attack, a compromise and assist in determining what the attacker did.
2. If the system is compromised can we be sure that the logs are complete? The attacker may have erased or modified the logs on the system compromised. We should be centrally logging to a very secure system to improve our confidence that the logs will remain in tact.

Works Cited

Garfinkel, Simson, and Gene Spafford. Practical UNIX and Internet Security 2nd Edition United States: O'Reilly & Associates, 1991.35-37.

Douba, Salim. Networking Unix United States: Sams Publishing, 1995. 82-83,84

References

Held, Gilbert and Kent Hundley. Cisco Security Architectures, CCNA. 1999
Maximum Linux Security. 2000
Waite, Mitchell and Stephen Prata. New C Primer Plus Second Edition. 1993
Wall, Larry and Randal L. Schwartz. Programming Perl. 1991.