SANS Institute
Information Security Reading Room

Incident Handling for SMEs (Small to Medium Enterprises)

Terry Morreale

Incident Handling for SMEs (Small to Medium Enterprises)

GCIH Gold Certification

Author: Terry Morreale, terry@atrust.com

Adviser: Dominicus Adriyanto

Accepted: May 12, 2008

Table of Contents

1. Introduction

Incident handling is more than managing a breach instigated by an outside intruder. It is the ability to manage a variety of incidents that range from a minor virus infestation to a major loss of data and productivity initiated by a malicious user inside or outside the organization. Many organizations consider themselves safe from incidents because they process data that is not particularly useful to outside parties, they are so small that they can't imagine someone in the outside world finding them much less attacking them, or they simply do not believe they have enough resources to worry about an incident unless they find one. All of these assumptions are fallacies. One particularly harsh reality is that according to the 2007 FBI Computer Crime Survey, insider abuse of network access or email was the most prevalent security problem at 52% to 59% of total incidents. Additionally, the average annual loss reported due to security incidents has skyrocketed to $350,424. These are staggering numbers, especially given the amount of resources that organizations typically spend on externally facing vs. internally facing security. Thus, it is critical that organizations take a holistic approach to security that allows for all types of threats, not just the ones that get the most media attention. Sooner or later, all organizations have an incident – and the organizations that are well prepared are the ones that will come out of the

incident with the least amount of damage.

The remainder of this paper describes the six step process heralded by SANS as the recommended way to deal with an incident when it does occur. However, this paper primarily focuses on small to medium enterprises. Such organizations have limited IT resources and frequently have little to no ability to dedicate any of those precious resources to planning for or handling an incident. This paper will analyze each of the recommended steps and make modified suggestions as to how to handle an incident. Additionally, modified tools can be found at the end of this paper tailored to the needs of a small to medium enterprise.

2. Overview of the Six Steps (Skoudis, 2006)

2.1 Preparation

The first step in handling any incident is to be well prepared. When an incident does occur, it is frequently mired in panic and fear – so it is best to know exactly what to do before it happens. While standing in front of a system that is being attacked is not a good time to be making decisions about what the best course of action is. Traditionally, this step includes many policy elements as well as monthly reports, incident team selection, emergency action plans, communication plans and packages

of software that can all be used in the event of an incident. Later in this paper we will examine these in more detail, highlighting those most useful for a small to medium enterprise.

2.2 Identification

The second step in incident management is the identification phase. In this phase, the organization gathers data, analyzes it, and then determines whether an incident has occurred. The incident handler must calmly assess the situation, be ready to communicate, and be ready to handle all evidence such that it can later be used in a court of law if necessary.

2.3 Containment

The goal of the containment phase is to prevent any further damage. If an incident has occurred, it is likely that some amount of damage has already been incurred. In this phase, the damage is contained such that it cannot spread to other data, systems or networks. The initial activities that will occur in this phase include those such as disconnecting the network or power cables, modifying firewall rules or changing DNS information. Once the spread has been stopped, then it is time to make a backup copy of the system that can be used for analysis. It is also a good idea to

make a second copy of the system that will be stored or used for forensic analysis. If the compromised system must remain in production, then additional containment efforts must occur. These are intended to temporarily keep the system up and running while another system is being built, or while decisions are being made as to the long-term plan for the compromised system.

2.4 Eradication

The eradication phase is quite difficult, as it requires the complete removal of any malicious code and data left by the intruder, but it also requires the complete closure of any holes that were used by the hacker to intrude in the first place. This cannot be done until the cause of the incident has been determined. Once the cause has been determined, the system can be rebuilt from a known good backup copy of the system. If no backup can be found, then the system must be reinstalled from scratch (including the OS!).

2.5 Recovery

In the recovery phase, operations return to normal. The system has either been rebuilt from scratch or rebuilt from a backup, and it is ready to be validated for production. This includes verifying the system is secure and will not fall prey to the

same or similar attacks once it has been put online.

2.6 Lessons Learned

The final stage of incident handling is to learn from our previous mistakes. No organization is perfect – therefore it is critical to take the time required to evaluate the incident after it is over. What caused it? Have we configured this and other systems such that this will not happen again? Traditionally, a report is created and a meeting occurs where the information is reviewed. Later, we will look at this step from the perspective of a small to medium enterprise and make recommendations as to the best course of action for such an organization.

It is also important to note the laws surrounding disclosure of computer security breaches. There is an excellent tool located at http://www.guardianedge.com/resources/breach-disclosure.php that allows you to select a location and find applicable laws. There are additionally federal statutes such as the Gramm-Leach-Bliley Act (GLBA) that target specific industries. It is well worth the time to examine your local and industry requirements ahead of time, so you are prepared and have an action plan when an incident occurs.

3. Analysis of the Six Steps for Small to Medium Enterprises

3.1 Preparation

As we discussed previously, preparation is key to successfully handling any incident. Small to medium enterprises will have a slightly different preparation schedule than larger enterprises with more resources. In this section we describe the essentials of preparation.

3.1.1 Know who will handle the incident

The first step in being prepared is knowing who will handle the incident. Some small to medium enterprises will choose to handle incidents in house, some will choose a trusted third party as their incident team and some will choose some combination of internal and external resources. It is critical that this split be defined before an incident occurs to make sure it is handled in the best way possible.

Once this decision has been made, then incident handling instructions should be created. These instructions will serve as an authoritative source for the incident handlers and will include comprehensive, step-by-step instructions.

3.1.2 Develop Incident Handling Instructions (Applied Trust Engineering, 2005)

Incident handling instructions are documents used by all individuals involved in

handling the incident and should leave as few decisions as possible that need to be made while in the midst of an incident. The pieces listed below are complete incident handling instructions, and include portions of each of the six steps. A sample incident handling instruction shell is included in the appendix of this document. Also note that sections 3.2 – 3.6 will feed the incident handling instructions document. Elements include the following:

3.1.2.1 Call list

The call list is who to call when. This should be a list of people that need to be involved in the incident. If you are handling the incident with internal resources, it should include your primary and secondary technical people who have been trained in incident management. It should include management as well. Small and medium business managers should be educated on IT security and understand the process that their organization will be using in the event of a breach. Additionally, it should include local law enforcement contacts in the event your incident has implications that affect the community at large. If you are including a trusted third party to handle all or part of your incidents, then contact information should be included for them as well. Finally, if you are sometimes using internal resources and sometimes using a trusted partner, it should be clear whom to contact and when. Perhaps your organization has

determined that virus incidents will be handled internally, and intrusions will be handled externally; then that designation should be clear in your incident handling instructions.

3.1.2.2 Initial response

Initial response is what to do the moment you suspect an incident. This should include questions that help the incident handler determine if there really is an incident. It should also include instructions for each type of system within the organization. For example, the organization may want to disconnect its financial database from the network if there is an incident, but it might want to leave its web server connected. Finally, initial response should always be crafted with forensics in mind. Any action that is being taken by the incident handler should be taken in a way that all chain-of-custody information is preserved. As mentioned above, part of the initial response is communicating with the right people. Be sure to have very clear designations about whom to contact and when.

3.1.2.3 Response strategy

Now that it is confirmed an incident has occurred, and the appropriate people have been notified, what do you do? It is going to depend on how your organization has chosen to handle incidents. If you have a trusted third party handling all of your

incidents, your response strategy is simply to hand over the reins to the partner. If you are handling some or all incidents, then the response strategy will need to be defined and tested. This section comprises several of the six steps, and will thus be defined in more detail later in the document.

3.1.2.4 Recovery

This section includes information on how to get back to a fully operational state and includes steps 5 and 6 from the six overall steps. It includes operational specifics for each key system within the organization. In this section, even if you use a trusted third party for incident management, you will need to coordinate that party with your internal technical resources. This section is the bridge between the emergency state of the incident and returning to a steady state after the incident has been handled.

3.1.2.5 Lessons Learned report

Include a brief report that describes the incident and what actions are being taken moving forward to prevent a repeat of the incident.

3.1.3 Tools for the Preparation phase

In addition to being prepared for the incident by knowing who will handle it and what steps that person will follow, the organization also needs to have put systems in to place that will provide the incident handler with the data he needs. This section will give concrete examples of tools that can be used.

3.1.3.1 Event logs

One of the most important sources of data when an incident occurs is the event log. To make logs easily usable, and to protect them from modification by an intruder, it is a good idea to send all the logs to one central log host. There are several low-cost or free packages available that make this easy to do – and there are several pricey solutions as well. One good solution for diverse environments is Snare (http://www.intersectalliance.com). Snare provides agents for many operating systems including Windows, several Unix options, as well as less frequently used systems such as Tru64. To use Snare, install an agent on each host that will generate logs. Then, choose a server as your centralized log host to receive and store all the logs. The agents are open source, so they are free of charge. The Snare server is not free however, so it can either be purchased, or a different server solution can be used. If your organization does not want to purchase the Snare server, one potential solution is to use set up a Linux log host and use syslog-ng to store logs from all the agent

machines. Syslog-ng is easy to set up and will collect logs. For documentation on how to install and configure syslog-ng, see http://www.balabit.com/dl/guides/syslog-ng-v2.0-guide-admin-en.pdf. Syslog-ng does not do any analysis however. Analysis will need to be completed separately. One good solution for log analysis is SEC (Simple Event Correlator). This tool is quite versatile but does require a relatively high amount of configuration. A very good paper by Risto Vaarandi gives some concrete examples (http://en.hakin9.org/attachments/pdf/hakin9_05_2006_10_EN_str28-39.pdf). Kiwi provides a free Windows based syslog server as well. It is quite configurable allowing the system administrator to handle alerts in a variety of ways. Some examples include logging alerts to a database, emailing alerts to appropriate administrators, paging administrators and triggering SNMP alerts. Kiwi additionally offers a free log management package for Windows, MacOS and Linux. Additional information as well as the product downloads can be found at http://www.kiwisyslog.com/.

Once you have selected a solution, you will want to configure the agents to send the events you are most interested in. Examples of interesting events include:

- successful and unsuccessful login attempts
- access to important files or directories

- process control
- changes to user rights
- account administration
- changes to the security policy
- system shutdown or restart

Servers are not the only systems on your network that generate useful logs. Routers are additionally full of interesting information, but again, the logs need to be stored on your centralized log host for quick and reliable access. Send the logs from your routers to your syslog-ng server and you can use SEC to look for interesting events such as unauthorized traffic (you see ftp traffic generated by one of your hosts even though you do not have ftp installed).

3.1.3.2 Network-based Intrusion Detection Systems (NIDS)

NIDSs are quite useful in preventing successful breaches of your organization's security systems, but they are also useful in investigating incidents. As with log centralization and analysis, there are several commercial products, but there are also

some good free or low-cost solutions as well. One of the most widely used network-based NIDSs is Snort (http://www.snort.org). The Snort software itself is actually an agent that listens on a host and generates alerts if a packet matches a particular signature. This alert is then stored for review by the system administrator. This alone is not typically enough. The alert needs to be accessible, and one of the common ways to do this is to install the BASE database system in conjunction with Snort. This database sorts alerts and allows administrators to view them and remove them or archive them as appropriate. Snort alone plus BASE is ok for very small installations, but for installations that require more than one sensor, or for administrators who would like a graphical user interface, a management console should be installed. Traditionally, SnortCenter has been used for this purpose, but this software package has not been updated in quite some time, so it would be wise to consider some alternatives. One good solution is the Activeworx Security Center (http://www.activeworx.com). The Security Center is a comprehensive solution that allows for simplified sensor and event management. It also conveniently offers a log management solution appropriate for meeting the needs described in the previous section. Another possible solution is that offered by the creators of Nessus. Tenable Network Security has a comprehensive security monitoring solution that, much like Activeworx, will incorporate sensor management with log management for a single,

easy to use solution. This solution is quite pricey, however.

For a completely free solution on a small network with only one Snort sensor and one syslog host, you can try Aanval (http://www.aanval.com). There is additionally a basic version that will support up to 3 sensors at a reasonable price.

Proper configuration of the sensors is important in making them useful to the organization. There is quite a bit of useful documentation on the snort website. You can access the documents at http://www.snort.org/docs/#docs. Poorly configured sensors will not provide value. The first step in ensuring a good solution is to have sensors in the critical locations of your network. If your organization is small enough to only have one segment, then one sensor may be enough. However, if your organization has multiple segments and a DMZ, you will need to install more than one sensor. It is a good idea to have a sensor on your DMZ that only watches DMZ traffic. This is because your DMZ sensor will pick up a lot of alerts! Your internal sensors will pick up fewer alerts, but these alerts are also the most alarming. If your DMZ sensor sees an attack attempt, you might not be surprised as it is accessible to the Internet. If, however, one of your internal sensors sees an attack from the outside world, you have reason to be concerned. Your internal hosts should be protected from direct Internet access in most cases (minus web browsing of course).

3.1.3.3 Host-based Intrusion Detection System (HIDS)

Another tool useful in analyzing an incident is a host-based intrusion detection system. This tool will verify your critical files and let you know if a file such as the ps executable on Unix or the lsass.exe file on Windows has been modified. Tripwire is one of the most commonly used commercial solutions, but it is very expensive. For a more cost effective solution, try Samhain (http://www.la-samhna.de).

As with the other tools described in this section, to use Samhain, you will install agents on your servers, and you will install a single server as your management console. Several pieces of software need to be installed to get Samhain up and running, but once it is installed, it is easy to configure and use. The Samhain client will be installed on each of the hosts that need to be monitored. It can be run on both Unix and Windows hosts. The Yule log server will collect the data and log reports from the clients and it will keep track of the configuration files for each agent. Reports are stored in a database, Oracle, MySQL, and PostgreSQL are supported. Finally, the Beltane web-based console will need to be installed on the server. It allows for simple management of the clients and reports. Samhain can be configured to check lots of

interesting things on your hosts. Some of the commonly watched items are:

- kernel integrity
- open ports
- hidden processes
- file integrity

The Samhain website offers detailed documentation. See http://www.la-samhna.de/samhain/s_documentation.html for more information.

3.1.3.4 Time synchronization

It is important to synchronize device time as devices that disagree about what time an event happened generate meaningless information. The Network Time Protocol (NTP) can be used to synchronize device clocks. Most systems ship with NTP already installed, but if it is not installed, it can be downloaded from the ntp.org project (http://www.ntp.org/downloads.html). A good architecture recommendation is to select one or two time servers that will synchronize with Internet based timeservers. These timeservers are referred to as Stratum 1 hosts, those servers directly connected to an

atomic clock. Then, the rest of the devices on the network should synchronize with the one or two local timeservers. This page provides some excellent detail on NTP architecture: http://en.wikipedia.org/wiki/Network_Time_Protocol.

3.2 Identification

In any organization, incident identification is the responsibility of more than just the IT staff. This is even more true in a small to medium enterprise. In smaller organizations the IT staff certainly will be likely to notice certain types of activity, but other employees might notice symptoms of an attack such as slow network or system performance. It is important to take these complaints seriously, as they can be indicators of an incident.

Incidents sometimes occur very quickly, so it is important that an incident handler be willing to occasionally sound a false alarm. An organization is better off reacting quickly to a suspected incident only to find out it was nothing, rather than not doing anything until severe damage has occurred. As soon as an incident is suspected, the handler should consult the incident handling instructions that were created in the preparation stage and start filling in information. It is critical that an incident handler write down as much as possible. If the handler is going too fast to

write down what they are doing, they are going too fast to handle the incident correctly. Notes are key to understanding an incident and being able to prevent similar incidents from occurring in the future.

SANS has created some very good incident identification pocket reference guides. These are available at the following URLs:

- www.sans.org/resources/winsacheatsheet.pdf
- www.sans.org/resources/linsacheatsheet.pdf

It is a good idea to print copies of these sheets and include them with the incident handling instructions. They make very specific recommendations and are organized as follows (SANS):

- Unusual processes and services
    - Unix
        - Look for unusual processes with ps –aux
        - Investigate unusual processes further with lsof –p [pid]

- Windows
  - Look for unusual processes with Task Manager (taskmgr.exe)
  - Look for unusual network services (net start)

- Unusual files and registry keys
  - Unix
    - Look for unusual SUID root files with find / -uid 0 –perm -4000 –print
    - Look for unusual large files with find / -size +10000k -print
    - Look for files named with dots and spaces:
      - Find / -name ``…`` -print
      - Find / -name ``.. `` -print
      - Find / -name ``. `` -print
      - Find / -name `` `` - print

- Windows
  - Check for major decreases in free disk space with dir c:\
  - Look for unusual large files by using the Search tool and selecting the size option to be at least 10000KB

- Unusual network usage
  - Unix
    - Look for promiscuous mode with ip link | grep PROMISC
    - Look for unusual port listeners with lsof –i and netstat –nap
    - Look for unusual ARP entries with arp –a
  - Windows
    - Look at file shares with net view 127.0.0.1
    - Look at who has open sessions with net session
    - Look at what sessions the machine has opened with net use

        - Look at NetBIOS activity with nbstat –S
        - Look for unusual listening ports with netstat –na
- Unusual scheduled tasks
    - Unix
        - Look for cron jobs with crontab –u root –l
        - cat /etc/crontab
        - ls /etc/cron.*
    - Windows
        - Look for scheduled tasks with at
        - Also check the scheduled tasks in the system tools
- Unusual accounts
    - Unix
        - Look in /etc/passwd for new accounts, especially those with

UID or GID of 0 with less /etc/passwd and grep :0: /etc/passwd

- Windows
  - Look for accounts in the administrators group with lusrmgr.msc

- Unusual log entries
  - Unix
    - Look for suspicious events such as “entered promiscuous mode”, a large number of authentication failures, RPC programs with a large number of strange characters, a large number of web errors
  - Windows
    - Run the event viewer with eventvwr.msc and look for suspicious events such as “event log service was stopped”, “windows file protection is not active”, “the telnet service has started”, or large numbers of failed login attempts

Also, use the tools that were discussed in the Preparation portion of this

document. If you have installed Snare, Snort, and Samhain, you should have a wealth of information at your disposal. You will easily be able to see in Samhain if critical files have changed. With Snort, you can see if malicious packets have traversed your network, with Snare, you can check for unusual events on your systems quickly and efficiently.

None of these items alone is an indicator of an incident. Rather, these pieces of information should be considered in total when deciding whether an incident has occurred.

Incident handling instructions include forms for handlers that help them keep track of information gathered during this phase. As the sample incident handling instructions direct, questions should be answered by the incident handler to help them determine the scope and severity of the incident. These questions will help the incident handler understand how to respond and what strategies should be followed moving forward after identification.

3.3 Containment

The containment phase is where an incident handler begins to make changes to the system or network. These changes are made with the goal of preventing the

incident from getting any worse. According to the six-step process, containment has three phases: short-term containment, system backup and long-term containment. These are also applicable to a small or medium business.

3.3.1 Short Term Containment

Short-term containment is the steps the incident handler takes to simply stop the attacker (person, virus, malicious software, other) from making more progress. The following steps will likely alert the attacker that the organization has discovered the breach. Thus, before taking any action, the organization needs to decide if it wants to try to analyze the situation without alerting the hacker and potentially "catch" him in the act, or if it wants to stop the incident immediately, and begin the eradication phases as soon as possible. If the organization is not worried about alerting the hacker to its investigation, some potential courses of action include (Skoudis, 2006):

- Disconnecting the network cable.
- Pulling the power cable (destroys volatile memory and may damage the hard drive).
- Isolating the system through switch port configurations or other

network management tools such as firewall filters.

- Changing DNS so the name points to another IP address (if the hacker is using a name).

As much as possible, all actions taken should be according to the pre-approved incident handling instructions. Having a developed plan allows incident handlers to immediately respond to urgent situations confidently and reliably. The incident handler needs to use the incident handling instructions to track incident information, then communicate the possible courses of action to management with recommendations. Any action the incident handler takes will impact the rest of the business, so it is very important that approval is received.

If it has been determined that the organization wants to try to investigate the situation without alerting the hacker, great care needs to be taken. Standard tools for analysis can alert a hacker quickly. Using ping, traceroute, nslookup, and other tools will notify the hacker if he is paying attention to what processes and commands are being run on the system. The best way to analyze a system is to make a backup copy of the system and analyze that machine.

If the breach is doing immediate severe damage, the business would be wise to

contain the system, even if it means notifying the hacker. However, many breaches are not so damaging in nature and can be monitored closely while analysis is occurring.

In all cases, one or more backups of the system should be made. The best backups for this situation are bit-by-bit copies of the hard drive. This can be done using a variety of tools such as dd or Ghost. DD is a powerful tool that must be used with care. It is quite easy to accidentally wipe out an entire drive by simply reversing the parameters for the source and target drives. There is a good tutorial here: http://www.linuxquestions.org/questions/linux-newbie-8/learn-the-dd-command-362506/. Norton provides excellent documentation for the Ghost product. Version 12 documentation can be found here: ftp://ftp.symantec.com/public/english_us_canada/products/ghost/12/manuals/ngh_12_user_guide.pdf. There are also manuals for versions 10 and 14 available online.

In addition to making a backup of the system, if the organization is planning to do forensic analysis, a forensic copy should be made. There are several tools available for this, but the one most widely used in a court of law is EnCase. EnCase is a very pricey software application, so most small or medium businesses will not want to pay for it. This is a good role for a trusted third party. A third party that has paid for

the software can be contracted to make the copy and even analyze it if the business decides to go that route.

Once backup copies and forensic copies have been made, it is time to do some analysis. At this point, an internal handler or a trusted third party can begin looking at logs and other data to make a recommendation as to the best long-term containment strategy. If the business decides it is ok to take the system off-line, then the handler can spend some time making sure the system is clean and patched for re-instatement. If the business decides that the system cannot be taken off-line, then the handler must find a way to clean the system as much as possible. This can be difficult and risky and should only be done as a last resort, as it is nearly impossible to say with 100% certainty that the hacker does not still have a backdoor or hole built into the system.

A few rules of thumb for neutralizing various types of incidents (Applied Trust Engineering, 2008):

Incident Type	Response
Theft of information	Review logs and systems to identify what information was taken and how

Incident Type	Response
Denial of Service	Reconfigure the operating system TCP/IP settings
	Reconfigure the firewall
	Reconfigure upstream routers to block or divert the attack
	Confirm the most recent data backup performed prior to the DoS attack
Malicious software/viruses	Isolate infected machines from the network or disable email or web services
	Mitigate by installing patches from vendor, reconfiguring email services, or modifying firewall configurations
	Forcibly update virus protection definitions

Incident Type	Response
Inappropriate Use	Collect all relevant information such as log files, email messages, network traffic etc.
	Create a forensic copy of the machine in question
	Burn evidence files to a CD for preservation
Physical Intrusion	Contact the police department if necessary

Incident Type	Response
	Investigate areas in question – - Verify machines require a root password - Verify the machine has a screensaver password - Check machines for signs of physical tampering - Check machine logs for unauthorized use

3.4 Eradication

Eradication is the complete removal of all artifacts left by the attacker, whether

that be a person or a program. This is a hard problem! The first step is to determine the cause of the incident. This is critical because no matter how much energy is put into cleaning a system, if the cause is not found, and thus the vulnerability that was exploited is not removed, the system will only get compromised again.

Forensic tools such as EnCase can help a great deal in this step. These tools can help the handler find malicious software, can help with log file analysis and can help the handler find hidden files the attacker might be using. While EnCase is used frequently within the law enforcement community, it is also quite expensive. A free option is the Autopsy Forensic Browser, a graphical interface for the commonly used Sleuth Kit (TSK). TSK, and therefore the Autopsy Forensic Browser, can be used to analyze disk images providing many of the same functions as EnCase. For example, an incident handler can find deleted files, display file system and meta-data, create timelines of activity, look up file hashes and organize files based on type (www.sleuthkit.org, Brian Carrier, 2003-2008). Good documentation as well as download links can be found at http://www.slethukit.org/proj.php).

There are two primary choices in the eradication phase. They are to restore the system from backup, or to rebuild the system from scratch. If the organization has a known good backup, and it is sure the backup was made prior to the breach, then the

system can be rebuilt using it. This is risky! Frequently attackers have breached the system long before they have been discovered. Statistics show that most incidents are not revealed for several months. An attacker may plant a backdoor and then not use it for many months. If this is the case, can the organization be sure that the backup occurred before the backdoor was planted? Make sure to look for malicious software on your backup copy. Check for rootkits, viruses and backdoors. Viruses can be dealt with relatively easily, but if you find a rootkit, the drive really should be formatted and the operating system re-installed from scratch.

Regardless of whether the system was rebuilt from a backup or rebuilt from scratch, the incident handler needs to improve the defenses of the system so it will not be re-compromised. Frequently this means installing patches, both operating system and application level. It can also mean changing firewalls, locking down services and changing accounts.

Before the system is put back online, it should be validated for security. The system could be scanned with a tool such as Nessus or MBSA to check for remaining vulnerabilities. Once the system has been validated, it should be backed up. Now you have a known good backup of the system. If it is compromised again, you know you have a clean starting point.

See the incident handling instructions at the end of this document for some recommendations on eradication steps.

3.5 Recovery

The system has already been validated for security reasons; it should also be validated for business function. It should be checked to verify all necessary functions of the system are working before being put back into production. Once both security and business functions have been validated, the system can be put back online.

The final step in recovery is to ensure that the system is being monitored such that future problems are detected in a timely fashion. Even in a small business, systems can be monitored easily. An administrator can review log files for individual systems, or a log collector can be used. One solution in a Windows environment is to use Snare to gather and analyze logs. In a Unix environment, syslog-ng can be used to consolidate logs, and the Simple Event Correlator can be used to assist with analysis. See the section above on Preparation, and install appropriate tools.

3.6 Lessons Learned

SANS best practices include developing an incident report as a part of the

lessons learned process. A detailed report may not be feasible in a small to medium enterprise, but even in a small company a short report is wise. In this report, the handler should clearly describe the following (Applied Trust Engineering, 2007):

1. What system(s) was compromised?
2. To the best of their knowledge, how did the compromise occur?
3. What steps were taken to contain the incident?
4. What steps were taken to clean the system?
5. What steps were taken to ensure the compromise is not repeated?

4. Conclusion

In conclusion, the six-step process can be modified such that it is effective for small to medium businesses. An overview of the steps and how a small to medium business can use them follow.

4.1 Preparation

Be prepared. Create incident handling instructions that can be used in the

incident. This is a critical step, especially for small businesses that do not have dedicated resources for incident management. Good incident handling instructions will help the incident get handled quickly and effectively.

4.2 Identification

Use the forms in the incident handling instructions to identify the incident and communicate it to the appropriate individuals both inside and outside the organization.

4.3 Containment

Based on the data in the identification phase, contain the incident appropriately. This can mean removing the system from the network. Make a backup copy of the system that can be used for analysis, and potentially make a forensic copy as well. Finally, decide whether the system will be cleaned or rebuilt. From a security perspective, it is always better to rebuild, but this might not be best for the business.

4.4 Eradication

Clean the system or rebuild it from scratch. Then, validate that the system is secure with a tool such as Nessus or MBSA. Also, make a backup of the cleaned system and store it in case another breach occurs.

4.5 Recovery

Put the newly cleaned or rebuilt system back into production and make sure it is being monitored to watch for future problems.

4.6 Lessons Learned

Finally, write a brief report that describes the situation and what has been done to prevent future breaches.

5. Appendix – Sample Incident Handing Instructions

5.1 Call List

Name	Department	Phone Numbers	Contact Scenario
John Smith	Trusted third party incident handler	(555) 555-1212 (555) 555-2323	Contact immediately upon suspicion of an incident
Betty Smith	CIO	(555) 444-1212 (555) 444-2323	Contact once the incident has been confirmed
Jim Jones	Lead System Administrator	(555) 444-3434 (555) 444-4545	Contact immediately upon suspicion of an incident
Mary Jones	ISP contact	(555) 333-1212 (555) 333-2323	Contact if the incident requires ISP intervention
Tom Johnson	Local police chief	(555) 222-1212	Call if the incident has wider impact on the

		(555) 222-2323	community

5.2 Initial Response

Incident Handler Information:
Name: Contact Information: Date and Time:
Incident Information:
System name: Type of incident suspected: Other systems that may also be affected: Actions that have already been taken:

Description of events that led to this investigation:

System Information:
Describe the risk of this incident (ie: is sensitive data at risk? how much damage is already known to have occurred?)
Analysis Information:
What analysis has been done?

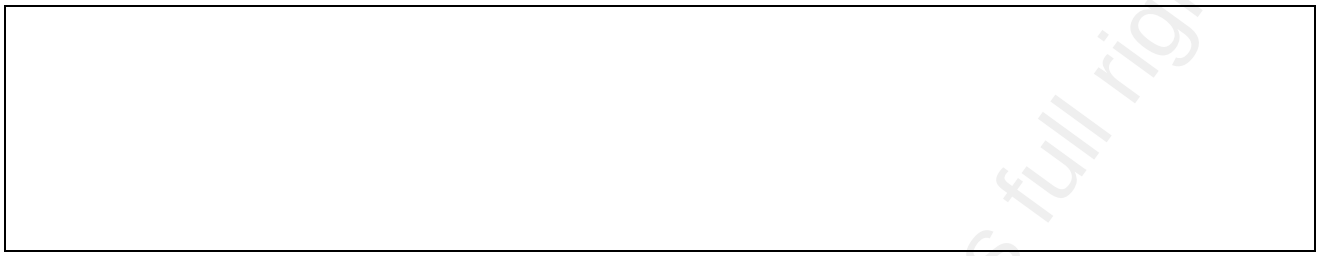
What are the recommended next steps?

5.3 Response Strategy

Communication plan:
Document who has been involved in communications:
Approval:
Recommended next steps: Remove the system from the network (or isolate it)? Analyze system without removing it from the network? Approver name: Approver signature:
System backups:
System has been backed up?

Forensic copy of the system has been created?

Analysis:
What was determined to be the cause of the incident? What data supports this conclusion?



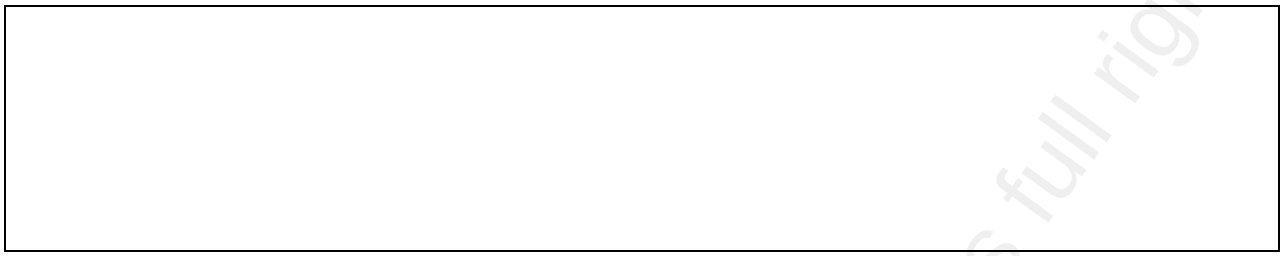
Eradication:
How will the system be recovered? System will be restored from backup? System will be rebuilt from scratch? Approver name: Approver signature: If the system will be restored from backup, what steps will be taken to ensure all artifacts left by the hacker are removed (ie: rootkits, viruses, backdoors)?

Recovery:
Validate system security: Scan with Nessus or MBSA produces a clean report? All current OS patches have been applied? All current application patches have been applied and will continue to be applied? All unnecessary services have been disabled? All system passwords are complex? Antivirus/antimalware software has been installed and will be regularly updated? Validate system function: All required system functions are intact? System is ready to be put back in production? Approver name:

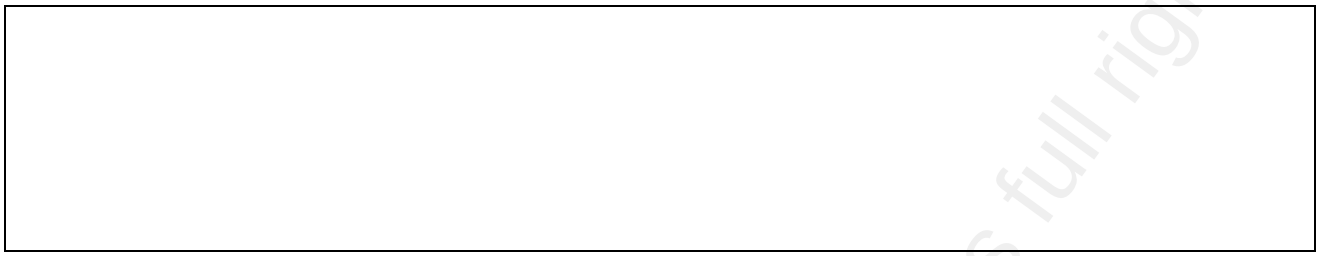
Approver signature:

5.4 Lessons Learned Report

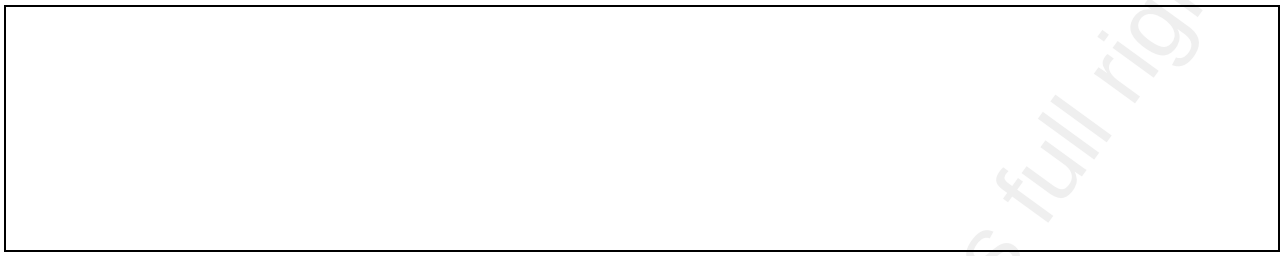
Incident description:
Affected systems:



Incident cause:
Actions taken to eradicate the compromise:



Actions taken to prevent future compromise:
Recommended additional actions:



6. References

SANS & Skoudis, E. (2006). Security 504 – Hacker Techniques, Exploits & Incident Handling

SANS. Intrusion Discovery Cheat Sheets

FBI (2007). 2007 FBI Computer Crime Survey

Applied Trust Engineering (2005). Incident Handling Ledger

Applied Trust Engineering (2005). Incident Handling Guide

Applied Trust Engineering (2006). The Barking Seal

Applied Trust Engineering (2008). Incident Handling Checklist