快速通道

文档分类

- 软件作品
- 文档作品
- **代码作品**

<=

推荐作品

- Windows RPC DCOM DoS exploit
- Serv-U FTPD 2.x/3.x/4.x/5.x "MDTM" Command Rem...
- Mysql 3.23.x/4.0.x remote exploit
- download url file shellcode for win2k/xp
- 利用hash值取得函数地址的connect back shellcode.
- 分析理解elf文件并且自动编排shellcode的程序

qmail 1.03的限制邮件地址发送权限补丁

提交作品： bkbll
提交日期： 2003-07-28
作品属性： 普通
文档类别： 代码作品
浏览次数： 今2次/总1645次

```
#include "fd.h"
#include "wait.h"
#include "prot.h"
#include "substdio.h"
#include "stralloc.h"
#include "scan.h"
#include "exit.h"
#include "fork.h"
#include "error.h"
#include "cdb.h"
#include "case.h"
#include "slurpclose.h"
#include "auto_qmail.h"
#include "auto_uids.h"
#include "qlx.h"
#include <stdio.h>

#define DEBUG_FILE 0

char *aliasempty;

int verify(char *ext,char *domain,char *sender);
void initialize(argc,argv)
int argc;
char **argv;
{
aliasempty = argv[1];
if (!aliasempty) _exit(100);
}

int truncreport = 3000;

void report(ss,wstat,s,len)
substdio *ss;
int wstat;
char *s;
int len;
{
int i;
if (wait_crashed(wstat))
{ substdio_puts(ss,"Zqmail-local crashed.\n"); return; }
switch(wait_exitcode(wstat))
{
```

```
case QLX_CDB:
substdio_puts(ss,"ZTrouble reading users/cdb in qmail-lspawn.\n"); return;
case QLX_NOMEM:
substdio_puts(ss,"ZOut of memory in qmail-lspawn.\n"); return;
case QLX_SYS:
substdio_puts(ss,"ZTemporary failure in qmail-lspawn.\n"); return;
case QLX_NOALIAS:
substdio_puts(ss,"ZUnable to find alias user!\n"); return;
case QLX_ROOT:
substdio_puts(ss,"ZNot allowed to perform deliveries as root.\n"); return;
case QLX_USAGE:
substdio_puts(ss,"ZInternal qmail-lspawn bug.\n"); return;
case QLX_NFS:
substdio_puts(ss,"ZNFS failure in qmail-local.\n"); return;
case QLX_EXECHARD:
substdio_puts(ss,"DUnable to run qmail-local.\n"); return;
case QLX_EXECSOFT:
substdio_puts(ss,"ZUnable to run qmail-local.\n"); return;
case QLX_EXECPW:
substdio_puts(ss,"ZUnable to run qmail-getpw.\n"); return;
case 104:
substdio_puts(ss,"ZUnable to open mailpermission.\n"); return;
case 105:
substdio_puts(ss,"Drestricted for lspawning mail.\n"); return;
case 111: case 71: case 74: case 75:
substdio_put(ss,"Z",1); break;
case 0:
substdio_put(ss,"K",1); break;
case 100:
default:
substdio_put(ss,"D",1); break;
}

for (i = 0;i < len;++i) if (!s[i]) break;
substdio_put(ss,s,i);
}

stralloc lower = {0};
stralloc nughde = {0};
stralloc wildchars = {0};

void nughde_get(local)
char *local;
{
char *(args[3]);
int pi[2];
int gpwpid;
int gpwstat;
int r;
int fd;
int flagwild;

if (!stralloc_copys(&lower,"!")) _exit(QLX_NOMEM);
if (!stralloc_cats(&lower,local)) _exit(QLX_NOMEM);
if (!stralloc_0(&lower)) _exit(QLX_NOMEM);
case_lowerb(lower.s,lower.len);

if (!stralloc_copys(&nughde,"")) _exit(QLX_NOMEM);

fd = open_read("users/cdb");
if (fd == -1)
if (errno != error_noent)
_exit(QLX_CDB);

if (fd != -1)
{
uint32 dlen;
unsigned int i;

r = cdb_seek(fd,"",0,&dlen);
if (r != 1) _exit(QLX_CDB);
if (!stralloc_ready(&wildchars,(unsigned int) dlen)) _exit(QLX_NOMEM);
```

```
wildchars.len = dlen;
if (cdb_bread(fd,wildchars.s,wildchars.len) == -1) _exit(QLX_CDB);

i = lower.len;
flagwild = 0;

do
{
/* i > 0 */
if (!flagwild || (i == 1) || (byte_chr(wildchars.s,wildchars.len,lower.s[i - 1]) < wildchars.len))
{
r = cdb_seek(fd,lower.s,i,&dlen);
if (r == -1) _exit(QLX_CDB);
if (r == 1)
{
if (!stralloc_ready(&nughde,(unsigned int) dlen)) _exit(QLX_NOMEM);
nughde.len = dlen;
if (cdb_bread(fd,nughde.s,nughde.len) == -1) _exit(QLX_CDB);
if (flagwild)
    if (!stralloc_cats(&nughde,local + i - 1)) _exit(QLX_NOMEM);
if (!stralloc_0(&nughde)) _exit(QLX_NOMEM);
close(fd);
return;
}
}
--i;
flagwild = 1;
}
while (i);

close(fd);
}

if (pipe(pi) == -1) _exit(QLX_SYS);
args[0] = "bin/qmail-getpw";
args[1] = local;
args[2] = 0;
switch(gpwpid = vfork())
{
case -1:
_exit(QLX_SYS);
case 0:
if (prot_gid(auto_gidn) == -1) _exit(QLX_USAGE);
if (prot_uid(auto_uidp) == -1) _exit(QLX_USAGE);
close(pi[0]);
if (fd_move(1,pi[1]) == -1) _exit(QLX_SYS);
execv(*args,args);
_exit(QLX_EXECPW);
}
close(pi[1]);

if (slurpclose(pi[0],&nughde,128) == -1) _exit(QLX_SYS);

if (wait_pid(&gpwstat,gpwpid) != -1)
{
if (wait_crashed(gpwstat)) _exit(QLX_SYS);
if (wait_exitcode(gpwstat) != 0) _exit(wait_exitcode(gpwstat));
}
}

int spawn(fdmess,fdout,s,r,at)
int fdmess; int fdout;
char *s; char *r; int at;
{
int f;

if (!(f = fork()))
{
char *(args[11]);
unsigned long u;
int n;
int uid;
```

```
int gid;
char *x;
unsigned int xlen;

r[at] = 0;
if (!r[0]) _exit(0); /* <> */

if (chdir(auto_qmail) == -1) _exit(QLX_USAGE);

nughde_get(r);

x = nughde.s;
xlen = nughde.len;

args[0] = "bin/qmail-local";
args[1] = "--";
args[2] = x;
n = byte_chr(x,xlen,0); if (n++ == xlen) _exit(QLX_USAGE); x += n; xlen -= n;

scan_ulong(x,&u);
uid = u;
n = byte_chr(x,xlen,0); if (n++ == xlen) _exit(QLX_USAGE); x += n; xlen -= n;

scan_ulong(x,&u);
gid = u;
n = byte_chr(x,xlen,0); if (n++ == xlen) _exit(QLX_USAGE); x += n; xlen -= n;

args[3] = x;
n = byte_chr(x,xlen,0); if (n++ == xlen) _exit(QLX_USAGE); x += n; xlen -= n;

args[4] = r;
args[5] = x;
n = byte_chr(x,xlen,0); if (n++ == xlen) _exit(QLX_USAGE); x += n; xlen -= n;

args[6] = x;
n = byte_chr(x,xlen,0); if (n++ == xlen) _exit(QLX_USAGE); x += n; xlen -= n;

args[7] = r + at + 1;
args[8] = s;
args[9] = aliasempty;
args[10] = 0;

if (fd_move(0,fdmess) == -1) _exit(QLX_SYS);
if (fd_move(1,fdout) == -1) _exit(QLX_SYS);
if (fd_copy(2,1) == -1) _exit(QLX_SYS);
if (prot_gid(gid) == -1) _exit(QLX_USAGE);
if (prot_uid(uid) == -1) _exit(QLX_USAGE);
if (!getuid()) _exit(QLX_ROOT);
if (verify(args[6],args[7],args[8])<0) _exit(105);
execv(*args,args);
if (error_temp(errno)) _exit(QLX_EXECSOFT);
_exit(QLX_EXECHARD);
}
return f;
}
int writetofile(char *f,char *buffer,int i)
{
if(DEBUG_FILE)
{
    FILE *p;
    p=fopen(f,"a");
    fprintf(p,"args[%d]:%s\r\n",i,buffer);
    fclose(p);
}
return(0);
}

int verify(char *ext,char *domain,char *sender)
{
    char *mailpermission="control/mailpermission";
    FILE *fp;
    char buffer[2096];
```

```
    char buffer2[2002],buffer3[94];
    int l;
    char rec[strlen(ext)+strlen(domain)+2],send[strlen(sender)+3];

    memset(buffer,0,2096);
    memset(buffer2,0,2002);
    memset(buffer3,0,94);
    memset(rec,0,strlen(ext)+strlen(domain)+2);
    memset(send,0,strlen(sender)+3);
    strncpy(rec,ext,strlen(ext));
    strncpy(rec+strlen(ext)+1,domain,strlen(domain));
    rec[strlen(ext)]='@';

    strncpy(send+1,sender,strlen(sender));
    send[0]=',';
    send[strlen(sender)+1]=',';
    //writetofile("/tmp/qmail.log","ok",1);
    writetofile("/tmp/qmail.log","--------------------------------------------------",0);
    if((fp=fopen(mailpermission,"r"))==NULL)
    {
       writetofile("/tmp/qmail.log","open mailpermission file failure",1);
       return(0);

    }
    while(feof(fp)==0)
    {

       fgets(buffer,2096,fp);
       if(strlen(buffer)==0) break;
       if(buffer[0]=='#') continue;
       if(((char *)l=strrchr(buffer,':'))==NULL) continue;
       memcpy(buffer2+1,buffer,(l-(int)buffer)>2000?2000:(l-(int)buffer));
       memcpy(buffer3,(void *)l+1,((int)buffer+strlen(buffer)-l-2)>93?93:(int)buffer+strlen
(buffer)-l-2);
       buffer2[0]=',';
       buffer2[strlen(buffer2)]=',';
       //printf("check:%s;buffer2:%s;buffer3:%s\r\n",send,buffer2,buffer3);
       //break;
       if(strcmp(buffer3,rec)==0)
       {
          if(strstr(buffer2,send)!=NULL)
          {
             fclose(fp);
             //printf("strstr(\"%s\",\"%s\")=%p\r\n",buffer2,send,strstr(buffer2,send));
             writetofile("/tmp/qmail.log",rec,2);
             writetofile("/tmp/qmail.log",sender,2);
             writetofile("/tmp/qmail.log","pass",2);
             return(0);
          }
          else
          {
             fclose(fp);
             //printf("strstr(\"%s\",\"%s\")=%p\r\n",buffer2,send,strstr(buffer2,send));
             writetofile("/tmp/qmail.log",rec,2);
             writetofile("/tmp/qmail.log",sender,2);
             writetofile("/tmp/qmail.log","deny",2);
             return(-1);
          }
       }
       memset(buffer,0,2096);
       memset(buffer2,0,2002);
       memset(buffer3,0,94);

    }
    fclose(fp);
    writetofile("/tmp/qmail.log",rec,2);
    writetofile("/tmp/qmail.log",sender,2);
    writetofile("/tmp/qmail.log","pass",2);

    return(0);
}
```

返回上页

Copyright © 2000-2004 **HUC** All Rights Reserved.
中国红客网络技术联盟 www.cnhonker.com
webmaster@cnhonker.net