
Subject: strcpy issue

Posted by [forlano](#) on Wed, 20 Jul 2016 15:44:16 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello,

the following code involving a silly STRCPY crash my application in release mode (upp 9994, 64 bit, gcc , c++11, lubuntu 16.04)

```
class Team : Moveable<Team> {
public:
...
    char hstatus[50];
...
    Team();
    ~Team(){;}
};

.....

sql.Execute("SELECT * FROM TEAMS order by N");
while(sql.Fetch()) {
    Team& t = VTeam.Add( (int)sql[N] );
    ...
    strcpy (t.hstatus, (String) sql[HSTATUS] ); //<=== crash!
    ...
}
```

instead works in debug mode.

On windows, 32 bit, old upp no c++11, it works too.

hstatus cannot be a String (I need to modify its content char by char when needed).

What is wrong in that innocent line?

Thanks,

Luigi

Subject: Re: strcpy issue

Posted by [forlano](#) on Wed, 20 Jul 2016 20:28:30 GMT

[View Forum Message](#) <> [Reply to Message](#)

forlano wrote on Wed, 20 July 2016 17:44

What is wrong in that innocent line?

Solved!

hstatus was shorter than the other string!
This bug remained silent so far.

Sorry for disturbing for this silly problem.
Luigi

Subject: Re: strcpy issue
Posted by [mirek](#) on Thu, 21 Jul 2016 05:17:07 GMT
[View Forum Message](#) <> [Reply to Message](#)

Why do not you just use String?

Mirek

Subject: Re: strcpy issue
Posted by [forlano](#) on Thu, 21 Jul 2016 10:54:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Thu, 21 July 2016 07:17Why do not you just use String?

Mirek

Because the compiler complain when I try to change a single character inside the string, like

```
String s = "000000000000000000000000";  
s[4] = '1';
```

Is there a way to do it?
Luigi

Subject: Re: strcpy issue
Posted by [dolik.rce](#) on Thu, 21 Jul 2016 11:37:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

forlano wrote on Thu, 21 July 2016 12:54
String s = "000000000000000000000000";
s[4] = '1';

Is there a way to do it?
This should work:
String s = "000000000000000000000000";
s.Set(4, '1');

Honza
