Subject: crypt
Posted by reborn on Sat, 23 Jun 2012 13:52:06 GMT
View Forum Message <> Reply to Message

I am not familiar at all with string encryption and decryption.

I have the following function:

```
void encrypt (char e[] )
{
 for( int i=0; e[i] != '\0'; ++i )
 {
  ++e[i];
 }
}


void decrypt( char * ePtr )
{
 for( ; * ePtr != '\0'; ++ ePtr )
 {
  --(* ePtr);
 }
}
```

It's obviously completely pathetic.

Anyone care to improve this? Something that can boast of industry standard would be excellent.

Subject: Re: crypt
Posted by iRANian on Sat, 23 Jun 2012 14:10:51 GMT
View Forum Message <> Reply to Message

What is it needed for? If you're releasing a binary people can just copy and paste the ASM for the encryption and decryption functions or hook them and have the output for them get logged to a console window.

Subject: Re: crypt
Posted by danpaul88 on Sat, 23 Jun 2012 15:09:02 GMT
View Forum Message <> Reply to Message

Xor encryption is simple but effective. Google it

Subject: Re: crypt
Posted by reborn on Sat, 23 Jun 2012 16:21:42 GMT
View Forum Message <> Reply to Message

iRANian wrote on Sat, 23 June 2012 10:10What is it needed for? If you're releasing a binary people can just copy and paste the ASM for the encryption and decryption functions or hook them and have the output for them get logged to a console window.

It would be in a released binary, yes.
I am storing a user name and password to a settings file, and storing it as plain text isn't acceptable.

danpaul88 wrote on Sat, 23 June 2012 11:09Xor encryption is simple but effective. Google it

Thank you.

Subject: Re: crypt
Posted by Sir Kane on Sun, 24 Jun 2012 21:11:57 GMT
View Forum Message <> Reply to Message

Keep in mind that encrypting text will require you to save the result base64 or something else encoded, because encryption can result in 0 values.

Subject: Re: crypt
Posted by Jerad2142 on Tue, 03 Jul 2012 17:41:13 GMT
View Forum Message <> Reply to Message

If your just doing ascii encryption keep in mind that it's from 0 - 127. So if you exceed 127 you'll have to loop it down to 0. If you drop under 0 you have to loop it back around to 127. Otherwise when you save it you'll be left with nothing useful.

However, as danpaul88 said, Google is an excellent place to check out too, you can go and get any encryption algorithm off of there in open source pretty much (wiki usually has source with examples of the algorithms as well).