Subject: Re: Encrypted storage with streaming (OpenSSL, AES)
Posted by koldo on Mon, 22 Feb 2010 07:50:10 GMT
View Forum Message <> Reply to Message

Mindtraveller wrote on Mon, 22 February 2010 08:31koldo wrote on Mon, 22 February 2010 09:541) Does it mean that AES cannot be used for saving user files with user defined password ?

However there are programs that include this possibility with AES. For example 7zip offers AES-256 encryption http://www.7-zip.org/7z.html.

2) Is there a standard way to convert a 8 chars user defined password into an useful 256 AES bits key ?
1) Cryptography is no miracle, it's just math. If you use weak password, you get weak protection, and no algorithm saves you from it. This means if you want stable and strong protection, you must use stable and strong key. The one of few options here is to use key generated by OpenSSL itself.

You have to consider user password as worst type of key. Also, many passwords are too plain and dumb: 123, 111, 123456, etc. This is bad for cryptography.

Russian programmer Igor Pavlov who wrote 7zip, has chosen to use compromise solution. He takes user password, calculates SHA-256 function for it (AFAIK U++ has its realization too). Then he adds some calculations/changes to that 256-bit value and the final value is used as a key for AES encryption.

This represents fair protection, which is very much stronger than using user password as key, but at some rate weaker protection than with OpenSSL-generated key. In a number of uses it is rather good and satisfactory protection. Also it allows using protection without storing user password itself which is very good practice. But frankly speaking I haven't heard of SHA output as extremely cryptographically strong combination of bytes. This algorithm has another application field (generating unique digest "far" from original bytes).

2) AFAIK there is no "standard" way to convert user password to key. The best way is to use OpenSSL generated key. You may of course use any function like SHA-256 but you must be aware of the crytpographic strongness/weakness you give to user.

Excellent explanation

I will follow your advice. Anyway, could you add a function to convert an username password into a "fair" protection ?. Thanks

I have checked your demo and now it works well. In a big program where I have applied it, it works well too .

You have done more changes than just a fix . You have removed dependencies to packages Web and Web/SSL.

This afternoon I will upload it to Bazaar. In some hours I will propose a possible application of your useful functions.

Great job !