

Forum:[Hack Forums](#)

Thread: [FileNinja - File/Folder Encrypter](#)

Decim - February 20, 2017, 7:48 am

FileNinja

Wrote this for personal use, wanted to encrypt some photos/text documents.
Won't work with very large files right now.

Source - <https://github.com/arcanecfg/FileNinja>

Drop suggestions and improvements below, thanks.

- A

Zilog80 - February 20, 2017, 11:05 am

Waiting for [Ninja](#) reply ?

.cYRil. - February 20, 2017, 12:48 pm

(Feb 20, 2017 12:05 PM)Zilog80 Wrote: [\[CITE\]](#) Waiting for [Ninja](#) reply ?
Exactly , lol.

Decim - February 21, 2017, 7:05 am

Thanks for the constructive comments.

WingsOfAltair - February 21, 2017, 9:48 am

This is actually pretty good and could be useful at certain points, in my opinion that is.
Good job.

jujuger - February 21, 2017, 11:14 am

[AES.Mode = CipherMode.ECB;](#)

OP you just dun goofed cryptographically speaking.

Electronic Code Book is a broken Stream Cipher

Blocks of identical text are encrypted identically. (Yes, this is bad extremely bad)

With ECB, the text "chocolate" would always gets encrypted to "sjkskhbqp". So if your plaintext was "I eat chocolate I like chocolate" it would get encrypted to "O pbq sjkafhbqp O hoxp sjkafhbqp" see the problem?

<http://crypto.stackexchange.com/questions/20941/why-shouldnt-i-use-ecb-encryption>

To be blunt, since you are using ECB, OP, your encryption program is completely broken.

Use CTR mode or CBC mode instead of ECB

Decim - February 22, 2017, 12:48 pm

(Feb 21, 2017 12:14 PM)jujuger Wrote: [\[CITE\]](#) `AES.Mode = CipherMode.ECB;`

OP you just dun goofed cryptographically speaking.

Electronic Code Book is a broken Stream Cipher

Blocks of identical text are encrypted identically. (Yes, this is bad extremely bad)

With ECB, the text "chocolate" would always gets encrypted to "sjkskhbqp". So if your plaintext was "I eat chocolate I like chocolate" it would get encrypted to "O pbq sjkafhbqp O hoxp sjkafhbqp" see the problem?

<http://crypto.stackexchange.com/questions/20941/why-shouldnt-i-use-ecb-encryption>

To be blunt, since you are using ECB, OP, your encryption program is completely broken.

Use CTR mode or CBC mode instead of ECB

Thanks for that piece of info.

But I just quickly tested it again and that doesn't seem to be the case. Try it out yourself and let me know?

jujuger - February 22, 2017, 3:06 pm

(Feb 22, 2017 01:48 PM)+Arcane Wrote: [\[CITE\]](#) Thanks for that piece of info.

But I just quickly tested it again and that doesn't seem to be the case. Try it out yourself and let me know?

It isn't a problem with code in .NET or C#

it is a problem with security

Decim - February 22, 2017, 6:21 pm

(Feb 22, 2017 04:06 PM)jujuger Wrote: [\[CITE\]](#) It isn't a problem with code in .NET or C#

it is a problem with security

I understand what you're saying. But if that's the case, shouldn't encrypting two simple plain text files containing a few identical words with the same key result in their corresponding ciphertexts looking similar? Is that not what makes it semantically insecure?

TheRealDykeGamer - June 4, 2017, 7:59 am

This program is very useful, I've used it and had no problems with it!

Dotexe - June 4, 2017, 9:25 am

Thank you for that Arcane.

Ninja - June 4, 2017, 10:09 am

You'll hear from my lawyer soon.

StaticCG - June 4, 2017, 3:59 pm

(Jun 4, 2017 10:09 AM)Ninja Wrote: [\[CITE\]](#) You'll hear from my lawyer soon.

?

On the real, great application :)

Alone. - June 5, 2017, 3:10 am

Nice release,
Its an useful share.

HUDCO - June 5, 2017, 3:13 am

(Jun 4, 2017 10:09 AM)Ninja Wrote: [\[CITE\]](#) You'll hear from my lawyer soon.Thats a nice one.

Ninja - June 5, 2017, 9:54 pm

I looked over your code, and I saw you could re-use some blocks of code, and I saw you were inconsistent with access modifiers. Just 2 things that caught my eye.

mrnikbobjeff - June 10, 2017, 4:38 am

(Feb 22, 2017 07:21 PM)Arcane. Wrote: [\[CITE\]](#) I understand what you're saying. But if that's the case, shouldn't encrypting two simple plain text files containing a few identical words with the same key result in their corresponding ciphertexts looking similar? Is that not what makes it semantically insecure?

Kinda, he is right with the cipher mode being absolutely flawed for this scenario.[The wikipedia article](#) does better explaining than me. My guess is that you had a text file where the blocksize did not coincide with the message, e.g. the text

"germangermangermangermangermangermangermangermangermangermangermangermangermangerman" should coincide with the blocksize of 16, thus splitting the text in two chunks which repeat itself (verifiable [here](#).)

Washi - June 10, 2017, 9:06 am

Source is very readable. Nice naming on your variables. A few pointers that you might want to consider:

[FileNinja.cs:103](#): You can shorten that entire if statement to:

PHP Code

```
<?php
```

```
txtKey.UseSystemPasswordChar = !txtKey.UseSystemPasswordChar;
```

[FileNinja.cs:155](#) and [FileNinja.cs:218](#): Any particular reason why you are declaring a file name as an object, only to use it with ToString later? Why not just declare it as a string?

[FileNinja.cs:199](#): Consider using one of the overloads of [Path.Combine](#) instead of doing the formatting yourself using string concatenation.

I also noticed that you use a lot of [Control.Invoke](#) method calls in order to avoid cross-threading problems. While this is not a huge problem, it's not really efficient to do it for each line of code. An example would be at [FileNinja:262](#), where you used Invoke twice to change properties of two controls. Since these controls are created on the same thread, it suffices to call Invoke once like so:

PHP Code

```
<?php
```

```
this.Invoke(new Action(() =>
```

```
{
```

```
    lbMain.Items.Remove(topDir);
```

```
    progressBar.Visible = false;
```

```
});
```

[AES.cs](#): Don't forget to dispose the instances for the [RijndaelManaged](#), [MD5CryptoServiceProvider](#)

and the [ICryptoTransform](#) instances in both the encrypt and decrypt method. You can use a using statement for that.

Just my two cents ?

AmericanExpress - June 29, 2017, 8:55 am

If you ever decide to implement a "secure file removal" feature you could overwrite the target files using the 'Gutmann Method' (35 passes) so the original data is overwritten with a bunch of gibberish and then mess up any file properties you can a few times (e.g. file name) and then finish off with `NtDeleteFile` (since you'll be able to remove the file using that function even if it is in use by another process, unlike with the Win32 API `WriteFile` which performs those checks beforehand) (use `P/Invoke` for `ntdll.dll`).

You could do it in C/C++ and use `(dllexport)` and then `P/Invoke` from C# project if it was easier that way.

Just an idea. ?

Ninja - June 29, 2017, 10:31 pm

(Jun 29, 2017 08:55 AM)AmericanExpress Wrote: [\[CITE\]](#) If you ever decide to implement a "secure file removal" feature you could overwrite the target files using the 'Gutmann Method' (35 passes) so the original data is overwritten with a bunch of gibberish and then mess up any file properties you can a few times (e.g. file name) and then finish off with `NtDeleteFile` (since you'll be able to remove the file using that function even if it is in use by another process, unlike with the Win32 API `WriteFile` which performs those checks beforehand) (use `P/Invoke` for `ntdll.dll`).

You could do it in C/C++ and use `(dllexport)` and then `P/Invoke` from C# project if it was easier that way.

Just an idea. ?

I love how you always give ideas based on Windows API and low level programming and what not, but you forget that %99 of the programming section has no idea what you are talking about.

AmericanExpress - June 30, 2017, 7:12 am

(Jun 29, 2017 10:31 PM)Ninja Wrote: [\[CITE\]](#) I love how you always give ideas based on Windows API and low level programming and what not, but you forget that %99 of the programming section has no idea what you are talking about. I cannot deny that you are probably right haha, well hopefully someone still finds it helpful one day.