

You are here: [Welcome to mbirth's Wiki](#) » [Code Snippets](#) » [Python](#) » [AES / Rijndael](#)

## AES / Rijndael

### Table of Contents

- AES / Rijndael
  - AesEncryptCtr / AesDecryptCtr
    - Usage
    - See Also

Usually, you should use [PyCrypto](#) from the [python-crypto](#) package. But if you want to code in Python3, there's no fast [hybrid<sup>1\)</sup>](#) implementation of such a library.

Using Google, you will most probably stumble on [Bram Cohen's Rijndael implementation in pure Python.<sup>2\)</sup>](#) I took his code and made it Python3 ready by replacing all `xrange()` by `range()`, all divisions (`/`) by integer-divisions (`//`) and made the `string.join()` working. There were no more changes necessary.

See the working Python class here: [rijndael.py](#) (10.69 KiB, [1285 downloads](#))

Another Rijndael implementation I found was [pyRijndael](#). After changing the two `long()` to `int()` and adding parentheses to all the `prints` at the end of the file, it worked fine with Python3.

## AesEncryptCtr / AesDecryptCtr

[Chris Veness](#) had created a [JavaScript implementation](#) of AES in counter operation mode some time ago. He also ported this script to [PHP](#) so that you can interchange information between those two systems.

I ported the same library to Python to let Python talk to a [PHP](#) server in an encrypted way.

Download it here: [aes.py](#) (7.59 KiB, [1021 downloads](#))

## Usage

```
import aes
text = "Hello, world!"
password = "itsmysecret"
blocksize = 256 # can be 128, 192 or 256
crypted = aes.encrypt( text, password, blocksize )
# do something
decrypted = aes.decrypt( crypted, password, blocksize )
```

## See Also

- [AES / Rijndael](#) — JavaScript implementation of this algorithm
- [AES / Rijndael](#) — [PHP](#) implementation of this algorithm

### AROUND THE WEB



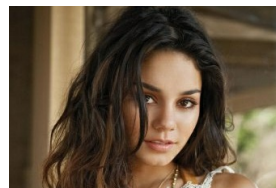
**Must Have Appliances Being sold For Next To Nothing**  
Lifefactopia



**Top 10 Turn-Offs for Women**  
Lifescript



**50 Best Pizzas in America: One from Every State**  
Zagat



**12 Celebrities You Didn't Know Were Asian**  
Answers.com

5 Comments

mbirth's Wiki

Login

Sort by Newest

Share Favorite



Join the discussion...



**Gabor** • a year ago

Resolved. The error was in my code...

1 ^ | v • Reply • Share



**Gabor** • a year ago

Hello

Welcome to mbirth's Wiki

Willkommen auf mbirth's Wi

Know-How

Code Snippets

apache

AutoHotkey

C #

CMD/BAT

CSS

Excel

Java

JavaScript

PHP

Python

AES / Rijndael

TrayIcon with wxWidg

SH

VBA

Software

### Other pages

- [Homepage](#)
- [Piwik](#)
- [SiteBar Bookmarks](#)



1



NetApp®  
Unbound  
Cloud

[netapp.com/Cloud](#)

Download Our Cloud  
Study & Learn A New  
Way to Manage Private  
Cloud!



PMI®  
Strategic  
Initiative

Free IP  
Address  
Manager

Network  
Security  
Software

Django Web  
Framework



1 hour,

I'm trying to use AESEncryptCtr / AESDecryptCtr in php / python by the following way:

I'm encrypting my string in php, and put that into a cookie:

```
$encrypted = AesCtr::encrypt($content,$password,256);  
setrawcookie("user", $encrypted, time()+3600);
```

Than I make a http request in python, read the content of the cookie, and try to decrypt like this:

```
import requests  
import aes  
...  
r = requests.get(url)  
encrypted = r.cookies[user]  
decrypted = aes.decrypt(encrypted,password,256)
```

the code runs without errors on both sides, but I get a meaningless "decrypted" string at the end. I know that data transfer by cookie is not very elegant, please don't comment that I have debugged the

[see more](#)

^ [v] • Reply • Share ›



**gadelat** • 2 years ago

your AESEncryptCtr / AESDecryptCtr does not work :(

```
gadelat@gadelat-Latitude-E6410:~/opera/temporary_downloads$ python3  
Python 3.2.2 (default, Sep 5 2011, 21:17:14)  
[GCC 4.6.1] on linux2
```

Type "help", "copyright", "credits" or "license" for more information.

```
>>> import aes
```

```
>>> text = "Hello, world!"
```

```
>>> password = "itsmysecret"
```

```
>>> blocksize = 256 # can be 128, 192 or 256
```

```
>>> crypted = aes.encrypt( text, password, blocksize )
```

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

File "aes.py", line 148, in encrypt

key = Cipher(pwBytes, KeyExpansion(pwBytes))

File "aes.py", line 112, in KeyExpansion

w = [0] \* Nb\*(Nr+1)

TypeError: can't multiply sequence by non-int of type 'float'

1 ^ [v] • Reply • Share ›



**Juan Fran Blanco** • 2 years ago

Great library, many thanks.

^ [v] • Reply • Share ›



**Robert** • 3 years ago

Does this work on Windows too?

^ [v] • Reply • Share ›



Subscribe



Add Disqus to your site

DISQUS

1) i.e. mostly C-code, partly Python-code

2) You know Bram Cohen, don't you?

snippets/python/aes-rijndael.txt · Last modified: 2013-03-16 17:27:03 (external edit)

Show pagesource

Old revisions

Media Manager

Login

Sitemap

Back to top

Except where otherwise noted, content on this wiki is licensed under the following license: CC Attribution-Noncommercial-Share Alike 3.0 Unported

RSS XML FEED

CC BY-NC-SA

DONATE

PHP POWERED

W3C XHTML 1.0

W3C CSS

DOKUWIKI

HATE ADDICT

BASS DRIVE

ICRA

Web2PDF

converted by Web2PDFConvert.com