- 目录
- 前言
- 关键字Lambda
- 第一个Python程序
- Python编辑器
- 安装
- 数学计算
- 数据类型
- 字符串和编码
- 使用list和tuple
- 条件判断
- 模式匹配match
- 循环
- 使用dict和set
- 函数
- 调用函数
- 定义函数
- 函数的参数
- 递归函数
- 切片
- 迭代
- 列表生成式
- 生成器
- 迭代器
- 函数式编程
- 匿名函数lambda
- 装饰器decorator
- 偏函数partial
- 模块
- 使用模块
- 安装第三方模块
- 面向对象编程
- 类和实例
- 访问限制
- 继承和多态
- 获取对象信息
- 实例属性和类属性
- 面向对象高级编程
- 定制类
- 枚举类
- 使用元类
- 错误处理try
- 调试
- 单元测试
- 文档测试
- IO编程
- 文件读写
- StringIO和BytesIO
- 操作文件和目录
- 序列化
- 进程和线程
- 多进程multiprocessing
- 多线程threading
- 使用threadlocal
- 分布式进程
- 正则表达式
- 常用内建模块
- datetime
- collections
- base64
- struct
- hashlib
- hmac
- itertools
- contextlib
- urllib
- XML
- HTMLParser
- 常用第三方模块
- virtualenv
- 图形界面
- 网络编程
- 电子邮件
- 访问数据库
- MySQL开发
- 数据库
- 实战
- FAQ
- 期末总结

关于作者

廖雪峰 关注 @廖雪峰

廖雪峰的官方网站

廖雪峰
自己的Python课程

Python商业黑客全解码
以天下生命相要不到的黑客！

Python黑客 | 数据分析
Python高级开发 | 深度学习

找廖雪峰老师

廖雪峰老师
自己的Java课程

Java高级架构师
更专业 更权威
源码分析专家 | 性能调优专家
高并发分布式架构 | 性能优化专家

找廖雪峰老师

廖雪峰 精选项目
Python 商业黑客全解码
Python黑客 | 数据分析 | 深度学习
高并发分布式架构 | 性能优化专家
Java高级架构师
找廖雪峰老师

ACM金牌得主
世界排名第一 | 特邀嘉宾导师团
入门零基础名师授课 | 理解底层原理
无offer退全款

廖雪峰推荐
JAVA进阶教程
原价1599元
0元领取
立即领取

阿里云
高性能云服务器自住5折
ECS | 弹性计算 | 最低1.5折起

python免费公开课
直播时间
周四晚8点，在线直播+录播回放，从零基础到中级实现工程师
查看详情

学生党专享福利 ¥0起
每天1GB流量 | 1核1G
立即购买

阿里云1元限时抢，最低至0.1元/月
新人专享1元起 | 每天1GB流量 | 1核1G
立即抢

评论

来不及开始就进正确的
[illegible] commented on Jul 19, 2019 2:29 PM, Last updated on January 12, 2020 2:17 AM

期待再更新，degistmod加解密模块和SSL，但是要等了解的时间内就到，有可以等待编辑器

编辑 删除 回复

[illegible] commented on Jul 19, 2019 7:02 PM, Last updated on April 29, 2019 7:02 PM

可以试试加上下面的代码：b = hmac.new(key,message,digestmod='MD5').hexdigest()或digest()，参数改成'MD5'

[illegible] commented on Jul 19, 2019 7:02 PM, Last updated on April 29, 2019 7:02 PM

可以试试加上下面的代码：b = hmac.new(key,message,'MD5').b = hmac.new(key,message,'SHA').默认可用b = hmac.new(key,message).默认采用SHA算法

编辑 删除 回复

[illegible] commented on April 5, 2019 7:37 PM, Last updated on April 5, 2019 7:37 PM

def login(username, password): user = db[username] return user.password == hmac.hmac(key, password).hexdigest()

编辑 删除 回复

[illegible] commented on Apr 29, 2019 7:02 PM, Last updated on April 29, 2019 7:02 PM

```
# 测试
assert login('michael', '123456')
assert login('bob', '123456')
assert login('alice', '123456')
assert login('michael', 'wrong-pass')
assert not login('michael', '123456')
assert not login('alice', '123456')
print('ok')
```

▶ Run

小结

使用Python模块hmac模块实现了HMAC算法，比较简一下hmac的usage方式：使用hmac算法比较hmac摘要是否正确，需先对明文明文message，不同的key和产生不同的hmac。

接近完善是可以做的是哪种，就应用哪种函加密的话：

关于作者

廖雪峰 关注 @廖雪峰

← 上一篇
下一篇 →

评论

[illegible] commented 2 months ago, Edited 2 months ago

期待再更新，degistmod加解密模块和SSL，但是要等了解的时间内就到，有可以等待编辑器

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

可以试试加上下面的代码：b = hmac.new(key,message,'MD5').b = hmac.new(key,message,'SHA').默认可用b = hmac.new(key,message).默认采用SHA算法

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

def login(username, password): user = db[username] return user.password == hmac.hmac(key, password).hexdigest()

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

import random import time

#定义密钥和明文消息

def hmac_md5(key, s): return hmac.new(key.encode('utf-8'),s.encode('utf-8'),digestmod='MD5').hexdigest();

#创建用户名和密码，用户名username，password，key的盐值 class User(object): def init(self,username,password): self.username=username self.key=''.join(random.randint(48, 122) for i in range(20))

#hmac_md5(self.key,str(time.time())).hexdigest() #使用当前时间作为盐值

self.password=hmac_md5(self.key,password)

#创建db用户数据集 db=dict(user1=User('bob','123456'),user2=User('alice','abc456'))

def login(username,password):

#print('%s,%s,%s',(username,password,db[username].key,db[username].password,hmac_md5(db[username].key,password)))

if username in db.keys():

return db[username].password == hmac_md5(db[username].key,password)

else:

return False

assert login('bob', '123456')

assert login('alice', 'abc456')

assert not login('bob', 'abc456')

assert not login('alice', '123456')

print('ok')

▼ Read More

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

#

def login(username, password): if username in db: user = db[username] return user.password == hmac.hmac(key, password).hexdigest() else: return False

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

可以试试加上下面的代码：b = hashlib.md5

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago, Edited 2 months ago

来不及开始就进正确的

[illegible] commented 2 months ago, Edited 2 months ago

小结

使用第三方模块hmac模块实现了HMAC算法，比较简一下hmac的usage方式：使用hmac算法比较hmac摘要是否正确，需先对明文明文message，不同的key和产生不同的hmac。

接近完善是可以做的是哪种，就应用哪种函加密的话：

关于作者

廖雪峰 关注 @廖雪峰

← 上一篇
下一篇 →

评论

[illegible] commented 2 months ago, Edited 2 months ago

期待再更新，degistmod加解密模块和SSL，但是要等了解的时间内就到，有可以等待编辑器

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

可以试试加上下面的代码：b = hmac.new(key,message,'MD5').b = hmac.new(key,message,'SHA').默认可用b = hmac.new(key,message).默认采用SHA算法

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

def login(username, password): user = db[username] return user.password == hmac.hmac(key, password).hexdigest()

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

import random import time

#定义密钥和明文消息

def hmac_md5(key, s): return hmac.new(key.encode('utf-8'),s.encode('utf-8'),digestmod='MD5').hexdigest();

#创建用户名和密码，用户名username，password，key的盐值 class User(object): def init(self,username,password): self.username=username self.key=''.join(random.randint(48, 122) for i in range(20))

#hmac_md5(self.key,str(time.time())).hexdigest() #使用当前时间作为盐值

self.password=hmac_md5(self.key,password)

#创建db用户数据集 db=dict(user1=User('bob','123456'),user2=User('alice','abc456'))

def login(username,password):

#print('%s,%s,%s',(username,password,db[username].key,db[username].password,hmac_md5(db[username].key,password)))

if username in db.keys():

return db[username].password == hmac_md5(db[username].key,password)

else:

return False

assert login('bob', '123456')

assert login('alice', 'abc456')

assert not login('bob', 'abc456')

assert not login('alice', '123456')

print('ok')

▼ Read More

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

#

def login(username, password): if username in db: user = db[username] return user.password == hmac.hmac(key, password).hexdigest() else: return False

编辑 删除 回复

[illegible] commented 2 months ago, Edited 2 months ago

可以试试加上下面的代码：b = hashlib.md5

编辑 删除 回复

[illegible] commented