CTF学习笔记——Havefun&easy_tornado

原创

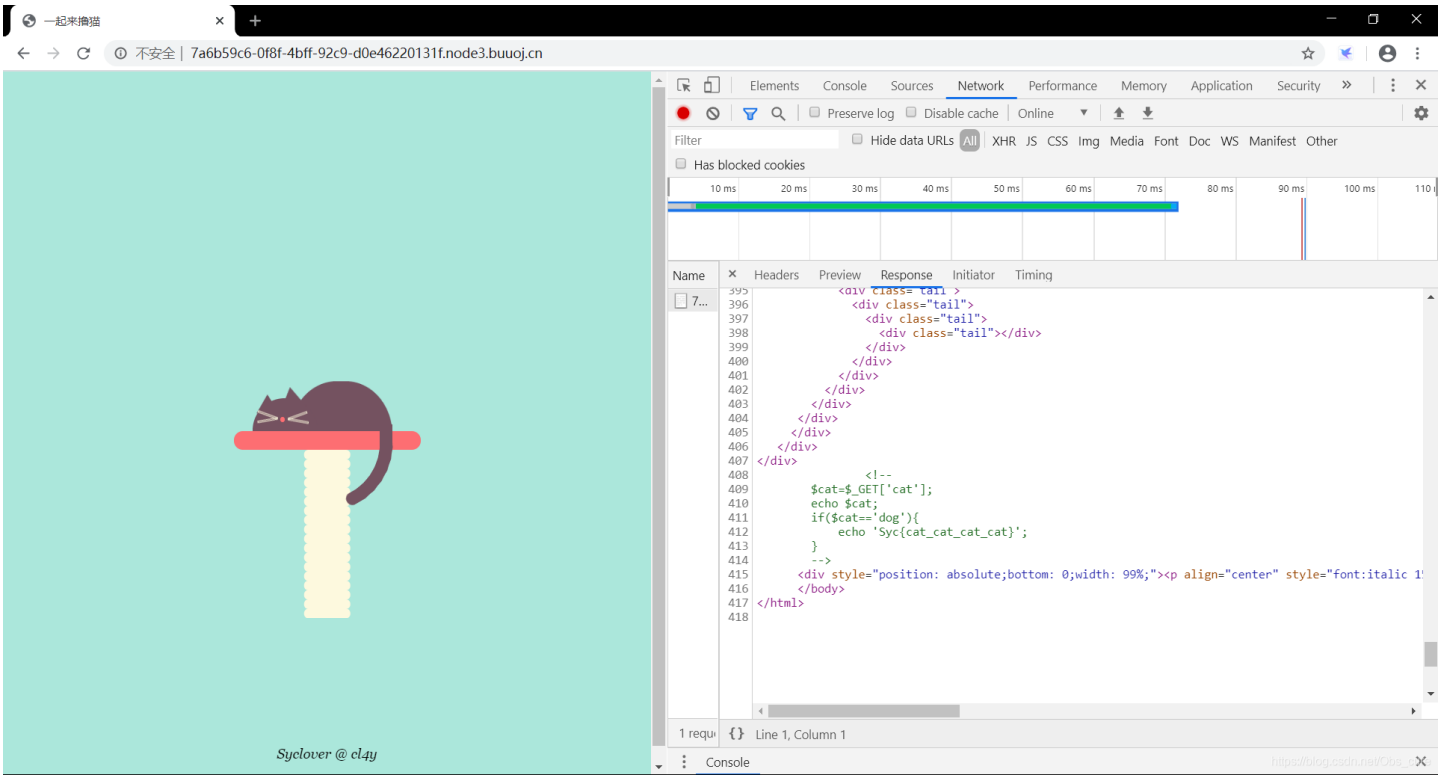
Obs_cure 于 2020-08-23 20:45:11 发布 494 收藏

文章标签： 网络安全

本文链接：https://blog.csdn.net/Obs_cure/article/details/108186377

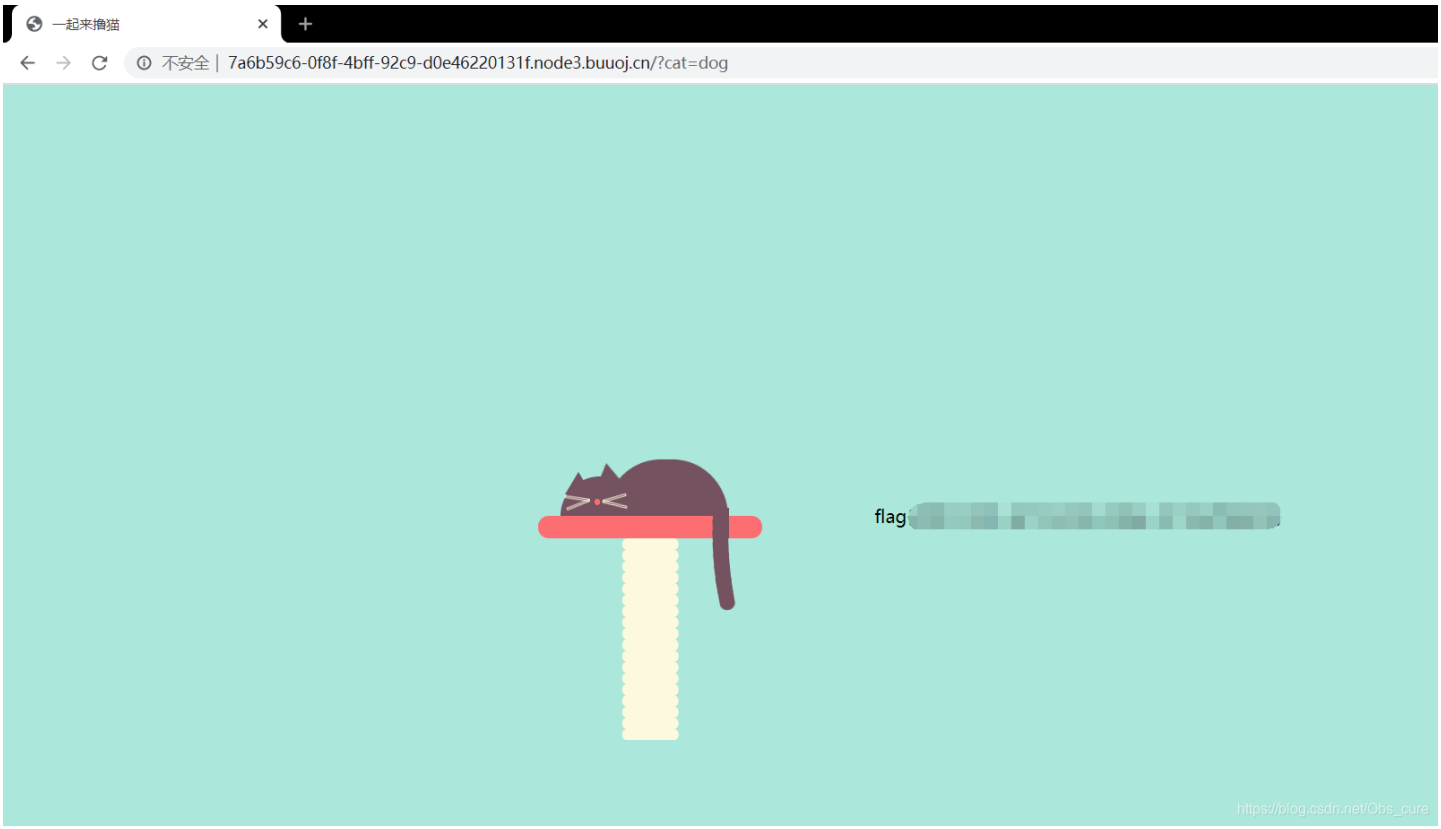
一、[极客大挑战 2019]Havefun

1.题目



2.解题步骤

虽然作为一个php废物，但是看到下面注释里写的明显是要提交一个get变量，而且很明显有一个dog的提示，就尝试输入一下：



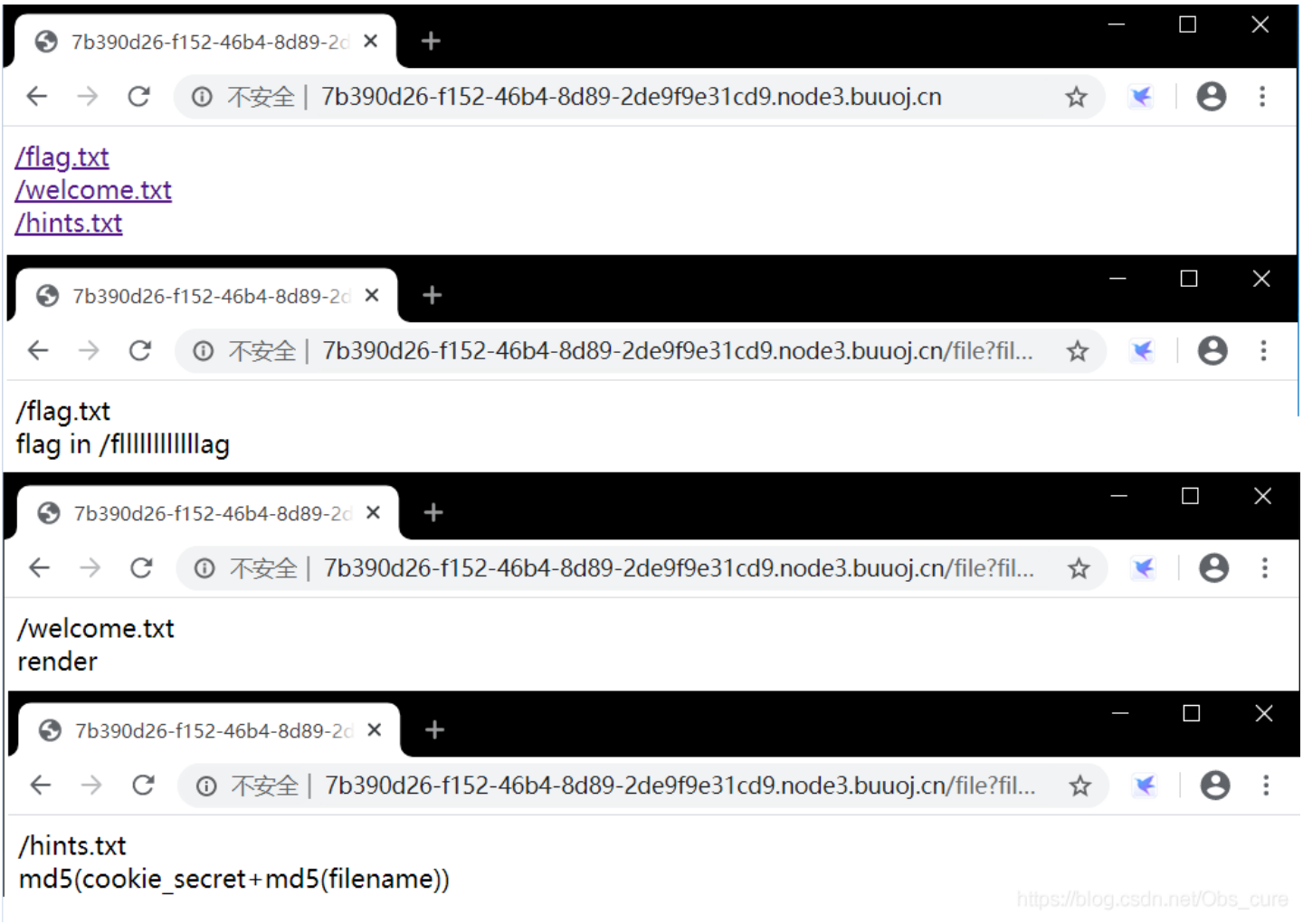
成功爆出flag。

3.总结

这是第一次自己不看writeup做出来的web题，泪目。回头看了一些writeup发现这道题在给cat变量赋值的时候会显示出来，如果赋dog会出flag，但是输入其他值并没有回显，不清楚为什么。

二、[护网杯 2018]easy_tornado

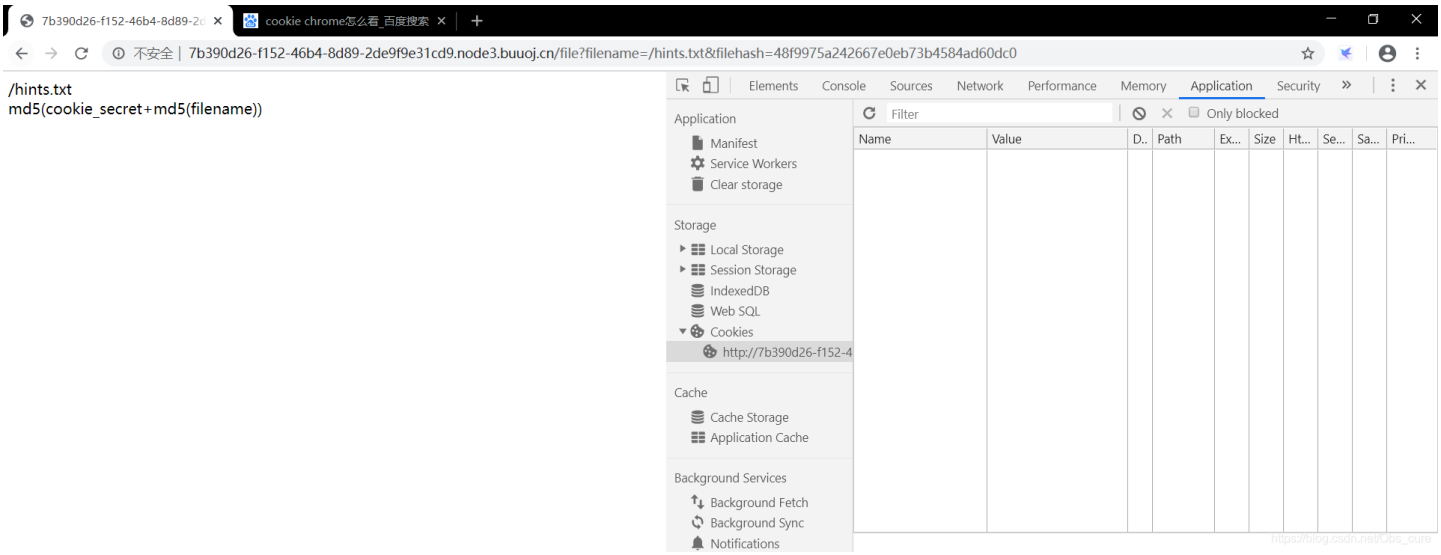
1.题目



题目界面是三个文件夹，依次进入后是这样的信息。

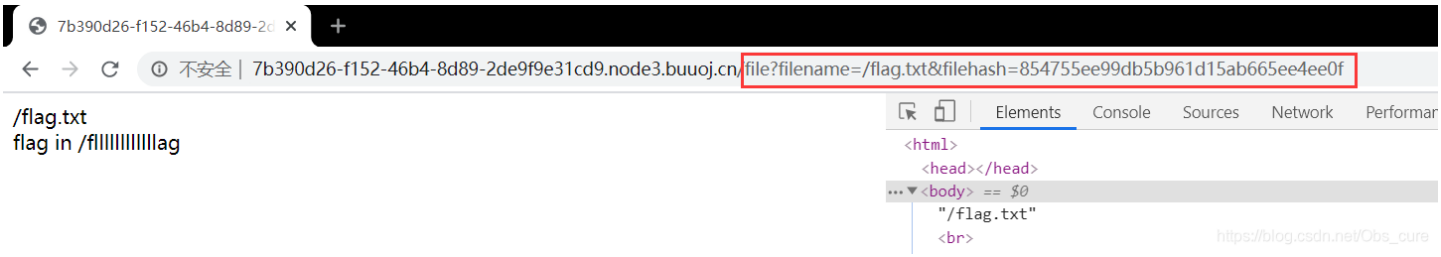
2.解题步骤

第三个文件夹很明显的提示cookie中有东西，所以先去找一下



唔…百度了半天怎么看cookie 然后发现什么都没有…还是老老实实的学技术吧~没有头绪，看看writeup

师傅说，这三个文件是三个提示。比如说第一个flag文件，提示的是flag所在的文件名。



从这张图可以看，在访问文件信息的时候，都是文件名&一个文件的哈希值。

而第三个文件中，正好提示了文件哈希的计算方法。

接下来就是如何获取文件的cookie_secret了。题目的名字是tornado

Tornado - Web服务器

Tornado是使用Python开发的全栈式（full-stack）Web框架和异步网络库，最早由Friendfeed开发。通过使用非阻塞IO，Tornado可以处理数以万计的开放连接，是long polling、We...

码云指数为18，超过32%的开源项目

软件类型：HTTP服务器 | 授权协议：Apache | 开发语言：Python

Star：7 | Fork：0

最新版本v4.5.3 代码仓库 相关博客 软件文档

https://www.oschina.net/p/tornado?hmsr=aladdin1e1

可见是去构造一个方法去访问cookie_secret的内容。师傅们是直接去看的官方文档...遗憾的是小白看了半天也没找到是怎么用handler.settings去访问到cookie_secret的。

发行说明

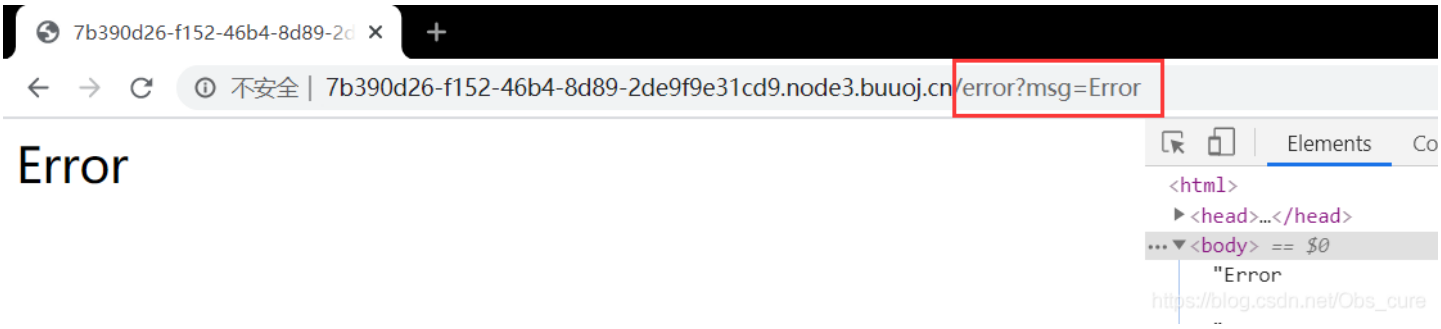
https://docs.python.org/3/library/http.cookies.html#http.cookies.Morsel。

RequestHandler.clear_cookie（名称：str，路径：str ='/'，域：可选[str] = None ）→无 [资源]

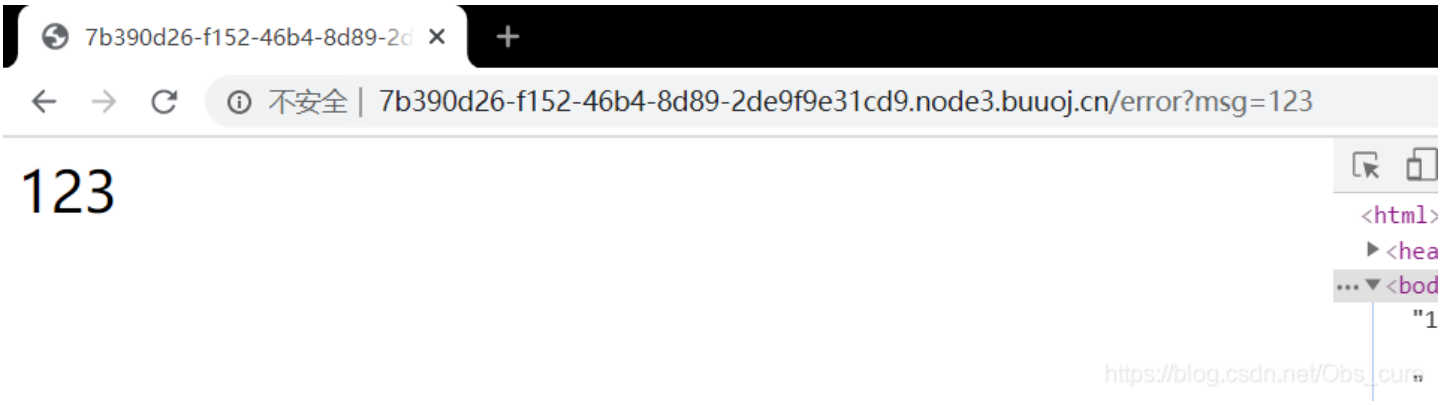
翻了好多writeup，这里直接照搬师傅的原话了

> 在tornado模板中，存在一些可以访问的快速对象,这里用到的是handler.settings，handler 指向RequestHandler，而RequestHandler.settings又指向self.application.settings，所以handler.settings就指向RequestHandler.application.settings了，这里面就是我们的一些环境变量。[原题复现]2018护网杯(WEB)easy_tornado(模板注入) - 笑花大王 - 博客园

蒙圈了半天，佩服当年做题的师傅…
好了，现在取得cookie_sercet的方法也拿到了，现在应该去思考如何执行了。通过测试发现如果在文件名称和哈希不对的情况下，会跳出错误的页面：

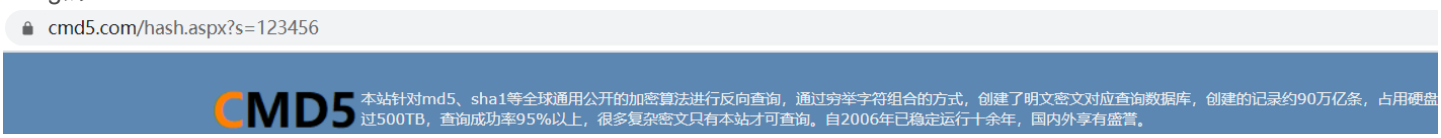


可以清晰的看到有一个?msg=Error。那要是换一下其他massage呢？



可见msg的功能是打印。那让他打印一下cookie_secret呢

这里有点懵的是要两个大括号…可能跟py的语法有关…?总之是拿到了cookie_secret的值了，然后根究前文的构造方法算一下flllllag的md5。



首页 解

q一键登录

Pass: /flllllllllllag unicode $[HEX...

Salt: HEX

Hash:

加密

Result:
base64: L2ZsbGxsbGxsbGxsbGxhZw==
md5: 3bf9f6cf685a6dd8defadabfb41a03a1
md5_middle: 685a6dd8defadabf
md5(md5($pass)): 8b7d24d2cc2bdc830c26bf26854ec7c9
md5(md5(md5($pass))): b7927cf6df61da9bf9d27f6692fb3e95
md5(unicode): 6ef05a98e55e760e40ad89798ce635e5

CMD5 本站针对md5、sha1等全球通用公开的加密算法进行反向查询，通过穷举字符组合的方式，创建了明文密文对应查询数据库，创建的记录约90万亿条，占用硬盘超过500TB，查询成功率95%以上，很多复杂密文只有本站才可查询。自2006年已稳定运行十余年，国内外享有盛誉。

首页 解密

Pass: b9568655-e76b-4543-ba3e-4dbdd522188a unicode $[HEX...

Salt: HEX

Hash:

加密

Result:
base64: Yjk1Njg2NTUtZTc2Yi00NTQzLWJhM2UtNGRiZGQ1MjIxODhh
md5: 0f78587da2bd7db1ad66c4be6f44be4a
md5_middle: a2bd7db1ad66c4be
md5(md5($pass)): ec382c33325b3a9e2fd0935bcd9db0ab
md5(md5(md5($pass))): 83449b9652836b9ebf330533dbdcb3a0
md5(unicode): c700a12623d6aa8f325ef33f3264724c
md5(base64): D3hYfaK9fbGtZsS+b0S+Sg==
mysql: 2c17826c54eb178e
mysql5: 81cc419ac7a337cb836e5e6b75f222e300abb769
ntlm: 47ad409526acd978f08e10ecb6c68445

cmd5.com/hash.aspx?s=123456

CMD5 本站针对md5、sha1等全球通用公开的加密算法进行反向查询，通过穷举字符组合的方式，创建了明文密文对应查询数据库，创建的记录约90万亿条，占用硬盘超过500TB，查询成功率95%以上，很多复杂密文只有本站才可查询。自2006年已稳定运行十余年，国内外享有盛誉。

首页 解密范围

q一键登录

Pass: b9568655-e76b-4543-ba3e-4dbdd522188a3bf9f6c unicode $[HEX...

Salt: HEX

Hash:

加密

Result:
base64: Yjk1Njg2NTUtZTc2Yi00NTQzLWJhM2UtNGRiZGQ1MjIxODhhM2JmOWY2Y2Y2ODVhNmRkOGRlZmFkYWJmYjQxYTAzYTE=
md5: 38b2e194feea162467096094dc77db6c
md5_middle: feea162467096094
md5(md5($pass)): 3908cfc9f80ea5433e22615d6f2ed0d0
md5(md5(md5($pass))): 7f3384948f32d0e864d055605a1056bc
md5(unicode): ccc4317292971ee5a6c83e452944f93d
md5(base64): OLLhlP7qFiRnCWCU3HfbbA==
mysql: 43750b6163f2dcea
mysql5: 93e3cc76a26d1c7b7b3cacda0ea8e447a329d4fd
ntlm: 72025433b30816528105c8effd81085f

3.总结

1. 这算做的第一道模板注入的题吧，感觉翻文档挺痛苦的。查了半天也不清楚是怎么用句柄访问到的，还要靠师傅给翻译一下…
2. {{}}两个大括号的闭合方式第一次见…个人感觉是先闭合msg的赋值语句，然后是查询语句？查了一下python的大括号专门是括字典数据类型的，应该跟py没有关系…可能是那个tornado的语法吧…
3. 学会了md5这些加密的处理方法，还算有点收获吧。

4.参考资料

- [原题复现]2018护网杯(WEB)easy_tornado(模板注入) - 笑花大王 - 博客园
- [护网杯 2018]easy_tornado_qq_43622442的博客-CSDN博客_[护网杯 2018]easy_tornado
- 学习笔记2.护网杯 2018 easy_tornado - 简书