|  |  |  |  |
| --- | --- | --- | --- |
| **计算机网络安全 实验报告** | | | |
| **实验名称** | 实验3 **hash 算法** | | |
| **日期** | 2018/11/9 | **专业班级** | 16信安1班 |
| **实验人** | 刘开凯 | **学号** | 29 |

一、实验目的

熟hash的加密原理。

二、实验内容

1.小明入侵网站后获得了管理员的密文，由于太高兴了手一抖把密文删除了一部分，只剩下前10位3b79b7ea7f，小明根据社工知道管理员的密码习惯是key{4位的数字或字母} ，所以管理员的密码是？（SHA512）  
2. 解决字谜问题。已有单词本，现给出某一个单词的乱序形式，比如，给出“nitysarato”，要求设计算法得出原来的单词是“stationary”。并求“mrptu”原来的单词是？附件给出的是37万+单词的词库（文本和Json形式），可以自由选用。  
提示：对于n个字母组成的单词有n!可能性，暴力搜索运算量太大。可考虑设计一Hash函数，为每个单词计算一个Hash值，然后比对hash值（可能碰撞），如果成功再严格比对字母。如果单词本词汇量特别大，考虑设置两个或多个hash函数。

三、实验步骤和设计思想

利用hash算法的各个性质加密

四、程序清单

 五、实验调试记录

import hashlib

str\_key = 'key{XXXX}'

str\_sha512\_fore = '3b79b7ea7f'

list\_base = '1234567890ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz'

list\_key = list(str\_key)

for i in range(len(list\_base)):

list\_key[4] = list\_base[i]

for j in range(len(list\_base)):

list\_key[5] = list\_base[j]

for k in range(len(list\_base)):

list\_key[6] = list\_base[k]

for l in range(len(list\_base)):

list\_key[7] = list\_base[l]

str\_sha = str(hashlib.sha512(''.join(list\_key).encode('utf-8')).hexdigest())

if str\_sha[0:10] == str\_sha512\_fore:

print(''.join(list\_key), str\_sha)

print("end")

2. # -\*-coding:utf-8-\*-

def new\_hash(text, count):

sum = 0

for i in text:

sum += pow(ord(i), count)

return sum

def main():

question = 'face'

value = new\_hash(question, 3)

lis = []

f = open('words\_alpha.txt', 'r')

for i in f:

i = i.strip()

value2 = new\_hash(i, 3)

if value2 == value:

lis.append(i)

print(lis)

return lis

if \_\_name\_\_ == "\_\_main\_\_":

main()

1. 实验结果及其分析

1. C:\Users\KevinLeak\AppData\Local\Microsoft\Windows\INetCache\Content.Word\2018-12-07_104423.png

2.

C:\Users\KevinLeak\AppData\Local\Microsoft\Windows\INetCache\Content.Word\2018-12-07_105456.png

七、实验心得

还是要看着老师的代码，再敲一遍吗，才能熟悉逻辑，对于hash算法，大概明白了他的加密方式