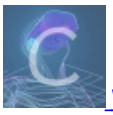
WeChall-writeup-3-11.14

原创

MeliodasC 于 2018-11-15 21:56:01 发布 436 收藏

分类专栏： WeChall 文章标签： WeChall writeup 安全



啊哈，这次只隔了一天，是昨天刚写完的writeup。这次收集到了几个好用的小工具，会在下面贴出来~

Training: Crypto - Transposition I

首先按照题意学习了一下 transposition ciphers，手动推算了一下这题是六个字符为一组。如下面的程序来调整字符的位置。

```
#include<iostream>
using namespace std;
int main()
{
   string a;
   while(getline(cin,a))
   {

      for(int i=0;i<a.size();i+=6)
      {
         cout<<a[i+1]<<a[i]<<a[i+3]<<a[i+2]<<a[i+5]<<a[i+4];
      }
      cout<<endl;
      if((6-(a.size()%6))!=0)  cout<<6-(a.size()%6)<<endl;
   }
   return 0;
}
```

程序中有一行输出划分后不符合六个一组的，多出来的字符位数。在提交答案时去掉多出来的位数即可。

Training: Crypto - Substitution I

https://www.cnblogs.com/ECJTUACM-873284962/p/7872114.html 可以写程序按照这个网站上的方法手工来推

或者利用quipqiup来自动模拟上述过程（真的好快！）

Training: Crypto - Caesar II

思想还是凯撒密码，但是k的大小从0-25变为了0-127.所以编程跑代码吧~总的思路是：十六进制->十进制->移位->（ASCII代表的）实际字符。发现作者有点调皮2333，把C替换成空格就好啦。

```cpp
#include<iostream>
#include<string>
#include<math.h>
#include<stdio.h>
using namespace std;
int ch[300];
int main()
{
    string ss;
    cin>>ss;
    for(int i=0;i<ss.size();i++){
        if(ss[i]!='C')
            cout<<ss[i];
            else
                cout<<' ';
    }
    cout<<endl;
    string s;
    int sum=0;
    int T=245;
    int cnt=0;
    while(T--)
    {
        sum=0;
        cin>>s;
        sum+=(s[0]-'0')*16;
        if(s[1]>='A'&&s[1]<='F')
            {
                sum+=(s[1]-'A'+10);
            }
        else{
            sum+=(s[1]-'0');
        }
        ch[cnt]=sum;
        cnt++;
    }
    for(int i=1;i<=128;i++){
        for(int j=0;j<cnt;j++){
            char tmp=(ch[j]+i)%128;
            cout<<tmp;
        }
        cout<<endl;
    }
    cout<<endl;
    return 0;
}
```

虽然这个代码可以直接看到解题结果，但其实有一段是后面加的喔~如果自己实在写不出来参考了这个代码，要看懂它的意思喔！

Training: Crypto - Digraphs

题意是一个字母加密成两个字符。做题前线猜了一下，前三十个是单词Congratulations。从这个信息点开始扩充，手动推算整个文本信息（但我还是有几个大写字母不确定，不影响这题最后的答案）

每重新打开一次题目密文都会变…等找个时间写一下代码，大概思路是替换了已经确定的字母以后，按照英语单词，猜测其余的字母。小写全部能够确定以后把整个文本都替换出来，得到solution。

P.S.我觉得自己手工推的方法有点慢，有时间写下代码。

Training: MySQL I

遇到的第一道sql注入题！之前就学习过这个知识点，但是没有进行实操。让我们来看看源代码叭！（跃跃欲试）

首先点开作者贴心的放的高亮版本23333，抓住这句话：

$ query = “SELECT * FROM users WHERE username=‘

$username\ ANDpassword =$ password’”;

在username处，利用 # 将password的验证部分注释掉，就可以通过啦。

Training: MySQL II

仔细一想倒是跟数据库有很大关系，查了一下可以用UNION和SELECT来完成这次注入。主要思想是在输入用户名是向数据库加入一条我们自己写的数据，密码填加入的这条数据中设定的代码。就可以骗过验证。

123 ’ union select 1 , ‘admin’ , md5(‘password’) #

Guesswork

看见群里大家在讨论这题【误】不会写爆破…就各种猜：

Do not re-use important passwords!

I think you are not even a legit user, since you post news items :WEIRD:

P.S.有没有大佬告诉一下不这样抖机灵的解法。

No Escape

看这道题好像提交的人挺多的就顺手开始看了【大误】

题目说要求有一个人的票数达到111票，试了一下手工点可以计票，但是题目里说明设置了在票数到达100的时候会清空【我没手工去做，那样去验证也太烦了吧！】于是开始看代码，看一看关于计票的部分。

在这一行上可以做文章：

$ query = “UPDATE noescvotes SET `$who` = `$who` +1 WHERE id=1”;index.php?vote_for=bill `=111--`

Tip：一开始反单引号【`】打不对，一直在打单引号【‘】= =