

# vyctf-wp

ID:xuanbin

邮箱:[1499679521@qq.com](mailto:1499679521@qq.com)

## 嘘拟鸡

经典的虚拟机，构建了一个栈和一个栈顶指针，每个字符进行四次加密

第一次加密异或key"ch4ng\_t14o\_r4p\_d4\_l4n\_qiu\_zh1\_y1n!"

第二次加密异或一个由运算得来的key

第三次加密乘0x11，由于储存位数会产生溢出，一般做法是乘法逆元，偷懒逐个在最高位加一复原应该也行

第四次加密减0xB

```
#include<iostream>
using namespace std;
int main()
{
    unsigned char flag[] =
    {
        0x28, 0x5B, 0x34, 0x28, 0xC0, 0x24, 0x13, 0x46, 0xBA, 0xAF,
        0x6B, 0xC1, 0xED, 0xCF, 0x59, 0x7A, 0x98, 0x16, 0xC0, 0x65,
        0x26, 0xE2, 0x48, 0xCF, 0x6A, 0x24, 0x37, 0x5B, 0x66, 0xD2,
        0x14, 0x42, 0xFE, 0x10
    };
    unsigned char key1[]="ch4ng_t14o_r4p_d4_l4n_qiu_zh1_y1n!"; //第一步逆运算异或
    unsigned char key2[34]={0}; //第二步逆运算对key异或
    for(int i=0;i<34;i++)
    {
        int temp1=22;
        int temp2=i;
        while ( temp2 )
        {
            unsigned char v3 = temp2 & temp1;
            temp1 ^= temp2;
            temp2 = 2 * v3;
        }
        key2[i]=temp1;
    }

    for (int i=0;i<34;i++) //第四步逆运算
    {
        flag[i]+=0xB;
    }
    for(int i=0;i<34;i++) //第三步逆运算
    {
        int temp=flag[i];
        while(temp%17!=0)
        {
            temp+=256;
        }
    }
}
```

```
    }  
    flag[i]=temp/17;  
}  
for(int i=0;i<34;i++)  
{  
    flag[i]^=key2[i];  
}  
for(int i=0;i<34;i++)  
{  
    flag[i]^=key1[i];  
    cout<<flag[i];  
}  
//vyctf{Vm_ji_ni_tai_mei~~~Oh_baby!}  
}
```