XCTF php2

原创

记忆湛蓝天空 于 2019-05-13 20:29:53 发布 2760 收藏 5
文章标签： XCTF php

初次进入环境，出现如下图所示界面

Can you anthenticate to this website?

先在域名中输入index.php欲要查看主页面，结果依旧是上述界面，此时，将php改为phps，尝试查看php源码，如下图所示

not allowed!

"); exit(); } $_GET[id] = urldecode($_GET[id]); if($_GET[id] == "admin") { echo "

Access granted!

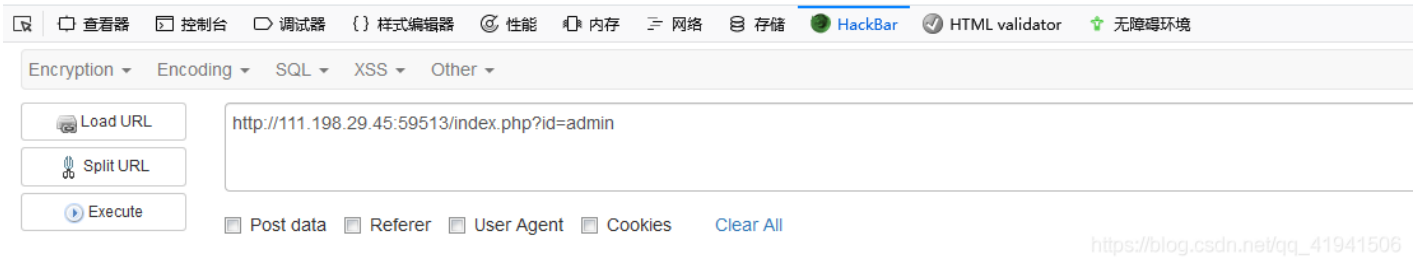
"; echo "

Key: xxxxxxx

"; } ?> Can you anthenticate to this website?

出现源码，代码审计，可以看出，get接受id=admin时，会出现key值，即可能出现flag，但是，当输入“?id=admin”时

not allowed!

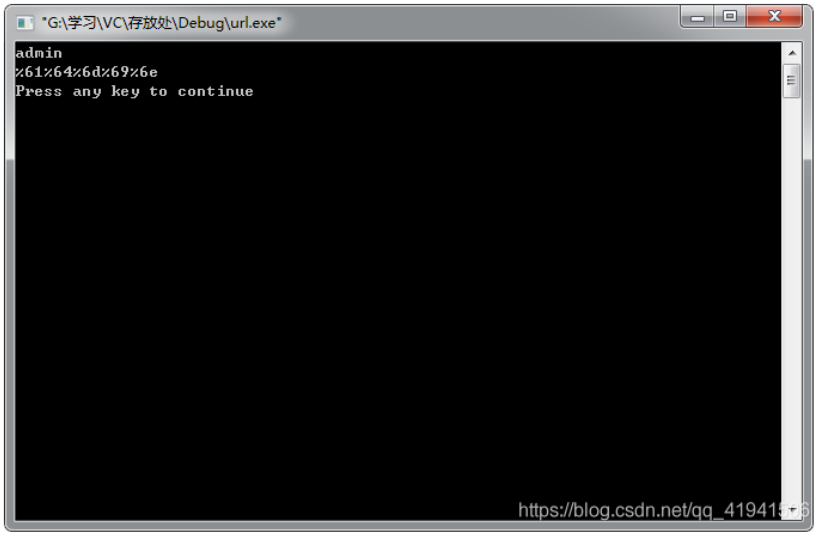


会提示错误，故此，想到可能admin 被设为了敏感字符，造成了只要识别到admin就会报错，因在url中提交，所以，我们使用url编码对“admin”进行编码，网上找到的URL编码均不能进行编译，故个人写了如下c++代码来执行编码

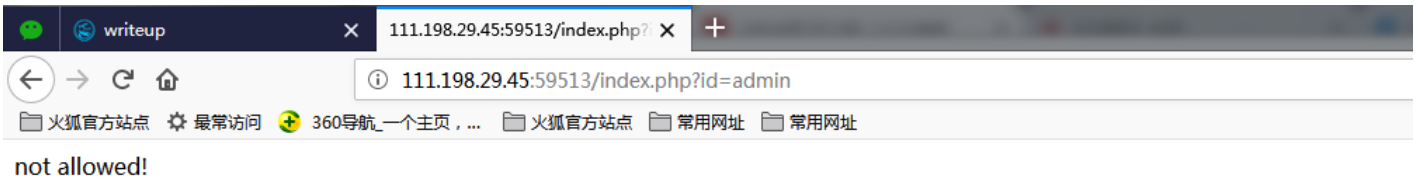
```
#include<iostream>
#include<string>
#include<string.h>
using namespace std;
void main()
{
    char a[27];
    string b[27]=
{"%61","%62","%63","%64","%65","%66","%67","%68","%69","%6a","%6b","%6c","%6d","%6e","%6f","%70","%7
    "%7a"};
    string c;
    string d[27];
    for(int i=0;i<26;i++)
    {
        a[i]=char(97+i);
    }
    cin>>c;
    for(i=0;i<c[i]!='\0';i++)
    {
        for(int j=0;j<26;j++)
        {
            if(c[i]==a[j])
                d[i]=b[j];
        }
    }
    for(i=0;i<c[i]!='\0';i++)
        cout<<d[i];
    cout<<endl;
}
```

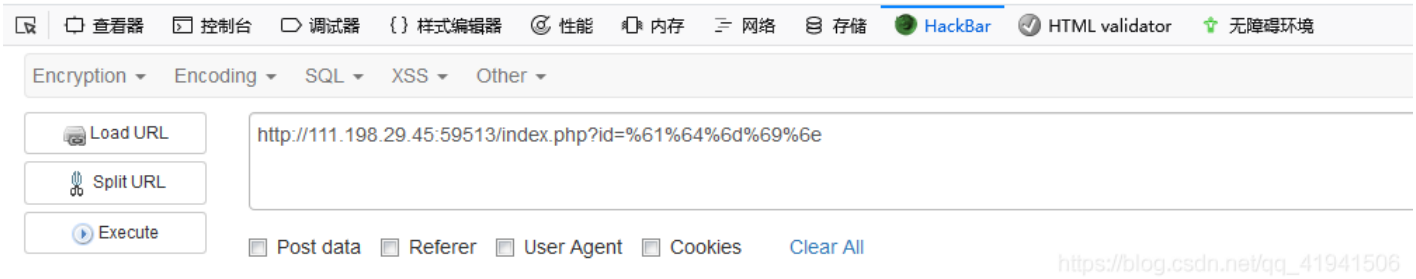
代码运行结果如下

```
#include<iostream>
#include<string>
#include<string.h>
using namespace std;
void main()
{
    char a[27];
    string b[27]={"%61","%62","%63","%64","%65","%66","%67","%68","%69","%6a","%6b","%6c","%6d","%6e","%6f","%70","%71","%72","%73","%74","%75","%7
    "%7a"};
    string c;
    string d[27];
    for(int i=0;i<26;i++)
    {
        a[i]=char(97+i);
    }
    cin>>c;
    for(i=0;i<c[i]!='\0';i++)
    {
        for(int j=0;j<26;j++)
        {
            if(c[i]==a[j])
                d[i]=b[j];
        }
    }
    for(i=0;i<c[i]!='\0';i++)
        cout<<d[i];
    cout<<endl;
}
```

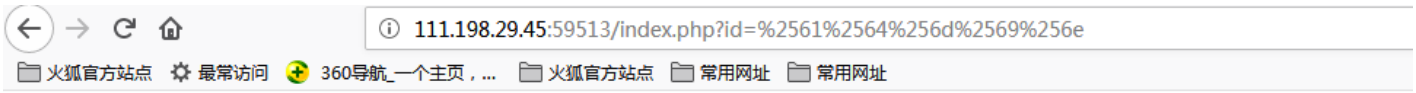


将编码结果放入到URL中，发现还是会报错，此处报错原因是因为PHP代码中的“urldecode”函数，该函数会对输入的ID进行解码，故而一次编码的admin无法绕过php代码的检测





不死心的我上网查询进行了二次URL编码，得到如下结果“%2561%2564%256d%2569%256e”，将二次编码过后admin放入到URL中，得到flag



Access granted!

Key: cyberpeace{1f555375442298e89bdedafdaf668cf7}

Can you anthenticate to this website?