CCTF重邮（绿盟）杯_web150

原创

焚卷觅光 于 2017-07-26 22:35:04 发布 738 收藏 2

分类专栏： CTF 代码审计 PHP 文章标签： php 代码审计 CCTF web150 writeup

本文链接：https://blog.csdn.net/wenliheng0/article/details/76167089


CTF 同时被 3 个专栏收录

5 篇文章 0 订阅

订阅专栏


代码审计

2 篇文章 0 订阅

订阅专栏


PHP

2 篇文章 0 订阅

订阅专栏

CCTF重邮（绿盟）杯_web150

本题是一个代码审计题,主要考察点还是PHP的弱验证。查看源代码可以在底部发现登陆规则。

```php
if(isset($_POST['login']))
 {
    if(isset($_POST['user']))
    {
        if(@strcmp($_POST['user'],$USER))//USER是被隐藏的复杂用户名
        {
            die('user错误! ');
        }
    }
    if (isset($_POST['name']) && isset($_POST['password']))
    {
        if ($_POST['name'] == $_POST['password'] )
        {
            die('账号密码不能一致! ');
        }
        if (md5($_POST['name']) === md5($_POST['password']))
        {
            if(is_numeric($_POST['id'])&&$_POST['id']!=='72' && !preg_match('/\s/', $_POST['id']))
            {
                    if($_POST['id']==72)
                        die("flag{xxxxxxxxxxxxx}");
                    else
                        die("ID错误2! ");
            }
            else
            {
                die("ID错误1! ");
            }
        }
        else
            die('账号密码错误! ');
    }
 }
```

0X00

第一个点的user提示是一个复杂用户名，然后它是用`@strcmp($_POST['user'],$USER)`判断的，所以构造时要构造`user[]`（字串），用burp暴力破解常见的用户名，得到`user[]=admin`。

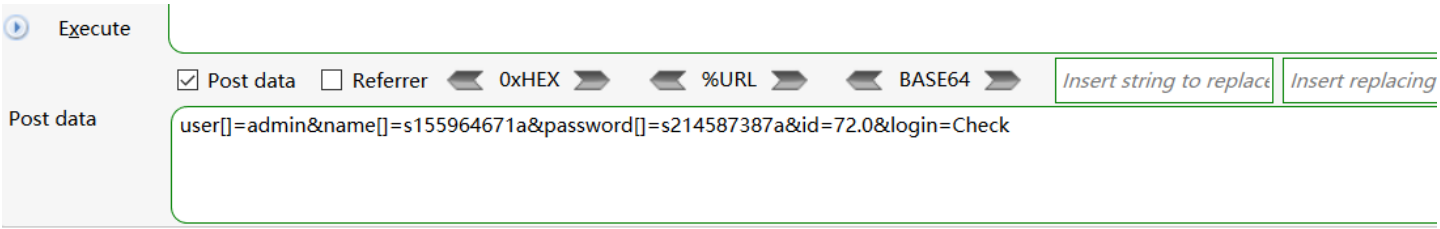
0X01

而下面要验证的是一个`name`不等于`password`，同时`name`和`password`的md5值要一致。
这里就要说一个PHP比较坑的地方了，***当你的字符串的MD5值是以0e开头的，PHP会默认他是int型的0，而0=0就是true。***
于是可以构造：`name[]=s155964671a&password[]=s214587387a`

0X01

最后一个地方要过的是，在之前作为字符串儿判断时ID不等于`'72'`。然后在下面判断数字形式的时候要等于`72`，这里考察的是一个PSP的弱验证。只需要构造`id=72.00`

最后payload及flag如图：



代码审计是网络安全的重要学科，不严格的代码很容易产生漏洞，包括不安全的函数、未经校验的参数输入、不安全的控件等为“地球最好的语言”，简单方便，也是弱类型语言。其存在大量的漏洞，在多次安全竞赛中不断出现。

学习资料，

工具：无。

用户：
账号：
密码：
ID：
Check

flag{9f6e6800cfae7749eb6c486619254b9c}