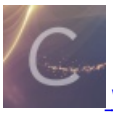
第四届全国网络空间安全技术大赛部分 Writeup

原创

Str3am 于 2018-10-21 16:19:31 发布 1256 收藏 1

分类专栏： Web CTF



Web1

首先是一个登录注册界面，注册登录进去后提示 flag 在 flag.php。

访问 flag.php 提示 `admin can get the flag`

发现一个修改密码的页面 change_passwd.php ，根据 cookie 中的 user 确定修改密码的用户，造成任意密码修改漏洞（大佬们还写了脚本更改 admin的密码，这道题还是一个拼手速的游戏？）。

修改 user 为 admin ，更改密码进入管理界面。

上传文件马上被删除，没什么用。利用点在远程图片地址，输入地址，服务器会访问输入的地址，必须以 `http:` 开头，得到的数据以 jpg 文件存到服务器上，没有对输入的地址进行任何限制，造成 ssrf 漏洞。

输入地址 `http://127.0.0.1/flag.php`

再访问图片得到 `flag{dbf6e52d69973dd16d87d4a8c3816ca9}`

比较坑的一点是最先尝试了地址的 ip ， `http://117.34.117.216/flag.php` ，仍然提示 `admin can get the flag` ，可能后台是通过判断访问 ip 来判断的。

Web2

进去首先是一个假的登录界面，并不会发送用户名和密码数据

用 dirsearch 扫描目录，发现 .git 文件泄漏

还原代码如下

主要在 upload.php

```
<?php
    function Administrator($value){
        if(empty($_COOKIE['in_adminid']) || empty($_COOKIE['in_adminexpire']) || $_COOKIE['in_adminexpire']!==md
5($_COOKIE['in_adminid'].$_COOKIE['in_adminname'].$_COOKIE['in_adminpassword'].$_COOKIE['in_permission'])){
            return False;
        }
        setcookie("in_adminexpire",$_COOKIE['in_adminexpire'],time()+1800);
        if(!empty($_COOKIE['in_permission'])){
            $array=explode(",",$_COOKIE['in_permission']);
            $adminlogined=false;
            for($i=0;$i<count($array);$i++){
                if($array[$i]==$value){$adminlogined=true;}
            }
            if(!$adminlogined){
                return False;
            }
        }else{
            return False;
        }
        return true;
    }
    if (Administrator(2)){
        if(isset($_FILES['file'])){
            $filename = './img/img'.rand().'.jpg';
            move_uploaded_file($_FILES["file"]["tmp_name"],$filename);
            header('Refresh:3,url=index.php?file=upload.php');
            echo "Upload $filename Success!";
            die;
        }
    }else{
        header('Refresh:3,url=index.php?file=login.html');
        echo "Who are you!";
        die;
    }
?>
<!DOCTYPE html>
<html lang="zh-CN">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
```

```
    <link rel="icon" href="../../favicon.ico">
    <title>图床后台</title>
    <link href="https://cdn.bootcss.com/bootstrap/3.3.7/css/bootstrap.min.css" rel="stylesheet">
    <link href="starter-template.css" rel="stylesheet">
  </head>
  <body>
    <script src="https://cdn.bootcss.com/jquery/1.12.4/jquery.min.js"></script>
    <script src="https://cdn.bootcss.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
    <form class="form-horizontal" action="upload.php" method="post" enctype="multipart/form-data">
    <fieldset>
      <div id="legend" class="">
        <legend class="">添加图片</legend>
      </div>
    <div class="control-group">
          <!-- Text input-->
          <label class="control-label" for="input01">图片名</label>
          <div class="controls">
            <input placeholder="请输入Message标题" class="input-xlarge" type="text" name="title">
          </div>
        </div>

    <div class="control-group">
          <label class="control-label">附件</label>
          <!-- File Upload -->
          <div class="controls">
            <input class="input-file" id="file" type="file" name='file'>
          </div>
        </div>



    <div class="control-group">
          <label class="control-label">预览</label>
          <!-- Button -->
          <div class="controls">
            <button class="btn btn-success">Submit</button>
          </div>
        </div>

    </fieldset>
  </form>
  </body>
</html>
```

首先检查 cookie，构造 cookie 如下

```
in_adminid 1
in_adminname admin
in_adminpassword admin
in_permission 2,1
in_adminexpire c6b0aa41bc44edc7c1e7dd4cc6ad4f9f
```

进去后是一个图片上传

上传完后给出图片的地址，跳转到 `http://117.34.116.192/index.php?file=login.html`，file 参数猜测是一个文件包含，试了试 css 里的内容，果然是一个 LFI

获取 shell，发现 f14g.php 是加密过的

github 代码解密一波

我的 C 刀下载文件会自动在开头加上 `work->`，很迷，半天才反应过来可能格式不对，QAQ

Misc1

binwalk 一波发现一个压缩包，含 stdgo.txt

打开是很多串 base64 编码的内容，尝试全部解码无果，py了一波提示，原来是 base64 隐写

原理是当编码后的字符存在 `=` 时，解码的时候会丢弃掉一部分最后一个非 `=` 的二进制位，一个 `=` 丢弃两个二进制位，就可以用丢弃的二进制位进行隐写

具体可以百度或者参考这篇文章：`https://www.tr0y.wang/2017/06/14/Base64steg/`

改了一下脚本,得到 `Ba5e_64OFive`：

```
def get_base64_diff_value(s1, s2):
    base64chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
    res = 0
    for i in xrange(len(s1)):
        if s1[i] != s2[i]:
            return abs(base64chars.index(s1[i]) - base64chars.index(s2[i]))
    return res

def solve_stego():

    with open('stego.txt', 'rb') as f:
        file_lines = f.readlines()

    bin_str = ''
    for line in file_lines:
        steg_line = line.replace('\n', '')
        norm_line = line.replace('\n', '').decode('base64').encode('base64').replace('\n', '')
        diff = get_base64_diff_value(steg_line, norm_line)

        pads_num = steg_line.count('=')
        if diff:
            bin_str += bin(diff)[2:].zfill(pads_num * 2)

        else:
            bin_str += '0' * pads_num * 2

    res_str = ''

    for i in xrange(0, len(bin_str), 8):

        res_str += chr(int(bin_str[i:i+8], 2))
    print res_str

solve_stego()
```