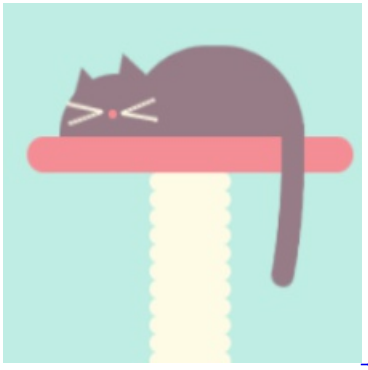
#每日一题 BUUCTF-[ACTF2020 新生赛]Include

原创

WHU_zhe 于 2021-04-14 00:33:08 发布 65 收藏

分类专栏： # BUUCTF CTF 文章标签： php filter 安全漏洞

本文链接：https://blog.csdn.net/yu748913833/article/details/115683384



BUUCTF 同时被 2 个专栏收录

2 篇文章 0 订阅

订阅专栏

CTF

2 篇文章 0 订阅

订阅专栏

BUUCTF-[ACTF2020 新生赛]Include

基础知识：`php://filter`

`php://filter`可以作为一个**中间流**来处理其他流，具有四个参数：

名称	描述	备注
resource=<要过滤的数据流>	指定了你要筛选过滤的数据流	必选
read=<读链的筛选列表>	可以设定一个或多个过滤器名称，以管道符（|）分隔	可选
write=<写链的筛选列表>	可以设定一个或多个过滤器名称，以管道符（|）分隔	可选
<；两个链的筛选列表>	任何没有以read=或write=作前缀的筛选器列表会视情况应用于读或写链	

如下方代码所示：

```php
<?php
#这里没有指定过滤器
readfile("php://filter/resource=example.etc");
?>
```

以上的代码将会通过中间流输出对应的`example.etc`文件的内容

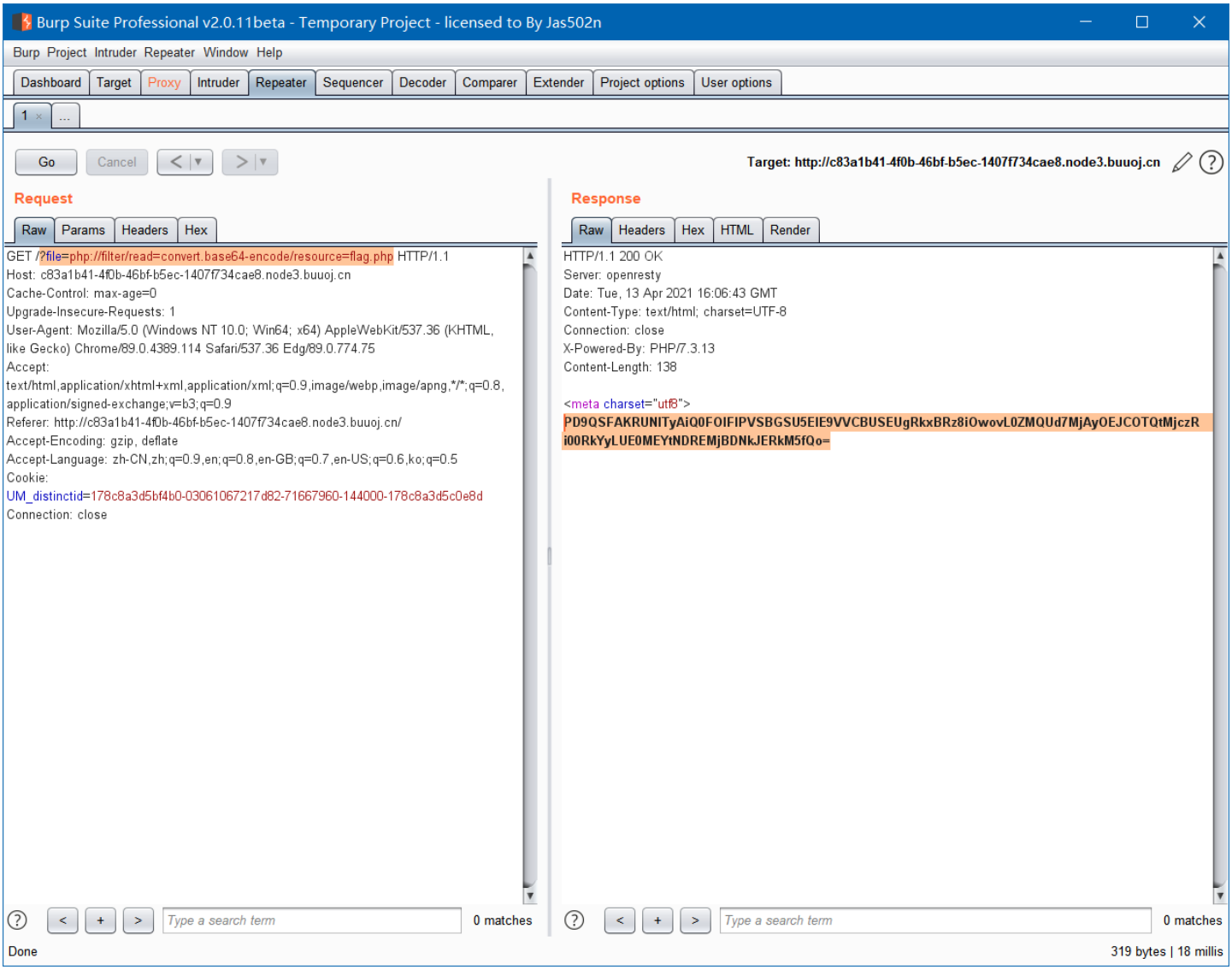
```php
<?php
readfile("php://filter/read=convert.base64-encode/resource=example.etc");
readfile("php://filter/read=string.toupper/resource=example.etc");
readfile("php://filter/read=string.toupper|string.rot13/resource=example.etc");
?>
```

以上三行代码分别会实现这些功能：（1）将对应的文件进行base64编码之后输出（2）将对应的文件转换为大写后输出（3）将对应的文件转换为大写并且使用rot13编码后输出

回到问题本身，我们的目的是读取`flag.php`的源码，但是在url中直接对该文件进行访问无法得到结果，估计是后台将该文件的源码进行了执行，而并非将文件源码输出

同时文件名的提交方式为get，所以我们猜测后台中有文件包含漏洞，需要想办法得到对应文件的源码，所以采用`php://filter`，将文件`flag.php`进行base64编码之后输出，提交的payload为：

`?file=php://filter/read=convert.base64-encode/resource=flag.php`，所得到的返回值如下图所示：



将该返回值进行base64解码得到flag

```
<?php
echo "Can you find out the flag?";
//flag{2028bb94-273f-4ff2-a40f-44d20c6bdfc9}
```

查看源码发现的确是对这个`flag.php`进行了执行，但是我们需要的flag却是在注释之中