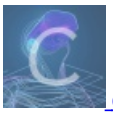
攻防世界-web writeup（xctf）

原创

影子019 于 2019-07-15 15:13:20 发布 1516 收藏 1

分类专栏： ctf_web 文章标签： web ctf xctf 基础题

本文链接：https://blog.csdn.net/qinying001/article/details/95943143



0x01 view source

flag放在源码之中，但是网页的脚本限制了鼠标作用，无法点击和选中，直接F12或者浏览器快捷键查看网页源码即可得到flag

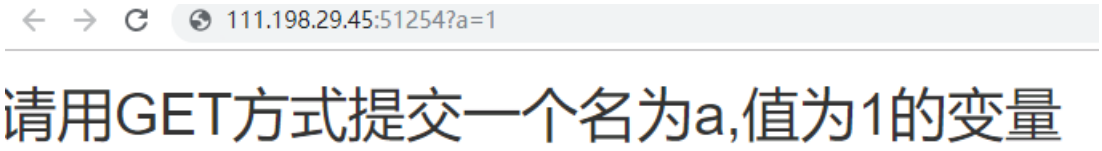
```
▼<script>
    document.oncontextmenu=new Function("return false")
    document.onselectstart=new Function("return false")
  </script>
```

flag：cyberpeace{e07dcafaeeb31df23b4d661dd4da56f9}

0x02 get_post

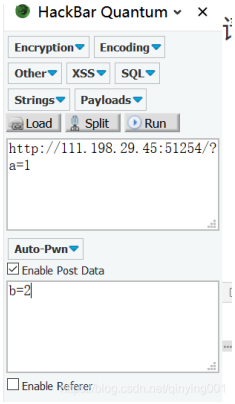
GET和POST是http协议的两种主要请求方式

GET请求直接把参数包含在url中



POST请求在网页没有输入的情况下，利用工具提交，如hackerbar

请用GET方式提交一个名为a,值为1的变量
请再以POST方式随便提交一个名为b,值为2的变量



flag：cyberpeace{c4e43c9c9d0f729358dd9417219a9da0}

0x03 robots

robots协议会在网站主目录产生一个robots.txt文件，打开文件，看到flag所在位置，访问flag_is_h3re.php获取flag

flag：cyberpeace{1b59446bc8e566382e01b0c209b899bd}

0x04 backup

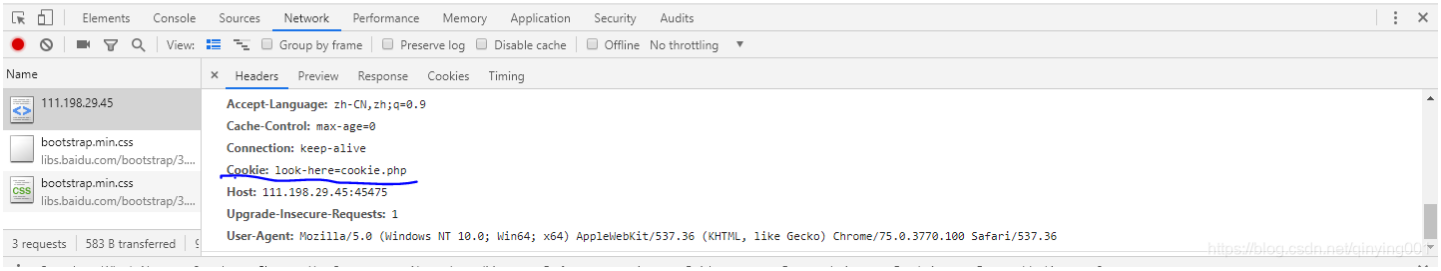
备份文件分三种
1.编辑器自动备份
2.版本控制系统备份
3.开发者主动备份
知道了主页的备份文件index.php.bak，访问得到文件，删除bak后缀再用浏览器打开php文件得到flag

flag: cyberpeace{4376485b1a095581d7fb57b8ab3bb924}

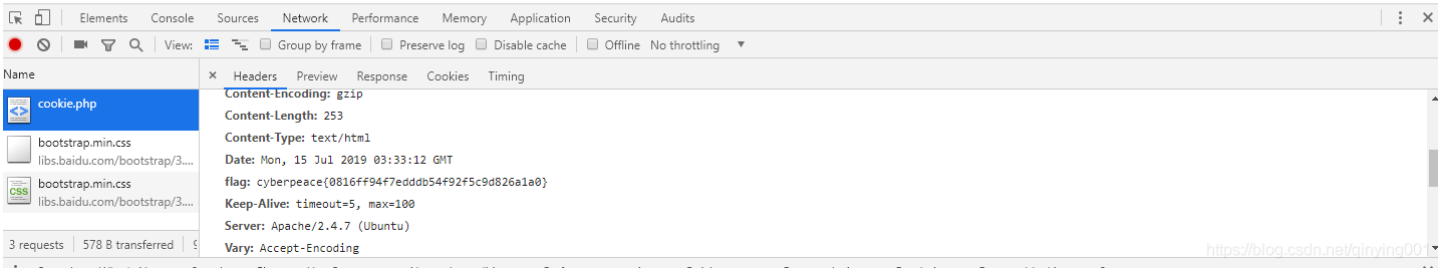
0x05 cookie

使用浏览器控制台找到cookie，按照其中信息访问cookie.php，得到flag

你知道什么是cookie吗?



See the http response



flag: cyberpeace{0816ff94f7edddb54f92f5c9d826a1a0}

0x06 disabled_button

修改前端代码，将按钮的disable属性去掉就可以点击了

flag: cyberpeace{de3c3c35166596311b23137ae6f3d33a}

0x07 simple_js

查看网页源代码

```
<html>
<head>
    <title>JS</title>
    <script type="text/javascript">
    function dechiffre(pass_enc){
        var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
        var tab  = pass_enc.split(',');
                var tab2 = pass.split(',');var i,j,k,l=0,m,n,o,p = "";i = 0;j = tab.length;
                        k = j + (l) + (n=0);
                        n = tab2.length;
                        for(i = (o=0); i < (k = j = n); i++ ){o = tab[i-l];p += String.fromCharCode((o = tab2[i]));
                                if(i == 5)break;}
                        for(i = (o=0); i < (k = j = n); i++ ){
                        o = tab[i-l];
                                if(i > 5 && i < k-1)
                                        p += String.fromCharCode((o = tab2[i]));
                        }
        p += String.fromCharCode(tab2[17]);
        pass = p;return pass;
    }
    String["fromCharCode"](dechiffre("\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30"));

    h = window.prompt('Enter password');
    alert( dechiffre(h) );

</script>
</head>

</html>
```


将fromCharCode中的16进制转为ascii字符，

55,56,54,79,115,69,114,116,107,49,50 ，但是输入之后还是不对，再细看源码，发现循环没有用到tab，也就是

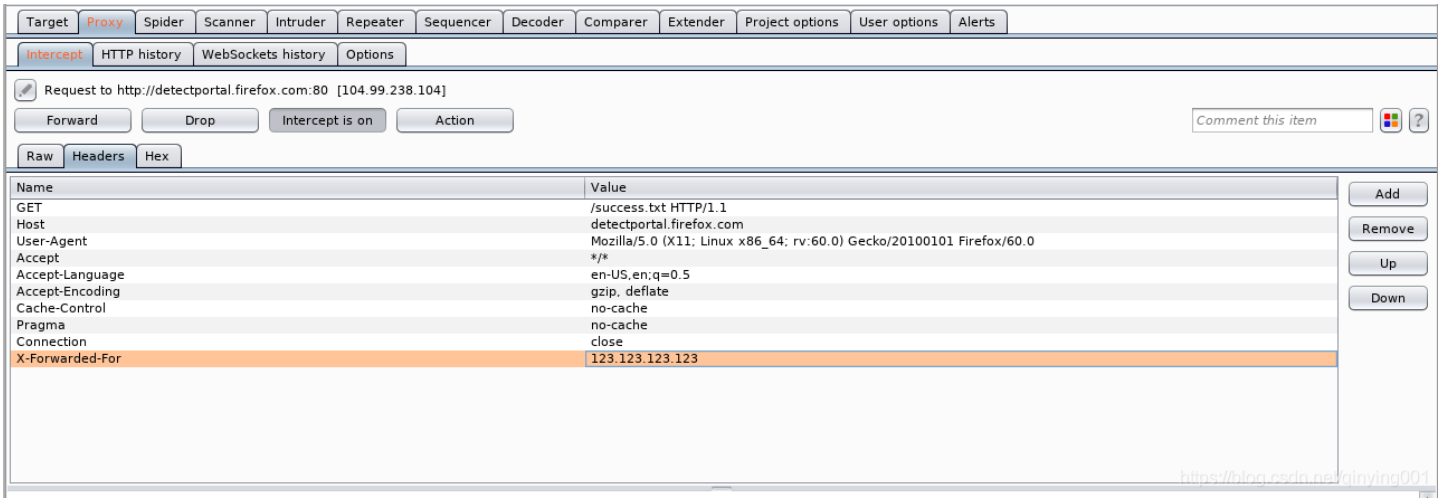
```
<html>
<head>
   <title>JS</title>
   <script type="text/javascript">
   function dechiffre(pass_enc){
      var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
      var tab  = pass_enc.split(',');
            var tab2 = pass.split(',');var i,j,k,l=0,m,n,o,p = "";i = 0;j = tab.length;
                  k = j + (l) + (n=0);
                  n = tab2.length;
                  for(i = (o=0); i < (k = j = n); i++ ){
                   o = tab[i-l];
                   //tab2改为tab
                   p += String.fromCharCode((o = tab[i]));
                     if(i == 5)break;
                     }
                  for(i = (o=0); i < (k = j = n); i++ ){
                  o = tab[i-l];
                        if(i > 5 && i < k-1)
                          //tab2改为tab
                              p += String.fromCharCode((o = tab[i]));
                  }
      p += String.fromCharCode(tab2[17]);
      pass = p;return pass;
   }
   //直接弹出答案
   var s = "\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\
x39\x2c\x35\x30";

   //h = window.prompt('Enter password');
   alert( dechiffre(s) );
</script>
</head>

</html>
```

或者直接写脚本将字符串中的十六进制转为字符

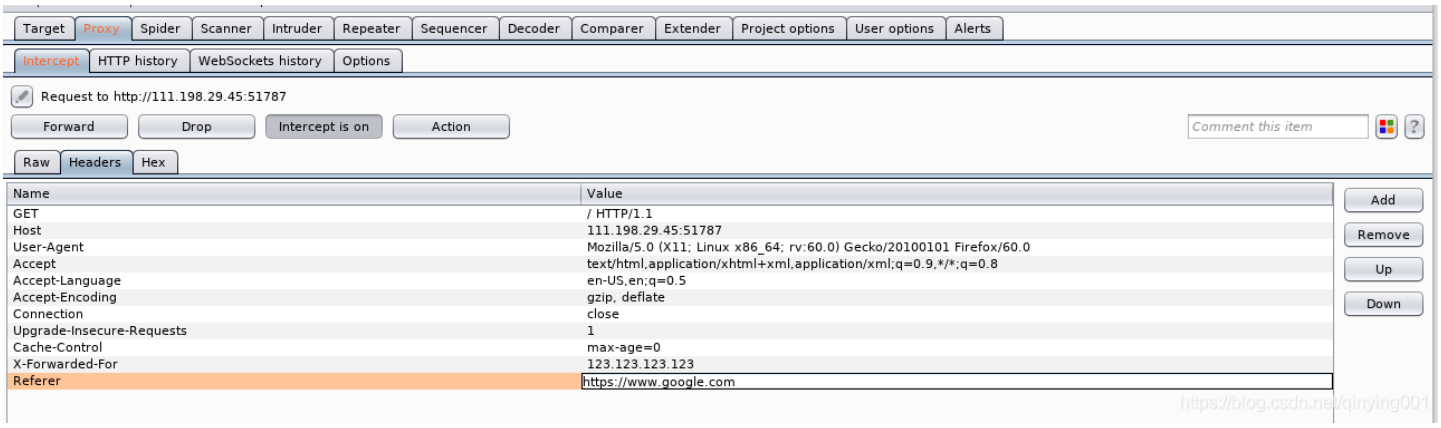
flag：cyberpeace{786OsErtk12}

0x08 xff_referer

打开网页后第一个要求是IP地址必须为123.123.123.123，根据http协议，其头字段的X-Forwarded-For字段是用来判别最原始的来源ip，即用burpsuit抓取http请求包，在头字段添加X-Forwarded-For字段就可以伪造来源ip



提交之后反回一个页面，要求必须来自https//:www.google.com。
http请求头中Referer字段便是用来告诉服务器该网页是从哪个页面链接过来，即需要在请求头中再加入一个Referer字段



提交得到flag

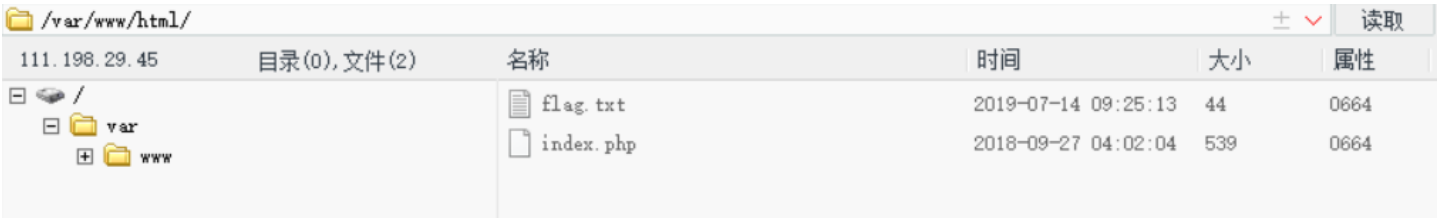
flag: cyberpeace{17bb357d42b6151576a75f2ef3089cdb}

0x09 webshell

网页描述的是一个一句话木马的webshell，根据题意，需要连接这个webshell

利用菜刀连接

找到flag



flag：cyberpeace{8333cbdfb5aa36d1238a005b1241a6cb}

0x0A command_execution

一个web程序，调用系统的ping命令，限制参数数量为3，题目描述之中书名这个程序没有写waf，没有过滤参数，就可以考虑用linux管道执行命令寻找flag

& find / 列出系统中所有文件

```
ping -c 3 & find /
/
/boot
/run
/run/sendsigs.omit.d
/run/utmp
/run/network
/run/resolvconf
/run/resolvconf/resolv.conf
/run/resolvconf/interface
/run/resolvconf/interface/original.resolvconf
/run/lock
/run/lock/apache2
/run/motd.dynamic
```


在网页页面中找到flag所在的目录，用cat读取文件

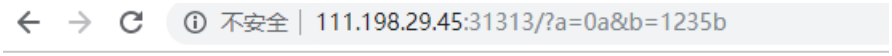
```
ping -c 3 & cat /home/flag.txt
cyberpeace{25b31692838e8403383f9916e03e0705}
```

flag: cyberpeace{25b31692838e8403383f9916e03e0705}

0x0B simple_php

```php
<?php
show_source(__FILE__);
include("config.php");
$a=@$_GET['a'];
$b=@$_GET['b'];
if($a==0 and $a){
        echo $flag1;
}
if(is_numeric($b)){
        exit();
}
if($b>1234){
        echo $flag2;
}
?>
```

根据网页代码，需要通过url接受两个参数，代码中的判断条件比较有意思，第一个是要求参数a为0的同时又不能为0，第二个条件则是要求b不能为数字，但又得大于1234。
利用php变量的特性，可以利用数字后面添加字符满足条件获取flag。




```php
<?php
show_source(__FILE__);
include("config.php");
$a=@$_GET['a'];
$b=@$_GET['b'];
if($a==0 and $a){
        echo $flag1;
}
if(is_numeric($b)){
        exit();
}
if($b>1234){
        echo $flag2;
}
?>
```

Cyberpeace{647E37C7627CC3E4019EC69324F66C7C}

flag: Cyberpeace{647E37C7627CC3E4019EC69324F66C7C}

0x0C weak_auth

根据提示，利用admin登录，burpsuit暴力破解，得到密码登录。