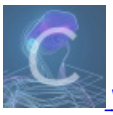
任意文件读取与下载漏洞

原创

Qwzf 于 2019-11-14 22:52:37 发布 12916 收藏 47

分类专栏： Web Web常见漏洞 文章标签： Web

本文链接：https://blog.csdn.net/qq_43625917/article/details/102617154



本文首发在先知社区

0x00前言

上周参加了一个线上赛。有个Web题的WriteUp说是任意文件下载。由于之前没学过，所以就没有想到。现在学习一下

0x01为什么产生任意文件读取与下载漏洞

一些网站的业务需要，可能提供文件查看或下载的功能，如果对用户查看或下载的文件不做限制，就能够查看或下载任意的文件，可以是源文件，敏感文件等等。

0x02任意文件读取漏洞

任意文件读取是属于文件操作漏洞的一种，一般任意文件读取漏洞可以读取配置信息甚至系统重要文件。严重的话，就可能导致SSRF，进而漫游至内网。

漏洞产生原因

- 存读取文件的函数
- 读取文件的路径用户可控，且未校验或校验不严
- 输出了文件内容

任意文件读取

```
<?php
$filename=”test.txt”;
readfile($filename);
?>
```

```
<?php
$filename=”test.txt”;
echo file_get_contents($filename);
?>
```

文件读取函数

`readfile()`、`file_get_contents()`、`fopen()`中，`$filename`没有经过校验或者校验不合格，用户可控制变量读取任意文件，如`/etc/passwd`、`./index.php`、`/config.ini`。

0x03任意文件下载漏洞

一些网站由于业务需求，往往需要提供文件下载功能，但若对用户下载的文件不做限制，则恶意用户就能够下载任意敏感文件，这就是文件下载漏洞。

漏洞产生原因

- 有读取文件的函数
- 读物文件的路径用户可控，且没有经过校验，或者校验不严格
- 输出文件内容
- 一个正常的网站，存在一个下载文件的功能，同时还会从浏览器接收文件名字

文件下载的两种方式

1、直接下载：

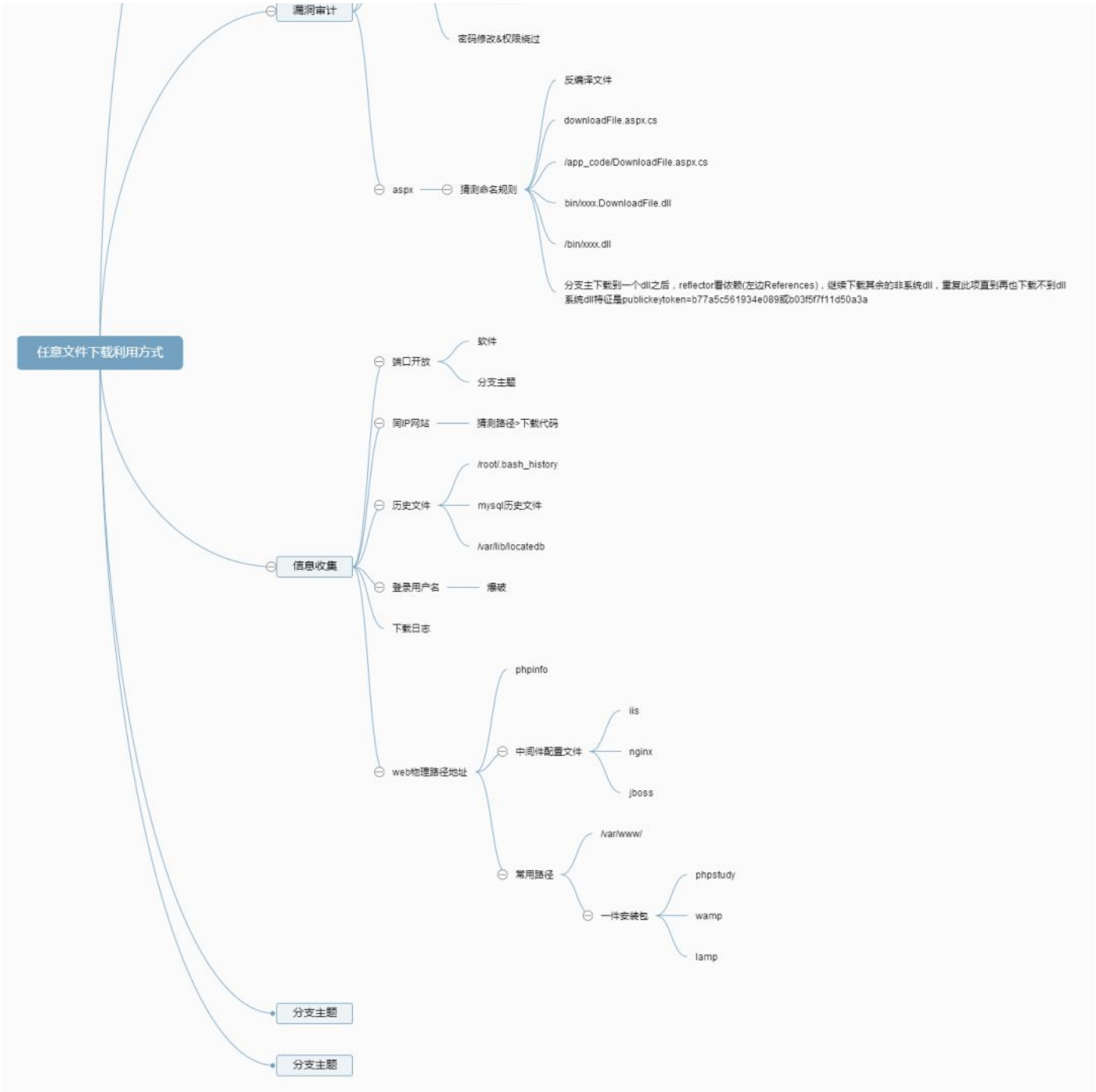
```
<a href="http://www.a.com/xxx.rar">下载</a>
```

2、增加header头

```
<?php
    $filename = $_GET['filename'];
    echo '<h1>讲开始下载文件! </h1><br /><br />';
    echo file_get_contents($filename);

    header('Content-Type: imgage/jpeg');
    header('Content-Disposition: attachment; filename='.$filename);
    header('Content-Lengh: '.filesize($filename));
?>
```

漏洞利用方式



利用思路

- 下载常规的配置文件，例如: ssh,weblogic,ftp,mysql等相关配置
- 下载各种`.log`文件，从中寻找一些后台地址，文件上传点之类的地方，如果运气好的话会获得一些前辈们的后门。
- 下载web业务文件进行白盒审计，利用漏洞进一步攻入服务器。

尝试读取`/root/.bash_history`看自己是否具有root权限。
如果没有，就只能利用`../`来回跳转读取一些`.ssh`下的配置信息文件。
读取mysql下的`.bash_history`文件。来查看是否记录了一些可以利用的相关信息。然后逐个下载需要审计的代码文件，但是下载的时候变得很繁琐，只能尝试去猜解目录，然后下载一些中间件的记录日志进行分析。

一些常见利用方式

java+oracle环境

可以先下载/WEB-INF/classes/applicationContext.xml 文件，这里面记载的是web服务器的相应配置，然后下载/WEB-INF/classes/xxx/xxx/ccc.class对文件进行反编译，然后搜索文件中的upload关键字看是否存在一些api接口，如果存在的话我们可以本地构造上传页面用api接口将我们的文件传输进服务器。

也可以先下载网站的配置文件，在根目录/WEB-INF/Web.xml的(一般都有很多内容,有时含有数据库连接用户名和密码等关键信息)。

具有root权限

在linux中有这样一个命令 locate 是用来查找文件或目录的，它不搜索具体目录，而是搜索一个数据库/var/lib/mlocate/mlocate.db。这个数据库中含有本地所有文件信息。Linux系统自动创建这个数据库，并且每天自动更新一次。

当我们不知道路径是什么的情况下，这个可以说是一个核武器了，我们利用任意文件下载漏洞将mlocate.db文件下载下来，利用locate命令将数据输出成文件，这里面包含了全部的文件路径信息。

locate 读取方法

```
locate mlocate.db admin
```

可以将`mlocate.db`中包含`admin`内容全部输出来。

```
/var/lib/mysql/phpmyadmin/pma__tracking.ibd
/var/lib/mysql/phpmyadmin/pma__userconfig.frm
/var/lib/mysql/phpmyadmin/pma__userconfig.ibd
/var/lib/mysql/phpmyadmin/pma__usergroups.frm
/var/lib/mysql/phpmyadmin/pma__usergroups.ibd
/var/lib/mysql/phpmyadmin/pma__users.frm
/var/lib/mysql/phpmyadmin/pma__users.ibd
/var/lib/phpmyadmin/blowfish_secret.inc.php
/var/lib/phpmyadmin/config.inc.php
/var/lib/phpmyadmin/tmp
/var/lib/ucf/cache/:etc:dbconfig-common:phpmyadmin.conf
/var/tmp/phpmyadmin.phpmyadmin.2019-07-21-16.25.mysql.qpjpN7
/var/www/html/phpmyadmin
/var/www/html/BWVS/admin
/var/www/html/BWVS/admin/delCom.php
/var/www/html/BWVS/admin/delUser.php
/var/www/html/BWVS/admin/index.php
/var/www/html/BWVS/admin/logCheck.php
/var/www/html/BWVS/admin/login.php
/var/www/html/BWVS/admin/manage.php
/var/www/html/BWVS/admin/manageCom.php
/var/www/html/BWVS/admin/manageUser.php
/var/www/html/DVWA/hackable/users/admin.jpg
/var/www/html/bbs/admin
```

利用这个文件可以获取到该服务器任何我们想要的内容并下载出来而不用一个一个去猜解目录，但是这个文件只有root用户才能读取。另一方面我们也可以利用linux内核的一个文件`/proc/self/cmdline`当前进程的`cmdline`参数，可以获取到路径信息。

总的来说，任意文件下载漏洞的利用主要是为了信息收集，我们通过对服务器配置文件的下载，获取到大量的配置信息、源码，从而根据获取的信息来进一步挖掘服务器漏洞从而入侵。

0x04任意文件读取与下载漏洞挖掘

> 1、web漏洞扫描器（awvs、appscan、openvas、nessus）
> 2、手动挖掘从连接和参数名查看

Google search

```
inurl:”readfile.php?file=
inurl:”read.php?filename=
inurl:”download.php?file=
inurl:”down.php?file=
```

连接：

`readfile.php?file=**.txt`

`download.php?file=**.rar`

参数名：

`&RealPath=`、`&readpath=`、`&FilePath=`、`&filepath=`、`&Path=`、`&path=`、`&Inputfile=`、`&inputfile=`、`&url=`、`&urls=`、`&Lang=`、`&dis=`、`&Data=`、`&data=`、`&readfile=`、`&filep=`、`&Src=`、`&src=`、`&menu=`、`META-INF=`、`WEB-INF`

0x05敏感信息

Windows：

> C:\boot.ini //查看系统版本
> C:\Windows\System32\inetsrv\MetaBase.xml //IIS配置文件
> C:\Windows\repair\sam //存储系统初次安装的密码
> C:\Program Files\mysql\my.ini //Mysql配置
> C:\Program Files\mysql\data\mysql\user.MYD //Mysql root
> C:\Windows\php.ini //php配置信息
> C:\Windows\my.ini //Mysql配置信息

Linux：

> /root/.ssh/authorized_keys //如需登录到远程主机，需要到.ssh目录下，新建authorized_keys文件，并将id_rsa.pub内容复制进去
> /root/.ssh/id_rsa //ssh私钥,ssh公钥是id_rsa.pub
> /root/.ssh/id_ras.keystore //记录每个访问计算机用户的公钥
> /root/.ssh/known_hosts
> //ssh会把每个访问过计算机的公钥(public key)都记录在~/.ssh/known_hosts。当下次访问相同计算机时，OpenSSH会核对公钥。如果公钥不同，OpenSSH会发出警告， 避免你受到DNS Hijack之类的攻击。
> /etc/passwd // 账户信息
> /etc/shadow // 账户密码文件
> /etc/my.cnf //mysql 配置文件
> /etc/httpd/conf/httpd.conf // Apache配置文件
> /root/.bash_history //用户历史命令记录文件
> /root/.mysql_history //mysql历史命令记录文件
> /proc/self/fd/fd[0-9]*(文件标识符)
> /proc/mounts //记录系统挂载设备
> /porc/config.gz //内核配置文件
> /var/lib/mlocate/mlocate.db //全文件路径
> /porc/self/cmdline //当前进程的cmdline参数

0x06任意文件读取与下载漏洞验证

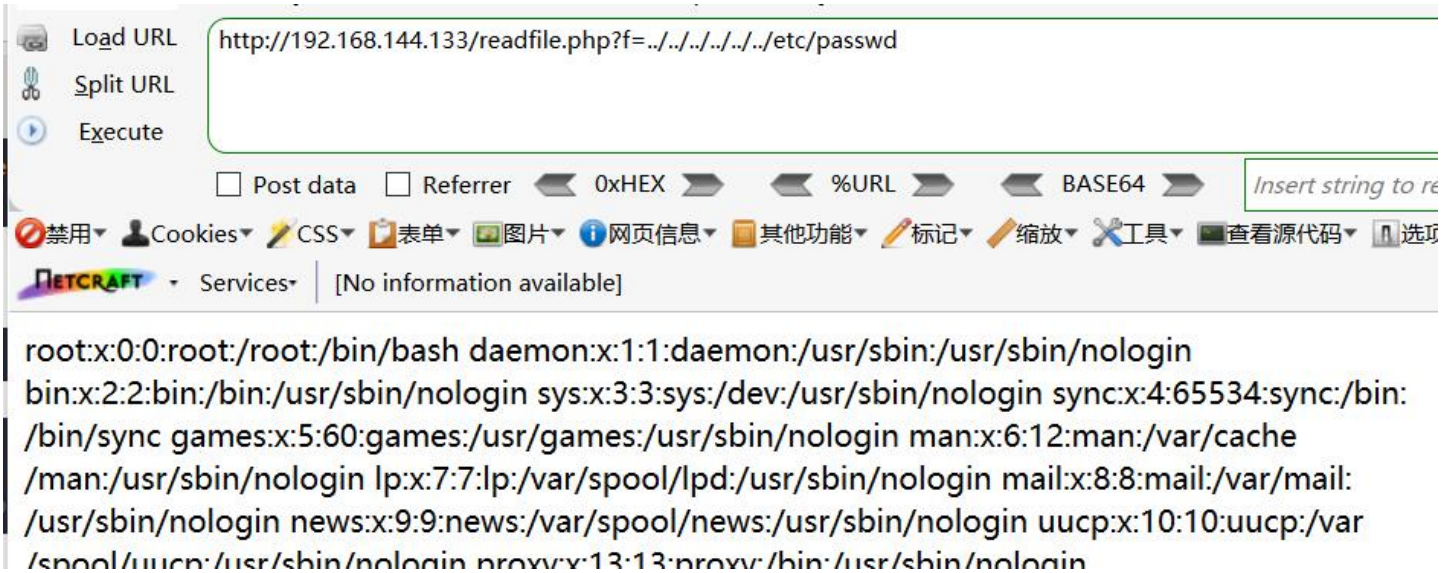
任意文件读取验证

示例代码：

```php
<?php
$filename=$_GET['f'];
echo file_get_contents($filename);
?>
```

测试：

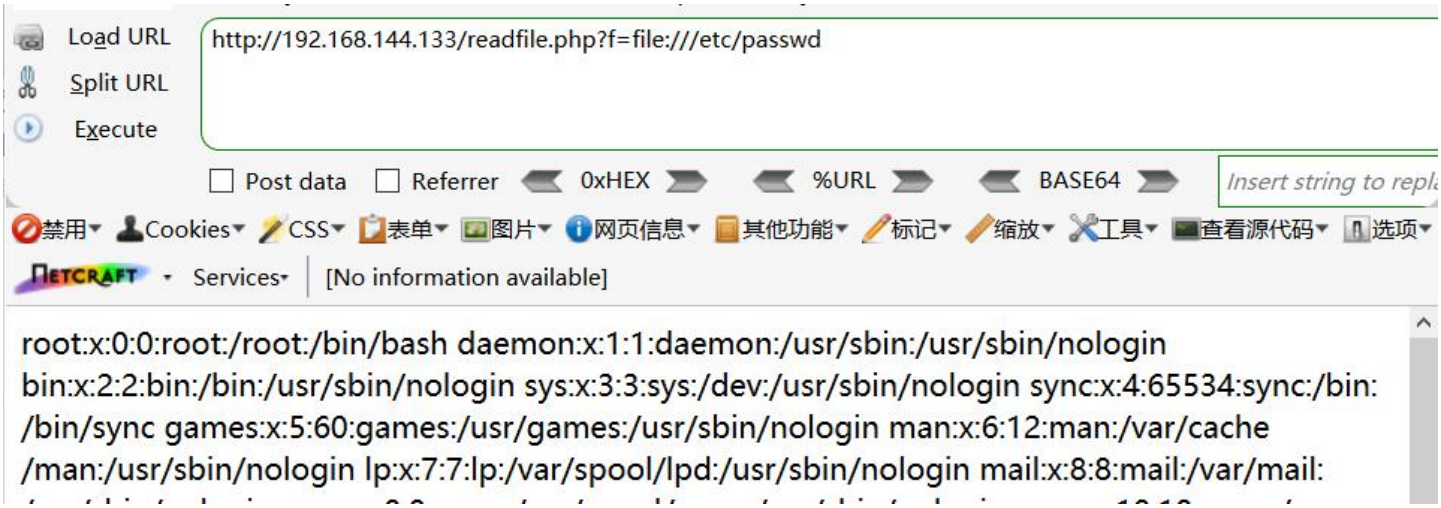
```
readfile.php?f=../../../../../../etc/passwd
readfile.php?file=../../../../../../../../../etc/passwd%00
```




```
readfile.php?f=../index.txt
```

`file://`伪协议，读取文件内容

```
readfile.php?f=file:///etc/passwd
```

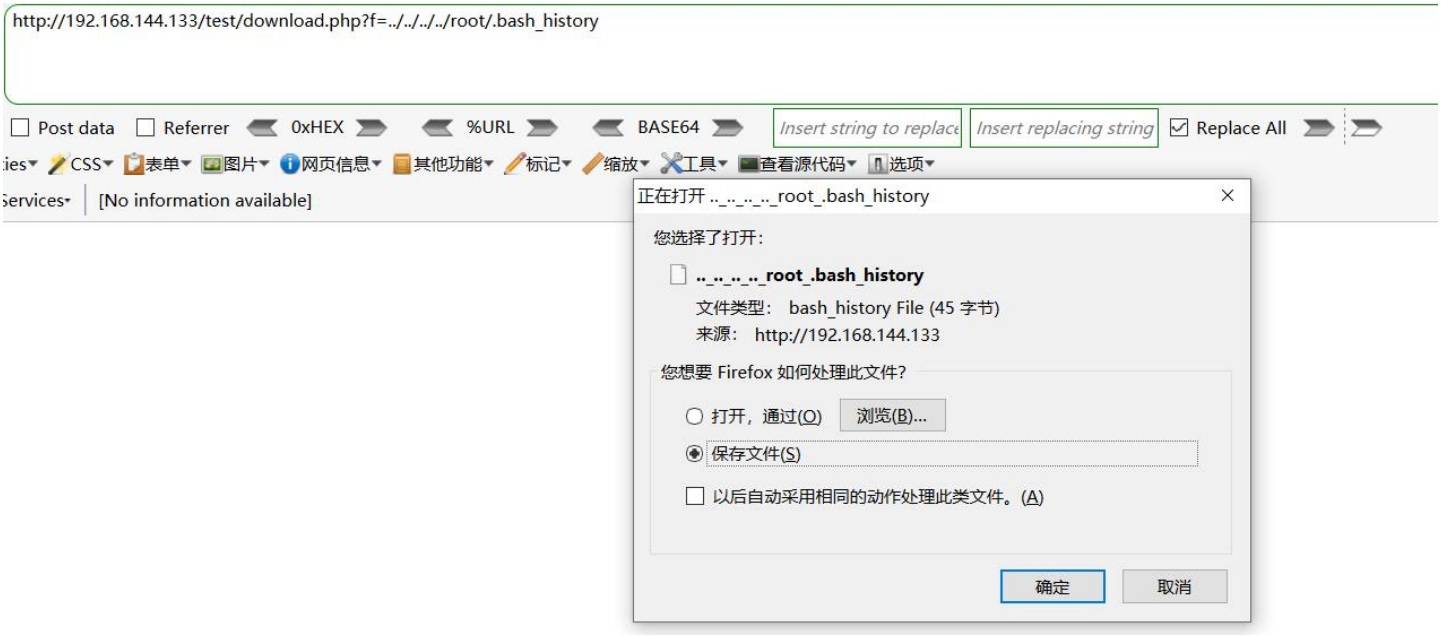


任意文件下载验证

示例代码：

```php
<?php
$filename = $_GET['f'];
echo '<h1>讲开始下载文件! </h1><br /><br />';
echo file_get_contents($filename);

header('Content-Type: imgage/jpeg');
header('Content-Disposition: attachment; filename='.$filename);
header('Content-Lengh: '.filesize($filename));
?>
```



当然，我下载这个文件并没有内容。

0x07漏洞判断

参数`f`的参数值为PHP文件时：

> 1.文件被解析，则是文件包含漏洞
> 2.显示源代码，则是文件查看漏洞
> 3.提示下载，则是文件下载漏洞

0x08漏洞防御修复

通用

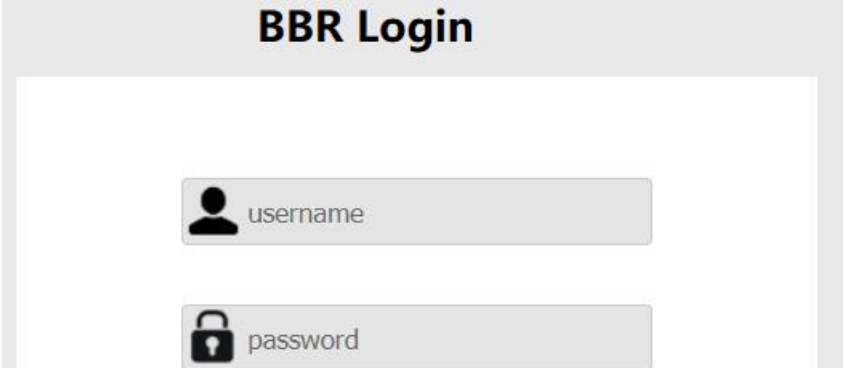
- 过滤 `.` 点，使用户在url中不能回溯上级目录
- 正则严格判断用户输入的参数
- `php.ini` 配置 `open_basedir` 限定文件访问范围

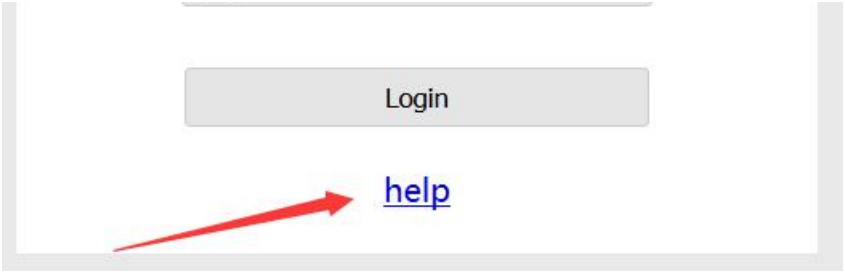
文件下载漏洞修复

- 将下载区独立出来，放在项目路径外，给每个下载资源固定的URL，而不是所有的下载资源都是统一的URL：`http://www.test.com/download?filename=文件名`
- 净化数据：对用户传过来的文件名参数进行硬编码或统一编码，对文件类型进行白名单控制，对包含恶意字符或者空字符的参数进行拒绝。
- web应用程序可以使用chroot环境包含被访问的web目录，或者使用绝对路径+参数来访问文件目录，时使其即使越权也在访问目录之内。www目录就是一个chroot应用。由chroot创造出的那个根目录，叫做“chroot监狱”(所谓”监狱”就是指通过chroot机制来更改某个进程所能看到的根目录，即将某进程限制在指定目录中，保证该进程只能对该目录及其子目录的文件有所动作，从而保证整个服务器的安全。
  详细具体chroot的用法，可参考：`http://blog.csdn.net/frozen_fish/article/details/2244870`
- 任意文件下载漏洞也有可能是web所采用的中间件的版本低而导致问题的产生，例如ibm的websphere的任意文件下载漏洞，需更新其中间件的版本可修复。
- 要下载的文件地址保存至数据库中。
- 文件路径保存至数据库，让用户提交文件对应ID下载文件。
- 用户下载文件之前需要进行权限判断。
- 文件放在web无法直接访问的目录下。
- 不允许提供目录遍历服务。
- 公开文件可放置在web应用程序下载目录中通过链接进行下载。
- 记录文件下载日志。

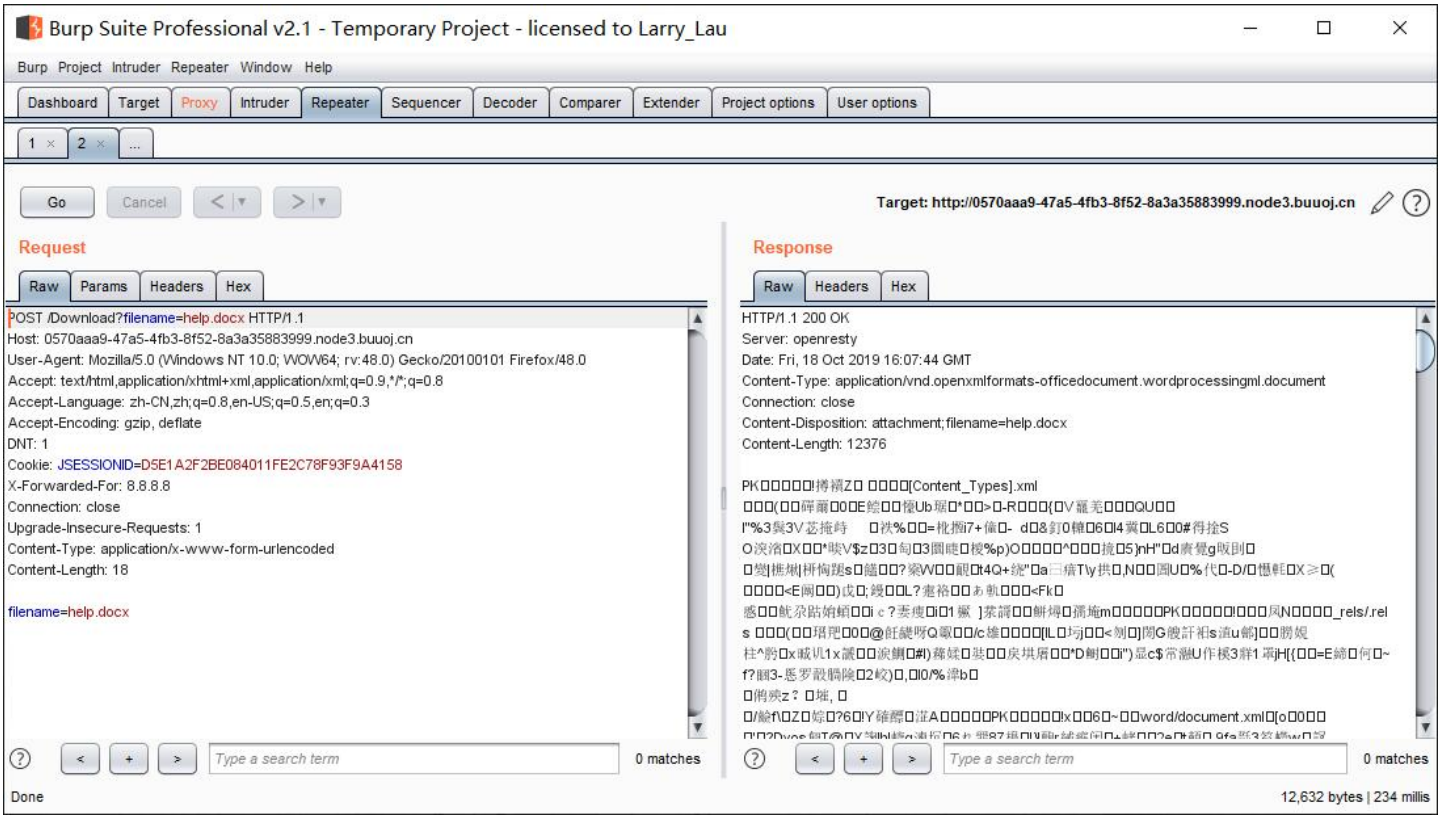
0x09漏洞利用实战

我学习任意文件读取与下载漏洞，就是因为遇到了一个任意文件读取与下载漏洞的Web题，所以在此实战一下
RoarCTF2019-Web：Easy Java

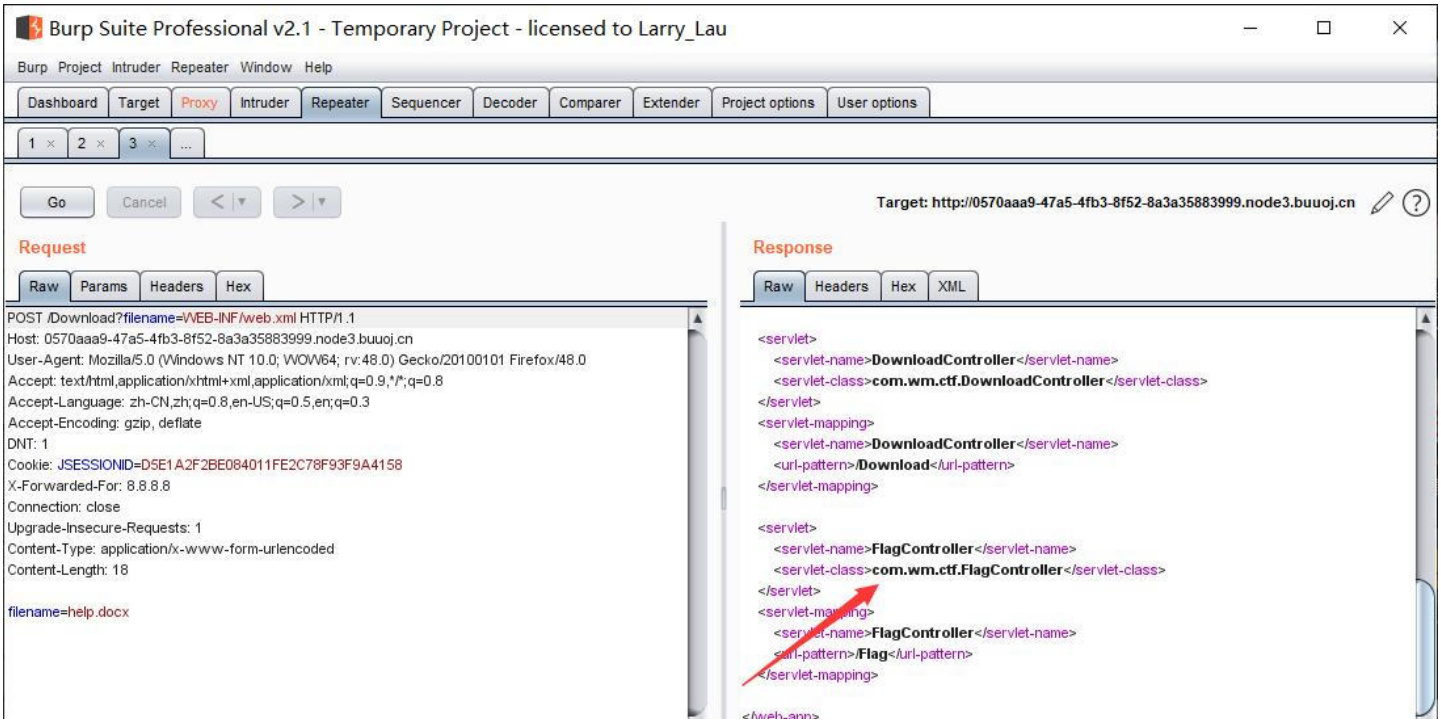




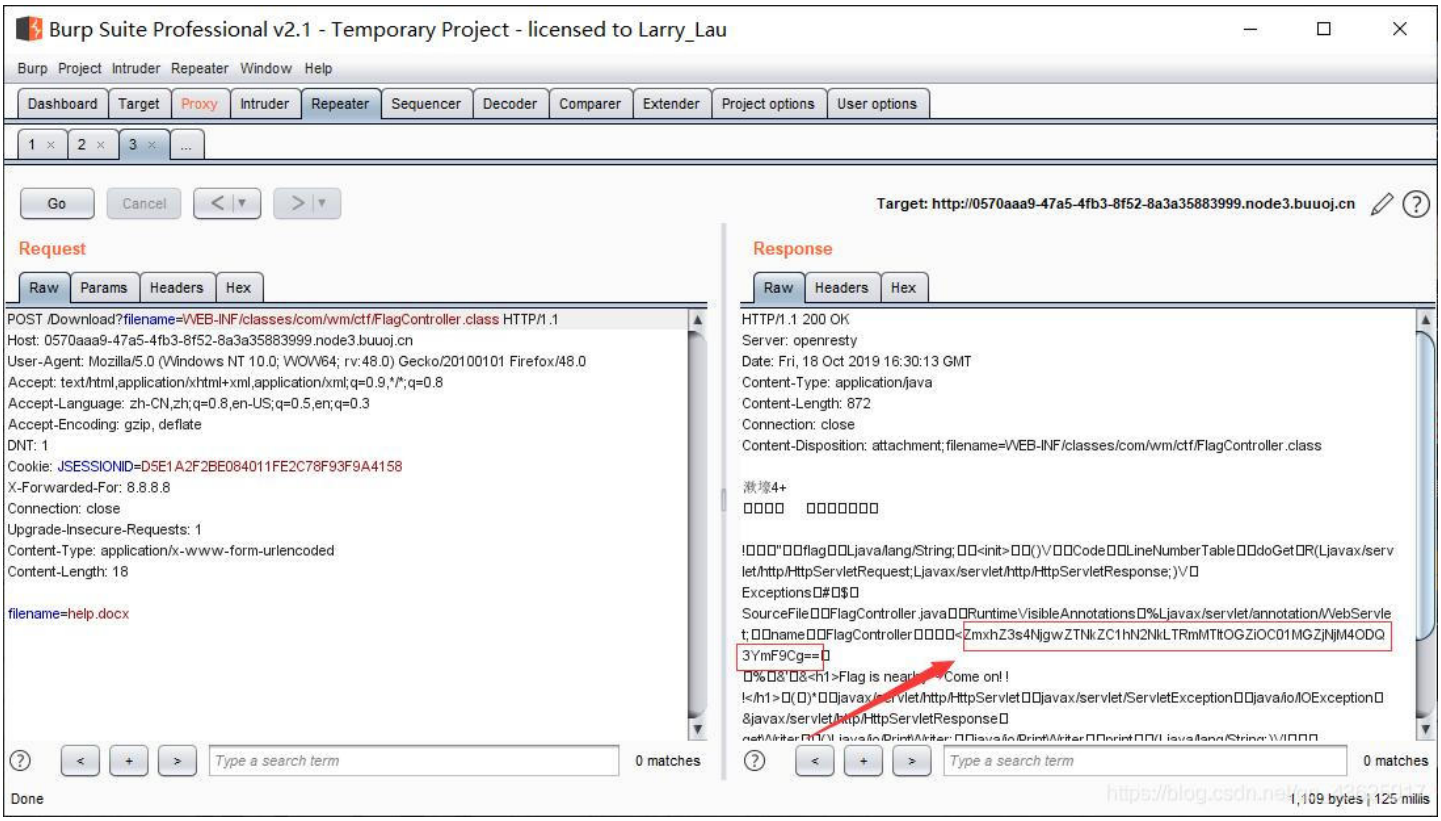
不是弱口令，也不能扫出目录。只有一个`help.docx`文件可以下载。于是可能是任意文件下载漏洞。
点击蓝字“help”，抓包，发包。发现GET方式一直什么都下载不了。后来修改为POST，就可以下载了。



因为题目提示java，所以可以先下载网站的配置文件，在根目录`WEB-INF/web.xml`



发现操作flag的关键文件位置，读取(或下载) `/WEB-INF/classes/` 下的flag的关键文件位置，又因为Java字节码类文件（ `.class` ）是Java编译器编译Java源文件（ `.java` ）产生的“目标文件”。
最终得出flag的关键文件位置为： `/WEB-INF/classes/com/wm/ctf/FlagController.class`



Base64解码得到flag

Base64编码转换

ZmxhZ3s4NjgwZTNkZC1hN2NkLTRmMTItOGZiOC01MGZjNjM4ODQ3YmF9Cg==

加密 解密 解密结果以16进制显示

flag{8680e3d[redacted]fc638847ba}