北京科技大学实验报告-网络安全与管理

学院：　　专业：　　班级：

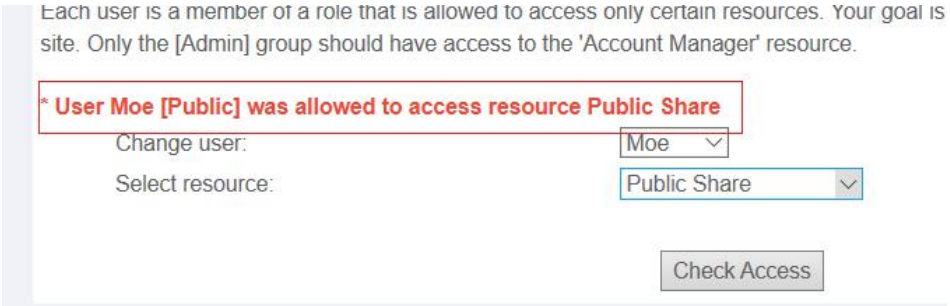
姓名：　　学号：　　实验日期：　　年　　月　　日

实验名称：基于 WebGoat7.1 的 Access Control Flaws（访问控制缺陷）/Using an Access Control Matrix（使用访问控制矩阵）的实验

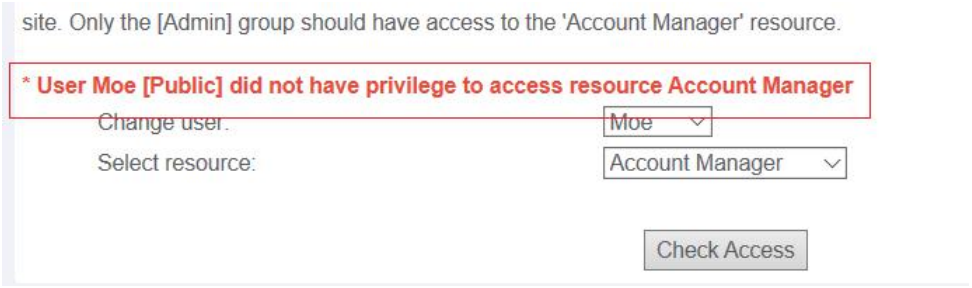
实验目的：在一个基于角色的访问控制方案中，角色代表了一组访问权限和特权。平台课程的要求即探索管理此网站的访问控制规则，只有 Admin 组才能访问“Account Manager”资源，找出 Admin 组，即找到对应角色及其权限。

实验仪器：WebGoat7.1 平台，Burpsuite v1.7.37 抓包工具，Microsoft Edge 44.18362.449.0 版本浏览器

实验原理（含源码分析）：一个用户可以被分配一个或多个角色。一个基于角色的访问控制方案通常有两个部分组成：角色权限管理和角色分配。一个被破坏的基于角色的访问控制方案可能允许用户执行不允许他的被分配的角色，或以某种方式允许特权升级到未经授权的角色的访问。只有 Admin 组才能访问“Account Manager”资源，如 MOE 用户只能访问“Public Share”组



MOE 用户是不允许访问该“Account Manager”组的



注意到如果角色有访问对应组的权限，则会出现“……was allowed to access resource 权限组”。从 Lesson Source Code 来看

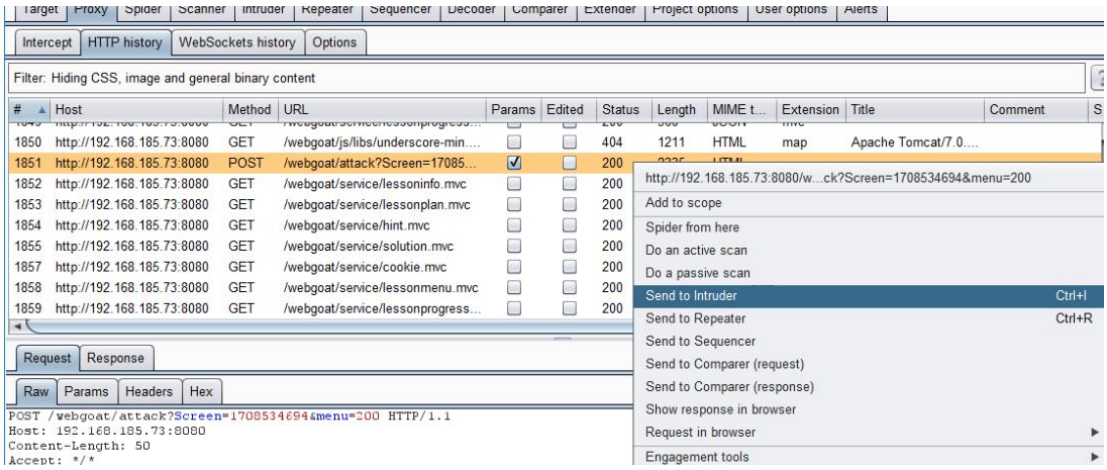
```
if (isAllowed(user, resource))
{
    if (!getRoles(user).contains("Admin") && resource.equals("Account Manager"))
    {
        makeSuccess(s);
    }
    s.setMessage("User " + user + " " + credentials + " was allowed to access resource " + resource);
}
else
{
    s.setMessage("User " + user + " " + credentials + " did not have privilege to access resource "
            + resource);
}
```

可以看到提交对象 user 和权限 resource 之后，getRolse(user).contain(“Admin”)判断 user 的角色是否属于 Admin 组，resource 是否等于 Account Manager，二者同时满足时，则在 HTTP 响应报文中给出“User”,“was allowed to access resource”,“resource”字段，否则给出“did not have……”字段，使用暴力破解的切入点即可以是筛选出符合条件的字段。

因此使用 Burpsuite 进行遍历破解，找到角色和权限组对应的两个 Payload，建立字典后遍历对应，过滤出 HTTP 响应报文中出现“was allowed to access resource Account Manager”的对象，其可以访问 Account Manager 组的资源，即其属于 User Manager 或 Admin。

实验内容与步骤：

1.用 Burpsuite 进行遍历破解

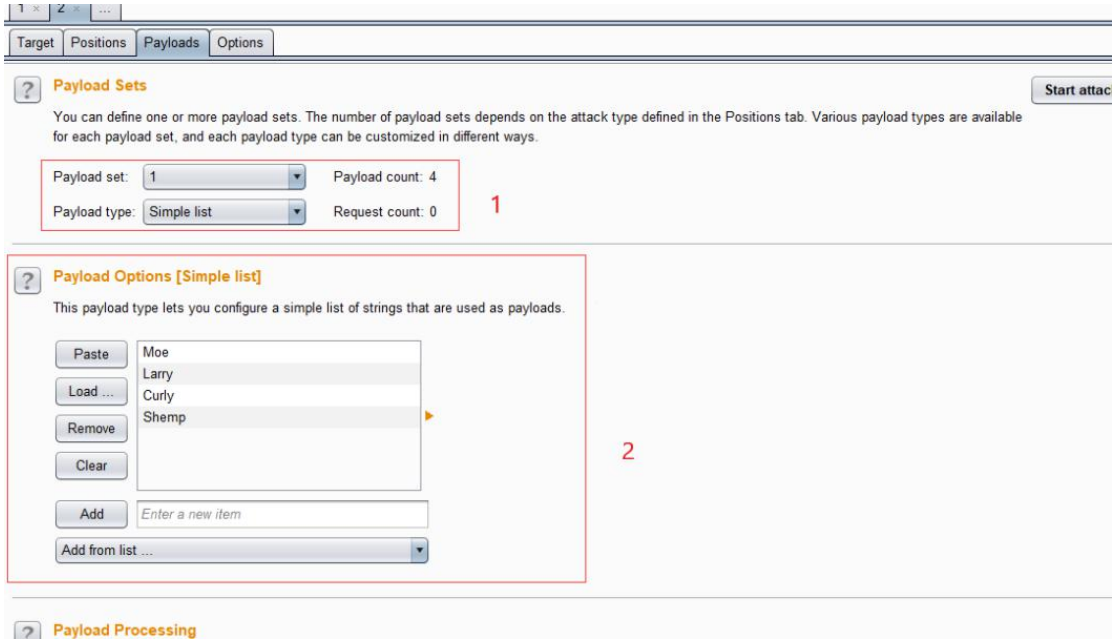


（1）攻击方式为多个 Payload 交叉替换的攻击方式

（2）找到的第一个 Payload

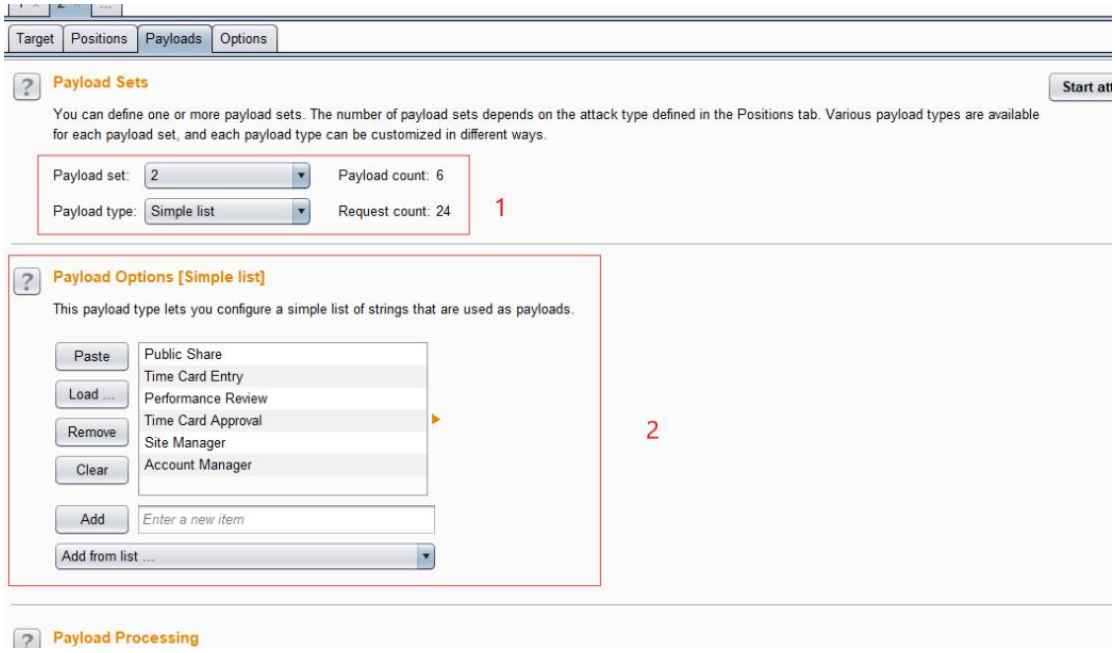
（3）找到的第二个 Payload

2. 找到两个 Payload 后建立字典，准备进行遍历对应



（1）设置 Payload1 的报文字典类型

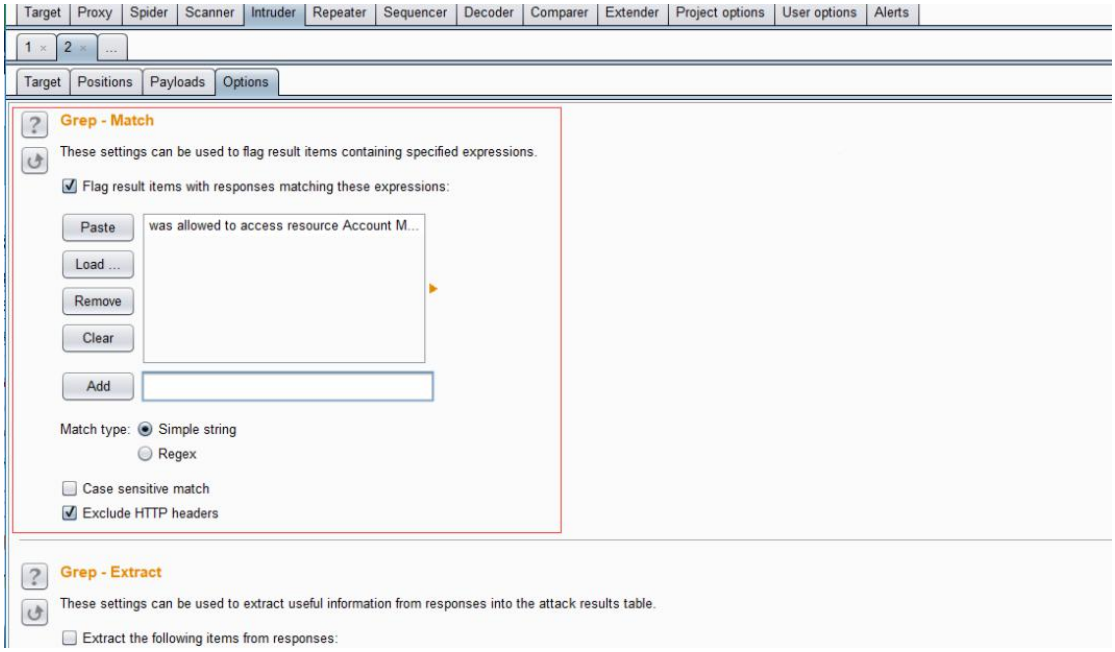
（2）设置 Payload1 的报文字典内容



（1）设置 Payload2 的报文字典类型

（2）设置 Payload2 的报文字典内容

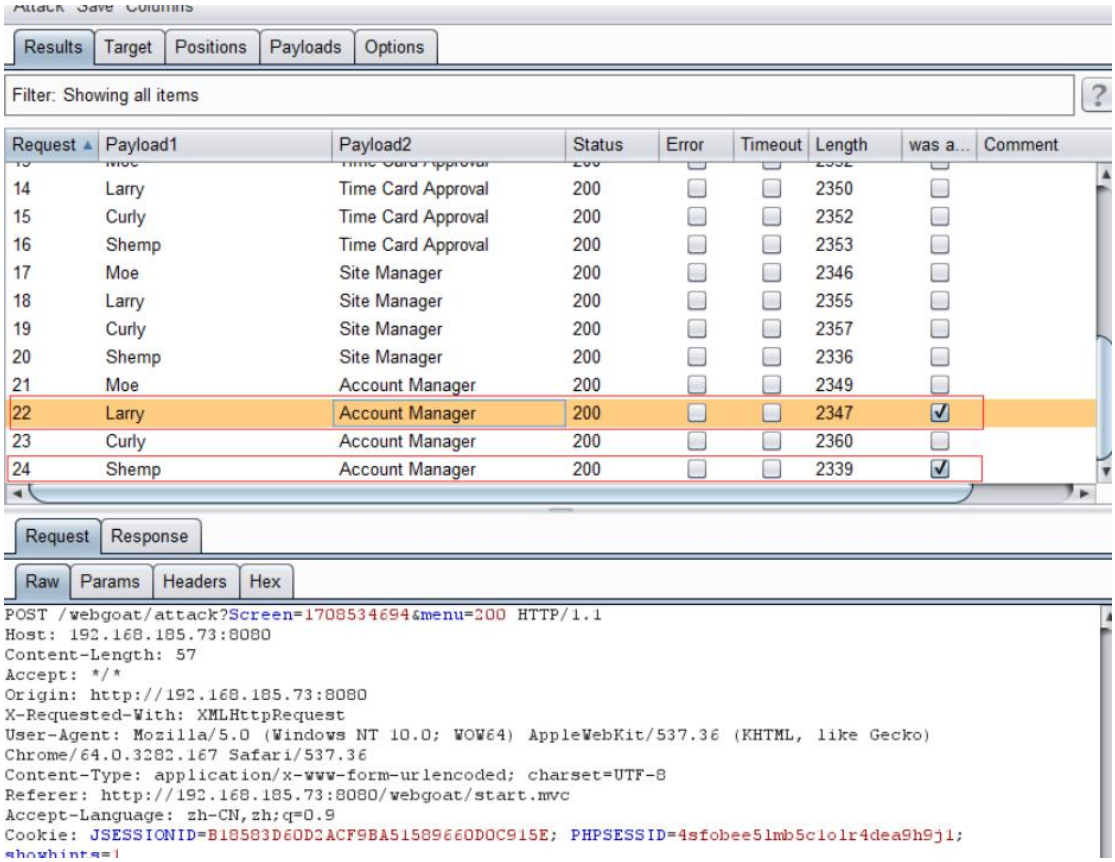
3. 设置过滤 HTTP 响应报文中是否出现“was allowed to access resource Account Manager”



实验数据：

在 Burpsuite 中抓到多条 HTTP 响应，选中其中的 POST 响应

启动过滤后的攻击结果

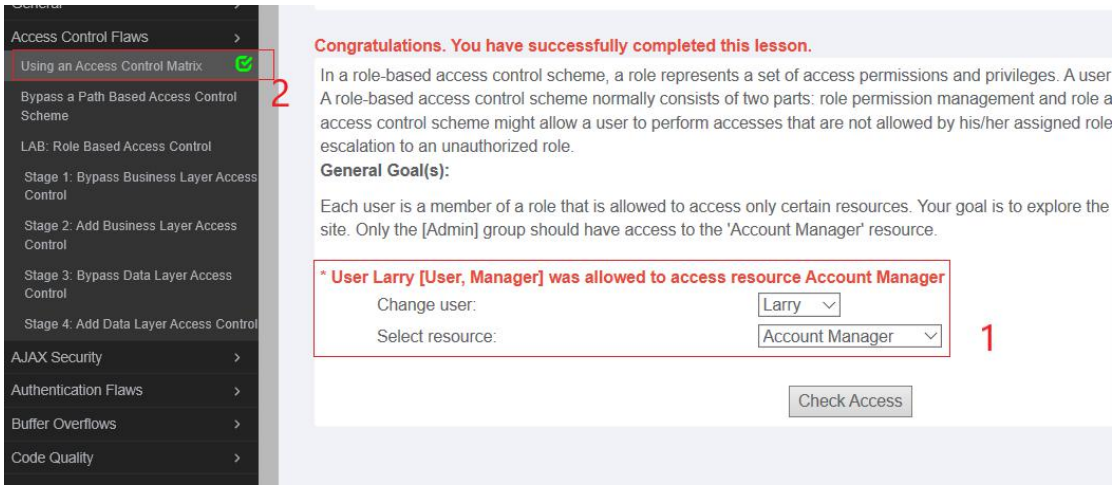


实验数据处理：可以看到，红框中的两条 was allowed to access resource Account Manager 字段被勾选，说明 Larry 和 Shemp 可以访问 Account Manager 组的资源，他们属于 User Manager 或 Admin。

Larry 允许访问该“Account Manager”，在 WebGoat 平台界面中选择并提交后，查看测试结果，看是否可以访问 Account Manager 组权限。

实验结果与分析：

在 WebGoat 平台界面中选择并提交后，测试结果如下



1：Larry 允许访问该“Account Manager”，攻击成功

2：课程后打勾说明已完成

平台课程的要求即探索管理此网站的访问控制规则，只有 Admin 组才能访问“Account Manager”资源，找出 Admin 组。使用 Burpsuite 进行遍历破解，建立字典后，用过滤的方法筛选出 HTTP 响应报文中出现的关键语句，找出其可以访问权限组的对象，即找到对应角色及其权限，虽然不能允许用户以没有分配他的角色或以某种方式获得的未经授权的角色进行访问，但可用相同的办法探索其他角色及权限，当 user 数量较多，权限分类较为复杂时，可以迅速找到访问控制规则，定位所需要的角色对象，为其他类型的攻击提供方便。

实验思考与扩展：

实验思考：本实验的完成是建立在对字典的字段内容全部知晓的前提下。使用 Burpsuite 进行遍历破解中需要对两个 payload 进行交叉对应和遍历，因此要保证两个 payload 字段的内容是可知且完整的。实验中是通过平台给出的字段下拉条来得知两种字段的全部内容的。

实验扩展：在找到访问控制规则之后，找到方法允许用户以没有分配他的角色或以某种方式获得的未经授权的角色进行访问，或进行一定修改。用 Burpsuite 抓包后修改报文的角色及其权限信息，修改角色能够访问的权限，实现未授权角色组的非法访问。