

Домашнее задание к уроку №7 Типы атак, часть 1. OWASP top 10.

Группа: Cyb07-onl
Студент: Парфимович Алексей

1. Выполнение заданий на ресурсе SQLBOLT.

sqlbolt.com

SQLBolt - Learn SQL - SQL Lesson 6: Multi-table queries with JOINS



SQLBolt
Learn SQL with simple, interactive exercises.

Interactive Tutorial

More Topics

SQL Lesson 6: Multi-table queries with JOINS

Up to now, we've been working with a single table, but entity data in the database is often broken down into pieces and stored across multiple orthogonal tables. This process known as *normalization*^[1].

Database normalization

Database normalization is useful because it minimizes duplicate data in the database and allows for data in the database to grow independently of each other (e.g. car engines can grow independent of each type of car). As a trade-off, c

All Lessons

[Introduction to SQL](#)

[SQL Lesson 1: SELECT queries 101](#)

[SQL Lesson 2: Queries with constraints \(Pt. 1\)](#)

[SQL Lesson 3: Queries with constraints \(Pt. 2\)](#)

[SQL Lesson 4: Filtering and sorting Query results](#)

[SQL Review: Simple SELECT Queries](#)

[SQL Lesson 6: Multi-table queries with JOINS](#)

[SQL Lesson 7: OUTER JOINS](#)

[SQL Lesson 8: A short note on NULLs](#)

[SQL Lesson 9: Queries with expressions](#)

2. Выполнение лабораторных работ из практики Brocken Access Control

2.1 User role controlled by request parameter

• <https://portswigger.net/web-security/access-control/lab-user-role-controlled-by-request-parameter>

This lab has an admin panel at /admin, which identifies administrators using a forgeable cookie. Solve the lab by accessing the admin panel and using it to delete the user carlos. You can log in to your own account using the following credentials: wiener:peter

После успешной авторизации в тестовом приложении предоставленным пользователем (wiener), в куки всех последующих запросов к серверу начинает передваться параметр Admin типа bool. Задача решается подменой в передаваемых куки значения false параметра Admin на значение true (для всех запросов при обращении к серверу).

User role controlled by request parameter



LAB Solved

WebSecurity Academy

User role controlled by request parameter

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

2.2 User id controlled by request parameter

• <https://portswigger.net/web-security/access-control/lab-user-id-controlled-by-request-parameter>

This lab has a horizontal privilege escalation vulnerability on the user account page.
To solve the lab, obtain the API key for the user carlos and submit it as the solution.
You can log in to your own account using the following credentials: wiener:peter

После успешной аутентификации пользователя, в запросах к серверу передается параметр запроса "id", в котором передается значение логина текущего пользователя (wiener).
Задача решается перехватом запроса к серверу и подменой значение в ппараметре "id" На логин целевого пользователя (carlos).

0 controlled b

User ID controlled by req

+

🔒

https://0a8500300436c9dd803b538f00d300c3.web-security-academy.net/my-account?id=wiener



WebSecurity Academy

User ID controlled by request parameter

LAB Solved

Back to lab description >>

Congratulations, you solved the lab!

Share your skills!   Continue learning >>

Home | My account | Log out

My Account

Your username is: wiener

Your API Key is: GTsMSWAIgG0DJMKn9AGIguQQjrTNE0Fm

Email

Update email

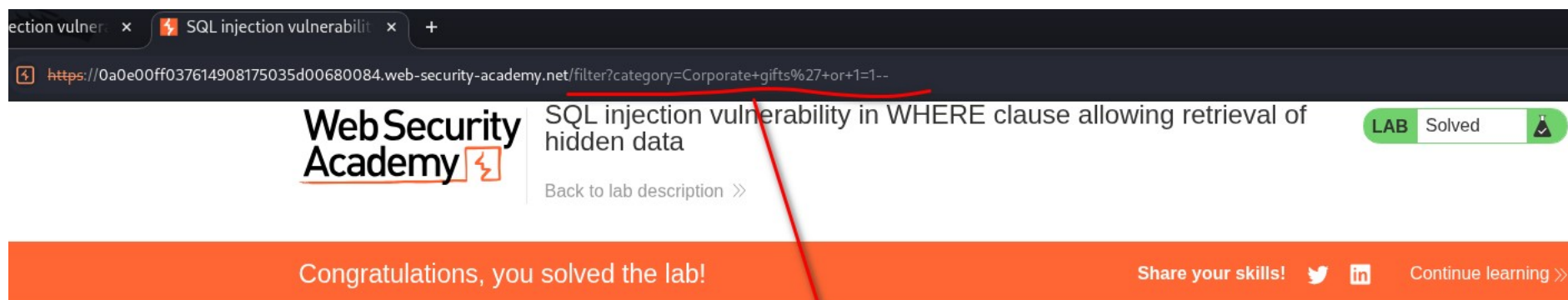
3. Выполнение лабораторной работы из практики *Injections*

• <https://portswigger.net/web-security/sql-injection/lab-retrieve-hidden-data>

Для экранирования кавычек в теле SQL запроса используются символы комментария "--"

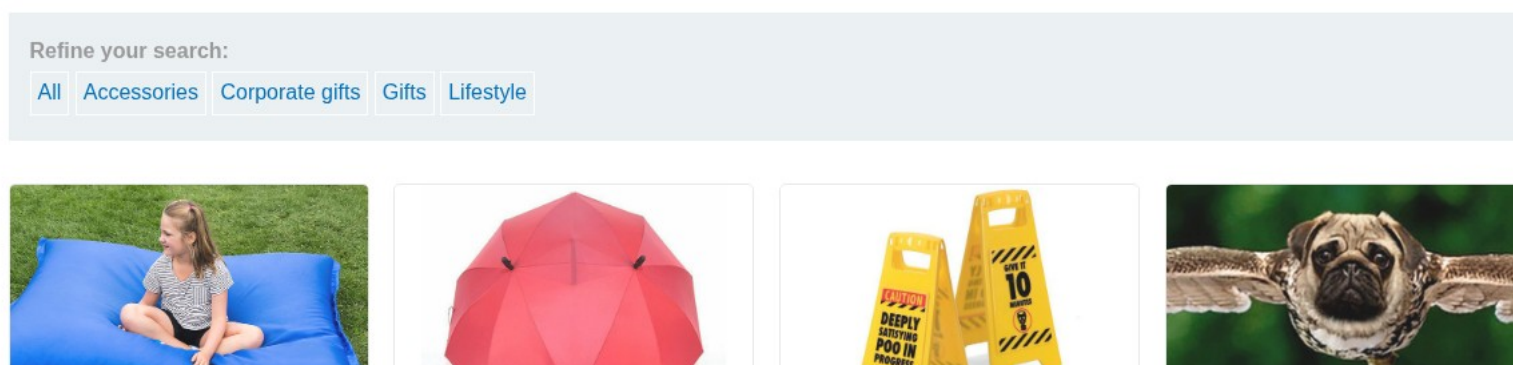
Например для исходного значения фильтра `filter?category=Gifts`

строка с инъекцией - `filter?category=Gifts'+or+1=1--`



WE LIKE TO
SHOP

Corporate gifts' or 1=1--



4. Выполнение лабораторной работы из практики Server-Side Request Forgery

• `https://portswigger.net/web-security/ssrf/lab-basic-ssrf-against-localhost`

Используется запрос POST /product/stock для проверки наличия продукта на складе
В запросе передается параметр stockApi, который содержит URL обращения к одному из складов,
например `http://stock.weliketoshop.net:8080/product/stock/check?productId=1&storeId=1`

Решение задачи состоит в эксплуатации этой уязвимости:

- Подменяем в запросе параметр stockApi адресом обращения к локальному ресурсу приложения - консоли администратора `http://localhost/admin`
- В консоли администратора получаем ссылку на URL для удаления пользователя `http://localhost/admin/delete?username=carlos`
- Подставляем полученный URL в запрос /product/stock

SRF against t... x

Basic SSRF against the lo... x

+

https://0a9700ef0403a70c807676c30033008b.web-security-academy.net/product?productId=1

WebSecurity Academy

Basic SSRF against the local server

LAB Solved

Back to lab description >>

Congratulations, you solved the lab!

Share your skills!

Continue learning >>

[Home](#) | [My account](#)

Potato Theater

\$52.49



5. Установка docker и развертывание контейнера с web-приложением JuicyShop

5.1 Установка Docker в Kali Linux

Шаг 1. Обновить систему:
sudo apt update && sudo apt upgrade -y

Шаг 2. Установить необходимые зависимости:
sudo apt install -y ca-certificates curl gnupg lsb-release

Шаг 3. Добавить официальный GPG-ключ Docker:
sudo mkdir -p /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg

Важно: Kali основан на Debian, поэтому необходимо использовать репозиторий Debian, а не Ubuntu.

Шаг 4. Добавить репозиторий Docker:
echo "deb [arch=\$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/debian bookworm stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

Примечание: команда для получения текщего релиза ОС \$(lsb_release -cs) в Kali возвращает kali-rolling, но репозиторий Docker не знает такой ветки, поэтому в команде используется bookworm (актуальный stable-релиз Debian на 2024-2025):

Шаг 5. Обновить список пакетов:
sudo apt update

Шаг 6. Установить Docker Engine:
sudo apt install -y docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

Шаг 7. Запустить и включить службу Docker:
sudo systemctl enable --now docker

Проверить статус:
sudo systemctl status docker

Проверить установку:
sudo docker run hello-world

На экране должно отобразиться сообщение:
Hello from Docker!
This message shows your installation appears to be working correctly.

Шаг 9. Запустить Docker без sudo (по умолчанию Docker требует sudo. Необходимо добавить пользователя в группу docker):
sudo usermod -aG docker \$USER

Примечание: перезапустить сессию чтобы изменения вступили в силу.

```
user@kali: ~  
File Actions Edit View Help  
Tasks: 9  
Memory: 23.5M (peak: 25.6M)  
CPU: 854ms  
CGroup: /system.slice/docker.service  
└─74623 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock  
Oct 05 13:31:35 kali dockerd[74623]: time="2025-10-05T13:31:35.735274718+03:00" level=info msg="CDI directory does not exist, skipping: failed to monitor for changes: no such file or directory" dir=/etc/cdi  
Oct 05 13:31:35 kali dockerd[74623]: time="2025-10-05T13:31:35.821923216+03:00" level=info msg="Creating a containerd client" address=/run/containerd/containerd.sock timeout=1m0s  
Oct 05 13:31:36 kali dockerd[74623]: time="2025-10-05T13:31:36.032948803+03:00" level=info msg="Loading containers: start."  
Oct 05 13:31:36 kali dockerd[74623]: time="2025-10-05T13:31:36.798267626+03:00" level=info msg="Loading containers: done."  
Oct 05 13:31:36 kali dockerd[74623]: time="2025-10-05T13:31:36.909225165+03:00" level=info msg="Docker daemon" commit=cd04830 containerd-snapshotter=false storage-driver=overlay2 version=28.5.0  
Oct 05 13:31:36 kali dockerd[74623]: time="2025-10-05T13:31:36.910209112+03:00" level=info msg="Initializing buildkit"  
Oct 05 13:31:37 kali dockerd[74623]: time="2025-10-05T13:31:37.004350900+03:00" level=info msg="Completed buildkit initialization"  
Oct 05 13:31:37 kali dockerd[74623]: time="2025-10-05T13:31:37.030041859+03:00" level=info msg="Daemon has completed initialization"  
Oct 05 13:31:37 kali dockerd[74623]: time="2025-10-05T13:31:37.030120308+03:00" level=info msg="API listen on /run/docker.sock"  
Oct 05 13:31:37 kali systemd[1]: Started docker.service - Docker Application Container Engine.  
  
user@kali:~  
$ sudo docker run hello-world  
Unable to find image 'hello-world:latest' locally  
latest: Pulling from library/hello-world  
17eec7bbc9d7: Pull complete  
Digest: sha256:54e66cc1dd1fcb1c3c58bd8017914dbed8701e2d8c74d9262e26bd9cc1642d31  
Status: Downloaded newer image for hello-world:latest  
  
Hello from Docker!  
This message shows that your installation appears to be working correctly.  
  
To generate this message, Docker took the following steps:  
1. The Docker client contacted the Docker daemon.  
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
   (amd64)  
3. The Docker daemon created a new container from that image which runs the  
   executable that produces the output you are currently reading.  
4. The Docker daemon streamed that output to the Docker client, which sent it  
   to your terminal.  
  
To try something more ambitious, you can run an Ubuntu container with:  
$ docker run -it ubuntu bash  
  
Share images, automate workflows, and more with a free Docker ID:  
https://hub.docker.com/  
  
For more examples and ideas, visit:  
https://docs.docker.com/get-started/  
  
user@kali:~  
$
```


5.2 развертывание контейнера с web-приложением JuicyShop

Загрузить образ контейнера из репозитория DockerHub:

```
docker pull bkimminich/juice-shop
```

Запустить контейнер из образа:

```
sudo docker run -d -p 3000:3000 bkimminich/juice-shop
```

Запустить приложение в браузере:

```
http://localhost:3000
```

