Udajuicer:
Threat Report

Haneen Alamoudi

11/6/2022

Purpose of this Report:

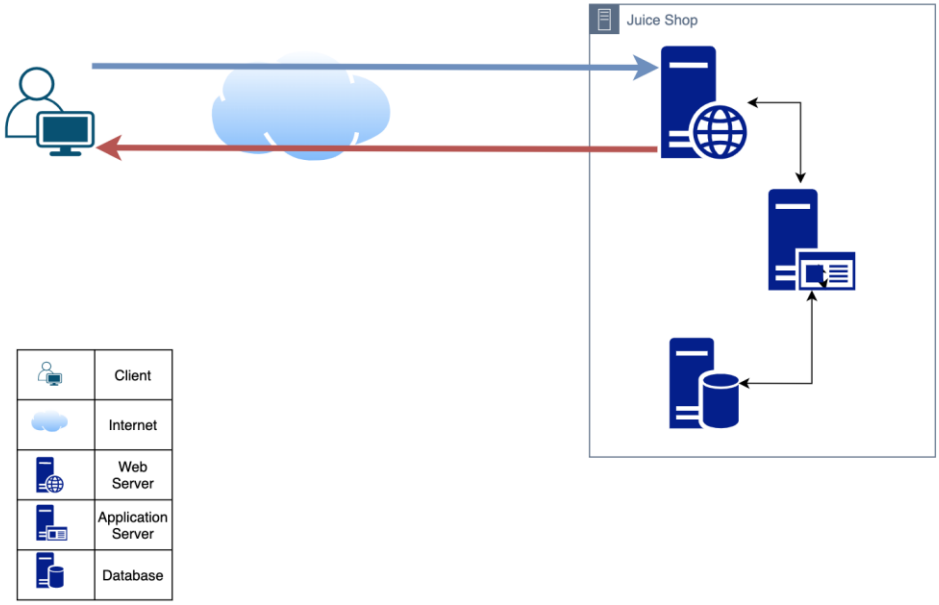
This is a threat model report for **Udajuicer**. The report will describe the threats facing Udajuicer. The model will cover the following:

- Threat Assessment
    - Scoping out Asset Inventory
    - Architecture Audit
    - Threat Model Diagram
    - Threats to the Organization
    - Identifying Threat Actors
- Vulnerability Analysis
- Risk Analysis
- Mitigation Plan

Section 1

Threat Assessment

1.1: Asset Inventory



Components and Functions:

- ***Web server:*** *A computer system that run web server software.*
- ***Application server:*** server that runs applications and processes request sent by the web server including communicating with the database.
- ***Database:*** Store Udajuicer data, and can be used to retrieve, process and update data.

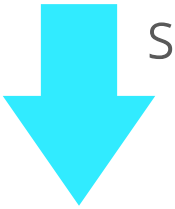
1.1: Asset Inventory

A Request Goes from Client to Server through the following steps

Client access the Udajuicer web application and place an order.

HTTPS Request

Web server communicate with the application server to process data.



SQL Query

Application server communicate with the database to retrieve, store or update data.

1.2 Architecture Audit

Flaws

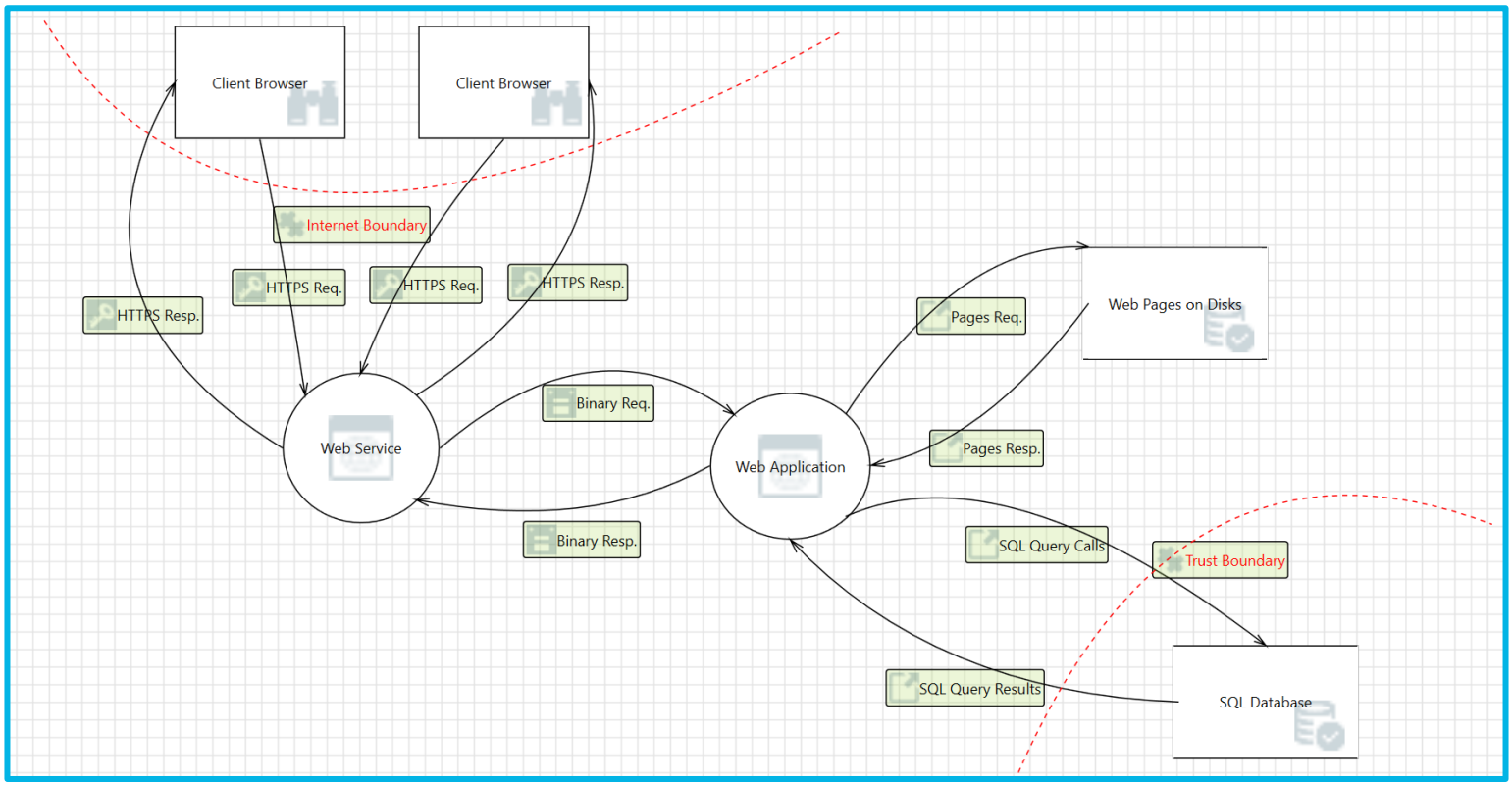
- ***Lack of firewalls deployments****, this include nework firewall as well as a web application firewall (WAF)*
- ***Lack of redundant architecture****, to ensure the high availability of the web application.*
- ***Lack of logs monitoring application****, (SIEM)*

1.3 Threat Model Diagram

Using OWASP Threat Dragon, build a diagram showing the flow of data in the Juice Shop application and identify 3 possible threats to the Juice Shop. Make sure to include the following components:

- **Client**
- **Web Server**
- **Application Server**
- **Database**

1.3 Threat Model Diagram



1.3 Threat Model Diagram

Some of the possible threats for Udajuicer web application are the OWASP Top 10 which consists of the following:

- **Injection**
- **Broken Authentication**
- **Sensitive Data Exposure**
- **XML External Entities (XXE)**
- **Broken Access Control**
- **Security Misconfiguration**
- **Cross-Site Scripting (XSS)**
- **Insecure Deserialization**
- **Using Components with Known Vulnerabilities**
- **Insufficient Logging & Monitoring**

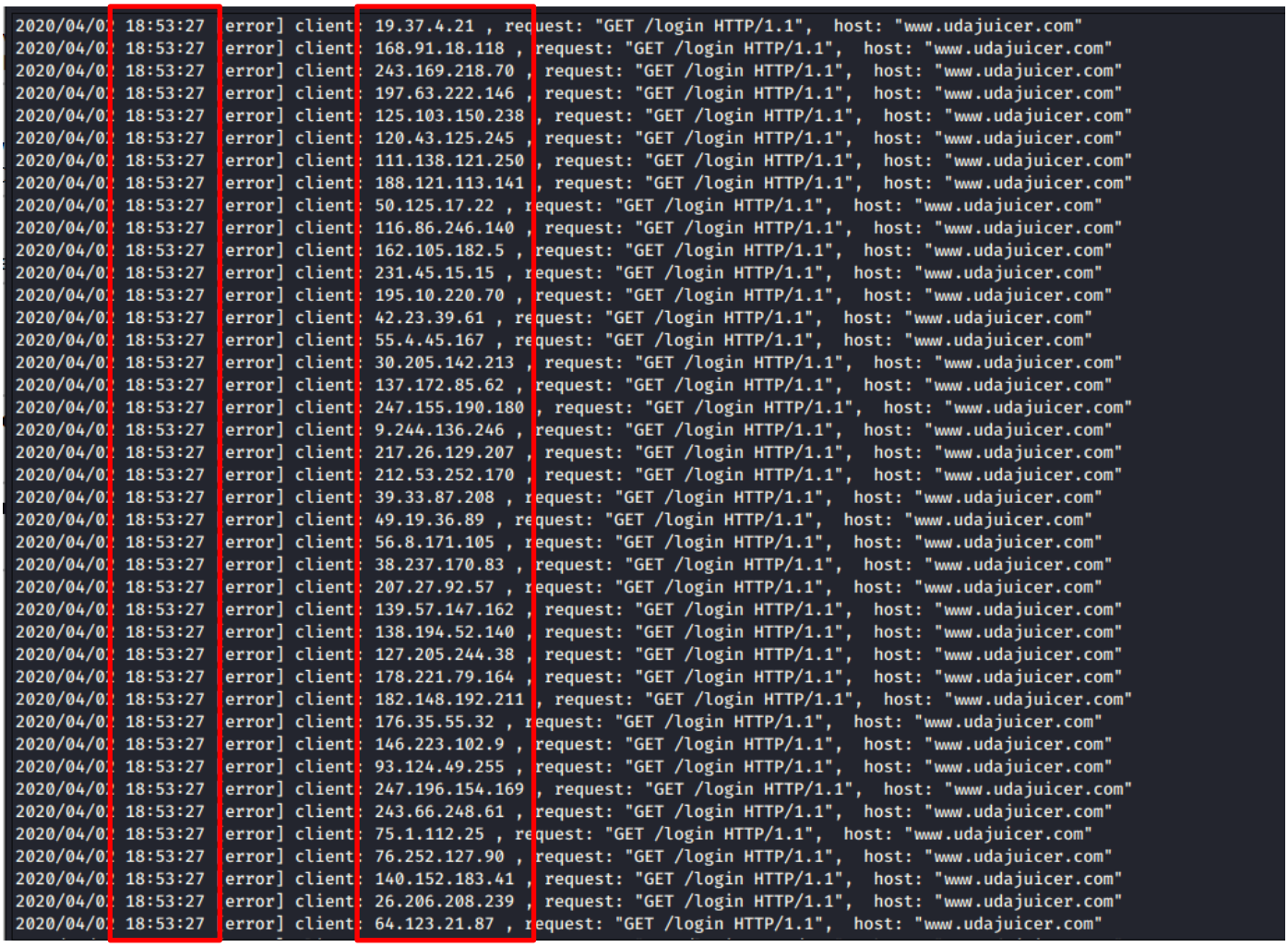
1.4 Threat Analysis

What Type of Attack Caused the Crash?

Distributed Denial of Service (DDOS) is the same attack from multiple machines that are controlled by the attackers.

What in the Logs Proves Your Theory?

Huge number of requests sent at the same time from different IP addresses.



1.5 Threat Actor Analysis

Who is the Most Likely Threat Actor?

Script Kiddie

What Proves Your Theory?

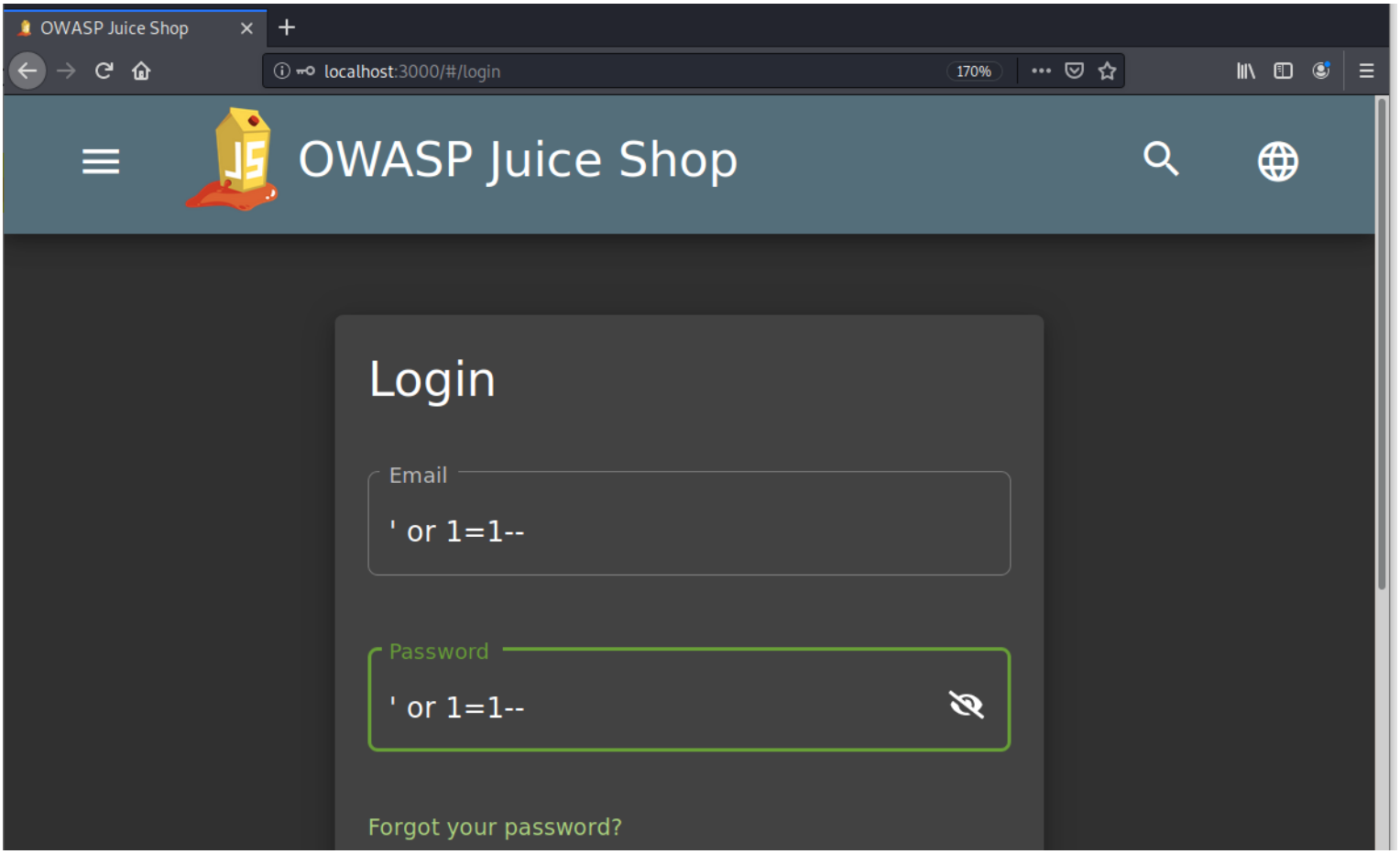
Udajuicer web application is build with minimal security masures this make the website easily hacked through amateur hackers with online templates. Also, there is no signs of known motivation of the attack on the website.

Section 2

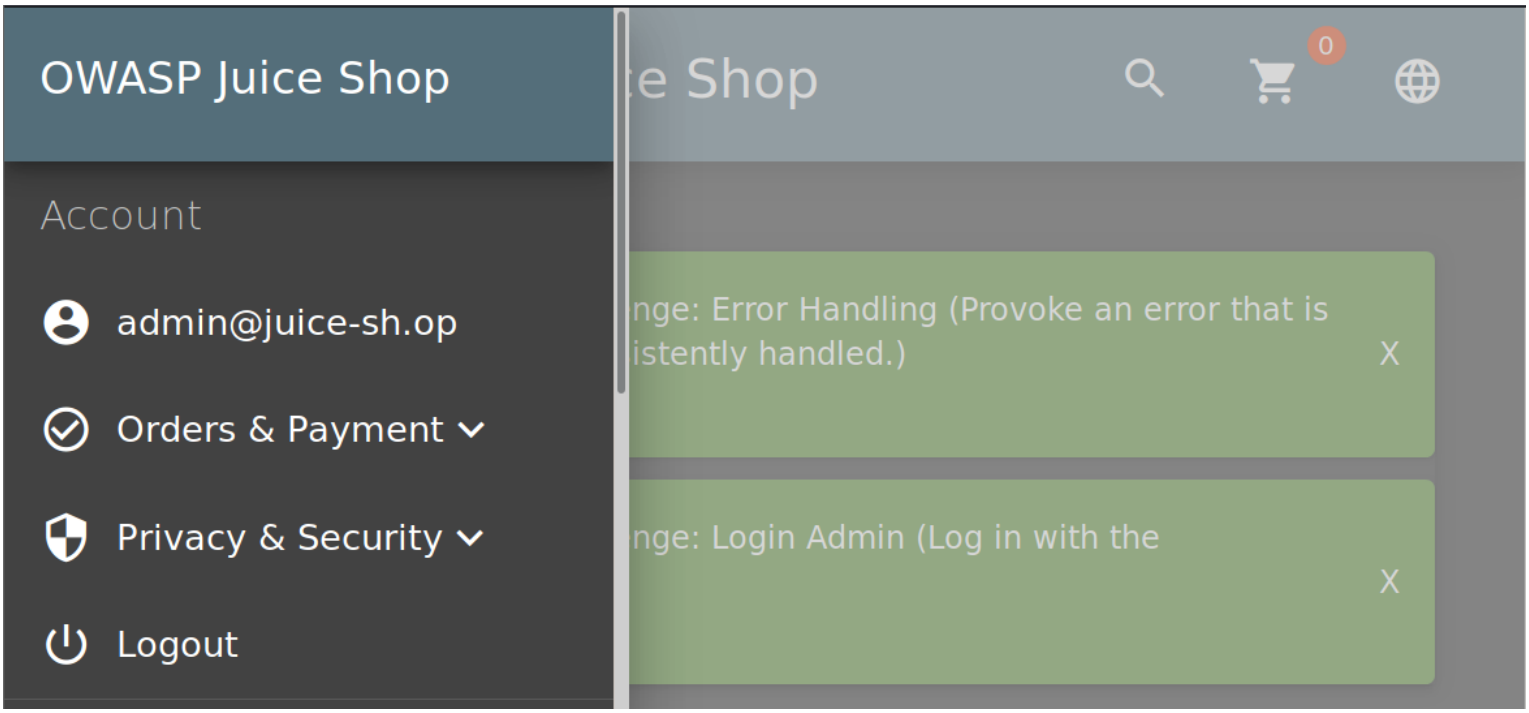
Vulnerability Analysis

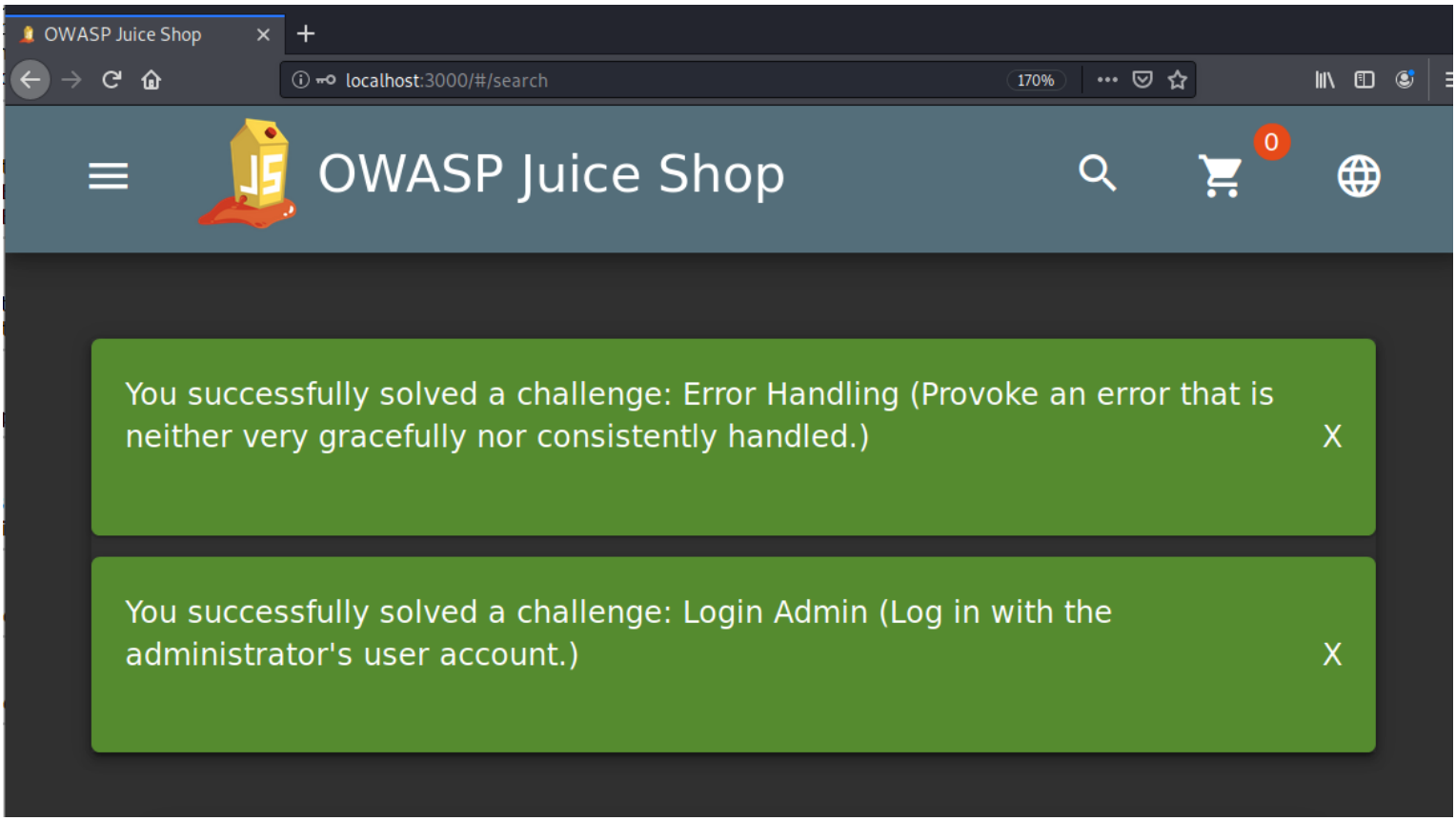
2.1 SQL Injection

Commands used to preform SQL Injection:

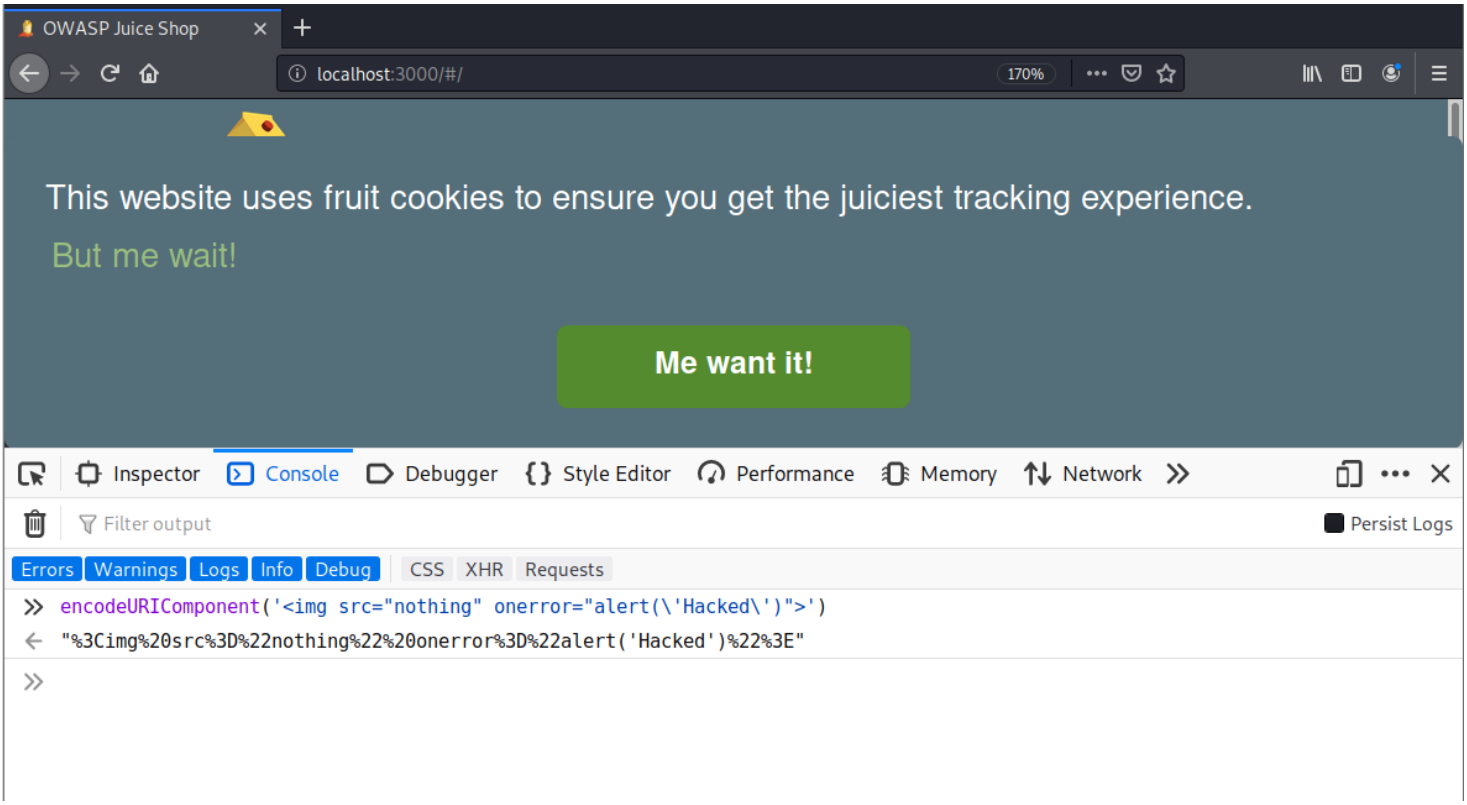


2.1 SQL Injection

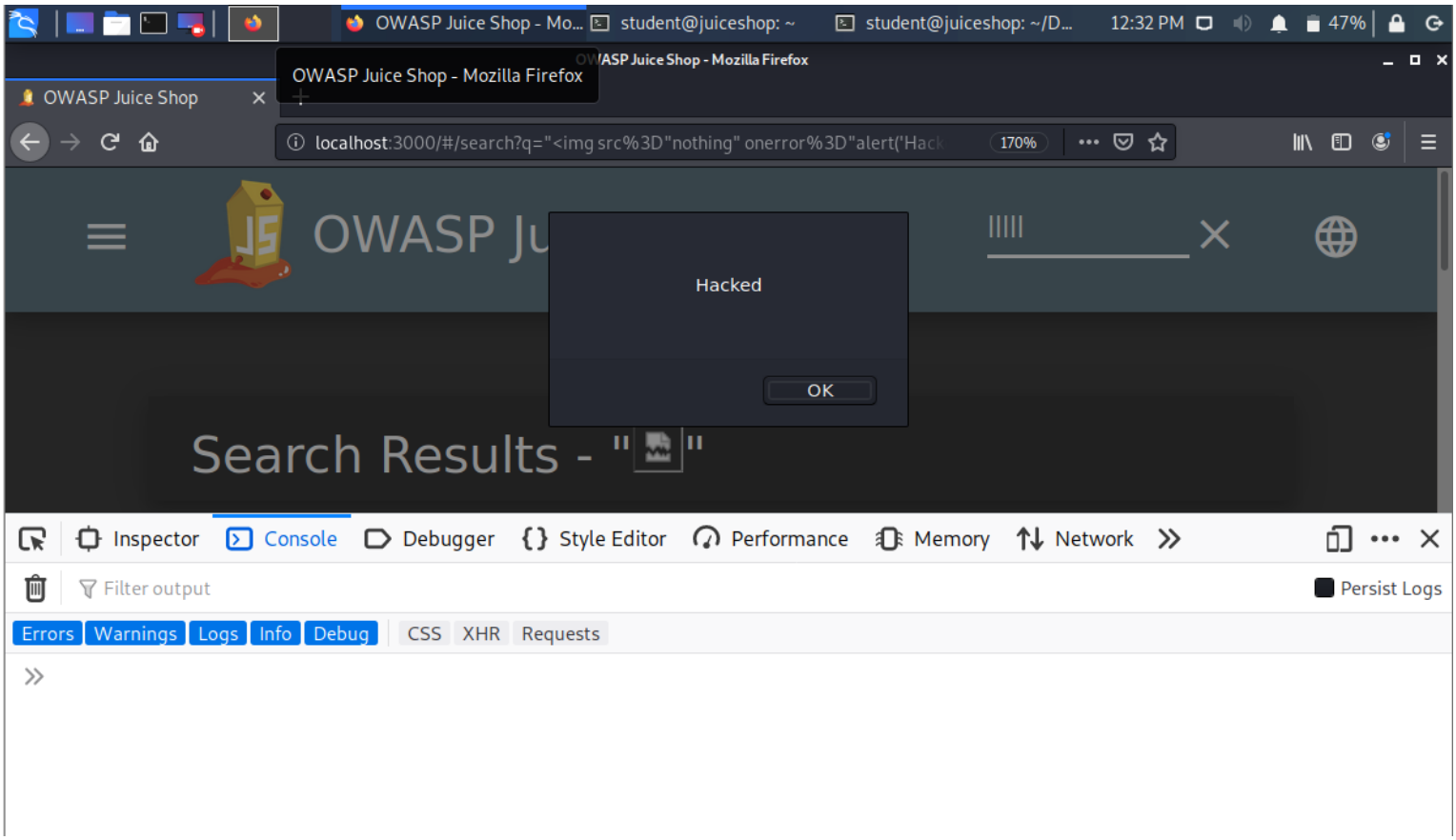




2.2 XSS


OWASP Juice Shop
localhost:3000/#/
170%
This website uses fruit cookies to ensure you get the juiciest tracking experience.
But me wait!
Me want it!
Inspector
Console
Debugger
Style Editor
Performance
Memory
Network
Filter output
Persist Logs
Errors
Warnings
Logs
Info
Debug
CSS
XHR
Requests
encodeURIComponent('<img src="nothing" onerror="alert(\'Hacked\')">')
"%3Cimg%20src%3D%22nothing%22%20onerror%3D%22alert('Hacked')%22%3E"

2.2 XSS


OWASP Juice Shop - Mo...
student@juiceshop: ~
student@juiceshop: ~/D...
12:32 PM
47%
OWASP Juice Shop - Mozilla Firefox
OWASP Juice Shop
localhost:3000/#/search?q="<img src%3D"nothing" onerror%3D"alert('Hack
170%
OWASP Ju
Hacked
OK
Search Results - "
Inspector
Console
Debugger
Style Editor
Performance
Memory
Network
Filter output
Persist Logs
Errors
Warnings
Logs
Info
Debug
CSS
XHR
Requests

Section 3

Risk Analysis

3.1 Scoring Risks

Risk	**Score** ***(1 is most dangerous, 4 is least dangerous)***
Distributed Denial of Service	1
Insecure Architecture	2
SQL Injection	3
XSS Vulnerability	4

3.2 Risk Rationale

Why Did You Choose That Ranking?

DDOS has the highest ranking as the website is currently down from a DDOS attack..

Building a secure architecture is a major step in solving any cyber security related issues, with a secure architecture many other vulnerability are eliminated or minimized such as DDOS.

Next, is SQL injection, as getting unauthorized access to the database as an admin could affect the company in many ways including loss of data, ransomware attach, exploiting customer data and personal record.

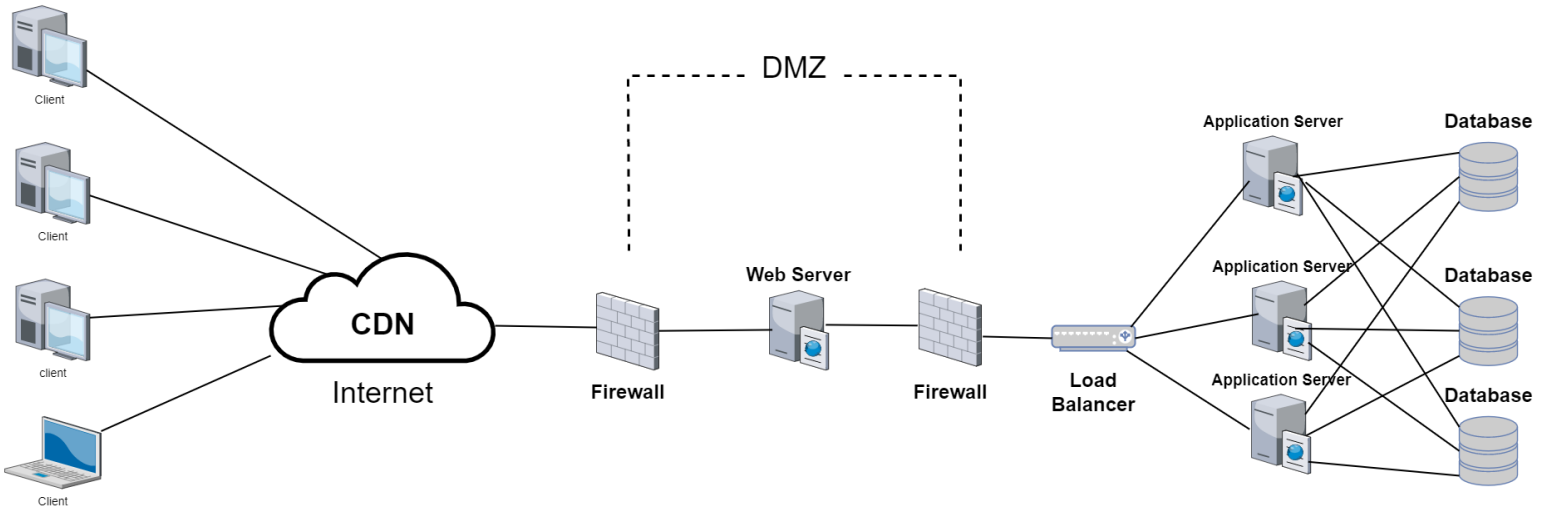
XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.

Section 4

Mitigation Plan

4.1 Secure Architecture

Proposed Secure Architecture:

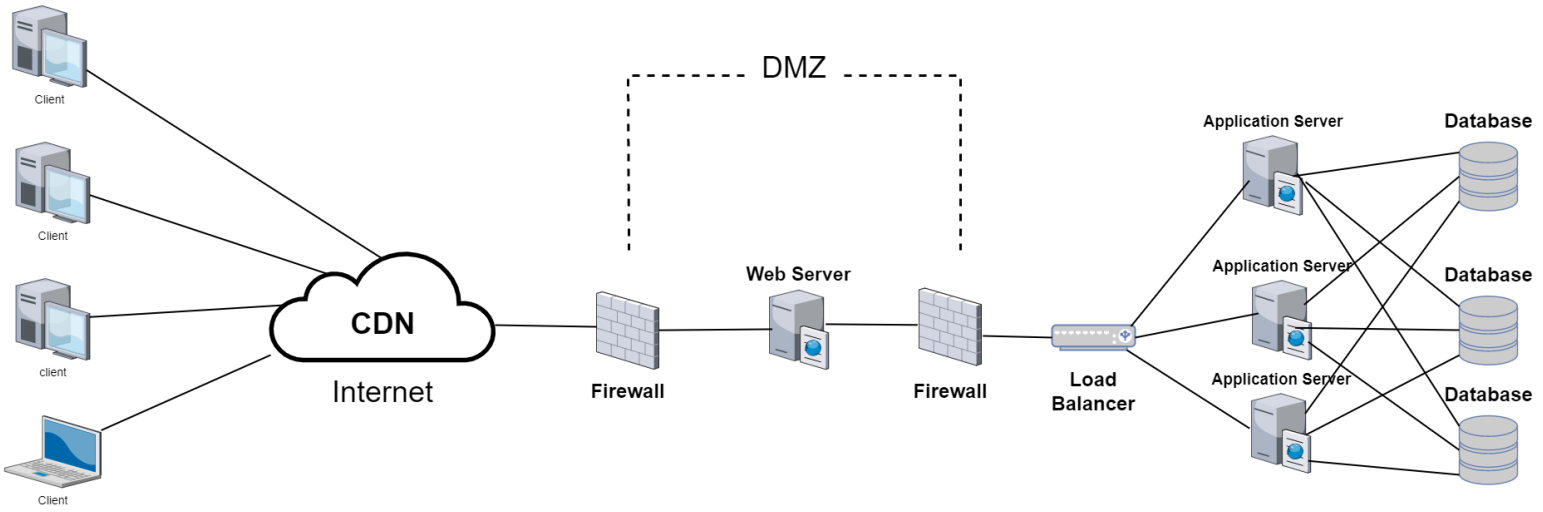


4.2 DDOS Attack Mitigation

To mitigate DDOS attach:
Implementing a secure architecture by using a Content Delivery Network (CDN), Load Balancers, and Firewalls.

CDN will help in redundance, so in case on of the server edge are down, other are used to respond to new requests

Also, Firewalls are used to secure communication with the internal network.



4.3 SQL Injection Mitigation

SQL injection is mitigated by :

1. Input Sanitization
2. Input Validation
3. Prepared Statements with Parameterized Queries
4. Escaping.

4.4 XSS Mitigation

XXS can be mitigated by:

1. Sanitizing User Input
2. Validating User Input
3. Escaping.