# **Penetration Test Report**

## **Executive Summary:**

### **Introduction:**

This report presents the findings of a penetration test conducted on group 16 Gotham project backend application deployed on an EC2 instance. The objective of this test was to assess the security of the application, focusing on the Elixir backend and PostgreSQL database. The testing scope included SSH and HTTP protocols.

**Key Findings:**

* Authentication Bypass Vulnerability:

No critical vulnerabilities were identified that could allow an attacker to bypass authentication and gain unauthorized access.

* SQL Injection Risk:

The application is not susceptible to SQL injection attacks due to sufficient input validation both on frontend and backend. Also, all access to the database via Api calls are secured by token verifications.

* Missing HTTPS Implementation:

The application currently lacks HTTPS, exposing sensitive data to potential interception.

* Outdated Software Components:

All software components are regularly updated by the SSM agent installed on the ec2 instance.

This agent pulls and install software updates and security patches.

**Recommendations:**

Enforce HTTPS:

Implement HTTPS to encrypt data in transit and protect against man-in-the-middle attacks.

This can be done by implementing SSL or TLS using a certificate on an application load balancer.

**Detailed Findings mechanisms:**

* Authentication Bypass Vulnerability:

Description: we ran series of request on different Api endpoints non unauthorized request had a response from the server.

Get All user's request: <http://13.51.249.253/users> response with unauthorized when no token valid token is passed to the request.

* SQL Injection Risk:

Description:

Running an SQL injection like <http://13.51.249.253/api/users/1> OR 1=1 to server returns bad request due to input validations on server side.

* Missing HTTPS Implementation:

Description:

Using http instead of https to communicate with server. Example http://13.51.249.253/api/users

Recommendation:

Enforce HTTPS:

Implement HTTPS to encrypt data in transit and protect against man-in-the-middle attacks.

This can be done by implementing SSL or TLS using a certificate on an application load balancer.

Conclusion:

In conclusion, this penetration test identified some vulnerabilities and areas of concern in the security of Gotham project. The recommendations provided are crucial for enhancing the overall security posture of the application.

It is recommended to address these findings promptly and conduct regular security assessments to stay ahead of emerging threats.