Task 1 Report

CITS3006 Project 2025
Task 1 Report

Team 17

Emily Han (23925907)
Aung Phone Hein (23738739)
Jake Blackburne (23782618) PoC
Vincent Ta (23975858)

Executive summary

Maybe half a page long, summary of what we did.

Web Server

We used an existing open source webpage as our vulnerable website to exploit.
OG Link: https://github.com/startbootstrap/startbootstrap-sb-admin-2

What changes were made to it:

Vulnerabilities

vulnerabilities created, the software they relate to, and provide descriptions of how they can be exploited.

Web vulnerabilities

What are they.

Vulnerability 1: Stored Cross-Site Scripting (Stored XSS)

What it is:
Stored XSS occurs when user-supplied content is saved by the application (database, file, etc.) and later rendered into pages viewed by other users without proper escaping or sanitization. The attacker stores JavaScript which runs in victims' browsers with that site's privileges (cookies, session tokens, DOM access).

Why it's dangerous:
An attacker can run arbitrary JavaScript in other users' browsers, steal session cookies, perform actions on behalf of users, display fake login dialogs, or chain into further attacks (CSRF + XSS). In an admin panel it's especially powerful because admin users may have high privileges.

How it works: :

Vulnerability 2: SQL Injection (SQLi)

What it is:

SQLi happens when an application builds SQL queries by concatenating or interpolating user input directly into SQL statements without parameterization, allowing attackers to alter SQL semantics (read data, bypass auth, modify DB).

Why it's dangerous:
An attacker can extract sensitive data, bypass authentication, modify or delete data, or escalate to remote code execution in some misconfigured DBs. Even simple injection against a local test DB demonstrates the concept.

How it works::

Vulnerability 3: Unrestricted File Upload / Dangerous File Types

What it is:
This occurs when an application allows users to upload files without validating file type, size, or content and saves them in a web-accessible directory. Attackers can upload HTML/JS (and then access it to run in victim browsers), server-side scripts (if the server will execute them), or large files to exhaust storage.

Why it's dangerous:
Uploading an HTML/JS file and then serving it can lead to stored XSS/drive-by attacks.
If the server executes uploaded files (PHP, JSP), an attacker could gain remote code execution.
Large uploads or many files can lead to DoS/storage exhaustion.

How it works:

Privilege escalation vulnerabilities

What are they?

Vulnerability 4 (Root)

Name, description

Vulnerability 5 (Root)

Name, description

Vulnerability 6

Name, description

Vulnerable image.

Link:

Setup Instructions

Type…

Demo video

Link:

Any timestamps or descriptions required.

Challenges & mitigations.

any challenges you encountered, and how you addressed them.

Task 2 Report

CITS3006 Project 2025
Task 2 Report

Team 17

Emily Han (23925907)
Aung Phone Hein (23738739)
Jake Blackburne (23782618) PoC
Vincent Ta (23975858)

Executive summary

type…

Exploited Box

Discovery and Exploitation

step-by-step description of discovery and exploitation

Commands and Tools

the commands and tools used,

Evidence

screenshots, log snippets, video timestamps

Assessed Boxes

Give a difficulty rating on a 1–10 scale (1 = very easy, 10 = very hard), and a brief justification (a few sentences) explaining the factors that influenced your score (e.g. complexity of the exploit chain, tooling required, time to discovery, need for manual analysis vs. automation).

Box 1: Name of Box

Type…

Box 2: Name of Box

Type…

Box 3; Name of Box

Type…

Caveats or Limitations

for example, if some parts of an exploit could not be executed due to time or environment constraints

Activity Log

Weekly Activity Tracker

Track the tasks done and decisions made each week.

Week 9

Emily: Created a Discord Server for team communication.
Emily: Created GitHub repo.
Emily: Created a skeleton structure for both reports.
Emily: Created an Activity Tracker to log the tasks completed for better task management.

First Team meeting on Wednesday at 2:30 PM.
Decided on PoC: Jake
Decided on Project details:
Server will be: An Open Sourced Web Server (Jake)
3 web Vulnerabilities will be:

- Stored Cross-Site Scripting (Stored XSS)
- SQL Injection (SQLi)
- Unrestricted File Upload / Dangerous File Types

3 privilege vulnerabilities will be:
Assigned Tasks to each member:

- Emily: (Web Vul)
- Aung: (Web Vul)
- Jake: (Priv Vul)
- Vincent: (Priv Vul)

Week 10

Next Team Meeting on Wednesday at 5:00 PM?

Week 11

Week 12