Ruben Boero working with Kai

STRIDE

**Spoofing**

- Monitor the network traffic to the http port (80) to get access to usernames and passwords. Then use the ill-gotten username and password to impersonate another user
    - Exclusively use https

**Tampering**

- Monitor network traffic until you find the admin of the tapir site log in. If they use http to login, a malicious person could gain access to the admin account and change a multitude of things about tapirs unlimited on the server side that only the admin have access to.
    - Exclusively use https
- Using ill-gotten usernames/passwords to modify a users blog posts to make them look bad
    - Exclusively use https (so the username/password can't be stolen in the first place)

**Repudiation**

- Because it's possible to get ahold of usernames and passwords via http, it is very hard to make a claim reputable, especially if the malicious person is using a public computer
    - Exclusively use https
- If a user has access to the chat logs, they could corrupt/destroy the logs
    - Exclusively use https to make it hard for usernames/passwords to get out
    - Protect the logs with a separate password that is not stored on the server

**Information Disclosure**

- Eve listening to user's usernames/passwords
    - Exclusively use https
- Monitoring port 80 to view private chat logs of users
    - Exclusively use https
- Because the site is run by a single person from 1 computer at their home, it's possible to find where that person lives, and gain access to the server that way.
    - Don't host the site from a single IP in your home
- Injection attack to get personal information
    - Code the site such that it is protected from injection attacks

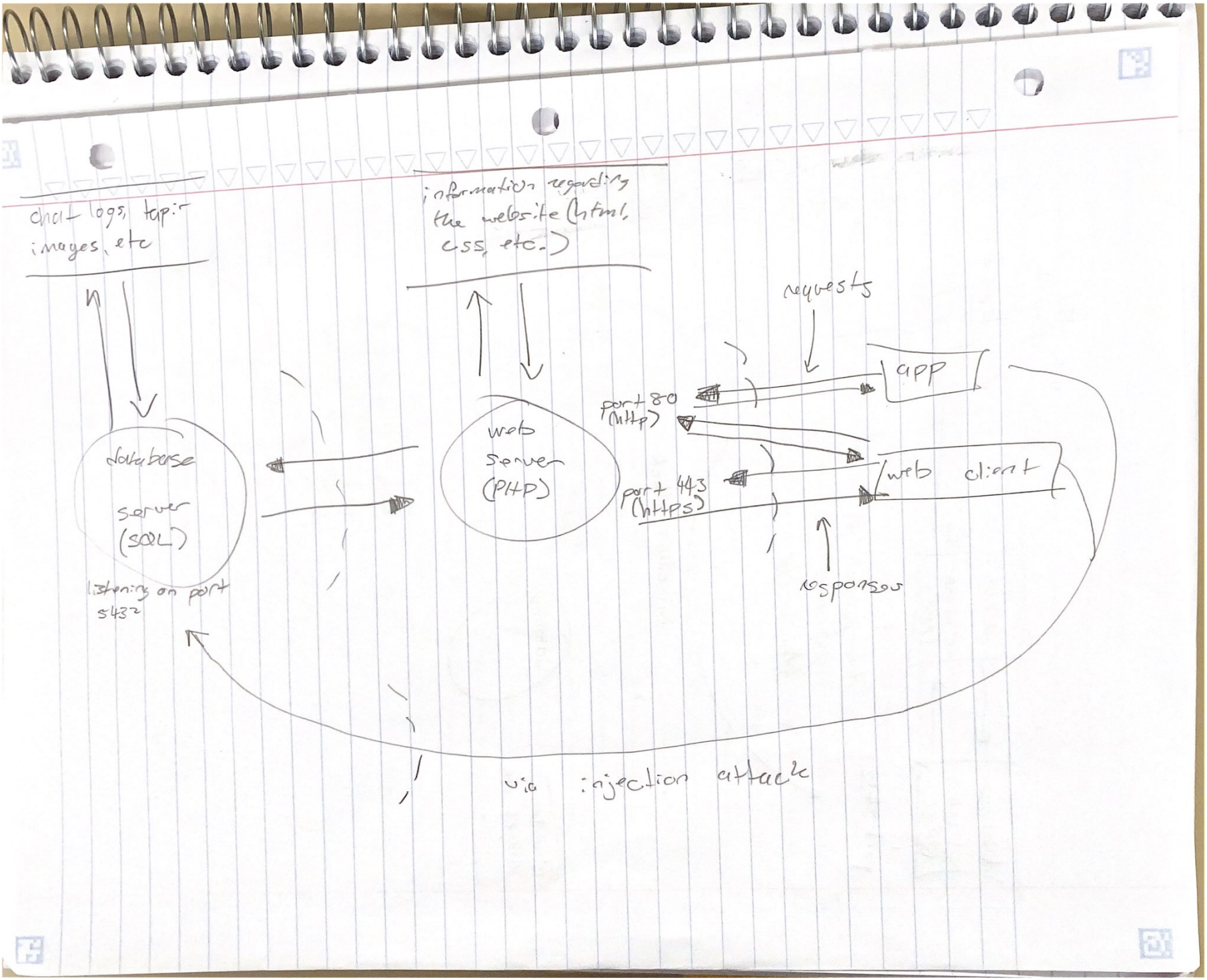
**Denial of Service**

- DDoS attack on the IP of the web server
    - Use a service such as AWS that offers DDoS protection

- By means of the http connection, a malicious person could find the address of the admin of tapirs unlimited, and the address associated with the admin's home computer (the one that runs the web server). This user could then destroy the home computer, killing the site.
    - Exclusively use https so the malicious user can't find the address
    - Use a service such as AWS that has backups and multiple locations for the server
- Injection attack to delete the database
    - Code the site such that it is protected from injection attacks

**Elevation of Privilege**

- A malicious user requesting sensitive information (that they wouldn't otherwise have access to) from the tapir API via a downgrade attack
    - Force the use of a secure version of TLS
- Carry out an injection attack to query the server for all the user's credit card information
    - Make sure the coding of the site isn't vulnerable to injection attacks


chat logs, tap: images, etc
information regarding the website (html, css, etc.)
requests
app
port 80 (http)
port 443 (https)
web client
database server (SQL)
listening on port 5432
web server (PHP)
responses
via injection attack