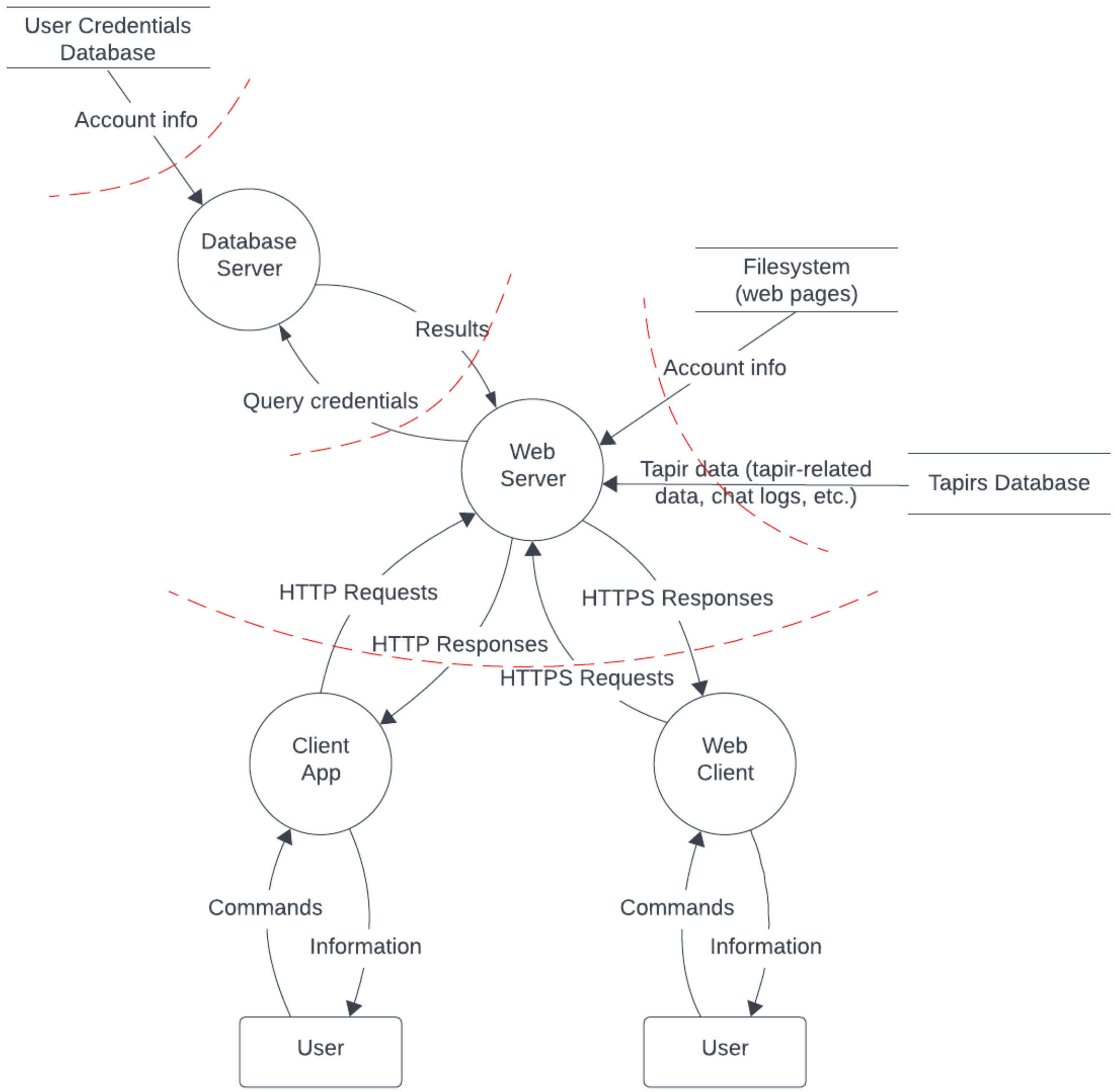
Data flow diagram for Tapirs Unlimited:



Threats against Tapir Unlimited and mitigations:

- An eavesdropper on a user's network reads communication between client app and web server. The eavesdropper grabs the user's username and password from the communication, and then uses the login credentials to view other personal information about the user stored on the server.
    - Fix by encrypting traffic to/from client app (ie. use HTTPS)
    - Falls under **I**nformation Disclosure - eavesdropper gains access to private data
- Malicious attacker intercepts communication between client app and web server when a user tries to set up their account. The user sends a request to set a new username and password, but the attacker intercepts the HTTP packet and changes the password. The user gets an account, but they no longer have access to the correct password for their own account.
    - Fix by encrypting traffic to/from client app
    - Falls under **T**ampering with Data - attacker modifies the password data that the user requested to be set
- If www.twincities.com goes down, the tapir image (currently most of the content on the website) will fail to load on tapirsunlimited.com during that time
    - Fix by storing the tapir image within the filesystem already accessed by the web server to get the web pages
    - Falls under **D**enial of Service - users can become unable to view the picture while the rest of the site stays up
- An attacker could reverse-ip Jeff's home network ip to find the database server's address. They could then query the database server externally and get a list of user information. If the information isn't hashed, then the attacker just has the plaintext information. But even if the information is hashed, the attacker should also have access to the hash function and can try to brute force some passwords.
    - Fix by only allowing the web server to be able to query the database server (and make sure an attacker doesn't get plaintext data by storing password hashes)
    - Falls under:
        - **I**nformation Disclosure - attacker gains access to private data
        - **S**poofing - attacker can subsequently use someone else's account for future activity
        - (possibly) **E**levation of Privilege - if the attack manages to get their hands on an admin password

- Attacker eavesdrops on an HTTP connection to learn a user's credentials, then executes a DDoS attack on the web server by attempting to login with the stolen credentials on a bunch of devices
  - Fix by encrypting traffic to/from client app
  - Falls under **R**epudiation - difficult to link attack back to attacker
- An attacker could write an email to an admin of Tapirs Unlimited, claiming that a data breach has occurred and that they need to change their password. If the admin falls for it, the email links them to a page that looks identical to the Tapirs Unlimited password reset page, but instead sends the information to the attacker (and ensures to ask the admin for the "old" password to verify authenticity to reset their password). The attacker can then use the acquired credentials to login to the web server with admin privileges (and could possibly make themselves an admin).
  - Fix by making admins aware of phishing attacks and create a separate communication channel (ie. via Slack) so that all admins can be aware if a legitimate data breach has occurred
  - Falls under **E**levation of Privilege - attacker gains access to admin privileges
- Someone could create an account posing as some famous figure (e.g. Eli Lilly) and attempt to convince other users that they're the real [insert famous figure]. If they're able to convince enough people, they might be able to cause real damage to the person (or organization) they're impersonating.
  - Fix by either verifying users with a certain size following, or by requiring users to provide ID upon account registration
  - Falls under **S**poofing - attacker poses as someone they're not