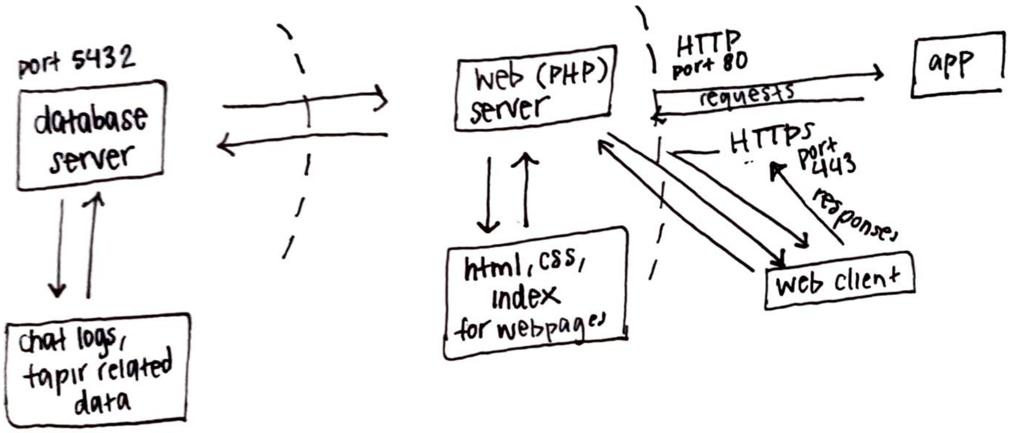
Vanessa Heynes
STRIDE based analysis

Category	Threat	Mitigation
Spoofing	An eavesdropper having access to usernames/passwords being sent across the network in plaintext can login as someone else	Using HTTPS instead of HTTP will encrypt the username and password and prevent the eavesdropper from being able to login
	Phishers can spoof the domain: tapirsunlimited.com and send spam emails posing as this site	Using an email security protocol like TLS would authenticate emails sent by tapirsunlimited.com and ensure that recipients are, in fact, receiving emails from the correct domain
Tampering	An attack on the web server consisting of the user changing the contents of the webpage(s)	Checking who is authorized to modify the site's contents through a digital signature or MAC will prevent a malicious user from being able to attack the web server
	An attacker sending a modified HTTP request header can inject payloads	Using HTTPS will ensure that the HTTP requests and responses are encrypted and will prevent the server from being susceptible to adversary in the middle attacks
Repudiation	An attacker can delete or modify logs to deny any other threats they've attempted	If an attacker can modify logs, this means they were able to gain admin access somehow. Using a digital signature to ensure that only permitted users can manage the logs will prevent malicious users from having admin privileges.
	An internal user with access to the database of all users' information would be able to exploit this and perform an internal attack. Since they	Hashing information like passwords and credit card details will allow tapirsunlimited to keep this data confidential as the actual

	most probably have admin privileges, they will be able to modify logs and deny their activities.	numbers have been put through a hash algorithm.
Information disclosure	An eavesdropper can gain information from error messages revealing sensitive information	Logs can allow the server to know when multiple failed attempts have been made. This could send an alert that there's suspicious activity to be wary of. From there, it would be beneficial to make sure the error message are simpler
	Keeping information like IP addresses and keys in source files or letting users accidentally gain knowledge to files or directory names	Keeping information encrypted through asymmetric and symmetric means will make it difficult for eavesdroppers to reverse engineer what the information means in plaintext
Denial of service	An attacker could deny access to the database server by flooding it with TCP packets and never fully completing the TCP handshake	The server can limit the number of requests it receives from the client.
	An attacker could perform a similar DDoS attack and cause a spike in traffic to a particular page on the site causing it to become overwhelmed	Spikes in traffic could send an alert that suspicious activity could be occurring. From there, it would be beneficial to limit the requests that can be made
Elevation of privilege	An attacker could perform an injection attack to get users' private information from the database server	Implement the site such that code is run with least privilege so processes are only ran at the most secure levels where it can no longer be prone to injection attacks
	Attackers gain username information if all usernames within an organization follow a particular pattern. An attacker could gain access if they're able to exploit an	Using more processes for authentication will allow admins to know if an attacker is trying to gain access

	admin’s password as well	

Data Flow Diagram


port 5432
database server
chat logs, tapir related data
web (PHP) server
html, css, index for webpages
HTTP
port 80
requests
app
HTTPS
port 443
responses
web client