Risk identification

Identify assets:

- Programs:
    - Back-end web application
    - Front-end web application
    - Prometheus
    - Grafana
    - DataDog log agent
- Servers / Environments:
    - Travis CI pipeline
    - Backend Digital Ocean droplet
    - Frontend Digital Ocean droplet
    - DataDog PaaS

Threat sources:

- Hacker / Cybercriminal
- Insider
- Competitors (other groups)
- Environment

Both hackers/cybercriminals, insiders, and competitors are related to as attackers in the subsequent sections.

Threats with risk scenarios:
For our identifications of threats, we are using OWASP's top 10 vulnerabilities for 2021 as well as other sources and personal experience.

1. Broken access control:
    a. An attacker forces the frontend application to target the user profile page URL without authorization.
    b. An attacker accesses Digital Ocean droplets or our Grafana dashboard.
2. Cryptographic Failures:
    a. An attacker eavesdrops on the communication between the frontend and the backend.
    b. An attacker de-hashes users' passwords stored in the database (attacker has access to the database).
3. Injection attacks:
    a. An attacker tries to perform SQL injection on the backend.
    b. An attacker tries to perform XSS on the frontend.
4. Insecure Design:
    a. An attacker tries to utilize the flaw in the application design to perform an attack.
5. Security Misconfiguration
    a. An attacker uses unnecessarily opened ports to attack the system.
    b. An attacker utilizes error stack traces to gain information about the system.
6. Vulnerable and Outdated Components:

    a. An attacker uses the common vulnerability of a system component.
    b. An attacker uses a common vulnerability of application libraries.
7. Identification and Authentication Failures
    a. An attacker performs a brute force attack to guess the user's password.
    b. An attacker tries to utilize default credentials to log in to a system component.
8. Software and Data Integrity Failures
    a. An attacker does a man-in-the-middle attack and modifies user requests on the fly.
9. Security Logging and Monitoring Failures
    a. An attacker performs a successful system attack undetected.
    b. An attacker gains access to DataDog and finds sensitive user information in the logs.
10. Server-Side Request Forgery
    a. An attacker tries the Remote Code Execution on the server behind the firewall.
11. Denial-of-service
    a. An attacker performs a denial-of-service attack on the system.
12. Social engineering
    a. An attacker uses social engineering techniques to persuade the development team to gain access to the system.
13. Lack of data recovery plans
    a. In the event of a successful attack or system corruption, a system doesn’t have backups.
14. Exposed application secrets
    a. An attacker accesses the group’s public repo and finds secrets in plaintext.

Risk analysis

Source	Likelihood	Impact	Risk	Actions
1A	L	M	L	-
1B	L	H	M	-
2A	H	M/H	H	TLS could be implemented for the frontend. Another option is to use VPC Network (a private network within DO that attacker doesn't have access to) for connections between BE and FE.
2B	M	H	H	Investigate the possibility of storing random salt alongside passwords in the database.
3A	L (1)	H	M	-
3B	L (3)	M	L	-
4A	L	M	L	-
5A	L	H	M	Go through the system and close unnecessarily opened ports.
5B	L	M	L	Verify that the stack trace is not attached to any BE responses, nor visible in the FE console.
6A	L (4)	M/H	L	-
6B	L (4)	M/H	L	-
7A	M	M	M	Investigate how to limit failed login attempts.
7B	L	M	L	-
8A	H	M/H	H	Same as 2A.
9A	L	L	L	-
9B	L	L	L	-
10A	L	H	M	Limit outbound ports on the firewall.
11A	L/M	M/H	M	Limit inbound ports on the firewall.
12A	L (2)	H	M	-
13A	M	H	H	Investigate how to do database backups.
14A	H	H	H	Hide remaining secrets and delete unused files.

(1) - Entity Framework with LINQ queries are not susceptible to traditional SQL injection attacks.

(2) - The group can only be persuaded with a sufficient amount of beer :)
(3) - Vue automatically escapes HTML content (https://v2.vuejs.org/v2/guide/security.html).
(4) - Pen test was performed with tools like Metasploit.