PRODUCT REQUIREMENTS

Objective:

To build a security product that scans containers with applications and their dependencies for vulnerabilities that the end user can observe.

Features:

1. Initially, the images need to be cloned one by one and then the scan needs to be performed on each scan and the scan results consist of
    a. Image name
    b. Image ID
    c. List of dependencies (Short)
    d. Vulnerabilities of each dependency or the application
    e. CVSS score
    f. Criticality (Critical, High, Medium, Low, Info)
    g. Scan details (scan type (auto or manual if manual who?), time of scan)

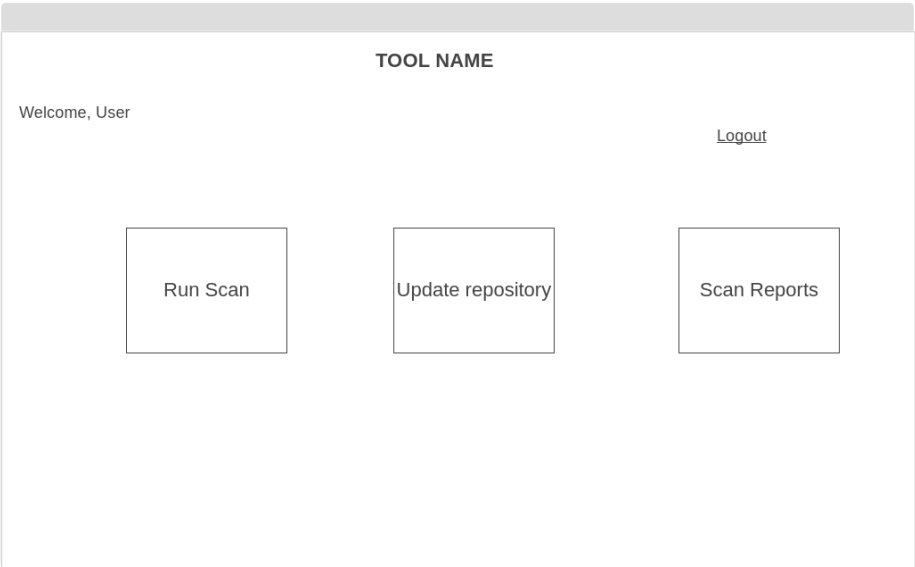
    Will should be stored in the DB.

2. The system needs to fetch the repository of the user and automatically perform a scan once any new changes are identified in the repositories a new scan.
3. If the identified vulnerability of the image is found to be having higher CVSS score then an email consisting of the details of the image, and vulnerability will be triggered automatically to the user.
4. User can search by vulnerability(CVE or type of vulnerability) which gives the list of images that this vulnerability is identified.
5. The user can manually update the repository which will fetch the repository instantly instead of the fetch happening at regular intervals.
6. The scan report should consist of the details of the vulnerabilities that the container has an to be added detailed info on how the vulnerability can be exploited or taken advantage of.
7. The application should also process the report into a PDF format which will be optimised for user understandability and informativeness.
8. The report generated should contain a snip of the code or the dependency.
9. The generated report should be manually checked once by the security analyst for minimising false positives.
10. A vulnerability database that updates periodically needs to be designed so that it becomes easy for us in the case of designing the system and identification of vulnerabilities as well as the remediations. This can be achieved by fetching data from “vulndb” etc.
11. The remediation and the info on the vulnerability can be taken from “Tenable” or something similar that can be added to the report for easy understanding for end users.

Stakeholders

- End User
- Product Manager
- Database Developer
- UI/UX Designer
- Full Stack Developer
- Networking Engineer
- Security Analyst

UX Flows: The developed wireframe can be observed here



TOOL NAME

Welcome, User

Search by

Search by Vulnerability | Search | Search by container | Search

Sort by

Criticality | Container Name | Recently Updated

Tabular representation of the data containing the following items

- Container Name
- Container ID
- Count of vulnerabilities
- Criticality of vulnerabilities
- List of dependencies
- CVSS score
- Scan details

TOOL NAME

Welcome, User

Scan Reports

Scan Report	Scan Date	Scan Time	View Download
Scan_Containerid_date_time	27-07-24	11:10 AM	View Download

System Requirements:

- 99% Uptime
- High computing capacity
- High storage capacity
- Active firewall

Assumptions:

- The end user has adequate knowledge to understand the basic security terminologies.

Constraints:

- Real-time access to the container repository needs to be taken.
- Time constraint increases with increasing repository size.
- High computational and storage is required.

Dependencies:

- An vulnerabilities database with CVSS scores which is updated pereodically.