Sentinel VS - Project Scope Document

Product: Sentinel AVS
Project Manager: Mario Ferrera
Date: 6/10/2024

Introduction

This document outlines the scope of the Sentinel VS software and lists key project requirements for the MVP. Information within this document acts as both a project management tool and an instrument for posterity.

Tables of Contents

1. Project Overview

Sentinel VS is an Automated Vulnerability Scanner tool that aims to identify and report potential security risks and gaps in an organization's applications. It operates on web applications and codebases, helping to improve site security. The goal of this project is

to create a user-friendly AVS tool for businesses looking to quickly and confidently assess their web application's security.

2. Objectives

The objectives of this project include:

- Develop a thoroughly tested and user-friendly AVS tool using Python, Docker, Kubernetes, and OWASP ZAP.
- Offer basic and advanced functionality for users including CLA's, configuration, and scanning types.
- Offer clear and hassle-free scanning reports generated within the tool's environment with the option for external reports to be generated via email or JSON file.
- Engineer standardized testing and debugging features to ensure the accuracy and quality of the tool's features.
- Refactoring throughout the software development lifecycle to improve the codebase, enhance security, and bring it in line with development standards.

3. Scope

The "scope" of a project refers to the immediate technical features, functions, and deliverables that need to be delivered with the MVP. In-Scope items are these MVP features while Out-of-Scope items are those that are not included in this project but are still useful to consider for future releases.

3.1 In-Scope Items

- Python-based AVS tool that runs scans on a web app in any command line interface.
- Able to view pertinent details including vulnerability descriptions, severity, and remediation steps.
- Able to detect a variety of common vulnerabilities and a few uncommon vulnerabilities as documented by OWASP.
- Able to utilize and switch between multiple scanning types and targets.
- Able to utilize filtering and prioritization of certain vulnerabilities.
- Able to configure scans based on several criteria including target URL, scan depth, and more.

- Able to configure the tools itself and have those setting persist after tool is closed.
- Able to generate local reports in common readable file formats including HTML, JSON, and CSV.
- Able to set up basic notification options such as email alerts.
- Includes basic documentation and user guides for installation, configuration, and general use.

3.2 Out-of-Scope Items

- Clean and modern frontend for the tool that allows for visualization of reports.
- Expansion into different targets and platforms such as networks and advanced cloud-based platforms.
- Expansion of tools to include new and rare vulnerabilities in cybersecurity.
- Advanced configuration for the tool.

4. Deliverables

1. AVS tool that runs scans on Windows and Linux computers for simple web apps.
2. Able to detect 10 different types of vulnerabilities.
3. Option to switch between two target types: simple websites and API's.
4. Option to utilize filtering and prioritization.
5. Option to configure scan for Target URL and Scan Depth
6. Option to change up to 4 different user settings and have them persist.
7. Option to generate local reports in HTML, JSON, or CVS format.
8. Option to set up basic email notifications for reports.
9. Includes basic documentation for installation, configuration, and general use.

5. Milestones and Timeline

The milestones for this project are broken into 7 key parts:

- Setup and Planning
- Scanner feature Development
- Advanced Features & Configuration Development
- Reporting & Results Development
- Testing & Debugging
- Documentation Production
- Deployment & Marketing

The expected timeline for this project is to take no more than 7-14 business days starting from this document's creation.

6. Assumptions

- The common vulnerabilities targeted by this project will not undergo any extreme changes within the community.
- The project manager will have the necessary time to complete all project requirements in the timeline.
- The scope of the project may only change slightly depending on the feasibility of certain features and discovered overhead introduced by cumbersome features.

7. Constraints

- The budget for the project is set at 0$ and will not be adjusted.
- The resources for this project are limited to open-source packages.

8. Stakeholders

- Mario Ferrera – Project Manager, Lead Software Developer, and QA Tester.