770 Chapel Street
New Haven, CT 06510

203-401-8768

mydae.org

Project Design Documentation Template:

Last updated: June 13th, 2025

1. Project Title & Version Control

All in one CLI pentesting tool.

Version Control
Version: DRAFT
Date: MM/DD/YYYY
Change Log: N/A

2. Project Summary (2–3 sentences)

I am creating an all in one command-line tool that automates target reconnaissance, active fuzzing of web applications, and all other steps of the pentesting process. This is especially useful for pentesters looking to organize and speed up their work.

3. Problem Statement / Use Case

In the pentesting world, a large sum of the work is repetitive and difficult to execute quickly without plugins. My tool combines all the software used in every step of the pentesting process and compresses it into a quick and easy to use CLI tool.

4. Goals and Objectives

1. A working and usable CLI tool that can run at least 2 tools in each step of the pentesting process.
2. Have clear commands and subcommands that run and navigate the project.
3. Create a configurable, customizable, and saveable environment for each user and their needs.

770 Chapel Street
New Haven, CT 06510

203-401-8768

mydae.org

5. Key Features / Functions

ve all the tools needed to perform the test. It will have a user-friendly command and sub-command systeThe project will create an all in one environment for the pentester to work in. It's designed to ham for ease of use. The main feature that makes this tool stand out is the customizable and saveable functions, removing the need for repetitive work. Overall, the tool to be used for ease of use and customizability.

6. Tech Stack and Tools

Python for the main framework. Linux for developing and testing.

Common Tools to Integrate

Recon:	theHarvester, Shodan, Amass, Sublist3r, whois
Scanning:	Nmap, Masscan, WhatWeb
Vuln Analysis:	Nessus (API), OpenVAS, Nikto, sqlmap
Exploitation:	Metasploit (via msfrpc), CrackMapExec, Hydra, Burp Suite API
Reporting:	Custom script → Markdown, PDF, HTML output

7. Architecture / Workflow Diagram

Algorithm and Flowchart for project

8. Timeline / Weekly Milestones

Week	Outcome
Week 1	Create an initial CLI framework and research tools for each step.
Week 2	Integrate the tools and APIs for the recon phase.

770 Chapel Street
New Haven, CT 06510

203-401-8768

mydae.org

Week 3	Build the output system and test the tool for errors and efficiency.
Week 4	Integrate the port scanning tools and accept user flags.
Week 5	Add fingerprinting tools and store the results.
Week 6	Add vulnerability scanning tools and log the results
Week 7	Label each vulnerability with a risk factor and group results by asset.
Week 8	Build the exploitation wrappers and test them.
Week 9	Add safeguards and the customization functionality to it.
Week 10	Design a tool that helps make a report.
Week 11	Polish it with error handling, colorized outputs, and help menus.
Week 12	Troubleshoot and push to github with the proper documentation.

9. Risks and Risk Mitigation

The largest issue when designing this project is implementing all the external tools without having compatibility issues. To mitigate this, I will add abstraction layers for each tool and give myself ample time for troubleshooting.

10. Evaluation Criteria

How will you know this is done well? List at least 3 measures of success.

11. Future Considerations

What long term maintenance or added functionality will be needed in the future