

ScanMasterX

Team No 10: UwUltimate Stardust Crusaders



Team Details

Lewis



Dennis



Eric



Alexa



Will



Project Overview

- Problem: The Rising Cost Of Cybersecurity Breaches
 - Average Data Breach Cost: \$4.45 million (Forbes)
 - Projected Cybercrime Damages: Over \$10 Trillion by 2025 (Forbes)
- Our Solution: ScanMasterX

Vision

- Lower The barrier to scan and analyze complex networks
- Prevent Data Breaches by identifying vulnerable services
- Perform Automated Security Testing
- Display the Topology and Security Score

Requirements List

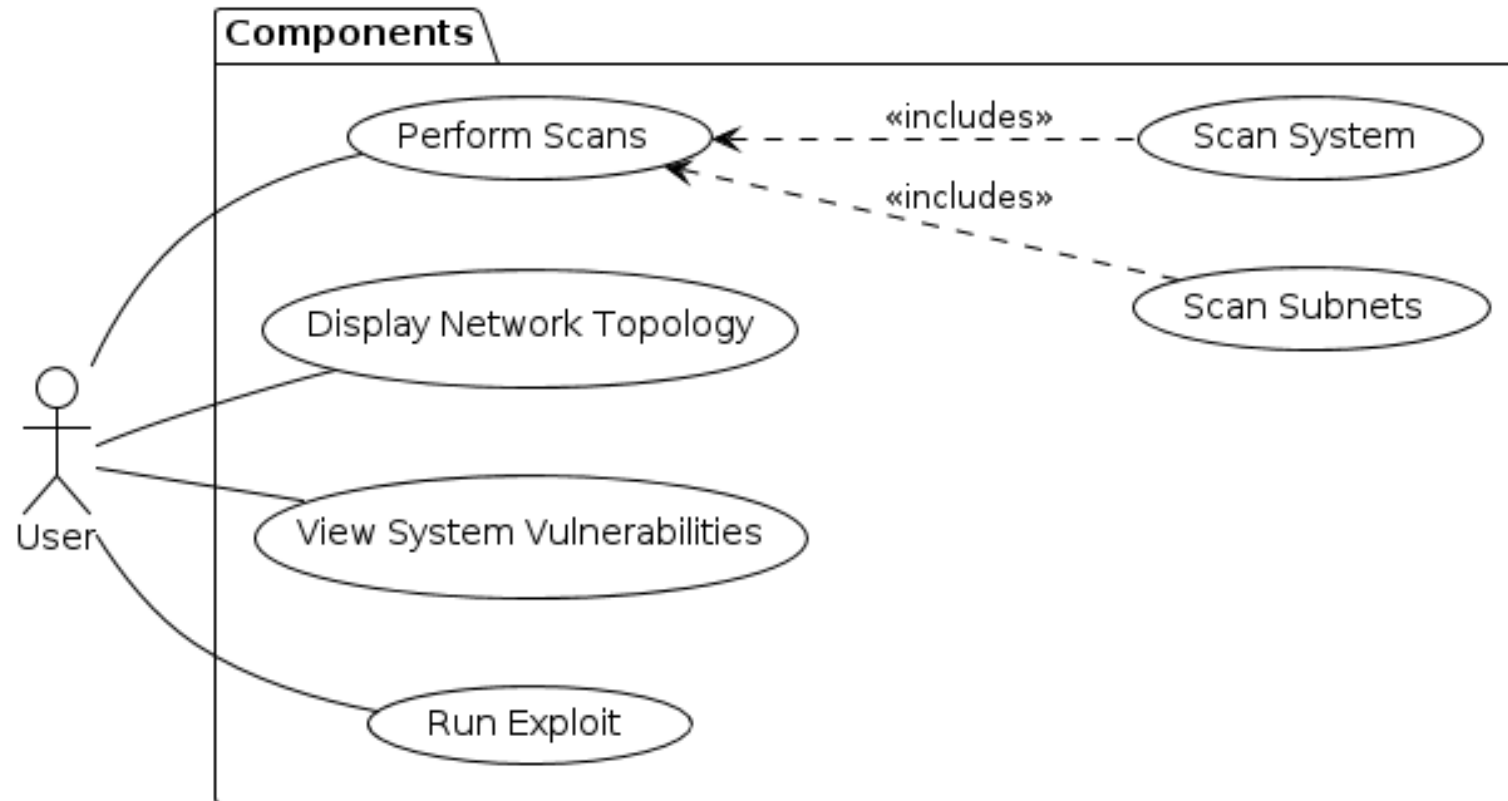
- There must be a feature to identify vulnerabilities on a server.
- There must be a component that will evaluate the security of a given server.
- The user must be able to select which device(s), which ports, or which group of ports, that they wish to test and scan.

Project Solution Approach

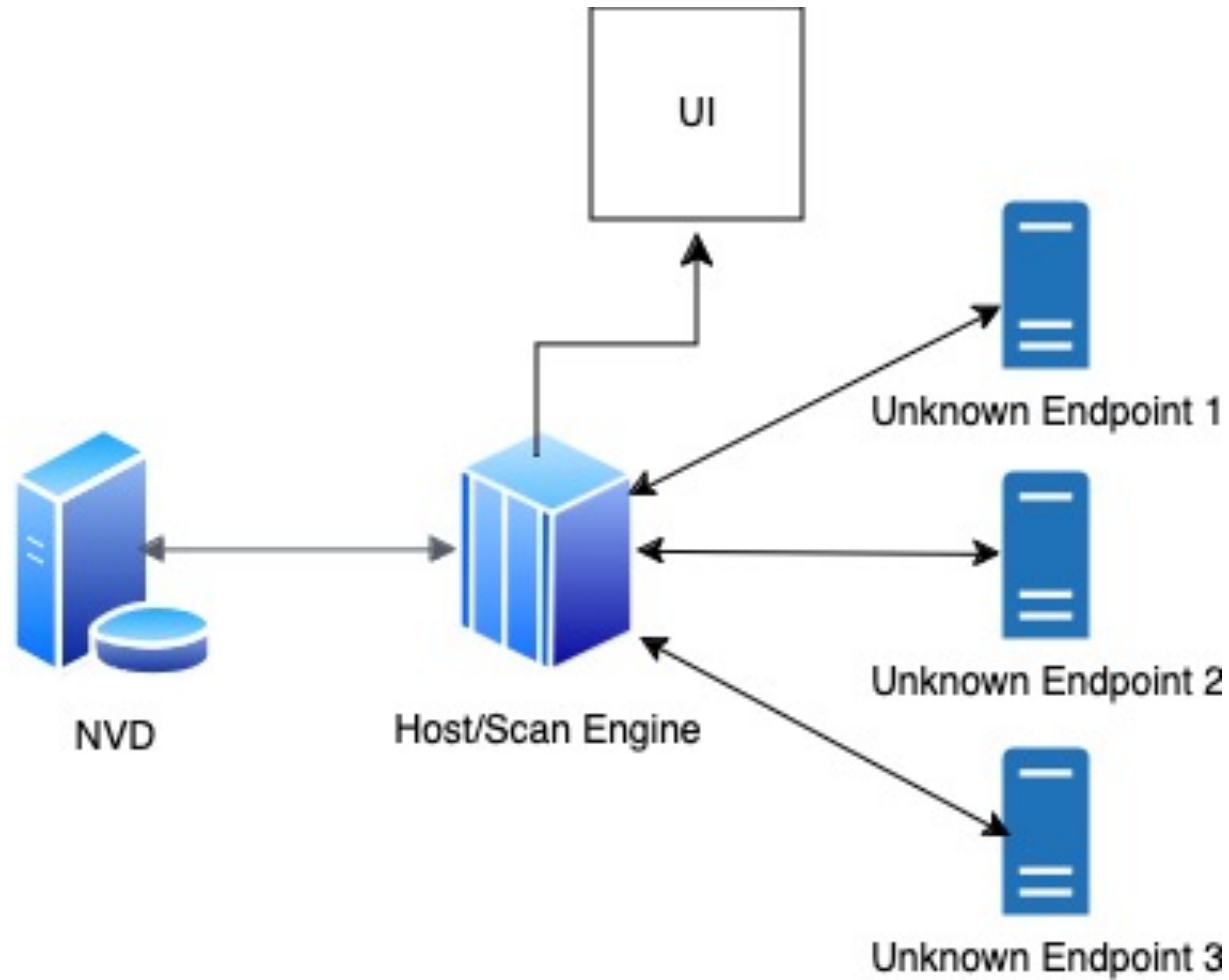
- Major Components
 - Custom scanner leveraging Nmap
 - GUI for user to interact with
- Tools, Frameworks, Platforms, Libraries
 - PYQT - UI framework for Python
 - NMAP – A CLI network scanner
 - CVE Database [TBD]



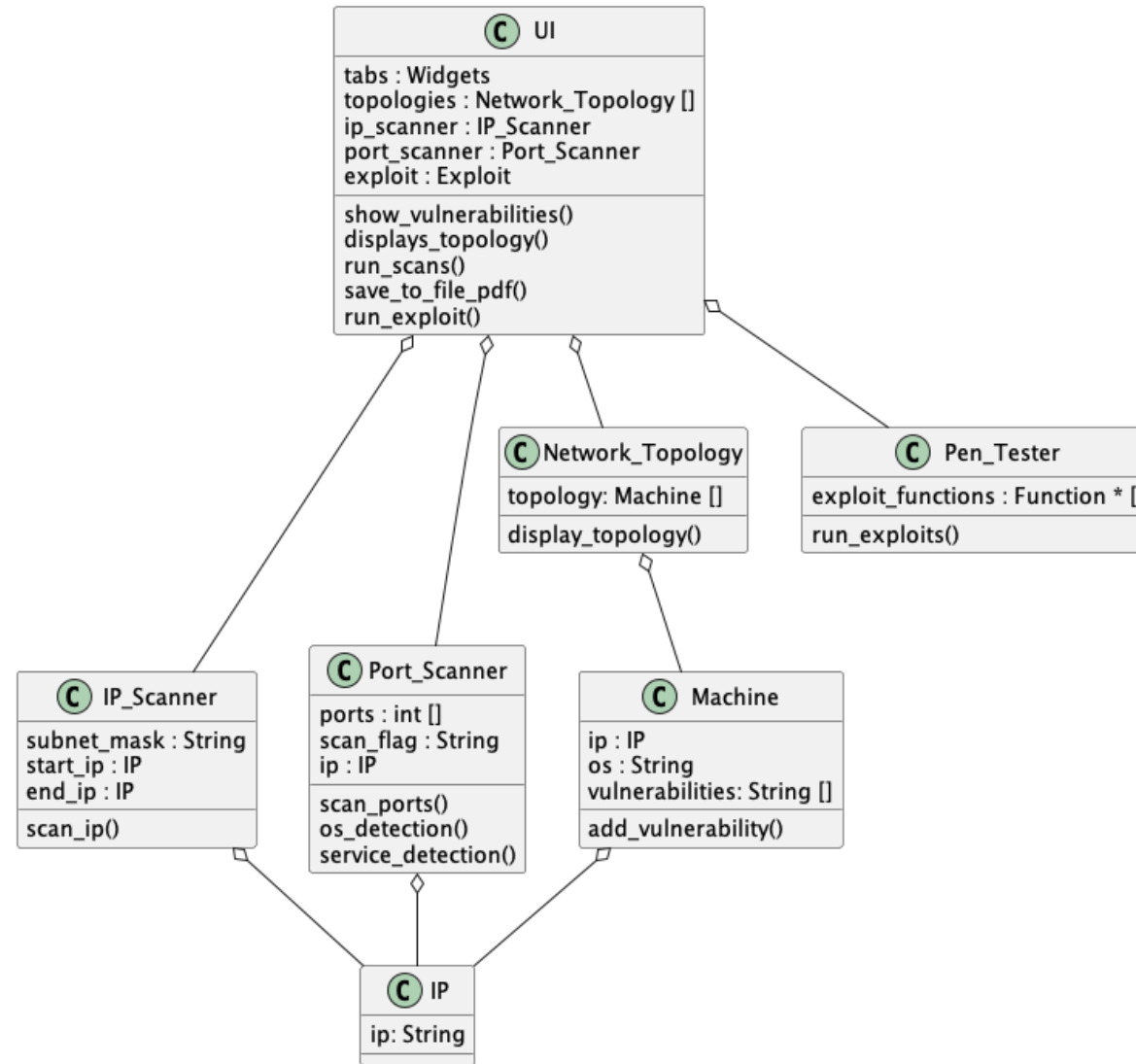
Use Case Modeling



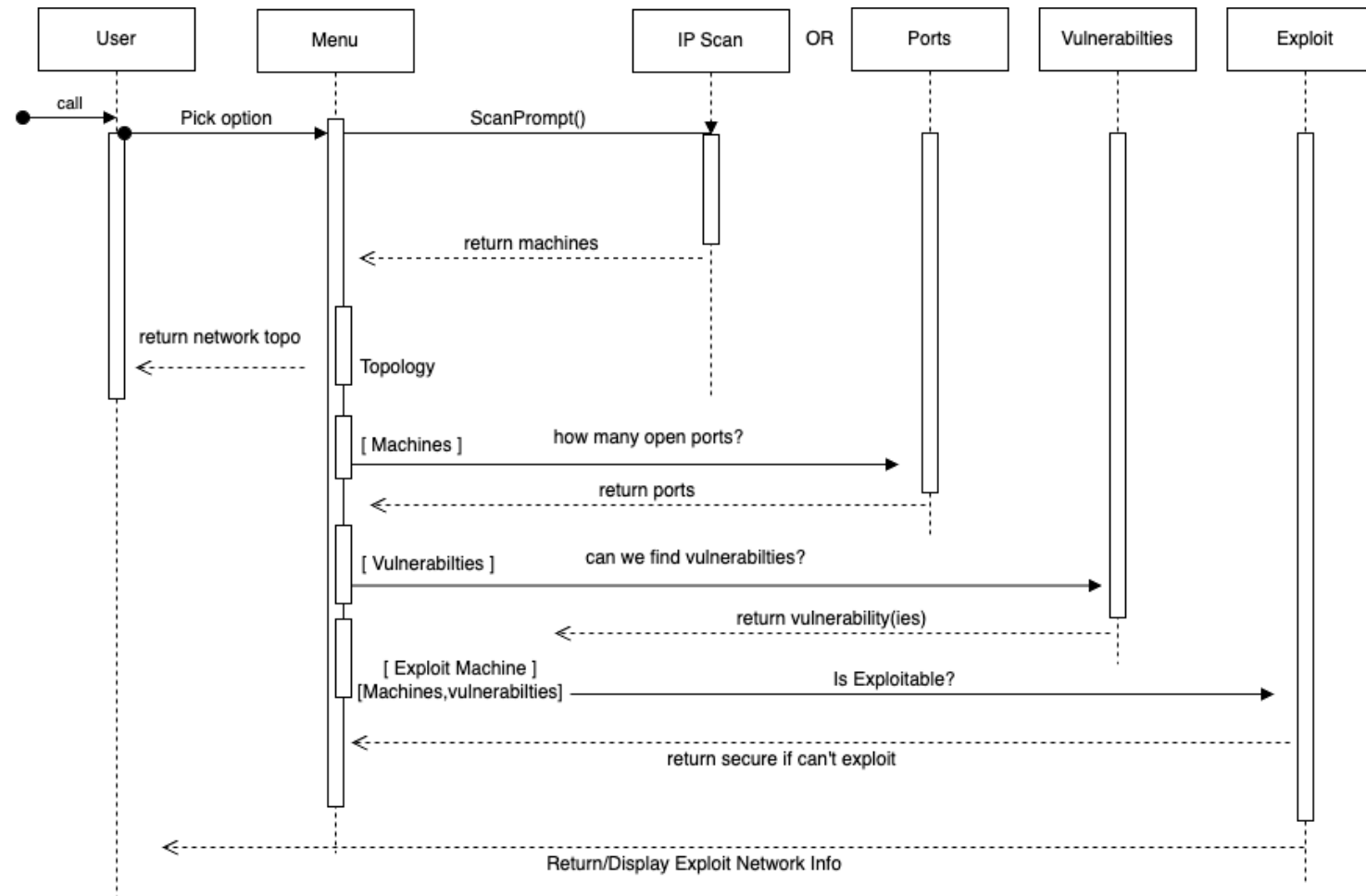
System Architecture Diagram



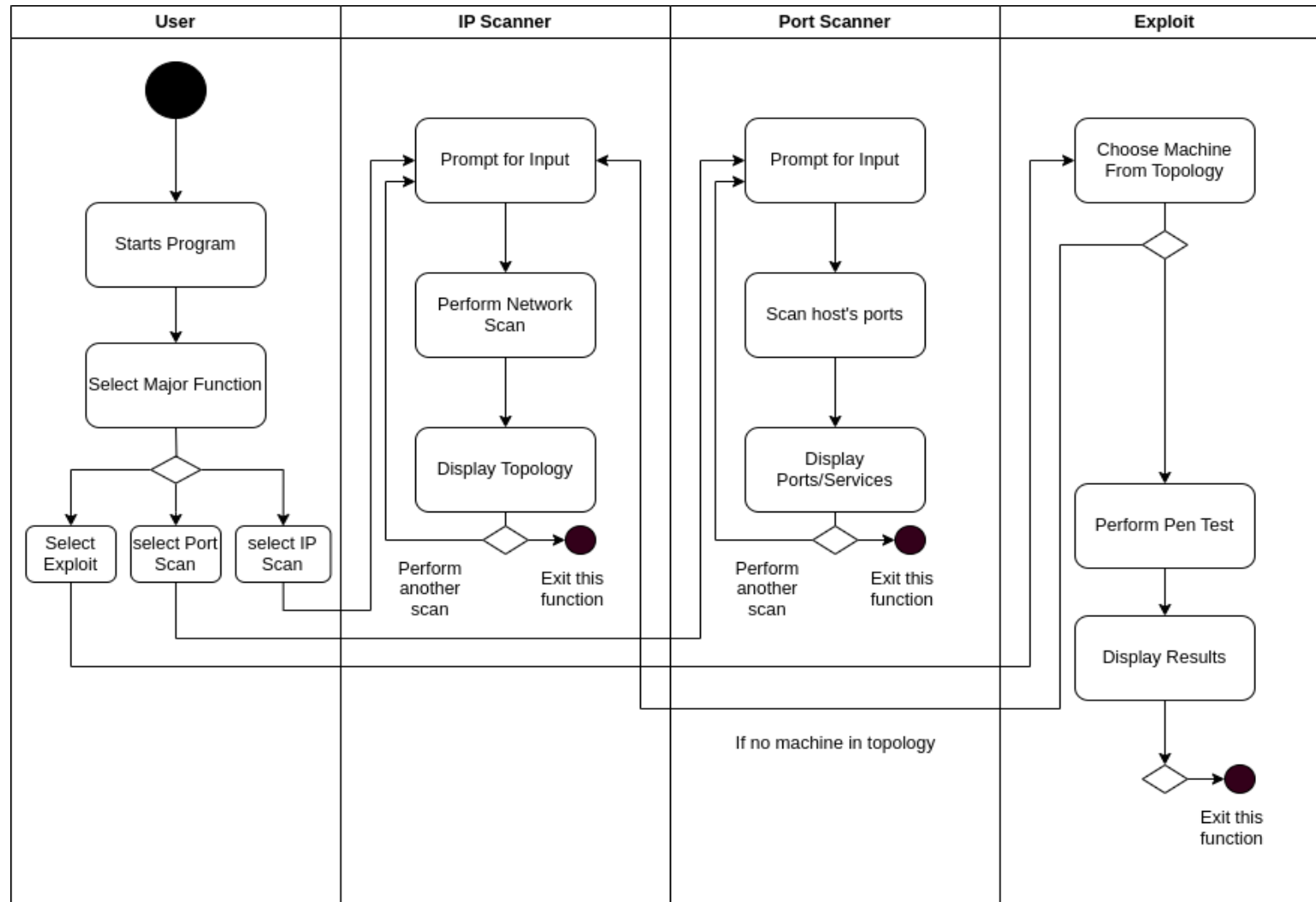
Structural Modeling: Class Diagram



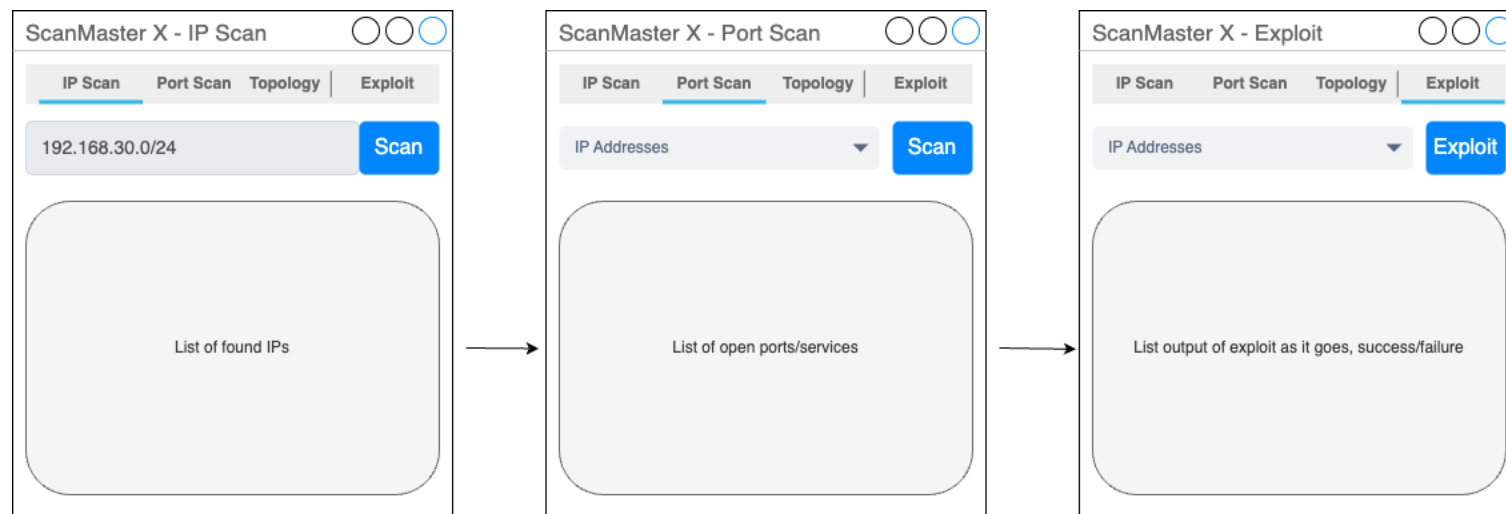
Behavioral Modeling: Sequence Diagram



Behavioral Modeling: Activity Diagram



UI



Demo Time on Exploits!

```
class Pen_Tester:

    def __init__(self):
        self.exploits = [self.login_to_ftp,]

    def run_exploits(self,machine):

        for exploit in self.exploits:
            result = exploit(machine.IP)
        return result

    def login_to_ftp(self,host):
        try:
            with ftplib.FTP(str(host)) as ftp:
                print("IM TRYING TO LOG IN")
                ftp.login()
                print("Login successful!")
                return True
        except ftplib.all_errors as e:
            print(f"Failed to connect or login: {e}")
            return False
```



What's Next??

- Development of the UI
- Displaying the Topology
- Identifying Vulnerabilities

