FIU | FLORIDA INTERNATIONAL UNIVERSITY

NETWORK SECURITY TOOLKIT

Johan Lopez Cabrera · Jean Maxi · Emmanuel Rodriguez · Angel M. Soto · Sasha G. Soto · Mentor: N/A · Instructor: Prof. Masoud Sadjadi

Problem

Entry-level users often need several separate tools just to perform basic security tasks.
Fragmented workflows make learning, testing, and documentation harder than they need to be.
Students and small lab teams benefit from one local toolkit for passwords, scans, and traffic inspection.
The project emphasizes clear output and safe, authorized use instead of complex offensive features.

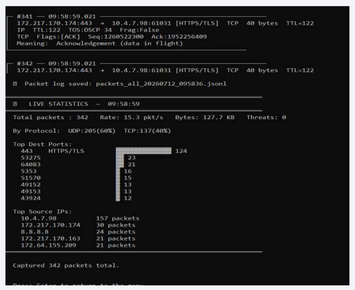
Our Solution

Single Python CLI with one consistent menu.
Password Generator: XKCD-style passphrases with configurable strength options.
Port Scanner: ghost, recon, and full scans plus service probing and JSON reporting.
Packet Analyzer: live capture, packet decoding, top protocols and ports, and JSONL logging.
Design focus: approachable workflow, evidence files, and responsible-use warnings.

Demo & Screenshots

Main menu / module selection

Scan summary with risk counts and JSON export



Packet analyzer statistics and saved JSONL log

Security & Risk (SO6)

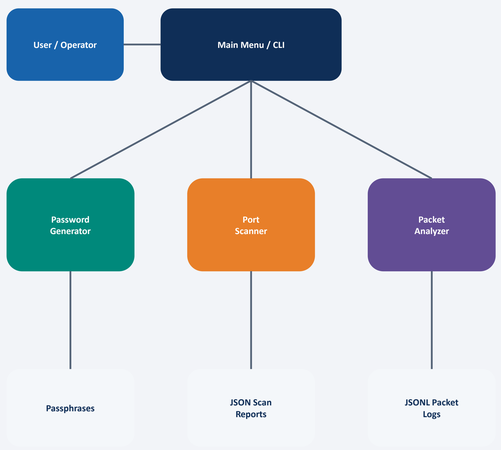
- Unauthorized use → explicit confirmation prompts and safe-use messaging.

Privacy exposure in captures → filter by IP/port and treat logs as sensitive artifacts.
Misleading scan interpretation → provide contextual hardening guidance, not just raw numbers.
Dependency / privilege issues → document Npcap and Windows requirements for reliable capture.
Contingency strategy → keep local demo targets and backup showcase assets ready.

Stakeholders & Requirements

Stakeholders: cybersecurity students, instructors, lab administrators, and small-network operators.
Functional requirements: passphrase generation, phased TCP scanning, service probing, live packet capture, and structured outputs.
Non-functional requirements: Windows-friendly CLI, repeatable demo flow, and ethical-use prompts.

Architecture



Shared prompts, console formatting, and safe-use warnings keep the workflow consistent.

Results & Evaluation

18 OPEN PORTS | **342** PACKETS | **JSON** OUTPUTS

Port scan on localhost produced 18 open ports across repeated tests; risk summary showed 2 critical, 1 high, and 15 medium results.
Packet analyzer sample captured 342 packets, 127.7 KB, and 15.3 packets/second with no signatures triggered in the sample run.
All core modules generated usable outputs and supported a stable Windows-based demo flow.

What's Next

Richer service fingerprinting
Expanded protocol and signature coverage
Installer / packaged executable
Dashboard for report review

Acknowledgments

KFSCIS Capstone Program · Prof. Masoud Sadjadi
Capstone II team deliverables and sprint documentation supported this final package.