Course Project

Instructor: Khaled Diab

Final Project: Objectives

- Learning new concepts
- Gaining hand-on experience
- Making an impact

Final Project

- This is your opportunity to explore or dig deeper in a specific security-related topic.
- Related to **systems** and **networking** topics
- Has to have an implementation component
- **Highly recommended** to discuss with me, TA, and/or in the discussion board

Checkpoints – Hard deadlines

- Jan 31st (next week):
  - Team formation
  - Initial project idea
  - Expect feedback from me

- Feb 12th:
  - Project proposal
  - Accept/reject decision
  - Rejected proposals need to be amended and accepted by Feb 22nd
  - Failure to get a project accepted → Zero-grade Project → Fail the class
  - Alert: Reading break starts on Feb 17th!

Checkpoints – Hard deadlines

- March 20th:
  - Project milestone presentation
  - Each group will present their progress and initial results

- April 8th:
  - Project demo/presentation/poster session
  - Public event

- April 15th:
  - Project code and report

Expected Report Structure

- Abstract
- Introduction: motivation and challenges
- Related work: how does your work compare to similar works?
- Problem statement

Expected Report Structure

- Proposed solution
  - Overview
  - Details
  - Analysis
  - Limitations
- Evaluation
  - Define control parameters
  - Define evaluation metrics
  - Cover the spectrum
- Conclusion and Learned Lessons

Project Ideas – Examples

- Reproducing (complex) Attacks and Defenses
- Reproducing research papers (related to security)
- Implementing security-related tools
- New research ideas
  - New attack/defense
  - New architecture or component

Grading

- Implementation: 30%
  - Working code
  - Code organization
  - Documentation
- Reproducibility: 20%
  - We will not fix your code.
  - You need to submit any required env. to reproduce your implementation
  - We will not create this env. as well
- Novelty: 30%
  - Measures the complexity of the project idea
  - I.e., if you have a working implementation, but the idea is quite simple, you will lose grades.
- Presentations: 10%
- Report: 10%

Open Source Code: Guidelines

- If your project idea is implemented somewhere else:
  - You cannot use that code; you need to implement it by yourself
- What type of libraries can I use?
  - A library that doesn’t directly implement your main/code idea
  - Helper utilities
- If in doubt, aske me.
  - Don’t wait till the end
  - If your implementation is marked as copied, this is as a cheating attempt

Examples

My idea is to create a network mapping tool, can I use `nmap`?

No

My idea is to reproduce “Paper X”. I found its source code online, can I use it?

No

Examples

My idea is to improve “Paper X”. I found its source code online, can I use it?

Check with me first

My idea is to create a ML-based anomaly detection for IDS, can I use `pytorch`?

Okay

Reproducing Attacks and Defenses

- DNS Rebinding Attacks
- SDN-related Attacks
  - The CrossPath Attack: Disrupting the SDN Control Channel via Shared Links
  - Attacking the Brain: Races in the SDN Control Plane
- Bypassing Virtualization/Sandboxing
- Side-channel Attacks

Reproducing research papers

- Examples:
  - BlindBox: Deep Packet Inspection over Encrypted Traffic
  - Embark: Securely outsourcing middleboxes to the cloud
  - The CrossPath Attack: Disrupting the SDN Control Channel via Shared Links
  - Attacking the Brain: Races in the SDN Control Plane
  - …

Implement/improve security-related tools

- One metric if you're improving an existing tool:
  - your code is merged to a popular open source tool

- Security-related dev tools:
  - Static and dynamic code analysis: Discover bugs and vulnerabilities
  - Compiler instrumentation

- Attack-based/enumeration tools:
  - nmap
  - ROPGadget

Implement/improve security-related tools

- Defense-based tools:
  - IDS, IPS, Firewall
- End user tools:
  - TOR (privacy)
  - VPN
- Security-related protocols

New/Other ideas

- New attack/defense
- Security issues in serverless/container platforms
- Are vNICs secure?
- Detecting malicious IoT behavior (large-scale, distributed env.)
- Attacks based on traffic analysis, e.g., Website fingerprinting
- Detecting caching policies in web/video servers
  - Find the worst-case scenario → launch DoS attack
  - Recall the PHP hash collision attack!
- Security of self-driving vehicles
  - Example