Yunsen Lei

Ph.D. Student at Worcester Polytechnic Institute
yunsenlei.dev | ylei3@wpi.edu

I am interested in solving challenging problems in networking and system security. My previous research focuses on bringing Software-Defined Networking to the end host to improve its deployment scalability. Currently, I am focused on threat detection and analysis. In particular, I build systems that facilitate the modeling and understanding of program execution to help organizations defend and respond to evolving cyber threats.

Education

Worcester Polytechnic Institute — *August 2020 – Present*
Ph.D. in Computer Science — *Advisor: Craig A. Shue*

Worcester Polytechnic Institute — *August 2018 – June 2020*
Master of Science in Computer Science

Xidian University — *August 2013 – June 2017*
Bachelor of Engineering in Information Security

Publications

Peer Reviewed

Shuwen Liu, Joseph P. Petitti, **Yunsen Lei**, Yu Liu, Craig A. Shue, "By Your Command: Extracting the User Actions that Create Network Flows in Android," *IEEE International Conference on Network of the Future* (NoF) 2023

Yunsen Lei, Julian P. Lanson, Craig A. Shue, Timothy W. Wood, "Attackers as Instructors: Using Container Isolation to Reduce Risk and Understand Vulnerabilities," *Conference on Detection of Intrusions and Malware & Vulnerability Assessment* (DIMVA) 2023

Matthew M. Tolbert, Elie M. Hess, Mattheus C. Nascimento, **Yunsen Lei**, Craig A. Shue, "Exploring Phone-Based AuthenticationVulnerabilities in Network Identity Systems," *International Conference on Information and Communications Security* (ICICS) 2022

Matthew A. Puentes, **Yunsen Lei**, Noëlle Rakotondravony, Lane T. Harrison, Craig A. Shue, "Visualizing Web Application Execution Logs to Improve Software Security Defect Localization," *IEEE Conference on Software Analysis, Evolution and Reengineering Workshop on Validation, Analysis and Evolution of Software Tests* (VST) 2022

Yunsen Lei, Julian P. Lanson, Remy M. Kaldawy, Jeffrey Estrada, Craig A. Shue, "Can Host-Based SDNs Rival the Traffic Engineering Abilities of Switch-Based SDNs?," *IEEE International Conference on Network of the Future* (NoF) 2020 (**best paper award**)

Yunsen Lei, Craig A. Shue, "Detecting Root-Level Endpoint Sensor Compromises with Correlated Activity," *EAI International Conference on Security and Privacy in Communication Networks* (SecureComm) 2019

Thesis

Yunsen Lei, "Towards Better Kernel and Network Monitoring of Software Actions", Master's thesis, Worcester Polytechnic Institute, 2020

Patent Application

Craig A. Shue, Lane Harrison, Julian Lanson, **Yunsen Lei**, Matthew Puentes, "Method and Apparatus for Identifying a Logic Defect in an Application", US Utility Patent Application Number 17/939,254, filing date September 7, 2022.

Craig A. Shue, Lane Harrison, Julian Lanson, **Yunsen Lei**, Matthew Puentes, "Software Defect and Incident Response Tool", US Provisional Patent Application Number 63/242,595, filing date September 10, 2021.

Corporate Sponsored Projects

Fastly, Inc. : Multi-Process Wasmtime *June 2022 - August 2022*

Wasmtime is a runtime that hosts server-side WebAssembly code. This project explored a method and implemented a prototype that splits Wasmtime into two parts: One part that interacts with Wasm guest code and another part in a different process that holds the host embedder.

- Implemented a mechanism in the Wasmtime codebase that supports parameter interpretations for the Wasm hostcalls.
- Designed and implemented APIs that support Wasm execution to interleave between the guest and host runtime.
- Delivered a proof-of-concept prototype of multi-process Wasmtime.

This project enables an efficient proxy of native API calls in Wasmtime. Furthermore, it significantly improves the security of Wasmtime by adding guest-to-guest and guest-to-host protection.

Academic Research Projects

Context and Flow Sensitive Syscall Filtering for PHP Applications *September 2022 - Present*

The `seccomp`-based syscall filtering is challenging to apply on web applications hosted by an interpreted language runtime. This project introduces a context-aware model that correlates system calls with their preceding application-level events. This fine-grained approach enhances security enforcement, allowing for more effective and targeted syscall filtering in PHP applications.

Key contributions:

- Developed a PHP extension that combines SystemTap's Statically Defined Tracing probes with PHP execution hooks, granting the system kernel extra visibility of API-level events.
- Engineered an eBPF-based dynamic profiler to produce accurate, context-rich execution profiles tailored for PHP applications.
- Developed a static analysis pipeline that generates syscall profiles for each PHP built-in API function and calculates a PHP script's syscall profile by mapping it to a script-level API call graph.
- Implemented a Linux kernel module as a `seccomp` alternative, capable of enforcing fine-grained syscall profiles at the sub-request level through a push-down automaton model.
- Developed a userspace library that exports APIs to instrument the PHP runtime and to control and configure the kernel enforcement module.

Preliminary results: Our methodology produces stringent syscall profiles that significantly restrict PHP application behavior at the sub-script level, all while maintaining comparable overhead to traditional `seccomp`-based approaches.

Single-Use Server for Enhanced Web Application Security *August 2019 - August 2022*

This project employs the principle of compartmentalization to engineer a server platform that dynamically provisions privilege-tailored containers as a single-use server instance for individual users. This approach substantially reduces the server's attack surface and elevates the overall security posture of the system.

Key Contributions:

- Engineered middleboxes for efficient user request dispatching, ensuring a consistent mapping of server instances using user identity.
- Incorporated a customized Kernel Same-page Merging (KSM) technique into the server provisioning pipeline, optimizing server memory utilization by 30%.
- Developed a high-performance PHP profiling extension that captures function execution contexts with around 1μs overhead on function execution time.
- Designed and implemented a log fusion and distillation system that combines PHP call graphs, network logs, and server profiling traces to reconstruct attack incident traces.
- Developed custom exploit scripts for Common Vulnerabilities and Exposures (CVEs) to assess the system's security improvements in a test environment.
- Implemented an `auditd` plugin that fuses `auditd` event with the system's control group information to label each event with container information.

Impact: This work is shown to effectively mitigate the confused deputy attack. In addition, the per-server instance logging capability provides developers with valuable data to help with vulnerability localization. These security benefits are achieved with modest network latency and linear memory overhead.

AppJudicator: Root Cause Analysis of Network Flows via UI Context *May 2021 - May 2022*

This project investigates methods for determining the root causes of network flows by correlating User Interface (UI) context with network activities on Android mobile systems.

Key Contributions:

- Adapted Android's VPN API to function in a `netfilterqueue`-like mode, enabling our app to serve as a non-root firewall capable of intercepting and forwarding packets.
- Contributed to the implementation of a component that fuses an application's intercepted network packets with its UI activities captured in the Android Accessibility Service.
- Engineered a controller component that analyzes both network flows and UI context collected from a client application running on the mobile device.
- Developed UI automation scripts via Appium to test the system across diverse application workflows.

Impact: Our approach enhances the precision of network sensors in identifying malicious flows by integrating UI context, thereby promising a more robust and accurate system for network flow analysis.

Host-Based Software-Defined Networking *August 2018 - May 2020*

Host-Based SDN is promising compared to traditional switch-based SDN in its rule cache scalability and flow visibility. This project examined the traffic engineering capabilities of host-based SDN and explored a collaborating flow reporting mechanism to enhance the system's robustness.

Key Contributions:

- Engineered a kernel network driver using Windows Filtering Platform for the Windows OS, capable of executing OpenFlow-compatible actions for packet modification and forwarding.
- Created an SDN controller module leveraging per-VLAN spanning tree protocol, allowing strategic packet tagging and path selection by the host SDN agent.
- Developed another SDN controller module that cross-references flow reports from multiple SDN agents to identify inconsistencies.

Impact: Our solution enables SDN controllers to interact with host agents within a 5ms network overhead, offering capabilities akin to traditional switch-based SDN. Additionally, it effectively identifies evasive or deceptive agents.

Skills

Research: Proficient in working with Software-Defined Networking and Network Function Virtualization paradigms. Proficient with system-level programming. Experienced in developing new technologies and program analysis tools to solve security-related problems.

Programming: C, Python, Rust, PHP, Shell, C++, Go, Kotlin, R

Frameworks: Windows Filtering Platform, asnycio, angr, Django

Tools: Linux, Latex, Git, Nginx, Docker, Adobe XD