Alexander J. Gaidis

Doctoral Researcher, Brown University CS

+1 (845) 554-8993
agaidis@cs.brown.edu
Providence, Rhode Island
United States
cs.brown.edu/~agaidis/
alexgaidis
ajgaidis
Scholar

Systems security: passionate about OS kernel protection and software hardening.

Education

Ph.D. in Computer Science — *Brown University*
GPA: 3.9/4.0 | Advisor: Vasileios P. Kemerlis — *Sept. 2021 – Present*

M.S. in Computer Science — *Brown University*
GPA: 4.0/4.0 — *Jan. 2019 - Dec. 2019*

B.A. in Computer Science — *Brown University*
GPA: 3.5/4.0 — *Sept. 2013 - Dec. 2018*

Full-Time Visiting Student — *MIT*
GPA: 3.9/4.0 — *Sept. 2017 - May 2018*

Work Experience

See "Research" below for more thorough explanations.

Crash Override — *Remote*
Graduate Intern — *June 2023 – Aug. 2023*

- Advisor: Theofilos Petsios
- Designed and implemented test case minimization system with Intel Processor Trace (PT)

Intel (IPAS STORM) — *Remote*
Offensive Security Research Intern — *May 2022 – Dec. 2022*

- Advisor: Joao Moreira
- Researched applications for Intel Memory Protection Keys (MPK)

Intel (IPAS STORM) — *Remote*
Offensive Security Research Intern — *June 2021 – Oct. 2021*

- Advisor: Joao Moreira
- Designed and implemented runtime library to improve hardware-assisted CFI schemes

Secure Systems Lab, Brown University — *Providence, Rhode Island*
Research Assistant — *Apr. 2020 – May 2021*

- Advisor: Vasileios P. Kemerlis
- Researched and improved `sysfilter`'s ability to handle different runtimes
- Developed a kernel exploit to break kR^X's execute-only memory and code diversification protections

Raytheon — *Portsmouth, Rhode Island*
Systems Engineering & Security Graduate Intern — *May 2019 – Aug. 2019*

- Advisor: Glenn Wojcik
- Engineered a minimal RHEL image for a sea-mine neutralization system

ABA English *Barcelona, Spain*
Software Developer *Jan. 2018 – Feb. 2018*

- Advisors: Brian Subirana & Maria Perillo
- Developed an English proficiency test for Amazon's Echo

Auto-ID Laboratory, MIT *Cambridge, Massachusetts*
Research Assistant *Aug. 2017 – Dec. 2018*

- Advisor: Brian Subirana
- Researched the democratization of IoT voice technologies (e.g., Amazon Echo)

Viral Communications Laboratory, MIT (Media Lab) *Cambridge, Massachusetts*
Research Assistant *Nov. 2017 – May 2018*

- Advisors: David Anderton & Andrew Lippman
- Researched bias in television news through sentiment analysis
- Developed an Ionic mobile phone application to help raise awareness of natural disasters

Setpoint Medical *Santa Clarita, California*
Research Intern *May 2014 – Aug. 2014*

- Advisor: Michael Faltys
- Designed experiments to evaluate RF heating by neuro-modulation devices

Teaching

(All comments below are in addition to holding 1:1 office hours with students.)

CSCI1380: Distributed Systems *Brown University*
Graduate Teaching Assistant *Spring 2023*
Advised on course development and high-level direction

CSCI1650: Software Security & Exploitation *Brown University*
Graduate Teaching Assistant *Fall 2022*
Helped teach 150 CS students software security & exploitation

CSCI1650: Software Security & Exploitation *Brown University*
Head Teaching Assistant *Fall 2019*
Led 5 TAs to help teach 100 CS students software security & exploitation

Artificial Intelligence *Northeastern University Software College (China)*
Teaching Assistant *Fall 2018*
Developed AI projects and homework on game theory, search algorithms, and more

CSCI E-11: The Frontiers of Computer Science *Harvard University Extension School*
Teaching Assistant *Spring 2018, Fall 2018*
Implemented novel IoT conversational learning tool for 200 students

Girls Who Code *Norton LifeLock*
Instructor *Summer 2017*
Used project-based approach to teach CS from Scratch to Java to high-schoolers

CSCI0040: Intro to Scientific Computing and Problem Solving *Brown University*
Teaching Assistant *Spring 2017*
Developed CS exams, problem sets, & projects for 65 engineering students

Publications

(★: *Tier-1 venue*)

Conference Papers

1. ★ **Alexander J. Gaidis**, Vaggelis Atlidakis, and Vasileios P. Kemerlis. SysXCHG: Refining Privilege with Adaptive System Call Filters. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2023
2. **Alexander J. Gaidis**, Joao Moreira, Ke Sun, Alyssa Milburn, Vaggelis Atlidakis, and Vasileios P. Kemerlis. FineIBT: Fine-grain Control-flow Enforcement with Indirect Branch Tracking. In *Proceedings of the International Symposium on Research in Attacks, Intrusions, and Defenses (RAID)*, 2023

Patents

1. Michael C. Gaidis and **Alexander J. Gaidis**. Magnetic domain wall shift register memory device readout, February 10 2015. US Patent 8,951,811
2. Michael C. Gaidis and **Alexander J. Gaidis**. Magnetic domain wall shift register memory device readout, July 8 2014. US Patent 8,772,889

Workshops

1. **Alexander J. Gaidis**, Vaggelis Atlidakis, and Vasileios P. Kemerlis. SysXCHG: Refining Privilege with Adaptive System Call Filters, 2024
2. Di Jin, **Alexander J. Gaidis**, and Vasileios P. Kemerlis. BPF-Box: Hardening BPF against Transient Execution Attacks, 2024

Posters

1. Jamie Gabbay, **Alexander J. Gaidis**, Joao Moreira, Ke Sun, Alyssa Milburn, Vaggelis Atlidakis, and Vasileios P. Kemerlis. NOPout: Dynamic Pruning of Indirect Branch Targets for FineIBT, March 2024
2. Brian Subirana, **Alexander J. Gaidis**, Sanjay Sarma, Richard Cantwell, Jon Stine, Peter Oliveira-Soens, Esteva Tarragò, Ferran Hueto Puig, Prithvi Rajasekaran, and Alex Armengol Urpi. A Secure Voice Name System to Wake Smart Devices, February 2019

Research

SysXCHG *Secure Systems Lab, Brown University*

Software Hardening, Kernel Protection

- **Problem:** Userland processes in Linux are given unfettered access to the system call API, widening the attack surface of the kernel and providing attackers with a rich post-exploitation arsenal.
- **Contribution:** My colleagues and I developed a system call filtering enforcement mechanism called `SysXCHG` that enables programs to run in accordance with the principle of least privilege. At a high-level, `SysXCHG` allows system call filters to be dynamically changed at runtime on `execve`, adapting protection to the currently executing program. To ensure attackers cannot tamper with filters, potentially widening their privileges, `SysXCHG` embeds filters in their corresponding binaries and computes a cryptographic signature over them. `SysXCHG` also includes an optimized installation and filtering mechanism called `xfilter` that improves number-based system call filtering over the current state-of-the-art filtering scheme, `seccomp-BPF`. `xfilter` provides processes with different "views" of the system call table where filtering can occur.
- **Publication:** To appear in CCS 2023.
- **Artifact:** https://gitlab.com/brown-ssl/sysxchg

BPF-Box *Secure Systems Lab, Brown University*

Kernel Protection

- **Problem:** BPF increases the attack surface of the (Linux) kernel and has been used extensively as the underlying mechanism for mounting transient execution attacks against the kernel from userland.
- **Contribution:** My colleagues and I developed a novel security architecture for BPF, called BPF-Box, that protects against transient execution attacks by sandboxing BPF's runtime data in a manner similar to software fault isolation. Additionally, BPF-Box uses static analysis and various domain-specific properties to selectively remove enforcement checks to improve performance without sacrificing security. The design is thorough, considering the security of eBPF, cBPF, and BPF helper functions.
- **Publication:** Submitted to USENIX Security 2024.

SMABlock *Secure Systems Lab, Brown University*

Software Hardening, Kernel Protection

- **Problem:** Memory errors and speculative execution attacks have historically been thought of as two disjoint domains. However, recent attacks, such as speculative probing, have demonstrated that memory corruption and speculative execution vulnerabilities can be combined to bypass memory-safety-based mitigations (e.g., (K)ASLR).
- **Contribution:** My colleagues and I designed and implemented a compiler-assisted solution for blocking such attacks. At a high level, it works by first identifying potentially vulnerable indirect branches before subsequently hardening them with instrumentation that propagates artificial data dependencies to prevent them from dereferencing attacker-controlled pointers during speculative execution. This solution advances the state-of-the-art as it can be implemented solely in software; it does not require complete elimination of side channels or memory corruption vulnerabilities; and it does not completely stop speculation (as with serializing instructions like `lfence`), thus offering good performance.
- **Publication:** Submitted to USENIX Security 2024.

FineIBT *Secure Systems Lab, Brown University*

Software Hardening *IPAS STORM, Intel*

- **Problem:** Control-flow integrity (CFI) solutions built with Intel Indirect Branch Tracking (IBT) technology are too permissive, allowing hijacked control flow to target the entry of most functions.
- **Contribution:** With the help of Intel, I have been designing, building, and testing a fine-grained CFI approach using Intel IBT. The solution adds function prototype hash checks to the Intel hardware feature, ensuring that indirect branches can only target a subset of the total set of functions. Prototype hashes are loaded into a register at a given call site and checked after an "anchoring" `endbr` instruction in the prologue of the callee. Our minimal design is performant and works seamlessly across shared object boundaries.
- **Publication:** Appeared in RAID 2023.
- **Artifact:** https://gitlab.com/brown-ssl/fineibt

NOPout *Secure Systems Lab, Brown University*

Software Hardening *IPAS STORM, Intel*

- **Problem:** Fine-grained control-flow integrity (CFI) schemes built with Intel Indirect Branch Tracking (IBT) technology are too permissive when applied across shared-object boundaries. They often fail to sufficiently harden a system against control-flow hijacking attacksdue to all exported library functions being valid indirect call targets at compile-time.
- **Contribution:** To increase the granularity of IBT-based CFI solutions across shared-objects, I designed and built a run-time library that live-patches code to make unused functions unreachable. This required me to edit the compiler (Clang) and the linker (LLD) to collect and store extra metadata in binaries, which the library could use to make decisions during program startup. The main functionality of the library runs as a constructor, but objects loaded dynamically (e.g. using `dlopen`/`dlsym`) required me to design a tamper-proof scheme to safely edit non-writable data and code pages.
- **Publication:** Appeared in RAID 2023 (part of FineIBT).

MIT Voice Name System *Auto-ID Laboratory, MIT*
Democratized Voice

- **Problem:** IoT voice technologies such as the Amazon Echo or Google Home don't operate in an open, egalitarian way potentially impeding competition in future e-commerce markets.
- **Contribution:** We explored the idea of a generic "wake word" to replace "Alexa" or "Google" so Amazon or Google branding didn't accompany every online purchase through a third-party application. Additionally, we considered a new, device-agnostic way to route traffic from these devices so an order to a big-box retailer doesn't first pass through Amazon or Google's servers giving them the chance to poach customers. These new concepts played into our imagining of commerce of the future, and our prototypes of the technology were presented to the likes of Target, Mango, Coles, and Intel.

Awards

NDSS Student Support Grant Recipient 2024

Service

Paper Reviewing

TIFS IEEE Transactions on Information Forensics and Security 2023

Artifact Evaluation Committee

SEC USENIX Security Symposium 2024
NDSS Network and Distributed System Security Symposium 2024

University Service

Ph.D. Admissions Committee (Member) Dept. of Computer Science, Brown University 2023
Ph.D. Mentorship Program (Mentor) Dept. of Computer Science, Brown University 2022 – Pres.

Community Service

Wildlife Rehabilitation Lisinia Doğa, Burdur, Turkey Spring 2013

Mentorship

Doctoral Students

Simran Kadadi Dept. of Computer Science, Purdue University Summer 2017 – Present
Benjamin Spiegel Dept. of Computer Science, Brown University Fall 2022 – Spring 2023

Master's Students

Shukai Ni Dept. of Computer Science, Brown University Spring 2023 – Present

Undergraduate Students

Jamie Gabbay Dept. of Computer Science, Brown University Spring 2023 – Present
Sierra Rowley Dept. of Computer Science, Brown University Fall 2021 – Spring 2022
Ethan Greenberg Dept. of Computer Science, Brown University Summer 2020

Other Interests

Cycling Raced professionally with wins domestically and a top 20 at the international Tour of Tobago
Visual Art Charcoal works and experimental darkroom photography displayed on Brown/MIT campus
Music Played guitar and saxophone in paid gigs and built guitar effects pedals