Vasileios Kemerlis

Department of Computer Science
Brown University

115 Waterman Street
P.O. Box 1910
Providence, RI 02912-1910, USA
☎ *+1 (401) 863-5787*
🏠 *https://cs.brown.edu/~vpk*
✉ *vpk@cs.brown.edu*

Research Interests

My research explores the intersection of software, hardware, and OS security, focusing on kernel protection, software hardening, hardware-assisted defenses, automated vulnerability discovery through fuzz testing, and information flow tracking.

Education

July 2015 — **Ph.D. in Computer Science**, *Columbia University*, Department of Computer Science, Graduate School of Arts and Sciences, New York, NY, USA.
Thesis: *Protecting Commodity Operating Systems through Strong Kernel Isolation*
Advisor: Prof. Angelos Keromytis

February 2013 — **M.Phil. in Computer Science**, *Columbia University*, Department of Computer Science, Graduate School of Arts and Sciences, New York, NY, USA.

May 2010 — **M.S. in Computer Science**, *Columbia University*, Department of Computer Science, Fu Foundation School of Engineering and Applied Science, New York, NY, USA.
GPA: 4.1/4.33

June 2006 — **B.S. in Computer Science**, *Athens University of Economics and Business*, Department of Informatics, Athens, Greece.
GPA: 8.76/10 (rank: 1^{st}; top 1%)

Employment

2024–present — **Associate Professor**, *Department of Computer Science*, Brown University.

2015–2024 — **Assistant Professor**, *Department of Computer Science*, Brown University.

2008–2015 — **Research Assistant (graduate)**, *Network Security Lab*, Columbia University (advisor: Prof. Angelos Keromytis).

Summer 2013 — **Research Assistant**, *Extreme Computing Group*, Microsoft Research (mentors: Dr. Marcus Peinado, Dr. Weidong Cui).

Summer 2012 — **Research Assistant**, *Autonomic Management Group*, NEC Laboratories America (mentor: Dr. Zhichun Li).

2004–2007 — **Research Assistant (undergraduate)**, *Mobile Multimedia Lab*, Athens University of Economics and Business (advisor: Prof. George Polyzos).

Honors and Awards

October 2025 **Distinguished Artifact Award** (for PickleBall [C.1]),
ACM Conference on Computer and Communications Security (CCS).

October 2025 **Top Reviewer**, ACM Conf. on Computer and Communications Security (CCS).

November 2025 **Finalist** (top 10), Applied Research Paper Award (for IUBIK [C.2]),
Cyber Security Awareness Week (CSAW) Europe, Grenoble INP – ESISAR.

May 2025 **Master of Arts** (ad eundem), Brown University.

October 2024 **Distinguished Reviewer**, ACM Conf. on Computer and Comm. Security (CCS).

May 2024 **Runner-up**, Weirdest Machine of the Year Award (for EPF [C.11]),
IEEE Language-Theoretic Security Workshop (LangSec).

November 2023 **Top Reviewer**, ACM Conf. on Computer and Communications Security (CCS).

July 2023 **Distinguished Paper Award** (for BinWrap [C.12]),
ACM Asia Conference on Computer and Communications Security (ASIACCS).

March 2023 **CAREER Award**, National Science Foundation (NSF).

December 2022 **Finalist** (top 5), Artifacts Competition and Impact Award (for libdft [C.38, Z.2]),
Annual Computer Security Applications Conference (ACSAC).

July 2020 **Outstanding Reviewer**, International Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA).

March 2020 **Fellow**, Greek Diaspora Fellowship Program (GDFP), Institute of International Education | Fullbright Foundation – Greece | Stavros Niarchos Foundation.

November 2018 **Finalist** (top 10), Applied Research Paper Award (for CCR [C.20]),
Cyber Security Awareness Week (CSAW), NYU Tandon School of Engineering.

August 2015 **Nominee**, Most Innovative Research Award (for ret2dir [C.30]), Pwnie Awards.

November 2014 **1st place winner**, Applied Research Paper Award (for ret2dir [C.30]),
Cyber Security Awareness Week (CSAW), NYU Tandon School of Engineering.

November 2012 **Finalist** (top 10), AT&T Applied Research Paper Award (for kGuard [C.36]),
Cyber Security Awareness Week (CSAW), NYU Tandon School of Engineering.

July 2007 **Ericsson Award of Excellence in Telecommunications** (for B.S. thesis),
Ericsson Hellas.

December 2006 **Valedictorian**, Athens University of Economics and Business,
Department of Informatics.

June 2006 Graduated **summa cum laude**, Athens University of Economics and Business,
Department of Informatics.

Research Activities

2015–present **Secure Systems Lab (Director, Assoc. Professor)**, Brown University.

▶ Kernel Protection [C.2 – C.6]. Introduced **BeeBox** [SEC '24]: a novel security architecture that hardens (e)BPF against transient execution attacks (*i.e.,* Spectre-PHT, Spectre-STL), allowing the OS kernel to safely expose (e)BPF functionality to unprivileged applications. BeeBox sandboxes the (e)BPF runtime against speculative code execution in an SFI-like manner, and by using a combination of static analyses and domain-specific properties, it selectively elides enforcement checks, thereby improving performance without sacrificing security. I also co-designed **SafeSLAB** [CCS '24] and **ISLAB** [ASIACCS '24]. The former is a heap-hardening extension that aims to mitigate use-after-free vulnerabilities, in kernel code, via an innovative and efficient address aliasing approach. SafeSLAB assigns multiple virtual aliases to each kernel memory page in the system, and manages their access rights via Intel's MPK/PKU technology. This allows it to drastically reduce the number of page table modifications, while blocking dangling pointers effectively. ISLAB introduces a set of novel (SLAB-based) heap hardening techniques that aim to ensure the integrity of kernel-managed memory by segregating memory-management metadata, from corruptible memory objects, into shadow memory. Lastly, I co-designed **IUBIK** [S&P '25]: a kernel-memory compartmentalization scheme that hinders exploitation by isolating attacker-controlled data in shadow memory, preventing them from interacting with sensitive kernel objects, using ARM's MTE and PA technology.

▶ Software Hardening [C.1, C.3]. I designed **Eclipse** [CCS '24]: an innovative protection scheme against speculative memory-error abuse (SMA) attacks, which combine memory corruption vulnerabilities with Spectre-like primitives. Eclipse works by propagating artificial data dependencies onto sensitive data, preventing the CPU from using attacker-controlled data during speculative execution. It provides comprehensive protection against speculative probing and PACMAN, the two most prominent SMA attacks in x86(-64) and ARM. In addition, I co-designed **PickleBall** [CCS '25] to help Machine Learning (ML) engineers load pickle-based models safely. PickleBall statically analyzes the source code of a given ML library and computes a deserialization policy that specifies a safe load-time behavior for pickled models; it then enforces that policy via a drop-in, pickle-module replacement. PickleBall *received* the Distinguished Artifact Award at ACM CCS 2025.

2015–2024 **Secure Systems Lab (Director, Asst. Professor)**, Brown University.

▶ Kernel Protection [C.11, C.13, C.16, C.22, J.2]. The advent of strict memory isolation mechanisms between kernel and user space (*e.g.,* SMEP/SMAP, PXN/PAN) has resulted in the increased use of code-reuse techniques for exploiting memory errors in kernel code. To deal with this problem, I designed and co-developed **kR^X** [EuroSys '17, TOPS '19]: a kernel hardening scheme, which builds upon the concepts of execute-only memory and fine-grain code diversification, for combating ROP/JOP/COP and similar code-reuse attacks, including (in)direct JIT-ROP, without relying on a hypervisor or any other super-privileged component. In addition, to aid both OS kernels and userland applications protect critical and sensitive data against data-only attacks (*i.e.,* attacks that exploit memory safety vulnerabilities to corrupt, or leak, data without hijacking the control flow), I co-designed **xMP** [S&P '20]: a system for providing dynamic, intra-{kernel, process} memory isolation and pointer integrity, as an OS service. xMP offers selective memory protection similarly to Intel's MPK/PKU technology, but (a) it supports more protection domains (512 vs. 16), and (b) it does so without relying on features that are only available on high-end CPUs.

I also contributed to the design of μ**SCOPE** [RAID '21]: a framework for investigating opportunities for least-privilege separation in kernel code. μSCOPE replaces expert-driven, semi-automated analyses with a general methodology for exploring different security vs. performance design points, by adopting a quantitative approach to privilege analysis. Lastly, I introduced **EPF** [ATC '23]: a set of novel kernel exploitation techniques, which rely on abusing the (e)BPF infrastructure, for bypassing memory safety protections that revolve around strong kernel-userland separation. As part of this study, I also designed a series of defenses for hardening OS kernels against EPF-style attacks with minimal overhead. EPF was the *runner-up* for the Weirdest Machine of the Year Award at IEEE LangSec 2024.

► Software Hardening [C.7, C.8, C.12, C.15, C.17 – C.20, C.23 – C.25, C.27, J.1]. Code reuse has become the de facto technique for exploiting memory corruption vulnerabilities. To protect binary-only software against ROP/JOP/COP, or other similar code reuse attacks, including (in)direct JIT-ROP and BROP, I co-designed **Shuffler** [OSDI '16]: a system that continuously re-randomizes the code of a running program, including itself, thwarting end-to-end code-reuse attacks by rapidly obsoleting (leaked) code layouts. In the same vein, I contributed to the design of **CCR** [S&P '18] and **Nibbler** [ACSAC '19, DTRAP '20]: the former is a hybrid compiler-rewriter framework that enables fast and robust fine-grain code randomization on end-user systems, by augmenting binaries with transformation-assisting metadata; the latter is a debloating framework that erases unused code in binary shared libraries, boosting defenses like Shuffler. In addition, I co-designed **Egalito** [ASPLOS '20]: a layout-agnostic, binary re-compiler that performs precise binary analyses to modify/augment binaries without relying on patching or virtualization. Furthermore, I introduced **sysfilter** [RAID '20]: an Egalito-based tool for restricting the system call set available to userland processes in a precise, complete (*i.e.,* safe over-approximated), and scalable manner. Egalito, Nibbler, and sysfilter were *selected* by DoD/ONR to further explore for potential transition to practice (TTP) opportunities. sysfilter has also been *adopted* by Star Lab (*i.e.,* a subsidiary of Wind River) as part of their Linux-hardening solution (Kevlar Embedded Security). I also introduced **SysXCHG** [CCS '23]: an enforcement mechanism for system call filtering that enables programs to run in accordance to the principle of least privilege. SysXCHG empowers applications to run with "tight" system call filters, uninfluenced from any future-executed (sub-)programs, by allowing filters to be dynamically (and securely) exchanged at runtime. To harden PHP code against attacks that abuse deserialization vulnerabilities, I co-designed **Quack** [NDSS '24]: a static analysis framework that automatically restricts the set of classes allowed at deserialization statements, via means of novel duck typing inference, effectively limiting program-code reuse through object injection (*i.e.,* property-oriented programming). Moreover, to protect C++ binaries from `vtable` hijacking (a prevalent C++ exploitation technique), I co-designed **VTPin** [ACSAC '16]: a framework for armoring applications, which cannot be re-compiled, or modified, against `vtable` hijacking through use-after-free bugs. Lastly, I designed and co-developed **BinWrap** [ASIACCS '23], **NaClDroid** [ESORICS '16], and **DynaGuard** [ACSAC '15]: BinWrap protects Node.js applications against attacks that abuse memory safety vulnerabilities in binary add-ons; NaClDroid sandboxes native code in Android applications; and DynaGuard thwarts canary brute-force attacks. BinWrap *received* the Distinguished Paper Award at ACM ASIACCS 2023.

► Side Channels [C.28]. Co-invented the first cache-based side channel attack that can be entirely executed in JavaScript context [CCS '15]. Proposed a set of techniques for: (a) tracking browsing activity, even when the "private browsing" mode is used; (b) constructing covert channels inside the JavaScript sandbox; and (c) detecting certain hardware events.

▶ Fuzz Testing [C.10]. Introduced **IvySyn** [SEC '23]: the first fully-automated framework for discovering memory error vulnerabilities in Deep Learning (DL) frameworks. IvySyn leverages the statically-typed nature of native, low-level DL-framework APIs to automatically: (1) perform type-aware mutation-based fuzzing on C/C++ (DL-kernel) code; and (2) synthesize managed (Python) code that triggers the identified vulnerabilities via high(er)-level, {user, developer}-accessible APIs. IvySyn uncovered 61 previously-unknown security vulnerabilities in TensorFlow and PyTorch, and was *awarded* 39 unique CVEs (many of *high* severity).

▶ Hardware Security [C.9, C.14, C.21, P.1]. Co-designed **Polyglot** [HOST '17]: the first hardware-based instruction set randomization scheme that (a) utilizes strong encryption (AES and ECC), (b) supports code sharing, and (c) is applicable to the entire software stack (bootloader, hypervisor, OS kernel, user applications). Polyglot (naturally) protects against code injection attacks, but can also mitigate code reuse if combined with leakage-resilient code diversification. I also co-designed **EPI** [SEED '21]: a hardware-assisted scheme, for microcontrollers and embedded devices, which protects pointers against memory corruption-based attacks by leveraging efficient memory tagging and metadata-embedding techniques. Lastly, I introduced **FineIBT** [RAID '23]: a CFI enforcement mechanism that improves the precision of hardware-assisted CFI solutions, like Intel IBT, by instrumenting program code to reduce the valid/allowed targets of indirect forward-edge transfers. The instrumentation of FineIBT is compact, incurring low runtime and memory overheads, and generic, so as to support different CFI policies. FineIBT has been *adopted* by Intel and (a variant of it) was more recently *upstreamed* to the Linux kernel.

▶ Robotics Security [C.19]. Co-designed and performed the first large scale scan of the entire IPv4 Internet address space for exposed instances of the Robot Operating System (ROS; the most widely-used robotics software platform) [ICRA '19]. Identified numerous publicly-accessible ROS hosts that allowed access, in an unauthorized manner, to robotic sensors and actuators, and demonstrated (with permission) the risks to user safety and privacy by leaking image sensor information from, and actuating, physical robots present at major US universities.

2008–2015 **Network Security Lab**, Columbia University.

▶ Kernel Protection [C.30, C.36, M.1]. Modern OSes employ a virtual memory model that trades strong isolation for performance. I investigated the security ramifications of weak user/kernel address space separation, and designed and implemented **kGuard** [SEC '12, ;login: '12]: a system to protect Linux/BSD kernels from attacks that exploit the weak segregation of address spaces. In addition, I introduced **ret2dir** [SEC '14]: a new exploitation technique that enables the complete circumvention of numerous software and hardware kernel protection mechanisms, including Intel's SMEP/SMAP and ARM's PXN.

▶ Software Hardening [C.35]. Applications can be logically separated to parts that face different types of threats or suffer dissimilar exposure to a particular threat. Based on this observation, I co-developed Virtual Application Partitioning (**VAP**) [CCS '12]: a technique that allows the selective and targeted application of various protection mechanisms to different software parts. Furthermore, I introduced a methodology for automatically slicing software, using a binary monitor and an intrinsic application property (user authentication), to dynamically adapt the defences being deployed by switching between protection mechanisms like dynamic taint analysis and instruction-set randomization.

▶ Data Flow Tracking [C.31, C.38, C.39]. Dynamic data flow tracking (DFT), also referred to as information flow tracking, deals with tagging and tracking data of interest as they propagate during program execution. I designed and implemented **libdft** [VEE '12]: a dynamic DFT framework that unlike previous work is at once fast, reusable, and works with commodity software and hardware. I explored different approaches for implementing efficient instruction-level data tracking, introduced a performant and 64-bit capable shadow memory, and identified common pitfalls responsible for the excessive run-time overhead of similar tools. In addition, I co-developed a set of techniques to further reduce the slowdown of DFT frameworks, by combining static and dynamic analysis. **TFA** [NDSS '12] separates program logic from tracking logic, extracts the semantics of the latter, and uses compiler optimizations to eliminate redundant tracking. **ShadowReplica** [CCS '13] accelerates DFT, and other shadow memory-based analyses, by decoupling analysis from execution and using spare CPU cores to run them in parallel.

▶ Anonymity Systems [C.33]. **CellFlood** [ESORICS '13] is a DoS attack that I co-developed, against Tor onion routers, which exploits a design flaw in the way Tor software builds virtual circuits. I studied the feasibility and implications of CellFlood, and demonstrated that an attacker needs only a fraction of the resources required by a network DoS attack for achieving similar damage. Furthermore, I contributed to the design and implementation of an effective solution to the problem that relies on cryptographic client puzzles.

▶ Cloud Auditing [C.34, C.32]. I contributed to the design and implementation of **CloudFence** [RAID '13]: a framework that allows users to independently audit the treatment of their data, by third-party services, through the cloud provider that hosts these services. CloudFence is built on top a fine-grained DFT framework that I developed (libdft), and besides data auditing it enables service providers to confine the use of sensitive data in well-defined domains, offering protection against inadvertent leaks and unauthorized accesses.

▶ System/Network Deception [C.41, C.42, J.3, P.2 – P.6]. **BotSwindler** [RAID '10] is a bait-injection system designed to delude and detect crimeware, by forcing it to reveal itself during the exploitation of monitored information. I contributed to the design and implementation of BotSwindler, which relies upon an out-of-host software agent that drives user-like interactions inside a virtual machine, seeking to convince malware residing within a guest OS that has captured legitimate credentials. In addition, I co-developed a novel trap-based architecture for enterprise networks that detects "silent" attackers who are eavesdropping on network traffic [WiSec '10, JCS '12].

Summer 2013 **Extreme Computing Group**, Microsoft Research.

Design and implementation of **RETracer** [C.26]: a debugging extension that leverages type information for triaging crash dumps. RETracer [ICSE '16] offers enhanced crash dump classification by utilizing static taint analysis, opportunistic reverse (concrete) execution, and a new concept that I co-developed, named backward data-flow graphs. RETracer has been *adopted* by Microsoft as the primary tool for triaging crashes (*i.e.*, part of the Windows Error Reporting platform).

Summer 2012 **Autonomic Management Group**, NEC Laboratories America.
Worked on **`AAPL`** [C.29, P.4]. AAPL [NDSS '15] is a static analysis framework that uses data flow tracking to vet Android Apps for component hijacking vulnerabilities (permission leakage, unauthorized data access, intent spoofing). I designed and developed a novel conditional tracking scheme that leverages constant folding/propagation techniques for improving the accuracy and detection rate of the framework.

Spring 2007 **Web Info. Management Group**, Athens University of Economics and Business.
Architectural design and implementation of **`MODS`**: a secure location management system for tracking spatially distributed mobile data. Investigated the security aspects related to the mobile context, namely integrity and authenticity, and developed an efficient lightweight protocol for carrying out the proposed architecture.

2004–2007 **Mobile Multimedia Lab**, Athens University of Economics and Business.
Worked on the Peer-to-Peer Wireless Network Confederation (**`P2PWNC`**) project [C.45, C.47, M.2, D.1 – D.3]. P2PWNC focuses on the reciprocal provision of Internet access to mobile users through voluntary-controlled wireless access points.

Impact and Technology Transfer

`FineIBT` [C.9] Adopted by Intel and (a variant of it) upstreamed to the Linux kernel (April 2022 and onward).
`https://tinyurl.com/ha22jf54`

`IvySyn` [C.10] Uncovered 61 previously-unknown security vulnerabilities in TensorFlow and PyTorch (*i.e.*, the two most popular DL frameworks), and was awarded 39 unique CVEs.
`https://bit.ly/3RqlDZO`

`sysfilter` [C.15] Adopted by Star Lab (*i.e.*, a subsidiary of Wind River) as part of their Linux-hardening solution (Kevlar Embedded Security) for attack surface reduction (November 2021 and onward).
`https://tinyurl.com/ycky6v4w`

`RETracer` [C.26] Adopted by Microsoft as the primary tool for triaging crashes; part of the Windows Error Reporting (WER) platform (March 2015 and onward).
`https://goo.gl/t8CfHr`

`"Spy in the Sandbox"` [C.28]

- Apple limited the time resolution of WebKit's performance API (iOS 9 and onward).
`https://goo.gl/EqCK4a` | APPLE-SA-2015-09-16-1 | CVE-2015-5825
- Mozilla reduced the resolution of `performance.now` in Firefox (v41 and onward).
`https://goo.gl/QmAqII` | MFSA2015-114 | CVE-2015-7327
- Tor decreased the time precision of JavaScript in the Tor Browser (v5.0.1 and onward).
`https://goo.gl/BixTBT` | ticket #1517
- W3C TAG finding, *Unsanctioned Web Tracking* (Finding 17, July 2015).
`https://goo.gl/rKpRR9`

`ret2dir` [C.30]

- Linux hardened access to `/proc` for mitigating `ret2dir` attacks (v4.0 and onward).
`https://goo.gl/010alY`
- OpenBSD introduced `ret2dir`-specific mitigations (v5.7 and onward).
`https://goo.gl/qAmAhK`

Press and Media Coverage

○ Robotics Security [C.19]

14/10/2019 **Kaspersky.** *A glimpse into the present state of security in robotics.*
`https://bit.ly/2GlXJi0`

08/24/2018 **WIRED.** *The Serious Security Problem Looming Over Robotics.*
`https://bit.ly/2o7anER`

○ Software Hardening [C.24]

01/17/2017 **Network World.** *7 really cool network and IT research projects.*
`https://goo.gl/8wwKYj`

11/18/2016 **ACM TechNews.** *New Software Continuously Scrambles Code to Foil Cyberattacks.*
`https://goo.gl/GKCvhv`

○ Kernel Protection [C.30]

01/02/2017 **Linux Journal.** *What's New in Kernel Development.*
`https://bit.ly/3HJwlqS`

09/14/2016 **LWN.net.** *Exclusive page-frame ownership.*
`https://goo.gl/JiuOeo`

09/09/2014 **Dark Reading.** *Black Hat Europe 2014: Gullible Computers.*
`https://goo.gl/DniZ7O`

○ Side Channels [C.28]

04/21/2015 **The Register.** *JavaScript CPU cache snooper tells crooks EVERYTHING you do online.*
`https://goo.gl/UhvsT8`

04/20/2015 **Forbes.** *New Browser Hack Can Spy On Eight Out Of Ten PCs.*
`https://goo.gl/TX0kRq`

Software Artifacts

NOPoutNG [W.1]	`https://gitlab.com/brown-ssl/nopoutng`
PickleBall [C.1]	`https://github.com/columbia/pickleball`
IUBIK [C.2]	`https://github.com/tum-itsec/iubik`
Eclipse [C.3]	`https://gitlab.com/brown-ssl/eclipse`
SafeSLAB [C.4]	`https://github.com/tum-itsec/safeslab`
BeeBox [C.5]	`https://gitlab.com/brown-ssl/beebox`
ISLAB [C.6]	`https://github.com/tum-itsec/islab`
Quack [C.7]	`https://github.com/columbia/quack`
SysXCHG [C.8]	`https://gitlab.com/brown-ssl/sysxchg`
FineIBT [C.9]	`https://gitlab.com/brown-ssl/fineibt`
IvySyn [C.10]	`https://gitlab.com/brown-ssl/ivysyn`
EPF [C.11]	`https://gitlab.com/brown-ssl/epf`
BinWrap [C.12]	`https://github.com/atlas-brown/binwrap`
μSCOPE [C.13]	`https://gitlab.com/fierce-lab/uscope`

Nibbler [J.1]	https://gitlab.com/brown-ssl/libfilter
sysfilter [C.15]	https://gitlab.com/brown-ssl/sysfilter
xMP [C.16]	https://github.com/virtsec/xmp
Egalito [C.17]	https://gitlab.com/egalito
CCR [C.20]	https://github.com/kevinkoo001/CCR
kR^X [C.22]	https://gitlab.com/brown-ssl/krx
VTPin [C.23]	https://github.com/uberspot/vtpin
DynaGuard [C.27]	https://github.com/nettrino/dynaguard
XPFO [C.30]	https://www.cs.columbia.edu/~vpk/research/xpfo/
ret2dir [C.30]	https://www.cs.columbia.edu/~vpk/research/ret2dir/
kGuard [C.36]	https://www.cs.columbia.edu/~vpk/research/kguard/
libdft [C.38]	https://gitlab.com/brown-ssl/libdft
P2PWNC [M.2]	https://mm.aueb.gr/research/p2pwnc/

Teaching

Instructor

▶ All courses are new additions to the curriculum and were developed from scratch. (Numbers in parentheses indicate enrollment.)

CSCI 1650 **Software Security and Exploitation**, Brown University.
Fall 2025 (75), Fall 2024 (100), Fall 2023 (195), Fall 2022 (148), Fall 2021 (111), Fall 2020 (135), Fall 2019 (98), Fall 2018 (52), Fall 2017 (36), Fall 2016 (28)

CSCI 2951U **Topics in Software Security**, Brown University.
Spring 2025 (9), Spring 2024 (12), Spring 2021 (6), Spring 2020 (11), Spring 2018 (12), Spring 2017 (9), Spring 2016 (4)

Advising and Mentoring

Postdoctoral Researchers

2021–2023 **Vaggelis Atlidakis** (Dept. of Computer Science, Brown University).
Computing Innovation Fellow 2020
Post-graduation: University of Athens (Assistant Professor)

Doctoral Students

2024–present **Grigoris Ntousakis** (Dept. of Computer Science, Brown University).
Co-advisor: Prof. Nikos Vasilakis

2021–present **Alex Gaidis** (Dept. of Computer Science, Brown University).

2021–present **Neophytos Christou** (Dept. of Computer Science, Brown University).

2016–2024 **Di Jin** (Dept. of Computer Science, Brown University).
Thesis: *Hardening In-kernel Execution Environments against Memory-safety and Transient-execution Vulnerabilities*
Post-graduation: Brown University (Postdoctoral Researcher)

2017–2021 **Nicholas DeMarinis** (Dept. of Computer Science, Brown University).
Thesis: *Improving App. Security at Scale by Reducing System Call and Library Overprivilege*
Post-graduation: Brown University (Assistant Teaching Professor)

2014–2019 **Marios Pomonis** (Dept. of Computer Science, Columbia University).
Thesis: *Preventing Code Reuse Attacks On Modern Operating Systems*
Co-advisor: Prof. Angelos Keromytis
Post-graduation: Google (Security Software Engineer)

Master's Students

2024–2025 **Kelsie A. Edie** (Dept. of Computer Science, Brown University),
Automated Detection and Extraction of String Deobfuscation Functions in Malware Binaries via Static and Dynamic Analysis.
Post-graduation: U.S. Army Cyber Command (Cyber Officer, 1LT)

2021–2022 **Richard Abou Chaaya** (Dept. of Computer Science, Brown University),
`sysfilter+`*: Type-based System Call Filtering with Temporal Specialization.*
Post-graduation: L3Harris Trenchant (Security Research Engineer)

2017–2020 **Kent Williams-King** (Dept. of Computer Science, Brown University),
Hardening Commodity Software through Automated System Call Filtering and Binary Recompilation. [C.15, C.17, J.1]
Post-graduation: Elpha Secure (Software Engineer)

2019–2020 **Jearson Alfajardo** (Dept. of Computer Science, Brown University),
Assessing the Correctness of Debloating Binary Shared Libs with `libfilter`. [J.1]
Post-graduation: Juni Learning (Instructor)

2019–2020 **Jon Vexler** (Dept. of Computer Science, Brown University),
Characterization of Forward-edge Ctrl.-flow Integrity Targets in LLVM-comp. Linux.
Post-graduation: Epic (Software Developer)

2019–2020 **Changmin Teng** (Dept. of Computer Science, Brown University),
`NestFuzz`**:** *A Framework for Fuzzing Nested Virtualization Environments.*
Post-graduation: Amazon (Software Engineer)

2017–2018 **Sorin Vatasoiu** (Dept. of Computer Science, Brown University),
Breaking and Fixing the Linux Pseudo-Random Number Generator.
Post-graduation: Global Trading Systems (Software Engineer)

2015–2016 **Jordan Hendricks** (Dept. of Computer Science, Brown University),
`kGuard++`**:** *Improving the Performance of kGuard with Low-latency Code Inflation.*
Post-graduation: Joyent (Software Engineer)

2015–2016 **Pawel Sarbinowski** (Dept. of Computer Science, Aalto University),
`VTPin`: *Protecting Legacy Software from VTable Hijacking.* [C.23]
Co-advisor: Prof. Elias Athanasopoulos
Post-graduation: Microsoft (Software Engineer)

Undergraduate Students

2024–2025 **Yi Hao Tan** (Dept. of Computer Science, Brown University),
`CacheCFI`: Forward-edge Control-flow Integrity with Inline Branch-target Caches.
Post-graduation: M.Sc. in Cybersecurity (VU University Amsterdam)

2023–2025 **Jamie Gabbay** (Dept. of Computer Science, Brown University),
`NOPoutNG`: Dynamic Pruning of Indirect Branch Targets for FineIBT. [W.1]
Post-graduation: N/A

2024–2025 **José D. Sandoval** (Dept. of Computer Science, Brown University),
Systematization of Knowledge: Platforms, Pedagogies, and Practices for Effective ICS/OT Security Training.
Post-graduation: Sc.M. in Cybersecurity (Brown University)

2021–2022 **Sierra Rowley** (Dept. of Computer Science, Brown University),
Assessing the Effectiveness of System Call Filtering between Linux and FreeBSD.
Post-graduation: Naval Undersea Warfare Center (Cybersecurity Engineer)

2018–2019 **Benjamin Shteinfeld** (Dept. of Computer Science, Brown University),
`libfilter`: *Debloating Dynamically-linked Libraries through Binary Recompilation.* [J.1]
Post-graduation: Google (Software Engineer)

2018–2019 **Elisa Guerrant** (Dept. of Computer Science, Brown University),
Hardening the Linux Kernel Key Retention Service against Information Disclosure Vulnerabilities.
Post-graduation: M.Sc. in Cyber Security (ETH Zurich)

2017–2018 **Di Yang Shi** (Dept. of Computer Science, Brown University),
An Exposition of Adversarial Examples in Neural Networks.
Post-graduation: MemSQL (Software Engineer)

Other Mentoring Activities

2021–2025 **Marius Momeu** (Ph.D. student; Technical University of Munich),
Hardware-assisted Protection of Operating System Kernels. [C.2, C.4, C.6]

2014–2016 **João Moreira** (Ph.D. student; University of Campinas),
`kCFI`: *Fine-Grained Control-Flow Integrity for the Linux Kernel.* [Z.4]

2014–2015 **Theofilos Petsios** (Ph.D. student; Columbia University),
`DynaGuard`: *Armoring Canary-based Protections against Brute-force Attacks.* [C.27]

2011–2013 **Marco Barbera** (Ph.D. student; Sapienza University of Rome),
`CellFlood`: *Attacking Tor Onion Routers on the Cheap.* [C.33]

Service

Program Chair

RAID	Intl. Symposium on Research in Attacks, Intrusions and Defenses (co-Chair), 2026, 2027.
CCS	ACM Conference on Computer and Communications Security (co-Chair, *Software Security Track*), 2026.

Program Committee Member

SEC	USENIX Security Symposium, 2019–2026.
DAC	ACM/IEEE Design Automation Conference, 2025.
OSDI	USENIX Symposium on Operating Systems Design and Implementation, 2025.
CCS	ACM Conference on Computer and Communications Security, 2021–2025.
S&P	IEEE Symposium on Security and Privacy, 2022–2024.
NDSS	Network and Distributed System Security Symposium, 2024.
WWW	International World Wide Web Conference, 2017.
RAID	International Symposium on Research in Attacks, Intrusions and Defenses, 2016, 2020, 2025.
ACSAC	Annual Computer Security Applications Conference, 2017–2021, 2023, 2024.
ASIACCS	ACM Asia Conference on Computer and Communications Security, 2018.
DIMVA	International Conf. on Detection of Intrusions and Malware & Vuln. Asmt., 2017–2025.
WOOT	USENIX WOOT Conference on Offensive Technologies, 2018, 2022, 2024.
ISC	International Information Security Conference, 2016, 2019, 2024, 2025.
3S	Security for Space Systems Conference, 2024, 2025.
DSC	IEEE Conference on Dependable and Secure Computing, 2017, 2018.
ESSoS	International Symposium on Engineering Secure Software and Systems, 2017.
ROOTS	Reversing and Offensive-oriented Trends Symposium, 2017.
BAR	Workshop on Binary Analysis Research, 2020, 2022–2025.
EuroSec	European Workshop on Systems Security, 2016–2022.
CCSW	ACM Cloud Computing Security Workshop, 2017.

Journal Reviewer

TOCS	ACM Transactions on Computer Systems, 2024, 2025.
TDSC	IEEE Transactions on Dependable and Secure Computing, 2017–2025.
TIFS	IEEE Transactions on Information Forensics and Security, 2023–2025.
TOPS	ACM Transactions on Privacy and Security, 2022–2024.
COSE	Computers & Security, 2016, 2018, 2022, 2025.
DTRAP	ACM Digital Threats: Research and Practice, 2020, 2021.
S&P	IEEE Security & Privacy, 2020.
TMC	IEEE Transactions on Mobile Computing, 2018, 2019.

TCAD IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2017, 2018.

TCC IEEE Transactions on Cloud Computing, 2016, 2017.

JSC Journal of Supercomputing, 2023.

JSME Journal of Software: Evolution and Process, 2016.

Grant Reviewer

NSF National Science Foundation (*Panelist*), 2018, 2022–2024.

Dissertation Committee Member

November 2024 **Zhiyuan Zhang**, School of Comp. and Info. Systems, University of Melbourne.
"The Security Impact of Microarchitecture Optimizations"

August 2023 **Evangelia Anna Markatou**, Dept. of Computer Science, Brown University.
"Exploring Searchable Encryption Leakage from Range Queries"

April 2023 **Yingjie Xue**, Dept. of Computer Science, Brown University.
"Enabling Cross-Chain Transactions"

November 2022 **Sergej Proskurin**, School of Comp., Info. and Tech., Technical Univ. of Munich.
"Virtualization-assisted Dynamic Binary Analysis and Operating System Security"

June 2022 **Yueqi Chen**, College of Information Sciences and Technology, Penn. State University.
"Investigating Exploitable Design Patterns for More Advanced Protection Design"

April 2022 **Mohamed Tarek Ibn Ziad**, Dept. of Computer Science, Columbia University.
"Hardware-Software Co-design for Practical Memory Safety"

January 2022 **Andrea Mambretti**, Khoury College of Computer Sciences, Northeastern University.
"Execution Security in the Spectre Era"

October 2020 **Miguel A. Arroyo**, Dept. of Computer Science, Columbia University.
"Bespoke Security for Resource Constrained Cyber-Physical Systems"

August 2020 **David Williams-King**, Dept. of Computer Science, Columbia University.
"Improving Security Through Egalitarian Binary Recompilation"

May 2019 **Hyungjoon Koo**, Dept. of Computer Science, Stony Brook University.
"Practical Software Specialization against Code Reuse Attacks"

April 2019 **Evgenios Kornaropoulos**, Dept. of Computer Science, Brown University.
"Information Leakage in Encrypted Systems through an Algorithmic Lens"

November 2018 **Kanad Sinha**, Dept. of Computer Science, Columbia University.
"Repurposing Software Defenses with Specialized Hardware"

November 2018 **Ioannis Agadakos**, Dept. of Computer Science, Stevens Institute of Technology.
"Improving Software by Disabling Unused Code in Dynamically-linked Applications"

March 2018 **Theofilos Petsios**, Dept. of Computer Science, Columbia University.
"Compiler-assisted Adaptive Software Testing"

December 2016 **João Moreira**, Institute of Computing, University of Campinas.
"Protection Mechanisms against Kernel Control-Flow Hijacking Attacks"

University Service

2024–present Academic Director, Master of Science in Cybersecurity,
Dept. of Computer Science, Brown University.

2024–present Security Faculty Organizational Group (FOG) Leader,
Dept. of Computer Science, Brown University.

2020–present Brown Ethics And Responsible Conduct Of Research Education (BEARCORE) Liaison,
Dept. of Computer Science, Brown University.

2017–present Concentration Advising, Dept. of Computer Science, Brown University.

2022–2023 PhD Admissions Committee (Chair), Dept. of Computer Science, Brown University.

2015–2021 PhD Admissions Committee (Member), Dept. of Computer Science, Brown University.

2019–2020 Lecturer Search Committee, Dept. of Computer Science, Brown University.

Talks, Lectures, Presentations

Invited Talks

○ **Building Secure and Trustworthy Operating Systems** [C.4, C.5, C.11, C.16, C.22]

April 2025 Yale University, *Host*: Prof. Charalampos Papamanthou

January 2025 Northeastern University, *Host*: Prof. Ziming Zhao

November 2024 Boston University, *Host*: Prof. Manuel Egele

April 2024 Columbia University, *Host*: Prof. Steven Bellovin

December 2023 Southern University of Science and Technology, *Host*: Prof. Fengwei Zhang

○ **Trustworthy Software Systems** [C.8, C.10, C.15, C.17, C.18, C.20, C.21, C.24]

February 2024 Pomona College, *Host*: Prof. Alexandra Papoutsaki

August 2016 Columbia University, *Host*: Prof. Simha Sethumadhavan

○ **Agile Software Hardening** [C.15, C.17, C.18]

May 2022 Brown University, *Host*: Prof. Timothy Edgar

January 2022 Athens University of Economics and Business, *Host*: Prof. George Polyzos

○ **Protecting Commodity Software against Data-only Attacks** [C.16]

May 2021 Inria Rennes-Bretagne Atlantique Research Centre, *Host*: Prof. Guillaume Hiet

April 2021 Ohio State University, *Host*: Prof. Zhiqiang Lin

○ **Secure Operating Systems** [C.22, C.30, C.36]

February 2018 Wayne State University, *Host*: Prof. Fengwei Zhang

January 2018 Athens University of Economics and Business, *Host*: Prof. George Polyzos

July 2017 University of Athens, *Host*: Prof. Mema Roussopoulos

○ **Rethinking Kernel Isolation** [C.30]

October 2016 Athens University of Economics and Business, *Host*: Prof. George Polyzos

November 2014 Stevens Institute of Technology, *Host*: Prof. Georgios Portokalidis

October 2014 VU University Amsterdam, *Host*: Prof. Herbert Bos

September 2014 Georgia Institute of Technology, *Host*: Dr. Tielei Wang, Prof. Wenke Lee

○ **Lightweight Kernel Protection against Return-to-user Attacks** [C.36]

November 2012 AT&T Security Research Center, *Host*: Dr. Baris Coskun

July 2012 NEC Laboratories America, *Host*: Dr. Zhichun Li

Guest Lectures

April 2019 **Security Architectures**
Operating Systems (CSCI 1670), Brown University.
Instructor: Prof. Thomas Doeppner

April 2016 **The Role of the Operating System in the Era of Cyberwar**
Cybersecurity and International Relations (CSCI 1800), Brown University.
Instructor: Prof. John Savage

December 2015 **Kernel Security (in the Embedded World)**
Embedded and Real Time Software (CSCI 1600), Brown University.
Instructor: Prof. Steven Reiss

November 2014 **Kernel Security: Attacks and Defenses**
Secure Systems (CS 576), Stevens Institute of Technology.
Instructor: Prof. Georgios Portokalidis

October 2014 **Kernel Security: Building Trustworthy OSes**
Reliable Software (COMS E6121), Columbia University.
Instructor: Prof. Junfeng Yang

April 2010 **Packet Filters: Proposed Solutions and Current Trends**
Network Systems Design and Implementation (COMS W6998), Columbia University.
Instructor: Dr. Erich Nahum

May 2009 **Securing Networked Applications: The Role of Program Structure**
Network Security, Athens University of Economics and Business.
Instructor: Dr. Elias Efstathiou, Dr. Thanasis Papaioannou

Conference Presentations

May 2025 **Hardening the Software Supply Chain: Practical Post-Compilation Defenses**
Securing Autonomous Vehicle Ecosystems and Supply Chains (SAVES) Workshop (co-located with IEEE MOST), Newark, DE, USA.

December 2022 **`libdft`: Dynamic Data Flow Tracking for the Masses**
Annual Computer Security Applications Conference (ACSAC), Austin, TX, USA.

February 2017 **The Role of Low-level Software in the Era of Cyber Conflict**
Messaging, Malware and Mobile Anti-Abuse Working Group (M3AAWG),
San Francisco, CA, USA.

October 2014 **`ret2dir`: Deconstructing Kernel Isolation**
Black Hat Europe (BHEU), Amsterdam, Netherlands.
Video: `https://youtu.be/kot-EQ9zf9k`

August 2014 **`ret2dir`: Rethinking Kernel Isolation**
USENIX Security Symposium (SEC), San Diego, CA, USA.
Video: `https://goo.gl/Cxdz7C` | Audio: `https://goo.gl/ab3vIJ`

August 2012 **`kGuard`: Lightweight Kernel Protection against Return-to-user Attacks**
USENIX Security Symposium (SEC), Bellevue, WA, USA.
Video: `https://goo.gl/y3rvHK` | Audio: `https://goo.gl/JM2cgh`

March 2012 **`libdft`: Practical Dynamic Data Flow Tracking for Commodity Systems**
International Conference on Virtual Execution Environments (VEE), London, UK.

Professional and Industry Experience

Summer 2014 **Software Engineer**, *Oracle America Inc.*, Santa Clara, CA, USA.
Member of the Solaris Core Kernel team. *Tasks included*: patching the kernel of Oracle Solaris to add support for full Address Space Layout Randomization (ASLR), modifying the build environment for compiling the OS/Net userland binaries as position-independent, and evaluating the performance overhead of position-independent code.

2007–2008 **Software Engineer**, *MySapient Ltd.*, Athens, Greece.
Participated in the design and development of a massively multiplayer online game (MMOG) and a social network. *Tasks included*: designing, and implementing, in C++, a set of networking libraries, as well as a networked game server using a distributed and scalable architecture.

2006–2007 **System Administrator**, *Network Economics and Services Lab*, Athens University of Economics and Business, Athens, Greece.
Daily administration and maintenance of 12 workstations. *Tasks included*: strong security setup (firewall/VPN), administering the laboratory's domain, file, and print server, and website maintenance for the Web Information Management (WIM) research team.

2005–2007 **Student Consultant**, *Microsoft Hellas*, Athens, Greece.
Member of Developers Platform Evangelists (DPE) group. *Tasks included*: administering the departmental Microsoft Developer Network Academic Alliance (MSDNAA) subscription, organizing technical presentations (for students) involving Microsoft products, advising students entering Microsoft's worldwide "Imagine Cup" programming contest, and setting up and moderating the studentguru.gr community website.

2004–2005 **Chief System Administrator**, *Computer Science Lab*, Athens University of Economics and Business, Athens, Greece.
Daily administration and maintenance of 40 workstations for supporting the needs of the offered courses. *Tasks included*: administering the laboratory's domain, file, and print server, administering the departmental web and email server, administering the local wireless network, and authoring a QoS management software to support multiple user and service classes.

Funding

▶ Total: $6,327,191 (Brown share: **$2,091,798**)

[F.1] CAREER: Countering Emerging Software Threats with Adaptive Hardening, Debloating, and Hardware-assisted Protection. **PI**
National Science Foundation (NSF/SaTC), CNS-2238467,
$660,080 (Brown share: **$660,080**), 06/01/2023 – 05/31/2028.

[F.2] Fuzzing of RESTful Cloud Services. **PI**
(Computing Innovation Fellows 2020 program; Postdoc: Vaggelis Atlidakis),
National Science Foundation (NSF/CRA), CIF2020-BU-04,
$318,288 (Brown share: **$318,288**), 01/01/2021 – 05/31/2023.

[F.3] ABIDES: Adaptive BInary Debloating and Security. **Co-PI**
(PI: Georgios Portokalidis; co-PI: Junfeng Yang),
Office of Naval Research (ONR), N00014-17-1-2788,
$3,243,244 (Brown share: **$925,930**), 09/01/2017 – 08/31/2020.

[F.4] Hardware-Up Security: Anti-fragility and Automation. **Co-PI**
(PI: Simha Sethumadhavan; co-PIs: Luca Carloni, Subhasish Mitra),
Defense Advanced Research Projects Agency (DARPA), HR001118C0017,
$2,106,579 (Brown share: **$187,500**), 12/06/2017 – 06/05/2019.

Patents

[P.1] S. Sethumadhavan, K. Sinha, A. D. Keromytis, V. Pappas, and **V. P. Kemerlis**.
Diversified instruction set processing to enhance security.
U.S. Patent 10,237,059. Issued: Mar 19, 2019.

[P.2] B. M. Bowen, P. V. Prabhu, **V. P. Kemerlis**, S. Sidiroglou, S. J. Stolfo, and A. D. Keromytis. Methods, systems, and media for detecting covert malware.
U.S. Patent 9,971,891. Issued: May 15, 2018.

[P.3] S. J. Stolfo, A. D. Keromytis, B. M. Bowen, S. Herhskop, **V. P. Kemerlis**, P. V. Prabhu, and M. B. Salem. Methods, systems, and media for baiting inside attackers. U.S. Patent 9,501,639. Issued: Nov 22, 2016.

[P.4] Z. Li, Z. Wu, Z. Qian, G. Jiang, K. Lu, and **V. P. Kemerlis**. Duleak: a scalable app engine for high-impact privacy leaks.
U.S. Patent 9,245,125. Issued: Jan 26, 2016.

[P.5] S. J. Stolfo, A. D. Keromytis, B. M. Bowen, S. Herhskop, **V. P. Kemerlis**, P. V. Prabhu, and M. B. Salem. Methods, systems, and media for baiting inside attackers. U.S. Patent 9,009,829. Issued: Apr 14, 2015.

[P.6] B. M. Bowen, P. V. Prabhu, **V. P. Kemerlis**, S. Sidiroglou, S. J. Stolfo, and A. D. Keromytis. Methods, systems, and media for detecting covert malware. U.S. Patent 8,528,091. Issued: Sep 3, 2013.

Publications

▶ Google Scholar [https://goo.gl/DMK8AJ] – DBLP [https://goo.gl/dpaFrg] (✪: Tier-1 venue, ___: Advisee.)

Conference Proceedings

[C.1] ✪ A. D. Kellas, N. Christou, W. Jiang, P. Li, L. Simon, Y. David, **V. P. Kemerlis**, J. C. Davis, and J. Yang. PickleBall: Secure Deserialization of Pickle-based Machine Learning Models. In *Proceedings of the 32nd ACM Conference on Computer and Communications Security (CCS)*, Taipei, Taiwan, October 2025. [Acceptance rate: 13.9%]

Artifacts Available acm V1.1 · Artifacts Evaluated Functional acm V1.1 · Results Reproduced acm V1.1

[C.2] ✪ M. Momeu, A. J. Gaidis, J. v.d. Heidt, and **V. P. Kemerlis**. IUBIK: Isolating User Bytes in Commodity Operating System Kernels via Memory Tagging Extensions. In *Proceedings of the 46th IEEE Symposium on Security and Privacy (S&P)*, San Francisco, CA, USA, May 2025. [Acceptance rate: 14.8%]

[C.3] ✪ N. Christou, A. J. Gaidis, V. Atlidakis, and **V. P. Kemerlis**. Eclipse: Preventing Speculative Memory-error Abuse with Artificial Data Dependencies. In *Proceedings of the 31st ACM Conference on Computer and Communications Security (CCS)*, Salt Lake City, UT, USA, October 2024. [Acceptance rate: 16.9%]

Artifacts Available acm V1.1

[C.4] ✪ M. Momeu, S. Schnuckel, K. Angnis, M. Polychronakis, and **V. P. Kemerlis**. Safeslab: Mitigating Use-After-Free Vulnerabilities via Memory Protection Keys. In *Proceedings of the 31st ACM Conference on Computer and Communications Security (CCS)*, Salt Lake City, UT, USA, October 2024.
[Acceptance rate: 16.9%]

Artifacts Available acm V1.1 · Artifacts Evaluated Functional acm V1.1 · Results Reproduced acm V1.1

[C.5] ✪ D. Jin, A. J. Gaidis, and **V. P. Kemerlis**. BeeBox: Hardening BPF against Transient Execution Attacks. In *Proceedings of the 33rd USENIX Security Symposium (SEC)*, Philadelphia, CA, USA, August 2024. [Acceptance rate: 19.1%]

ARTIFACT EVALUATED usenix ASSOCIATION AVAILABLE | ARTIFACT EVALUATED usenix ASSOCIATION FUNCTIONAL | ARTIFACT EVALUATED usenix ASSOCIATION REPRODUCED

[C.6] M. Momeu, F. Kilger C. Roemheld, S. Schnuckel, S. Proskurin, M. Polychronakis, and **V. P. Kemerlis**. ISLAB: Immutable Memory Management Metadata for Commodity Operating System Kernels. In *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security (ASIACCS)*, Singapore, July 2024. [Acceptance rate: 22%]

[C.7] ✪ Y. David, N. Christou, A. D. Kellas, **V. P. Kemerlis**, and J. Yang. QUACK: Hindering Deserialization Attacks via Static Duck Typing. In *Proceedings of the 31st Network and Distributed System Security (NDSS) Symposium*, San Diego, CA, USA, February 2024. [Acceptance rate: 17.6%]

Artifact Evaluated NDSS Available | Artifact Evaluated NDSS Functional | Artifact Evaluated NDSS Reproduced

[C.8] ✪ A. J. Gaidis, V. Atlidakis, and **V. P. Kemerlis**. SysXCHG: Refining Privilege with Adaptive System Call Filters. In *Proceedings of the 30th ACM Conference on Computer and Communications Security (CCS)*, Copenhagen, Denmark, November 2023. [Acceptance rate: 19.2%]

Artifacts Available acm V1.1 | Artifacts Evaluated Functional V1.1 acm | Results Reproduced acm V1.1

[C.9] A. J. Gaidis, J. Moreira, K. Sun, A. Milburn, V. Atlidakis, and **V. P. Kemerlis**. FineIBT: Fine-grain Control-flow Enforcement with Indirect Branch Tracking. In *Proceedings of the 26th International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, Hong Kong, HK, October 2023. [Acceptance rate: 23.5%]

[C.10] ✪ N. Christou, Di Jin, V. Atlidakis, B. Ray, and **V. P. Kemerlis**. IvySyn: Automated Vulnerability Discovery for Deep Learning Frameworks. In *Proceedings of the 32nd USENIX Security Symposium (SEC)*, Anaheim, CA, USA, August 2023. [Acceptance rate: 29%]

ARTIFACT EVALUATED usenix ASSOCIATION AVAILABLE | ARTIFACT EVALUATED usenix ASSOCIATION FUNCTIONAL | ARTIFACT EVALUATED usenix ASSOCIATION REPRODUCED

[C.11] ✪ D. Jin, V. Atlidakis, and **V. P. Kemerlis**. EPF: Evil Packet Filter. In *Proceedings of the 29th USENIX Annual Technical Conference (USENIX ATC)*, Boston, MA, USA, July 2023. [Acceptance rate: 18.4%]

[C.12] G. Christou, G. Ntousakis, E. Lahtinen, S. Ioannidis, **V. P. Kemerlis**, and N. Vasilakis. BinWrap: Hybrid Protection against Native Node.js Add-ons. In *Proceedings of the 18th ACM Asia Conference on Computer and Communications Security (ASIACCS)*, Melbourne, Australia, July 2023.
[Acceptance rate: 18.8%]

[C.13] N. Roessler, L. Atayde, I. Palmer, D. McKee, J. Pandey, **V. P. Kemerlis**, M. Payer, A. Bates, A. DeHon, J. M. Smith, and N. Dautenhahn. μSCOPE: A Methodology for Analyzing Least-Privilege Compartmentalization in Large Software Artifacts. In *Proceedings of the 24th International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, San Sebastian, Spain, October 2021. [Acceptance rate: 23.9%]

[C.14] M. T. I. Ziad, M. Arroyo, E. Manzhosov, **V. P. Kemerlis**, and S. Sethumadhavan. EPI: Efficient Pointer Integrity for Securing Embedded Systems. In *Proceedings of the 1st IEEE International Symposium on Secure and Private Execution Environment Design (SEED)*, Virtual Event, September 2021.

[C.15] N. DeMarinis, K. Williams-King, D. Jin, R. Fonseca, and **V. P. Kemerlis**. sysfilter: Automated System Call Filtering for Commodity Software. In *Proceedings of the 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, San Sebastian, Spain, October 2020. [Acceptance rate: 25.6%]

[C.16] ✪ S. Proskurin, M. Momeu, S. Ghavamnia, **V. P. Kemerlis**, and M. Polychronakis. xMP: Selective Memory Protection for Kernel and User Space. In *Proceedings of the 41st IEEE Symposium on Security and Privacy (S&P)*, San Francisco, CA, USA, May 2020. [Acceptance rate: 12.3%]

[C.17] ✪ D. Williams-King, H. Kobayashi, K. Williams-King, G. Patterson, F. Spano, Y. J. Wu, J. Yang, and **V. P. Kemerlis**. Egalito: Layout-Agnostic Binary Recompilation. In *Proceedings of the 25th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, Lausanne, Switzerland, March 2020. [Acceptance rate: 18%]

Artifacts Available acm Artifacts Evaluated acm Functional

[C.18] I. Agadakos, D. Jin, D. Williams-King, **V. P. Kemerlis**, and G. Portokalidis. Nibbler: Debloating Binary Shared Libraries. In *Proceedings of the 35th Annual Computer Security Applications Conference (ACSAC)*, San Juan, Puerto Rico, December 2019. [Acceptance rate: 22.6%]

[C.19] ✪ N. DeMarinis, S. Tellex, **V. P. Kemerlis**, G. Konidaris, and R. Fonseca. Scanning the Internet for ROS: A View of Security in Robotics Research. In *Proceedings of the 36th IEEE International Conference on Robotics and Automation (ICRA)*, Montreal, Canada, May 2019. [Acceptance rate: 45%]

[C.20] ✪ H. Koo, Y. Chen, L. Lu, **V. P. Kemerlis**, and M. Polychronakis. Compiler-assisted Code Randomization. In *Proceedings of the 39th IEEE Symposium on Security and Privacy (S&P)*, San Francisco, CA, USA, May 2018. [Acceptance rate: 11.5%]

[C.21] K. Sinha, **V. P. Kemerlis**, and S. Sethumadhavan. Reviving Instruction Set Randomization. In *Proceedings of the 9th IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, McLean, VA, USA, May 2017. [Acceptance rate: 24.5%]

[C.22] ✪ M. Pomonis, T. Petsios, A. D. Keromytis, M. Polychronakis, and **V. P. Kemerlis**. kR^X: Comprehensive Kernel Protection against Just-In-Time Code Reuse. In *Proceedings of the 12th European Conference on Computer Systems (EuroSys)*, Belgrade, Serbia, April 2017. [Acceptance rate: 20%]

[C.23] P. Sarbinowski, C. Giuffrida, **V. P. Kemerlis**, and E. Athanasopoulos. VTPin: Practical VTable Hijacking Protection for Binaries. In *Proceedings of the 32nd Annual Computer Security Applications Conference (ACSAC)*, Los Angeles, CA, USA, December 2016. [Acceptance rate: 22.8%]

[C.24] ✪ D. Williams-King, G. Gobieski, K. Williams-King, J. P. Blake, X. Yuan, P. Colp, M. Zheng, **V. P. Kemerlis**, J. Yang, and W. Aiello. Shuffler: Fast and Deployable Continuous Code Re-Randomization. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, Savannah, GA, USA, November 2016. [Acceptance rate: 18.1%]

[C.25] E. Athanasopoulos, **V. P. Kemerlis**, G. Portokalidis, and A. D. Keromytis. NaClDroid: Native Code Isolation for Android Applications. In *Proceedings of the 21st European Symposium on Research in Computer Security (ESORICS)*, Heraklion, Greece, September 2016. [Acceptance rate: 21%]

[C.26] ✪ W. Cui, M. Peinado, S. K. Cha, Y. Fratantonio, and **V. P Kemerlis**. RETracer: Triaging Crashes by Reverse Execution from Partial Memory Dumps. In *Proceedings of the 38th International Conference on Software Engineering (ICSE)*, Austin, TX, USA, May 2016. [Acceptance rate: 19%]

[C.27] T. Petsios, **V. P. Kemerlis**, M. Polychronakis, and A. D. Keromytis. DynaGuard: Armoring Canary-based Protections against Brute-force Attacks. In *Proceedings of the 31st Annual Computer Security Applications Conference (ACSAC)*, Los Angeles, CA, USA, December 2015. [Acceptance rate: 24.4%]

[C.28] ✪ Y. Oren, **V. P. Kemerlis**, S. Sethumadhavan, and A. D. Keromytis. The Spy in the Sandbox: Practical Cache Attacks in JavaScript and their Implications. In *Proceedings of the 22nd ACM Conference on Computer and Communications Security (CCS)*, Denver, CO, USA, October 2015. [Acceptance rate: 19.8%]

[C.29] ✪ K. Lu, Z. Li, **V. P. Kemerlis**, Z. Wu, L. Lu, C. Zheng, Z. Qian, W. Lee, and G. Jiang. Checking More and Alerting Less: Detecting Privacy Leakages via Enhanced Data-flow Analysis and Peer Voting. In *Proceedings of the 22nd Network and Distributed System Security (NDSS) Symposium*, San Diego, CA, USA, February 2015. [Acceptance rate: 16.9%]

[C.30] ✪ **V. P. Kemerlis**, M. Polychronakis, and A. D. Keromytis. ret2dir: Rethinking Kernel Isolation. In *Proceedings of the 23rd USENIX Security Symposium (SEC)*, San Diego, CA, USA, August 2014. [Acceptance rate: 19%]

[C.31] ✪ K. Jee, **V. P. Kemerlis**, A. D. Keromytis, and G. Portokalidis. ShadowReplica: Efficient Parallelization of Dynamic Data Flow Tracking. In *Proceedings of the 20th ACM Conference on Computer and Communications Security (CCS)*, Berlin, Germany, October 2013. [Acceptance rate: 19.8%]

[C.32] V. Pappas, **V. P. Kemerlis**, A. Zavou, M. Polychronakis, and A. D. Keromytis. CloudFence: Data Flow Tracking as a Cloud Service. In *Proceedings of the 16th International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, Saint Lucia, October 2013. [Acceptance rate: 23.1%]

[C.33] M. V. Barbera, **V. P. Kemerlis**, V. Pappas, and A. D. Keromytis. CellFlood: Attacking Tor Onion Routers on the Cheap. In *Proceedings of the 18th European Symposium on Research in Computer Security (ESORICS)*, Egham, UK, September 2013. [Acceptance rate: 17.8%]

[C.34] A. Zavou, V. Pappas, **V. P. Kemerlis**, M. Polychronakis, G. Portokalidis, and A. D. Keromytis. Cloudopsy: an Autopsy of Data Flows in the Cloud. In *Proceedings of the 15th International Conference on Human-Computer Interaction (HCI)*, Las Vegas, NV, USA, July 2013.

[C.35] ✪ D. Geneiatakis, G. Portokalidis, **V. P. Kemerlis**, and A. D. Keromytis. Adaptive Defenses for Commodity Software through Virtual Application Partitioning. In *Proceedings of the 19th ACM Conference on Computer and Communications Security (CCS)*, Raleigh, NC, USA, October 2012. [Acceptance rate: 18.9%]

[C.36] ✪ **V. P. Kemerlis**, G. Portokalidis, and A. D. Keromytis. kGuard: Lightweight Kernel Protection against Return-to-user Attacks. In *Proceedings of the 21st USENIX Security Symposium (SEC)*, Bellevue, WA, USA, August 2012. [Acceptance rate: 19.4%]

[C.37] E. Athanasopoulos, **V. P. Kemerlis**, M. Polychronakis, and E. P. Markatos. ARC: Protecting against HTTP Parameter Pollution Attacks Using Application Request Caches. In *Proceedings of the 10th International Conference on Applied Cryptography and Network Security (ACNS)*, Singapore, June 2012. [Acceptance rate: 17.2%]

[C.38] **V. P. Kemerlis**, G. Portokalidis, K. Jee, and A. D. Keromytis. libdft: Practical Dynamic Data Flow Tracking for Commodity Systems. In *Proceedings of the 8th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE)*, London, UK, March 2012.

[C.39] ✪ K. Jee, G. Portokalidis, **V. P. Kemerlis**, S. Ghosh, D. I. August, and A. D. Keromytis. A General Approach for Efficiently Accelerating Software-based Dynamic Data Flow Tracking on Commodity Hardware. In *Proceedings of the 19th Network and Distributed System Security (NDSS) Symposium*, San Diego, CA, USA, February 2012. [Acceptance rate: 18%]

[C.40] **V. P. Kemerlis**, V. Pappas, G. Portokalidis, and A. D. Keromytis. iLeak: A Lightweight System for Detecting Inadvertent Information Leaks. In *Proceedings of the 6th European Conference on Computer Network Defense (EC2ND)*, Berlin, Germany, October 2010.

[C.41] B. M. Bowen, P. Prabhu, **V. P. Kemerlis**, S. Sidiroglou, A. D. Keromytis, and S. J. Stolfo. BotSwindler: Tamper Resistant Injection of Believable Decoys in VM-Based Hosts for Crimeware Detection. In *Proceedings of the 13th International Symposium on Recent Advances in Intrusion Detection (RAID)*, Ottawa, Canada, September 2010. [Acceptance rate: 23%]

[C.42] B. M. Bowen, **V. P. Kemerlis**, P. Prabhu, A. D. Keromytis, and S. J. Stolfo. Automating the Injection of Believable Decoys to Detect Snooping. In *Proceedings of the 3rd ACM Conference on Wireless Network Security (WiSec)*, Hoboken, NJ, USA, March 2010 (*short paper*). [Acceptance rate: 21.2%]

[C.43] K. Katsaros, **V. P. Kemerlis**, C. Stais, and G. Xylomenos. A BitTorrent Module for the OMNeT++ Simulator. In *Proceedings of the 17th IEEE/ACM International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, London, UK, September 2009.

[C.44] A. Kosmopoulos, I. Karamichali, **V. P. Kemerlis**, and G. C. Polyzos. Fueling Game Development in Mobile P2P Environments. In *Proceedings of the 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Athens, Greece, September 2007.

[C.45] P. A. Frangoudis, **V. P. Kemerlis**, D. C. Paraskevaidis, E. C. Efstathiou, and G. C. Polyzos. Experimental Evaluation of Community-Based WLAN Voice and Data Services. In *Proceedings of the 3rd International Mobile Multimedia Communications Conference (MobiMedia)*, Nafpaktos, Greece, August 2007.

[C.46] E. G. Giannopoulou, **V. P. Kemerlis**, M. Polemis, J. Papaparaskevas, A. C. Vatopoulos, and M. Vazirgiannis. A Large Scale Data Mining Approach to Antibiotic Resistance Surveillance. In *Proceedings of the 20th IEEE International Symposium on Computer-Based Medical Systems (CBMS)*, Maribor, Slovenia, June 2007.

[C.47] E. C. Efstathiou, F. A. Elianos, P. A. Frangoudis, **V. P. Kemerlis**, D. C. Paraskevaidis, E. C. Stefanis, and G. C. Polyzos. Public Infrastructures for Internet Access in Metropolitan Areas. In *Proceedings of the 1st International Conference on Access Networks (AccessNets)*, Athens, Greece, September 2006.

[C.48] **V. P. Kemerlis**, E. C. Stefanis, G. Xylomenos, and G. C. Polyzos. Throughput Unfairness in TCP over WiFi. In *Proceedings of the 3rd IFIP Conference on Wireless On Demand Network Systems and Services (WONS)*, Les Mènuires, France, January 2006.

Workshop Proceedings

[W.1] A. J. Gaidis, J. Gabbay, J. Moreira, and **V. P. Kemerlis**. NOPoutNG: Improving the Effectiveness of Hardware-assisted Control-flow Integrity via Dynamic Landing Pad Elision. In *Proceedings of the 18th Cyber Security Experimentation and Test Workshop (CSET)*, Honolulu, HI, USA, December 2025.

[W.2] G. Xylomenos, K. Katsaros, and **V. P. Kemerlis**. Peer Assisted Content Distribution over Router Assisted Overlay Multicast. In *Proc. of the 1st Euro-NF Workshop on Future Internet Architecture (FIA)*, Paris, France, November 2008.

Journal Articles

[J.1] I. Agadakos, N. DeMarinis, D. Jin, K. Williams-King, J. Alfajardo, B. Shteinfeld, D. Williams-King, **V. P. Kemerlis**, and G. Portokalidis. Large-scale Debloating of Binary Shared Libraries. *ACM Digital Threats: Research and Practice (DTRAP)*, 1(4), December 2020.

[J.2] M. Pomonis, T. Petsios, A. D. Keromytis, M. Polychronakis, and **V. P. Kemerlis**. Kernel Protection against Just-In-Time Code Reuse. *ACM Transactions on Privacy and Security (TOPS)*, 22(1), January 2019.

[J.3] B. M. Bowen, **V. P. Kemerlis**, P. Prabhu, A. D. Keromytis, and S. J. Stolfo. A System for Generating and Injecting Indistinguishable Network Decoys. *Journal of Computer Security (JCS)*, 20(2-3), January 2012.

Magazine Articles

[M.1] **V. P. Kemerlis**, G. Portokalidis, E. Athanasopoulos, and A. D. Keromytis. kGuard: Lightweight Kernel Protection. *USENIX ;login: Magazine*, 37(6), December 2012.

[M.2] P. A. Frangoudis, G. C. Polyzos, and **V. P. Kemerlis**. Wireless Community Networks: An Alternative Approach for Broadband Nomadic Network Access. *IEEE Communications Magazine*, 49(5), May 2011.

Tutorials

[T.1] D. Williams-King, V. Rajagopalan, J. Yang, G. Portokalidis, and **V. P. Kemerlis**. Rewriting Modern Binaries with Egalito. In *Total Platform Cyber Protection* (TPCP) *Software Security Summer School (SSSS)*, Office of Naval Research (ONR) and MITRE Corporation, November 2021.

Demo Papers

[D.1] E. C. Efstathiou, F. A. Elianos, P. A. Frangoudis, **V. P. Kemerlis**, D. C. Paraskevaidis, G. C. Polyzos, and E. C. Stefanis. Practical Incentive Techniques for Wireless Community Networks. In *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, Uppsala, Sweden, June 2006.

[D.2] E. C. Efstathiou, F. A. Elianos, P. A. Frangoudis, **V. P. Kemerlis**, D. C. Paraskevaidis, G. C. Polyzos, and E. C. Stefanis. The Peer-to-Peer Wireless Network Confederation Scheme. In *International Conference on Computer Communications (INFOCOM)*, Barcelona, Spain, April 2006.

[D.3] E. C. Efstathiou, F. A. Elianos, P. A. Frangoudis, **V. P. Kemerlis**, D. C. Paraskevaidis, G. C. Polyzos, and E. C. Stefanis. The Peer-to-Peer Wireless Network Confederation Scheme: Protocols, Algorithms, and Services. In *International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom)*, Barcelona, Spain, March 2006.

Non-academic Papers (Refereed)

[Z.1] Y. David, N. Christou, A. D. Kellas, **V. P. Kemerlis**, and J. Yang. QUACK: Hindering Deserialization Attacks via Static Duck Typing. In *Black Hat USA (BHUSA)*, Las Vegas, NV, USA, August 2025.

[Z.2] **V. P. Kemerlis**. libdft: Dynamic Data Flow Tracking for the Masses. In *Annual Computer Security Applications Conference (ACSAC)*, Austin, TX, USA, December 2022.

[Z.3] M. Pomonis, T. Petsios, A. D. Keromytis, M. Polychronakis, and **V. P. Kemerlis**. kR^X: Comprehensive Kernel Protection against Just-In-Time Code Reuse. In *Black Hat USA (BHUSA)*, Las Vegas, NV, USA, July 2017.

[Z.4] J. Moreira, S. Rigo, M. Polychronakis, and **V. P. Kemerlis**. Drop the ROP: Fine-Grained Control-Flow Integrity for the Linux Kernel. In *Black Hat Asia (BHASIA)*, Singapore, March 2017.

[Z.5] **V. P. Kemerlis**, M. Polychronakis, and A. D. Keromytis. ret2dir: Deconstructing Kernel Isolation. In *Black Hat Europe (BHEU)*, Amsterdam, Netherlands, October 2014.