TAPTI PALIT

www.github.com/taptipalit www.tpalit.blog tpalit@purdue.edu

SUMMARY

I am a postdoctoral researcher at Purdue University, working on multiple problems in the domains of systems, software security and static analysis. My Ph.D. research investigated methods for fine-grained protection of sensitive information against data leakage and transient execution attacks. I have extensive experience with the LLVM compiler and various static analysis techniques, and I am looking to apply my skills towards solving exciting problems in the systems and compilers domain.

EDUCATION

- **PhD in Computer Science** — *Oct 2021*
  Stony Brook University, Stony Brook NY — *GPA – 3.88/4.0*
  Thesis: Selective Data Encryption: A Scalable Defense against Sensitive Data Leakage
  Advisor: Dr. Michalis Polychronakis
- **Masters in Computer Science** — *Aug 2015*
  Stony Brook University, Stony Brook NY — *GPA – 3.73/4.0*
  Thesis: Benchmarking Network-Intensive Applications
  Advisor: Dr. Mike Ferdman
- **Bachelors in Computer Science** — *May 2009*
  Mumbai University, MH, India — *First Class with Distinction*

WORK EXPERIENCE

- CRA Computing Innovation Fellow — *(started)* Oct 2021
  Purdue University
  Supervisor: Dr. Pedro Fonseca
  Research on various topics related to systems, security, and program analysis.
- Research Assistant — Jan 2017 - Oct 2021
  Stony Brook University
  Supervisor: Dr. Michalis Polychronakis
  Research on sensitive data isolation and static program analysis.
- Research Engineer Intern — May 2018 - Aug 2018
  Zeropoint Dynamics
  Research on binary clone detection using symbolic execution.

RESEARCH AREAS

- **Invariant-Guided Pointer Analysis *(Ongoing)*:**
  - Developed a semi-automated framework that instruments a standard field-sensitive and context-insensitive Andersen's pointer analysis to record *hot-spots*, that lead to an explosion of the points-to set sizes in the application program.
  - Designed a set of *likely-invariants*, that are optimistic assumptions about the points-to relationships in the application, and generate both an *optimistic* and a *fallback* points-to graph for the application.
  - Implemented an LLVM-based, lightweight runtime monitoring system that monitors the state of these *likely-invariants* during program execution and falls back to the conservative analysis if the likely-invariants are violated.
  - Applied this system to harden Linux applications with Control Flow Integrity.
- **Lightweight Sandboxes for Linux Kernel Resource Isolation *(Ongoing)*:**
  - Designed an Intel-MPK based isolation mechanism that can isolate memory of mutually-untrusted userspace components.

- Developed a lightweight technique for isolating kernel resources without expensive kernel-mode switches.
- Performed performance debugging that resulted in a performance improvement of 58% over similar WebAssembly-based sandboxing systems.

- **Sensitive Data Encryption:** Built a compiler-based defense that protects in-memory sensitive data against data-leakage and Spectre-style transient-execution attacks.
  - Implemented and optimized various analysis and transformation passes using the LLVM compiler toolchain to automatically encrypt sensitive data in the application.
  - Implemented and enhanced the Steensgaard's, and the Andersen's pointer analysis techniques for our system.
  - Augmented the scalable Steensgaard's pointer analysis with dynamic data-flow tracking information to apply our encryption system to larger applications with minimal runtime overhead.
  - Enhanced the LLVM interpreter to collect and reuse the points-to relationships established during application initialization, to improve the precision of the static pointer analysis algorithm.
- **Attack Surface Reduction:** Built multiple systems to identify and remove *unneeded* application code and *software bloat*.
  - Developed static analysis techniques in the LLVM compiler toolchain to generate precise call-graphs of large server applications.
  - Developed analysis passes using the GCC toolchain to correctly map Glibc functions to system calls.
  - Identified the application's different *execution phases* and the code that is accessible from each phase, thus allowing for the removal of unnecessary code.
  - Applied these techniques to restrict unneeded, security-critical system calls in Docker containers using Seccomp profiles.
- **Development of FPGA-based low latency devices:** Developed FPGA prototypes of low-latency, persistent memory devices.
  - Investigated various access-mechanisms (direct-memory access, queue-based) for optimal performance.
  - Designed microbenchmarks to study and compare these access-mechanisms.
- **Cloud Benchmarking:** Developed realistic benchmarks for cloud workloads.
  - Developed new realistic benchmarks for network-intensive Web 2.0 and media-streaming workloads.
  - Integrated these benchmarks with the existing CloudSuite benchmarking suite.

TECHNICAL STRENGTHS

Static Analysis Tools	Angr, KLEE, SVF
Performance Debugging and Instrumentation Tools	Intel pintool, perf tool, Intel VTune
Systems Development	LLVM (Intermediate Representation), Linux kernel module development
Programming Languages	C/C++, Python, Java

PUBLICATIONS

- Dinglan Peng, Congyu Liu, **Tapti Palit**, Pedro Fonseca, Anjo Vahldiek-Oberwagner, Mona Vij. "uSwitch: Fast Kernel Context Isolation with Implicit Context Switches" *Under Major Revision* in 2023 IEEE Symposium on Security and Privacy. S&P 2023.
- Seyedhamed Ghavamnia, **Tapti Palit**, and Michalis Polychronakis. "C2C: Fine-grained Configuration-driven Code and System Call Debloating" in the 29th ACM Conference on Computer and Communications Security (CCS). November 2022.
- **Tapti Palit**, Jarin Firose Moon, Fabian Monrose, and Michalis Polychronakis. "DynPTA: Combining Static and Dynamic Analysis for Practical Selective Data Protection" in 2021 IEEE Symposium on Security and Privacy. S&P 2021.

- Seyedhamed Ghavamnia, **Tapti Palit**, Shachee Mishra, and Michalis Polychronakis. “Temporal system call specialization for attack surface reduction.” In 29th USENIX Security Symposium (USENIX Security 20). 2020.
- Seyedhamed Ghavamnia, **Tapti Palit**, Azzedine Benameur, and Michalis Polychronakis. “Confine: Automated system call policy generation for container attack surface reduction.” In Proceedings of the International Conference on Research in Attacks, Intrusions, and Defenses (RAID). 2020.
- **Tapti Palit**, Fabian Monrose, and Michalis Polychronakis. “Mitigating data leakage by protecting memory-resident sensitive data.” In Proceedings of the 35th Annual Computer Security Applications Conference. 2019.
- Shenghsun Cho, Amoghavarsha Suresh, **Tapti Palit**, Michael Ferdman, and Nima Honarmand. “Taming the killer microsecond.” In 2018 51st Annual IEEE/ACM International Symposium on Microarchitecture (MICRO). IEEE, 2018.
- **Tapti Palit**, Yongming Shen, and Michael Ferdman. “Demystifying cloud benchmarking.” In 2016 IEEE international symposium on performance analysis of systems and software (ISPASS). IEEE, 2016.
- Varun Agrawal, Abhiroop Dabral, **Tapti Palit**, Yongming Shen, and Michael Ferdman. “Architectural support for dynamic linking.” In Proceedings of the Twentieth International Conference on Architectural Support for Programming Languages and Operating Systems. 2015.

DIVERSITY AND INCLUSION WORK

- Member of the Diversity Committee in the Department of Computer Science at Stony Brook University
- Treasurer of the Women in PhD graduate club
- Instructor at High School WISE program for under-represented groups in Computer Science