G. Gary Tan

W358 Westgate Building
Computer Science & Engineering
Penn State University
University Park, PA 16802

gtan AT psu.edu
Phone: (814) 865-7364
Fax: (814) 865-3176
http://www.cse.psu.edu/~gxt29/

Biographical Information

Education

Princeton University, Department of Computer Science, 1999–2005
Doctor of Philosophy, Awarded in 2005. Advisor: Andrew W. Appel.
Master of Arts, Awarded in 2001.

Tsinghua University, Department of Computer Science, China, 1994–1999.
Bachelor of Engineering in Computer Science (with distinction), Awarded in 1999.
Bachelor of Economics, Awarded in 1999.

Professional experience

Pennsylvania State University, University Park, PA. Full professor of Computer Science & Engineering. 07/2020–present. Also a co-hire of Institute for Computational and Data Science (ICDS).

Pennsylvania State University, University Park, PA. Associate Prof. of Computer Science & Engineering. Tenured. 1/2016–06/2020. Also a co-hire of ICDS.

Intelligent Automation, Inc., Rockville, MD. Consultant. 2/2020–5/2020; 4/2018–9/2018, 4/2016–8/2017, and 3/2013–9/2014.

Lehigh University, Bethlehem, PA. Associate Prof. of Computer Science & Engineering. Tenured. 6/2015–12/2015.

Lehigh University, Bethlehem, PA. Assistant Prof. of Computer Science & Engineering. 8/2008–5/2015.

Microsoft Research, Redmond, WA. Consulting Researcher. 6/2007–6/2008.

Boston College, Chestnut Hill, MA. Asst. Prof. of Computer Science. 9/2005–6/2008.

NEC Labs America, Princeton, NJ. Research summer intern. 2004.

Microsoft Research, Redmond, WA. Research summer intern. 2002.

Research Interests

Software security, programming languages, formal methods, software engineering.

Honors and Awards

- **Best Paper Award**, Lightweight Kernel Isolation with Virtualization and VM Functions, 16th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE), 2020.
- **Ruth and Joel Spira Excellence in Teaching Award**, Penn State, 2018.
- **Distinguished Reviewer Award**, 39th IEEE Symposium on Security and Privacy (Oakland), 2018.
- **Outstanding Paper Award**, From Debugging-Information Based Binary-Level Type Inference to CFG Generation, 8th ACM Conference on Data and Application Security and Privacy (CODASPY), 2018.
- **Best Demo Award**, Sensitive Information Tracking in Commodity IoT, Florida Institute of Cybersecurity Research Annual Conference on Cybersecurity, 2018.
- **James F. Will Career Development Professorship**, Pennsylvania State University, 2016-2018.
- **National Science Foundation Career Award**, 2012.
- **Google Faculty Research Award**, 2010 and 2012.
- Faculty Fellowship, Boston College, 2008.
- Francis Upton Graduate Fellowship, Princeton Univ., 1999 – 2003.
- Tsinghua University Scholarship, 1995, 1996 and 1998.
- HSBC Bank Scholarship, 1997.
- Lenovo Cup Computer Programming Contest, First Prize, 1995.
- China National Olympic Contest on Information Science, First Prize, 1993.

Awards won by my students

- Ben Niu, ACM SIGSAC Doctoral Dissertation Award Runner-Up, 2016.
- Matthew Kilgore, Honorable Mention in CRA's Outstanding Undergraduate Award, 2015.
- Jason Croft, Honorable Mention in CRA's Outstanding Undergraduate Award, 2009.

Research Funding

Competitively awarded external research grants

- NSF CNS-1956032: Automatic Software Patching against Microarchitectual Attacks. Co-Principal Investigator. With Danfeng Zhang (PI), Mahumut Kandemir, and Dinghao Wu. $500,000. 2020–2023.
- DARPA HR0011-19-C-0073 (Defense Advanced Research Projects Office): SPARTA—the Secure Parser Toolkit for Assurance. Principal Investigator. Subcontractor to Galois Inc.. My group's portion: $1,020,327. 2019–2023.

- DARPA HR0011-19-C-0106 (Defense Advanced Research Projects Office): Secure Handling of Isolated Executables without Leaking Data (SHIELD). Principal Investigator. With Trent Jaeger (Co-PI). Subcontractor to Perspecta Labs. Penn state portion: $650,000. 2019–2024.
- NSF CNS-1900873: Automated IoT Safety and Security Analysis and Synthesis. Co-Principal Investigator. With Patrick McDaniel (PI). $1,199,869. My group's portion: $600,000. 2019–2023.
- NSF CNS-1801534: Threat-Aware Defense: Evaluating Threats for Continuous Improvement. Co-Principal Investigator. Collaborative research with Trent Jaeger and Matthias Payer. $1,200,000. My group's portion: $400,000. 2018–2022.
- NSF CCF-1723571: Lightweight Abstract Memory Features. Principal Investigator. Collaborative research with Mike Spear and Xiaochen Guo at Lehigh, and Aviral Shrivastava at Arizona State. Jointly supported by NSF and Intel. $2,000,000. My group's portion: $500,000. 2017–2020.
- ONR (Office of Naval Research) N00014-17-1-2539: Semantics-Directed Binary Reverse Engineering and Transformation Validation. Principal Investigator. $500,000. 2017–2020.
- NSF CNS-1408826: Retrofitting Software for Defense-in-Depth. Principal Investigator. Collaborative research with Trent Jaeger at Penn State, Vinod Ganapathy at Rutgers, and Christian Skalka at U. of Vermont. $1,200,000. My group's portion: $300,000. 2014–2018.
- DARPA (Defense Advanced Research Projects Office) N6600117C4052: Automatic Generation of Anti-Specifications from Exploits for Scalable Program Hardening. Principal Investigator. Subcontractor to Virginia Tech. My group's portion:$190,000. 2017–2018.
- AFRL (Air Force Research Laboratory) FA8750-14-C-0179: SLICE: Secure Lightweight Cloud Computing Environment. Principal Investigator. Subcontractor to Intelligent Automation Inc.. $30,000. 2014–2015.
- NSF CCF-1217710: Reusable Tools for Formal Modeling of Machine Code. Principal Investigator. Collaborative research with Greg Morrisett at Harvard. $477,495. My group's portion: $258,785. 2012–2015. REU supplement: $16,000.
- NSF CAREER CCF-1149211: User-Space Protection Domains for Compositional Information Security. Principal Investigator, NSF, $483,125, 2012–2017.
- Google Research Award: A Fully Certified Native Client Verifier. Principal Investigator. $50,100. 2012–2013.
- Google Research Award: Native Client with Trustworthy Verifier and Stronger Security. Principal Investigator. $60,000. 2010–2011.
- NSF CCF-0915157: Securing Multilingual Software Systems. Principal Investigator. Collaborative research with Greg Morrisett at Harvard. CCF-0915157. $480,131. My group's portion: $265,048. 2009–2012. REU supplement: $16,000.
- NSF IIS-0854606: Structuring, Reasoning, and Querying in a Very Large Medical Image Database. Principal Investigator. Collaborative Research with Xiaolei Huang and Dan Lopresti at Lehigh, and George Nagy at RPI. $392,000. My group's portion: $54,464. 2008–2011.

University internal research grants

- Lehigh Collaborative Research Opportunity (CORE) Grant. Principal Investigator (with Co-PI Parv Venkitasubramaniam from the ECE department). Quantitative Information Flow for

Security and Privacy in Software Systems, $36,770, 2014–2015.

Professional Membership

ACM member; IEEE senior member.

Publications and Creative Activities

Journal papers and book chapters

1. Sun, C., Ma, Y., Zeng, D., Tan, G., Ma, S., and Wu, Y. (2022). muDep: Mutation-based dependency generation for precise taint analysis on android native code. *IEEE Transactions on Dependable and Secure Computing*, page To appear

2. Singh, A., Dave, S., Zardoshti, P., Brotzman, R., Zhang, C., Guo, X., Shrivastava, A., Tan, G., and Spear, M. F. (2021). SPX64: A scratchpad memory for general-purpose microprocessors. *ACM Trans. Archit. Code Optim.*, **18**(1), 14:1–14:26. [paper]

3. Fan, Y., Bai, J., Lei, X., Lin, W., Hu, Q., Wu, G., Guo, J., and Tan, G. (2021b). PPMCK: Privacy-preserving multi-party computing for k-means clustering. *Journal of Parallel and Distributed Computing*, **154**, 54–63. [paper]

4. Fan, Y., Liu, S., Lei, X., Li, K.-C., Lin, W., and Tan, G. (2021a). One enhanced secure access scheme for outsourced data. *Information Sciences*, **561**, 230–242

5. Fan, Y., Bai, J., Lei, X., Zhang, Y., Zhang, B., Li, K.-C., and Tan, G. (2020a). Privacy preserving based logistic regression on big data. *Journal of Network and Computer Applications*, **171**, 102769

6. Huang, Z., Lie, D., Tan, G., and Jaeger, T. (2020). Using safety properties to generate vulnerability patches. *USENIX ;login*, **45**(4), 23–28. [paper]

7. Tian, K., Tan, G., Yao, D., and Ryder, B. (2020b). Prioritizing data flows and sinks for app security transformation. *Journal of Computers & Security*, **92**, 101750. [paper]

8. Tian, K., Yao, D., Rider, B., Tan, G., and Peng, G. (2020a). Detection of repackaged Android malware with code-heterogeneity features. *IEEE Transactions on Dependable and Secure Computing*, **17**(1), 64–77. [paper]

9. Fan, Y., Zhao, G., Li, K., Zhang, B., Tan, G., Sun, X., and Xia, F. (2020b). SNPL: One scheme of securing nodes in IoT perception layer. *Sensors*, **20**(4), 1090

10. Celik, Z. B., McDaniel, P., Tan, G., Babun, L., and Uluagac, A. S. (2019c). Verifying internet of things safety and security in physical spaces. *IEEE Security and Privacy*, **17**(5), 30–37. [paper]

11. Celik, Z. B., Fernandes, E., Pauley, E., Tan, G., and McDaniel, P. (2019b). Program analysis of commodity IoT applications for security and privacy: Opportunities and challenges. *ACM Computing Surveys (CSUR)*, **52**, 74:1–74:30. [paper]

12. Fan, Y., Lin, X., Tan, G., Liang, W., Lei, J., and Lei, X. (2019c). Tracechain: a blockchain-based scheme to protect data confidentiality and traceability. *Software: Practice and Experience*. [paper]

13. Fan, Y., Lin, X., Liang, W., Tan, G., and Nanda, P. (2019b). A secure privacy preserving deduplication scheme for cloud computing. *Future Generation Computer Systems*, **101**, 127–135

14. Fan, Y., Lin, X., Tan, G., Zhang, Y., Dong, W., and Lei, J. (2019a). One secure data integrity verification scheme for cloud storage. *Future Generation Computer Systems*, **96**, 376–385. [paper]

15. Li, H., Wang, Y., Yin, J., and Tan, G. (2019). SmartShell: Automated shell scripts synthesis from natural language. *International Journal of Software Engineering and Knowledge Engineering*, **29**(2), 197–220. [paper]

16. Yin, J., Tan, G., Li, H., Bai, X., Wang, Y., and Hu, S. (2019). Debugopt: Debugging fully optimized natively compiled programs using multistage instrumentation. *Science of Computer Programming*, **169**, 18–32. [paper]

17. Tan, G. and Niu, B. (2018). Protecting dynamic code. In P. Larsen and A.-R. Sadeghi, editors, *The Continuing Arms Race*, chapter 2, pages 25–60. [paper]

18. Fan, Y., Liu, S., Tan, G., and Lin, X. (2018a). CSCAC: one constant-size CPABE access control scheme in trusted execution environment. *International Journal of Computational Science and Engineering*, pages 162–168

19. Fan, Y., Liu, S., Tan, G., and Qiao, F. (2018b). Fine-grained access control based on trusted execution environment. *Future Generation Computer Systems*. [paper]

20. Tan, G. and Morrisett, G. (2018). Bidirectional grammars for machine-code decoding and encoding. *Journal of Automated Reasoning*, **60**(3), 257–277. [paper]

21. Tan, G. (2017). Principles and implementation techniques of software-based fault isolation. *Foundations and Trends in Privacy and Security*, **1**(3), 137–198. [paper]. [slides]

22. Yin, J., Tan, G., Bai, X., and Hu, S. (2015). WebC: Toward a portable framework for deploying legacy code in web browsers. *Science China Information Sciences*, **58**(7), 1–15. [paper]

23. Tan, G. (2015). JNI Light: an operational model for the core JNI. *Mathematical Structures in Computer Science*, **25**(4), 805–840. [paper]

24. Li, S. and Tan, G. (2014a). Exception analysis in the Java Native Interface. *Science of Computer Programming*, **89**, 273–297. [paper]

25. Sun, M., Tan, G., Siefers, J., Zeng, B., and Morrisett, G. (2013). Bringing Java's wild native world under control. *ACM Transactions on Information and System Security (TISSEC)*, **16**(3), 9:1–9:28. [paper]

26. Bai, S., Yin, J., Tan, G., Wang, Y., and Hu, S. (2011). FDTL: a unified flash memory and hard disk translation layer. *IEEE Transactions on Consumer Electronics*, **57**(4), 1719–1727. [paper]

27. Kim, E., Huang, X., and Tan, G. (2011). Markup SVG: An online content-aware image abstraction and annotation tool. *IEEE Transactions on Multimedia*, **13**(5), 993–1006. [paper]

28. Tan, G., Shao, Z., Feng, X., and Cai, H. (2011). Weak updates and separation logic. *New Generation Compting*, **29**(1), 3–29. [paper]

29. Ahmed, A., Appel, A., Richards, C., Swadi, K., Tan, G., and Wang, D. (2010). Semantic foundations for typed-assembly languages. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, **32**(3), 1–67. [paper]

30. Jiang, M., Zhu, X., Gielen, G. G. E., Drábek, E., Xia, Y., Tan, G., and Bao, T. (2002). Braille to print translations for Chinese. *Information & Software Technology*, **44**(2), 91–100. [paper]

Refereed conference and workshop papers

31. Huang, Y., Narayanan, V., Detweiler, D., Huang, K., Tan, G., Jaeger, T., and Burtsev, A. (2022b). Ksplit: Automating device driver isolation. In *16th USENIX Symposium on Operating Systems Design and Implementation (OSDI'22)*, page To appear. [paper]

32. Tizpaz-Niari, S., Kumar, A., Tan, G., and Trivedi, A. (2022). Fairness-aware configuration of machine learning libraries. In *44th International Conference on Software Engineering (ICSE)*, page To appear. [paper]

33. Kim, S. H., Sun, C., Zeng, D., and Tan, G. (2022). Binpointer: Towards precise, sound, and scalable binary-level pointer analysis. In *ACM SIGPLAN 2022 International Conference Compiler Construction (CC)*, page To appear. [paper]

34. Wang, Y.-P., Hu, X.-Q., Zou, Z.-X., Tan, W., and Tan, G. (2022). ROS-SF: A transparent and efficient ROS middleware using serialization-free message. In *23rd ACM/IFIP Middleware Conference*, page To appear

35. Beugin, Y., Burke, Q., Hoak, B., Sheatsley, R., Pauley, E., Tan, G., Hussain, S. R., and McDaniel, P. (2022). Building a privacy-preserving smart camera system. In *Privacy Enhancing Technologies Symposium (PETS)*, page To appear. [paper]

36. Huang, K., Huang, Y., Payer, M., Qian, Z., Sampson, J., Tan, G., and Jaeger, T. (2022a). The taming of the stack: Isolating stack data from memory errors. In *Network and Distributed System Security Symposium (NDSS)*. [paper]

37. Zhang, Y., Liu, X., Sun, C., Zeng, D., Tan, G., Kan, X., and Ma, S. (2021). ReCFA: Resilient control-flow attestation. In *Annual Computer Security Applications Conference (ACSAC)*, pages 311–322. [paper]

38. Jia, X., Kumar, A., and Tan, G. (2021). A derivative-based parser generator for visibly pushdown grammars. *Proceedings of the ACM on Programming Languages*, **5**(OOPSLA), 1–24. [paper]

39. Brotzman, R., Zhang, D., Kandemir, M., and Tan, G. (2021b). SpecSafe: Detecting cache side channels in a speculative world. *Proceedings of the ACM on Programming Languages*, **5**(OOPSLA), 1–28. [paper]

40. Zeng, D., Niu, B., and Tan, G. (2021). MazeRunner: Evaluating the attack surface of control-flow integrity policies. In *20th IEEE International Conference On Trust, Security And Privacy In Computing And Communications (TrustCom)*, page To appear

41. Kan, X., Sun, C., Liu, S., Huang, Y., Tan, G., Ma, S., and Zhang, Y. (2021). Sdft: A PDG-based summarization for efficient dynamic data flow tracking. In *21st IEEE International Conference on Software Quality, Reliability, and Security (QRS)*

42. Muntean, P., Viehoever, R., Lin, Z., Tan, G., Grossklags, J., and Eckert, C. (2021). iTOP: Automating counterfeit object-oriented programming attacks. In *International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, pages 162–176. [paper]

43. Kim, S. H., Sun, C., Zeng, D., and Tan, G. (2021). Refining indirect call targets at the binary level. In *Network and Distributed System Security Symposium (NDSS)*. [paper]

44. Brotzman, R., Zhang, D., Kandemir, M., and Tan, G. (2021a). Ghost thread: Effective user-space cache side channel protection. In *11th ACM Conference on Data and Application Security and Privacy (CODASPY)*, pages 233–244. [paper]

45. Huang, Z., Jaeger, T., and Tan, G. (2021). Fine-grained program partitioning for security. In *14th European Workshop on Systems Security (EuroSec)*, pages 21–26. [paper]

46. Paranjpe, A. and Tan, G. (2021). Bohemia: A validator for parser frameworks. In *7th Workshop on Language-Theoretic Security (LangSec)*. [paper]

47. Muntean, P., Neumayer, M., Lin, Z., Tan, G., Grossklags, J., and Eckert, C. (2020). rhoFEM: Efficient backward-edge protection using reversed forward-edge mappings. In *Annual Computer Security Applications Conference (ACSAC)*, pages 466–479. [paper]

48. Ahmed, M. S., Xiao, Y., Snow, K. Z., Tan, G., Monrose, F., and Yao, D. (2020). Methodologies for quantifying (re-)randomization security and timing under JIT-ROP. In *26th ACM Conference on Computer and Communications Security (CCS)*, pages 1803–1820. [paper]

49. Norris, M., Celik, Z. B., Venkatesh, P., Zhao, S., McDaniel, P. D., Sivasubramaniam, A., and Tan, G. (2020). IoTRepair: Systematically addressing device faults in commodity IoT. In *5th ACM/IEEE Conference on Internet of Things Design and Implementation (IoTDI)*, pages 142–148. [paper]

50. Narayanan, V., Huang, Y., Tan, G., Jaeger, T., and Burtsev, A. (2020). Lightweight kernel isolation with virtualization and VM functions. *16th ACM International Conference on Virtual Execution Environments (VEE)*, pages 157–171. [paper]

51. Liu, S., Zeng, D., Huang, Y., Capobianco, F., McCamant, S., Jaeger, T., and Tan, G. (2019). Program-mandering: Quantitative privilege separation. In *26th ACM Conference on Computer and Communications Security (CCS)*, pages 1023–1040. [paper]

52. Wang, Y.-P., Hu, X.-Q., Zou, Z.-X., Tan, W., and Tan, G. (2019). IVT: an efficient method for sharing subtype polymorphic objects. In *ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*, pages 130:1–130:22. [paper]

53. Muntean, P., Neumayer, M., Lin, Z., Tan, G., Grossklags, J., and Eckert, C. (2019). LLVM-CFI: Analyzing static control flow integrity protections. In *Annual Computer Security Applications Conference (ACSAC)*, pages 584–597. [paper]

54. Huang, Z., Lie, D., Tan, G., and Jaeger, T. (2019). Using safety properties to generate vulnerability patches. In *IEEE Symposium on Security and Privacy (S&P)*, pages 539–554. [paper]

55. Brotzman, R., Liu, S., Zhang, D., Tan, G., and Kandemir, M. (2019). CaSym: Cache aware symbolic execution for side channel detection and mitigation. In *IEEE Symposium on Security and Privacy (S&P)*, pages 364–380. [paper]

56. Celik, Z. B., Tan, G., and McDaniel, P. (2019a). IoTGuard: Dynamic enforcement of security and safety policy in commodity IoT. In *Network and Distributed System Security Symposium (NDSS)*. [paper]

57. Huang, Z. and Tan, G. (2019). Rapidly mitigating vulnerabilities with security workarounds. In *Workshop on Binary Analysis Research (BAR)*. [paper]

58. Muntean, P., Fischer, M., Tan, G., Lin, Z., Grossklags, J., and Eckert, C. (2018). tauCFI: Type-assisted control flow integrity for x86-64 binaries. In *International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, pages 423–444. [paper]

59. Celik, Z. B., Babun, L., Sikder, A. K., Aksu, H., Tan, G., McDaniel, P., and Uluagac, A. S. (2018a). Sensitive information tracking in commodity IoT. In *27th Usenix Security Symposium*, pages 1687–1704. [paper]

60. Celik, Z. B., McDaniel, P., and Tan, G. (2018b). Soteria: Automated IoT safety and security analysis. In *USENIX Annual Technical Conference (ATC)*, pages 147–158. [paper]

61. Fan, Y., Liu, S., Tan, G., Lin, X., Zhao, G., and Bai, J. (2018c). One secure access scheme based on trusted execution environment. In *17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications (TrustCom)*, pages 16–21. [paper]

62. Zeng, D. and Tan, G. (2018). From debugging-information based binary-level type inference to CFG generation. In *8th ACM Conference on Data and Application Security and Privacy (CODASPY)*, pages 366–376. **Outstanding paper award**. [paper]

63. Liu, S., Tan, G., and Jaeger, T. (2017). PtrSplit: Supporting general pointers in automatic program partitioning. In *24th ACM Conference on Computer and Communications Security (CCS)*, pages 2359–2371. [paper]

64. Tan, G. and Jaeger, T. (2017). CFG construction soundness in control-flow integrity. In *ACM SIGSAC Workshop on Programming Languages and Analysis for Security (PLAS)*, pages 3–13. [paper]

65. Tian, K., Tan, G., Yao, D., and Ryder, B. (2017). ReDroid: Prioritizing data flows and sinks for app security transformation. In *ACM Workshop on Forming an Ecosystem Around Software Transformation (FEAST)*, pages 35–41. [paper]

66. Tan, G. and Morrisett, G. (2016). Bidirectional grammars for machine-code decoding and encoding. In *8th International Conference on Verified Software: Theories, Tools, and Experiments (VSTTE)*, pages 73–89. [paper]

67. Guo, X., Shrivastava, A., Spear, M., and Tan, G. (2016). Languages must expose memory heterogeneity. In *Second International Symposium on Memory Systems (MEMSYS)*, pages 251–256. [paper]

68. Tian, K., Yao, D., Ryder, B. G., and Tan, G. (2016). Analysis of code heterogeneity for high-precision classification of repackaged malware. In *Workshop on Mobile Security Technologies (MoST)*, pages 262–271. [paper]

69. Niu, B. and Tan, G. (2015). Per-input control-flow integrity. In *22nd ACM Conference on Computer and Communications Security (CCS)*, pages 914–926. [paper]. [webpage]

70. Muthukumaran, D., Talele, N., Jaeger, T., and Tan, G. (2015). Producing hook placements to enforce expected access control policies. In *7th International Symposium on Engineering Secure Software and Systems (ESSoS)*, pages 178–195. [paper]

71. Ganapathy, V., Jaeger, T., Skalka, C., and Tan, G. (2014). Assurance for defense-in-depth via retrofitting. In *8th Layered Assurance Workshop (LAW)*. [paper]

72. Niu, B. and Tan, G. (2014b). RockJIT: Securing just-in-time compilation using modular control-flow integrity. In *21st ACM Conference on Computer and Communications Security (CCS)*, pages 1317–1328. [paper]

73. Li, S. and Tan, G. (2014b). Finding reference-counting errors in Python/C programs with affine analysis. In *European Conference on Object-Oriented Programming (ECOOP)*, pages 80–104. [paper]

74. Sun, M. and Tan, G. (2014). NativeGuard: Protecting Android applications from third-party native libraries. In *7th ACM Conference on Security & Privacy in Wireless and Mobile Networks (WiSec)*, pages 165–176. [paper]

75. Niu, B. and Tan, G. (2014a). Modular control-flow integrity. In *ACM Conference on Programming Language Design and Implementation (PLDI)*, pages 577–587. [paper]. [webpage]

76. Niu, B. and Tan, G. (2013b). Monitor integrity protection with space efficiency and separate compilation. In *20th ACM Conference on Computer and Communications Security (CCS)*, pages 199–210. [paper]

77. Zeng, B., Tan, G., and Erlingsson, Ú. (2013). Strato: A retargetable framework for low-level inlined-reference monitors. In *22nd Usenix Security Symposium*, pages 369–382. [paper]

78. Niu, B. and Tan, G. (2013a). Efficient user-space information flow control. In *8th ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, pages 131–142. [paper]

79. Su, X., Chuah, M.-C., and Tan, G. (2012). Smartphone dual defense protection framework: Detecting malicious applications in Android markets. In *8th International Conference on Mobile Ad-hoc and Sensor Networks*, pages 153–160. [paper]

80. Li, S., Liu, D. Y., and Tan, G. (2012). JATO: Native code atomicity for Java. In *Asian Symposium on Programming Languages and Systems (APLAS)*, pages 2–17. [paper]

81. Sun, M. and Tan, G. (2012). JVM-portable sandboxing of Java's native libraries. In *17th European Symposium on Research in Computer Security (ESORICS)*, pages 842–858. [paper]. [webpage]

82. Niu, B. and Tan, G. (2012). Enforcing user-space privilege separation with declarative architectures. In *Proceedings of the Sixth ACM Workshop on Scalable Trusted Computing (STC)*, pages 9–20. [paper]

83. Morrisett, G., Tan, G., Tassarotti, J., Tristan, J.-B., and Gan, E. (2012). Rocksalt: Better, faster, stronger SFI for the x86. In *ACM Conference on Programming Language Design and Implementation (PLDI)*, pages 395–404. [paper]. [webpage]

84. Zeng, B., Tan, G., and Morrisett, G. (2011). Combining control-flow integrity and static analysis for efficient and validated data sandboxing. In *18th ACM Conference on Computer and Communications Security (CCS)*, pages 29–40. [paper]

85. Li, S. and Tan, G. (2011). JET: Exception checking in the Java Native Interface. In *ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*, pages 345–358. [paper]

86. Fedynyshyn, G., Chuah, M. C., and Tan, G. (2011). Detection and classification of different botnet C&C channels. In *8th International Conference on Autonomic and Trusted Computing (ATC)*, pages 228–242. [paper]

87. Tan, G. (2010). JNI Light: An operational model for the core JNI. In *Asian Symposium on Programming Languages and Systems (APLAS)*, pages 114–130. [paper]

88. Kim, E., Huang, X., Tan, G., Long, L. R., and Antani, S. K. (2010). A hierarchical SVG image abstraction layer for medical imaging. In *SPIE Medical Imaging: Advanced PACS-based Imaging Informaticsand Therapeutic Applications*, volume 7628. [paper]

89. Siefers, J., Tan, G., and Morrisett, G. (2010). Robusta: Taming the native beast of the JVM. In *17th ACM Conference on Computer and Communications Security (CCS)*, pages 201–211. [paper]. [webpage]. [slides]

90. Tan, G., Shao, Z., Feng, X., and Cai, H. (2009). Weak updates and separation logic. In *Asian Symposium on Programming Languages and Systems (APLAS)*, pages 178–193. [paper]

91. Lopresti, D. P., Zhou, X., Huang, X., and Tan, G. (2009). Document analysis support for the manual auditing of elections. In *10th International Conference on Document Analysis and Recognition (ICDAR)*, pages 733–737

92. Li, S. and Tan, G. (2009). Finding bugs in exceptional situations of JNI programs. In *16th ACM Conference on Computer and Communications Security (CCS)*, pages 442–452. [paper]

93. Appel, A. W., Ginsburg, M., Hursti, H., Kernighan, B. W., Richards, C. D., Tan, G., and Venetis, P. (2009). The New Jersey voting-machine lawsuit and the AVC advantage DRE voting machine. In *Conference on Electronic Voting Technology/Workshop on Trustworthy Elections (EVT/WOTE)*. [paper]

94. Tan, G. and Croft, J. (2008). An empirical security study of the native code in the JDK. In *17th Usenix Security Symposium*, pages 365–377. [paper]. [slides]. [tech report]

95. Tan, G. and Morrisett, G. (2007). ILEA: Inter-language analysis across Java and C. In *ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*, pages 39–56. [paper]. [slides]

96. Tan, G., Chen, Y., and Jakubowski, M. H. (2006a). Delayed and controlled failures in tamper-resistant software. In *8th International Workshop on Information Hiding (IH)*, pages 216–231. [paper]. [slides]

97. Tan, G., Appel, A., Chakradhar, S., Raghunathan, A., Ravi, S., and Wang, D. (2006b). Safe Java Native Interface. In *IEEE International Symposium on Secure Software Engineering*, pages 97–106. [paper]. [slides]

98. Tan, G. and Appel, A. (2006). A compositional logic for control flow. In *International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI)*, pages 80–94. [paper]. [slides]

99. Marino, D., Chin, B., Millstein, T., Tan, G., Simmons, R. J., and Walker, D. (2006). Mechanized metatheory for user-defined type extensions. In *Workshop on Mechanizing Metatheory*

100. Ou, X., Tan, G., Mandelbaum, Y., and Walker, D. (2004). Dynamic typing with dependent types. In *Proceedings of IFIP 3rd International Conference on Theoretical Computer Science*, pages 437–450. [paper]

101. Tan, G., Appel, A., Swadi, K., and Wu, D. (2004). Construction of a semantic model for a typed assembly language. In *International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI)*, pages 30–43. [paper]. [slides]

102. Tan, G., Ou, X., and Walker, D. (2003). Enforcing resource usage protocols via scoped methods. In *Proceedings of 10th International Workshop on Foundations of Object-Oriented Languages (FOOL)*. [paper]

103. Jiang, M., Zhu, X., Xia, Y., Tan, G., Yuan, B., and Tang, X. (2000). Segmentation of mandarin braille word and braille translation based on multi-knowledge. In *5th International Conference on Signal Processing (ICSP)*, pages 2070–2073. [paper]

Other papers

- K. Tian, D. Yao, G. Tan. Android-Application Rewriting with Quantitative Information Flow Analysis. Poster at the 2016 Network and Distributed System Security Symposium (NDSS), Feb., 2016.
- B. Niu and G. Tan. Chobham: Taming JIT-ROP Attacks. Poster at the 2015 Network and Distributed System Security (NDSS) Symposium, Feb., 2015.
- B. Niu and G. Tan. uPro: A Compartmentalization Tool Supporting Fine-Grained and Flexible Security Configuration. Poster at the 18th ACM Conference on Computer and Communication Security (CCS), Oct. 2011.
- A. Appel, M. Ginsburg, H. Hursti, B. Kernighan, C. Richards, and G. Tan. Insecurities and Inaccuracies of the Sequoia AVC Advantage 9.00H DRE Voting Machine. Redacted version of expert report submitted in Gusciora v. Corzine, September 2008.

- J. Croft and G Tan. Security Analysis of the Native Code in Sun's JDK. In 23rd Annual Computer Security Applications Conference (ACSAC), work-in-progress session, Dec. 2007.
- G. Tan. A Compositional Logic for Control Flow and its Application in Foundational Proof-Carrying Code, Princeton University Ph.D. Thesis, July 2005.
- G. Tan, A. Appel, S. Chakradhar, A. Raghunathan, S. Ravi and D. Wang. Safe Heterogeneous Applications: Curing the Java Native Interface. Princeton University Technical Report, TR-715-04, Oct. 2004.
- G. Tan and A. Appel. A Typed Calculus for Machine Instructions and its Semantics in Higher-order Logic. Princeton University Technical Report, March 2004.
- X. Ou, G. Tan, Y. Mandelbaum and D. Walker. Dynamic Typing with Dependent Types (Extended Version). Princeton University Technical Report TR-695-04, April 2004.
- G. Tan and A. Appel. Semantics of Machine Instructions at Multiple Levels of Abstraction. Short paper at the 16th Symposium on Logic in Computer Science (LICS), 2001.

Patents

- G. Tan and B. Niu. Methods for Enforcing Control Flow of Computer Programs. US Patent No. 9,361,102, filed in Jun 2015, awarded in Jun 2016.
- Y. Chen and G. Tan. Tamper Response Mechanism. US Patent No. 7818799, filed in May 2006, awarded in Oct 2010.

Publicly released software

- PDG construction in LLVM. We released it in 2019 for LLVM 5.0 and 2020 for LLVM 9.0, at https://bitbucket.org/psu_soslab/program-dependence-graph-in-llvm/.
- MCFI: a low-overhead CFI implementation with support for dynamic library loading and just-in-time compilation. We released it in 2015 at https://github.com/mcfi.
- Robusta: a framework that allows JVM administrators to constrain native code with different trust levels, similar to how the security of Java code is configured. We released it in 2011 under the BSD license at http://www.cse.psu.edu/~gxt29/projects/gonative/
- RockSalt: a new machine-code verifier for Google's Native Client, with a formal correctness proof mechanized in Coq. We released it in 2012 under the GPL license at http://www.cse.psu.edu/~gxt29/projects/gonative/.

Presentations

Invited talks at professional conferences and summer schools

- **Keynote talk**. Towards Secure and Reliable IoT Applications. 2nd Workshop on the Internet of Things Security and Privacy (IoT S&P), Oct. 2019.
- **Invited talk**. Bidirectional and Executable Specifications of Machine Code Decoding and Encoding. Invited talk at the Fifth Workshop on Language-Theoretic Security (LangSec), San Francesco, May 2018.

- **Keynote talk**. Protecting Dynamic Code by Modular Control-Flow Integrity. International Workshop on Modularity Across the System Stack (MASS 2016), Mar. 2016.
- **Invited talk**. Reusable Tools for Formal Modeling of Machine Code. Invited talk at the Principles in Practice (PiP) Workshop associated with the 2014 POPL Conference, San Diego, Jan. 2014.
- **Invited lecture**. 2012 Summer School on Cryptography and Principles of Software Security, Binary-Level Software Security, Penn State University, May 2012.

Invited talks at university colloquiums and seminars

- Compiler-based Side Channel Detection and Mitigation. At the Dagstuhl Seminar on Secure Compilation, Dagstuhl, Nov, 2021.
- Recent Advances in Automatic Privilege Separation. DC-Area Anonymity, Privacy, and Security Seminar (DCAPS). Dec. 2019. Also at Microsoft Research Cambridge, Feb. 2020, and at Intel Security Forum, Apr, 2020.
- Checking IoT Apps for Property Violations. University of Louisiana at Lafayette. Feb. 2019; and China University of Petroleum, July 2019.
- Modular Control-Flow Integrity. At the Dagstuhl Seminar "The Continuing Arms Race: Code-Reuse Attacks and Defenses", Dagstuhl, July, 2015.
- A Compiler-Centric Approach to Software Security. Penn State. Apr. 2015.
- Control Flow Integrity: Efficiency and Modularity. Virginia Tech. Oct. 2014.
- Modular Control Flow Integrity. Zhejiang University, Jul 2014.
- Reusable Tools for Formal Modeling of Machine Code. Chinese Institute of Software, Jul 2014.
- Software Security at the Binary Level. Center for the Advancement of Research and Education at Rochester Institute of Technology, May 2012. Also at Peking University, Jun 2012. Also at Intelligent Automation, Inc. Jan 2013.
- Towards Verifiably Safe Machine Code. CyLab at Carnegie Mellon University, Mar. 2012.
- GoNative: Safe Native Code for Safe Languages. USTC-Yale Joint Research Center, Suzhou, China, Dec. 2010.
- Protecting Java from Native Code. IBM's T.J. Watson research center in Hawthorne, Feb. 2009. Also at Department of Computer Science and Technology, Tsinghua University, May 2009.
- Language-Based Security for Java-C Interoperation. UCLA Seminar, Jul. 2008.
- Interface Safety in Multilingual Software. Northeastern Programming Languages Seminar, Feb. 2008.
- Security Analysis of the Native Code in the JDK. Princeton University Computer-Science Security Lunch Seminar, Nov. 2007.
- Inter-Language Analysis across Java and C. Boston University Computer-Science Colloquium, Oct. 2007.
- Towards Reliable and Secure Software. Lehigh University Colloquium, Apr. 2006.

- Safe Java Native Interface. Triforce seminar, Harvard University, Mar. 2006.
- A Compositional Logic for Control Flow. The Church Project Seminar, Boston University, Oct. 2005.
- Reliable and Secure Software through Static Verification and Dynamic Checking. NEC Labs America. Apr. 2005.
- Structured Verification of Unstructured Machine Code. Toyota Technological Institute at Chicago. Feb. 2005.
- Construction of a Semantic Model for a Typed Assembly Language. Ottawa Carleton Logic Seminar, University of Ottawa, Nov. 2003.
- Protection Against Untrusted Code. Microsoft Research, Feb. 2002.

Paper presentations at professional conferences

- Program Partitioning for Secure Memory. In Intel Computer Assisted Programming for Heterogeneous Architectures (CAPA) Annual Meeting, virtual meeting, Oct. 2020.
- Program Partitioning for Secure Memory. In Intel Computer Assisted Programming for Heterogeneous Architectures (CAPA) Annual Meeting, Santa Clara, Sep. 2019.
- Semantics-Directed Binary Reverse Engineering and Translation Validation. In ONR Total Platform Cyber Protection (TPCP) Annual Meeting, Boston, June 2019.
- Program Partitioning for Secure Memory. In Intel Computer Assisted Programming for Heterogeneous Architectures (CAPA) Annual Meeting, Santa Clara, Sep. 2018.
- Semantics-Directed Binary Reverse Engineering and Translation Validation. In ONR Total Platform Cyber Protection (TPCP) Annual Meeting, Seattle, May 2018.
- Bidirectional Grammars for Machine-Code Decoding and Encoding. In 8th Working Conference on Verified Software: Theories, Tools, and Experiments (VSTTE), Jul. 2016. Also in Deep Spec Workshop, May 2016.
- Software Security: A Compiler-Based Perspective. School of Electrical Engineering and Computer Science Industrial and Professional Advisory Council (IPAC) meeting, Mar. 2016. Also at the 2016 Silicon Happy Valley conference.
- Modular Control-Flow Integrity. In the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI), Edinburgh, UK, Jun. 2014.
- Monitor Integrity Protection with Space Efficiency and Separate Compilation. In 20th ACM Conference on Computer and Communications Security (CCS), Berlin, Germany, Nov. 2013.
- Towards Safe Language Interoperation. Pre-Program-Committee-Meeting Workshop of OOPSLA. University of California at Irvine, May 2013.
- JATO: Native Code Atomicity for Java. Tenth Asian Symposium on Programming Languages and Systems (APLAS 2012), Kyoto, Japan, Dec. 2012.
- JNI Light: An Operational Model for the Core JNI. Eighth Asian Symposium on Programming Languages and Systems (APLAS 2010), Shanghai, China, Dec. 2010.
- Weak Updates and Separation Logic. New Jersey Programming Languages Seminar, Apr. 2010.

- Weak Updates and Separation Logic. Seventh Asian Symposium on Programming Languages and Systems (APLAS 2009), Seoul, South Korea, Dec. 2009.
- An Empirical Security Study of the Native Code in the JDK. Seventeenth USENIX Security Symposium (Security '08), San Jose, CA, Jul. 2007.
- ILEA: Inter-Language Analysis across Java and C. Twenty-second ACM Conference on Object-Oriented Programming, Systems, Languages & Applications (OOPSLA '07), Research Paper Track, Montreal, Canada, Oct. 2007.
- Delayed and Controlled Failures in Tamper-Resistant Software. Eighth Information Hiding (IH '07), Old Town Alexandria, Virginia, USA, Jul. 2006.
- Safe Java Native Interface. IEEE International Symposium on Secure Software Engineering (ISSSE 06), McLean, Virginia, USA, Mar. 2006.
- A Compositional Logic for Control Flow. Seventh International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI '06), Charleston, South Carolina, USA, Jan. 2006.
- Construction of a Semantic Model for a Typed Assembly Language. Fifth International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI '04), Venice, Italy, Jan. 2004.
- Enforcing Resource Usage Protocols via Scoped Methods. Tenth International Workshop on Foundations of Object-Oriented Languages (FOOL '03), New Orleans, Louisiana, USA, Jan. 2003.
- Semantics of Machine Instructions at Multiple Levels of Abstraction. NJ Programming Languages and Systems Seminars, AT&T Research, May 2001.

Teaching and Advising

Courses taught (unless noted otherwise, all courses are 3-credit courses)

- Spring 2020. CMPSC 461. Programming Language Concepts. Session 1: 130 students; instructor quality evaluation: 6.33/7. Session 2: 130 students; instructor quality evaluation: 6.43/7.
- Spring 2019. CMPSC 447. Software Security. 41 students; instructor quality evaluation: 6.6/7.
- Spring 2019. CSE 597. Special Topics on Binary-Level Program Analysis. 23 students; instructor quality evaluation: 6.3/7.
- Spring 2018. CMPSC 461. Programming Language Concepts. 143 students; instructor quality evaluation: 6.7/7.
- Spring 2017. CMPSC 461. Programming Language Concepts. 121 students; instructor quality evaluation: 6.5/7.
- Fall 2016. CSE 597. Special topics on theorem proving and static analysis. 5 students; instructor quality evaluation: 7/7.
- Spring 2016. CMPSC 443. Introduction to Computer and Network Security. 56 students; instructor quality evaluation: 6.4/7.

- Fall 2015. CSE 262. Programming Languages.
- Fall 2015. CSE 411. Advanced Programming Techniques.
- Spring 2015. CSE 262. Programming Languages.
- Fall 2014. CSE 262. Programming Languages.
- Fall 2014. CSE 411. Advanced Programming Techniques.
- Spring 2014. CSE 262. Programming Languages. 30 students; evaluation: 4.71/5.
- Fall 2013. CSE 262. Programming Languages. 69 students; evaluation: 4.29/5
- Fall 2013. CSE 411. Advanced Programming Techniques. 28 students; evaluation: 4.79/5
- Spring 2013. CSE 262. Programming Languages. 32 students; evaluation: 4.58/5.
- Fall 2012. CSE 262. Programming Languages. 49 students; evaluation: 4.56/5.
- Fall 2012. CSE 334/434. Software System Security. 19 students; evaluation: 4.79/5.
- Spring 2012. CSE 262. Programming Languages. 54 students; evaluation: 4.5/5.
- Fall 2011. CSE 262. Programming Languages. 28 students; evaluation: 4.90/5.
- Fall 2011. CSE 497. Advanced Programming Languages. 13 students; evaluation: 4.31/5.
- Fall 2010. CSE 262. Programming Languages. 18 students; evaluation: 4.85/5.
- Fall 2010. CSE 397/497. Software System Security. 21 students; evaluation: 4.93/5.
- Spring 2010. CSE 216. Software Engineering. 30 students; evaluation: 4.04/5.
- Fall 2009. CSE 397/497. Programming Language Design & Analysis. 8 students; evaluation: 5/5.
- Spring 2009. CSE 216. Software Engineering. 31 students; evaluation: 4.43/5.
- Fall 2008. CSE 397/497. Software System Security. 7 students; evaluation: 4.83/5.
- Fall 2007. CS361. Information Security. Boston College.
- Spring 2007. CS366. Principles of Programming Languages. Boston College.
- Fall 2006. CS390. Information Security. Boston College.
- Fall 2005. CS383. Algorithms. Boston College.

Current Ph.D. students

- Sun Hyoung Kim (Ph.D. student, fall 2017–now).
- Michael Norris (Ph.D. student, fall 2017–now).
- Yongzhe Huang (Ph.D. student, fall 2019–now).
- Ashish Kumar (Ph.D. student, fall 2019–now).
- Xiaodong Jia (Ph.D. student, fall 2019–now).
- Jialun Zhang (Ph.D. student, fall 2021–now).

Graduated Ph.D. students

- Dongrui Zeng. Graduated in Dec 2021. Thesis title: Evaluating the Attack Surface of Control Flow Integrity.
- Robert Brotzman Smith. Graduated in May 2021. Thesis title: Detecting and Mitigating Cache-Based Side-Channels.
- Shen Liu. Ph.D. Graduated in May 2020. Thesis title: Quantitative Privilege Separation with Pointer Supports.
- Ben Niu. Ph.D. Graduated in December 2015. Thesis title: Practical Control-Flow Integrity. Current Position: Research Software Development Engineer at Microsoft Research.
- Siliang Li. Ph.D. Graduated in May 2014. Thesis title: Improving Quality of Software with Foreign Function Interfaces using Static Analysis.
- Elizabeth Carter. Ph.D. Graduated in May 2014. Co-advised with Glenn Blank. Thesis title: An Intelligent Debugging Tutor For Novice Computer Science Students.

Past Postdocs

- Zhen Huang, 2018–2019; now Assistant Professor at DePaul University.
- Suman Saha, co-supervised with Greg Morrisett, 2013; now Assistant Professor at Illinois State University.
- Zhiyuan Wan, 2015; now at Zhejiang University.

Graduated Master's students

- Jialun Zhang, M.S., May 2021, Interval Parsing Grammar for File Format Specification.
- Ke Liang, M.S., May 2021, Inferring Aliasing and Buffer Size Relationship in C via Graph Neural Networks.
- Qingyuan Zhang, M.S., May 2021, A Symbolic Data Dependence Analysis with Abstract Interpretation.
- Eralp Sahin, M.S., Jul 2020, Automatic EDL Generation For Intel Software Guard Extensions.
- Qing Gong, M.S., May 2019. Extending Parallel Datalog with Lattice.
- Yongzhe Huang, M.S., May 2019. Automatic IDL Generation for Privilege Separation.
- Anish Prasad Paranjpe, M.S., May 2019. Bohemia: a Validator For Parser Frameworks.
- Hao Li, M.S., Aug 2018. System Call Trace Based Probabilistic Program Modeling for Exploitation Detection.
- Ashley Huhman, M.S., May 2018. Binary-Level Type Inference Using Datalog.
- Sheng-Hsiu Lin, M.S., May 2015. Alias Analysis in LLVM.
- Mengtao Sun, M.A., May 2012.
- Joseph Siefers, M.S., May 2010. Robusta: Taming the Native Beast of the JVM.

Member of Ph.D. thesis committees

- Zeyu Ding, Penn State University.
- Yuxin Wang, Penn State University.
- Spyridoula Gravani, University of Rochester.
- Peixuan Li, Penn State University.
- Lunpin Yuan, Penn State University, 2017. A Study of Android Security: From User-generated Data to User-generated Code.
- Z. Berkay Celik, Penn State University, 2019. Automated IoT Security and Privacy Analysis.
- Eunjung Yoon, Penn State University, 2019. Ensuring Service Integrity in Cloud Computing Environment.
- Ke Tian, Virginia Tech, 2018. Android Security Demystified: From Malware Detection to Post-detection Rewriting.
- Xinyang Ge, Penn State University, 2016. Enforcing execution integrity for software systems.
- Yujie Liu, Lehigh University, 2015. Crafting Concurrent Data Structures.
- Wenjia Ruan, Lehigh University, 2015. Accelerating Transactional Memory by Exploiting Platform Specificity.
- Rui Shi, Boston University, 2007. Types for Safe Resource Sharing in Sequential and Concurrent Programming.

Member of M.S. thesis committees

- Cong Ma, Quantifying and Mitigating Cache Side Channel Leakage with Differential Set, 2021.
- Yohan Beugin, Building a Secure and Privacy-Preserving Smart-Camera system, Penn State University, 2021.
- Kaiming Huang, DataGuard: Guarded Pages For Augmenting Stack Object Protections, Penn State University, 2020.
- Michael Steward, Global Permission Derivation Chain Granting and Revoking Permissions Using a Distributed Ledger, Penn State University, 2020.
- Adam Mohammed, Detecting Non-Constant Time Code in Cryptography Libraries using a Static Information Flow Analysis, Penn State University, 2018.
- Srikumar Sridhar, Testbed Design for Evaluation of Active Cyber Defense Systems, Penn State University, 2018.

Undergraduate research advising

- Honors thesis advising. Normen Yu (2022); Brian Ouzomgi (2021); Alyssa Jo Tice (2020) ; Corey Capooci (2018); Apurva Bhogale (2018).
- Undergraduate research assistants. Francesco Grossi (summer 2014). Sara Huser (summer 2014). Robert Brotzman Smith and Matthew Hartman (summer 2014). Matthew Kilgore (summer 2013; **Won Honorable Mention in 2015 CRA's Outstanding Undergraduate**

Award). Matthew Messersmith (summer 2013). Mark Kogan (spring, summer and fall 2012, summer 2013). Tyler Trephan (summer 2012). Alex Galakatos (summer 2011). David Stolfo (spring 2011). Evans Kosgei (summer 2010). Jason Croft (summer 2007–May 2008; **Won Honorable Mention in 2009 CRA's Outstanding Undergraduate Award**). Michael Dubinsky (summer 2006).

- Senior design projects. Irene Lau and Daniel Kramer (fall 2015). Lauren Mentzer and Lian Block (fall 2015). Rodney Christman (fall 2014). Seth Denburg and Ryan Ramirez (fall 2013).
- Independent studies. James Lamberti (fall 2014).

Services

Service to the professional community

- Advisory groups
  - DARPA Information Science and Technology (ISAT) study group, 2020–2023.
  - Steering committee member, The Joint Workshop on CPS & IoT Security and Privacy (CPSIoTSec), 2020–present.
- Editor/editorial review board membership for scholarly publications
  - Editorial Board Member, International Journal on Cybersecurity, 2017–now.
  - Associate Editor, GSTF Journal On Computing (JoC), 2014–now.
- Program chairs/co-chairs
  - Program Co-Chair, 8th Workshop on Language-Theoretic Security (LangSec), San Francisco, USA, 2022.
  - Program Co-Chair, 7th Workshop on Language-Theoretic Security (LangSec), San Francisco, USA, 2021.
  - Program Co-Chair, DARPA/ISAT workshop on Data-Oblivious Interdisciplinary Representation (DOPLR), 2020.
  - Program Co-Chair, 6th Workshop on Language-Theoretic Security (LangSec), San Francisco, USA, 2020.
- Conference organizing committees
  - Workshop Chair, ACM Conference on Computer and Communications Security (CCS), USA, 2020.
  - Poster Co-Chair, 2020 Network and Distributed System Security Symposium (NDSS), San Diego, USA.
  - Poster Co-Chair, 2019 Network and Distributed System Security Symposium (NDSS), San Diego, USA.
  - Web Chair, 2019 International Symposium on Code Generation and Optimization, Washington DC, USA.

- Member of conference program committees
  - IEEE Symposium on Security and Privacy (Oakland), 2022, 2021, 2018, 2017, and 2016.
  - Annual Network & Distributed System Security Symposium (NDSS), 2021, 2020, and 2014.
  - ACM Conference on Computer and Communications Security (CCS), 2016, 2015, and 2014.
  - 27th USENIX Security Symposium, 2018.
  - ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages & Applications (OOPSLA), 2013.
  - European Conference on Object-Oriented Programming (ECOOP), 2018.
  - World Wide Web Conference (WWW), Abuse, Security, and Privacy Track, 2011.
  - IEEE European Symposium on Security and Privacy (Euro S&P), 2020, 2021.
  - Asian Symposium on Programming Languages and Systems (APLAS), 2015 and 2013.
  - Workshop on Language-Theoretic Security (LangSec), 2018, 2017, and 2016.
  - World Conference on Information Security Applications (WISA), 2022, 2020, 2019, 2018, and 2017.
  - IEEE Secure Development Conference (SecDev), 2020, 2018 and 2017.
  - The Joint Workshop on CPS & IoT Security and Privacy (CPSIoTSec), 2021 and 2020.
  - International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE), 2022.
  - International Conference on Emerging Security Information, Systems and Technologies (SECURWARE), 2016, 2015, and 2014.
  - ACM Symposium on Information, Computer and Communications Security (AsiaCCS), 2015 and 2014.
  - IEEE International Conference on Cloud Computing Technology and Science (CloudCom), 2019, 2017, and 2015.
  - Military Communications Conference (MILCOM), 2021.
  - International Symposium on Emerging Information Security and Applications (EISA), 2021.
  - 20th Information Security Conference (ISC), 2017.
  - Workshop on Forming an Ecosystem Around Software Transformation (FEAST), 2017.
  - 14th International conference on Applied Cryptography and Network Security (ACNS), 2016.
  - International Workshop on Mobile Computing Security (MCS), 2015.
  - IEEE International Symposium on Security, Privacy and Anonymity in Internet of Things (SpaIoT), 2015 and 2014.
  - IEEE International Symposium on Security and Privacy in Internet of Things (SPIoT), 2013 and 2012.
  - IEEE International Workshop on Security and Privacy in Internet of Things (SPIoT), 2011.

  - 10th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), 2014.
  - 3rd International Conference on Certified Programs and Proofs (CPP), 2013.
  - Open64 Workshop at PLDI'12, 2012.
  - New Jersey Programming Languages Seminar, Program Chair and Host, October 2009.
  - Network and Information Security Symposium at CHINACOM, 2009 and 2010.
  - International Workshop on Distance Education Technologies (DET 2007).

- Session chairs
  - Session on Program Analysis in 2021 Network and Distributed System Security Symposium (NDSS).
  - Session on Systems Security in 2020 IEEE European Symposium on Security and Privacy (Euro S&P).
  - Session on Side Channels in 2020 Network and Distributed System Security Symposium (NDSS).
  - Session on Executing in Untrusted Environments in 2018 Usenix Security.
  - Session on Authentication in 2018 IEEE Symposium on Security and Privacy (Oakland).
  - Session on Systems Security and Authentication in 2017 IEEE Symposium on Security and Privacy (Oakland).
  - Session on Attacks Using a Little Leakage in 2016 ACM Conference on Computer and Communication Security (CCS).
  - Session on Understanding Android Apps in 2015 ACM Conference on Computer and Communication Security (CCS).
  - Session on Access Control in 2014 ACM Conference on Computer and Communication Security (CCS).
  - Session on Security and Optimization in 2013 International Conference on Object-Oriented Programming, Systems, Languages & Applications (OOPSLA).
  - Session on Trusted Computing Applications in 2012 ACM Workshop on Scalable Trusted Computing (ACM STC).

- External reviewer: ACM Transactions on Privacy and Security 2021; IEEE Transactions on Computers 2013, 2019; IEEE Security and Privacy Magazine 2018; Journal of Software special issue on Frontier of Programming Languages and Systems 2016; Science of Computer Programming 2016, 2018; IEEE Transactions on Dependable and Secure Computing (TDSC) 2015, 2018, 2022; Journal of Computer Security 2014; Applied Mathematical Sciences 2014; ACM Transactions on Computer Systems 2014; International Journal of Information Security 2014; IEEE Transactions on Parallel and Distributed Systems 2013; Higher-Order and Symbolic Computation 2012; Journal of Computer Science and Technology (JCST) 2012; PLDI 2011; POPL 2010; ESOP 2010; INFOCOM 2010; ACM Transactions on Programming Languages and Systems (TOPLAS) 2006, 2008, and 2010, 2021; Logical Methods in Computer Science (LMCS) 2010; IEEE Transactions on Software Engineering (TSE) 2007; International Journal of Foundations of Computer Science (IJFCS) 2006.

- Rapporteur, NSF, Convergence of Software Assurance Methodology and Trustworthy Semiconductor Design and Manufacture Workshop, Jan. 2013.
- NSF review panel, 2009, 2010, 2012, 2013, 2016, 2020.
- Organized a summer high-school teacher workshop on cyber security at Lehigh in 2012. The workshop helped teachers develop lesson plans for integration into their schools' technology curriculum.
- Panelist, "Understanding and Managing Cyber Crime: the Virtual Criminal", United Nations, DPI/NGO Briefing. Feb. 2011.
- Member of a team of computer scientists in a study of the software and hardware of the Sequoia AVC Advantage Voting Machine. This is in support of a NJ voting-machine lawsuit. Jul. 2008.

Service to School of EECS and the CSE Department of Penn State

- Departmental Promotion and Tenure Committee. Chair; 2020–2021. Member; 2017–2021.
- Departmental Awards Committee. Chair; 2020–2022.
- Departmental Colloquium. Chair; 2017–2020.
- Departmental Strategic Committee. Member; 2017–2023.
- Departmental Faculty Search Committee. Member; 2016–2017, 2019–2020, and 2021–2022.
- Departmental Graduate Committee. Member; 2020–2022.
- Departmental Curriculum Committee. Member; 2016–2017.
- School of EECS Promotion and Tenure Committee. Member; 2018–2022.
- School of EECS Steering Committee. Member; 2020–2022.
- Security and Programming Language (SEPL) Seminar Series. Organizer; 2016.

Service to Penn State

- CSRE (Center for Security Research and Education) Director Search Committee, Member; 2021–2022.
- Student Laptop Requirement Program Task Force, Member; 2021.
- Engineering faculty Council. College of Engineering, Member; 2016–2019.
- Institute for CyberScience Coordinating Committee, Member; 2016–2017, 2021–2022.
- Schreyer Honors College Application Faculty Reviewer, Member; 2016.
- Institute for CyberScience Seed Grant. Reviewer; 2016–2021.

Service to Lehigh University

- RCEAS college first-year advisor for engineering students, 2013–2015.
- University facility planning committee, member, 2014–2015.
- Departmental Professor of Practice (POP) search committee. Member, 2014.
- RCEAS college faculty search committee, Smart Grid Cluster, 2013.

- RCEAS college committee for Stout Dissertation Award, 2013.
- Departmental publicity and web committee. Chair, 2011–2015. Member, 2009–2010.
- Departmental computer facilities committee. Chair, 2014–2015. Member, 2011–2014.
- Departmental graduate admission committee. Member, 2013–2014, 2008–2009.
- Departmental colloquium committee. Chair, 2009–2010.
- Departmental curriculum committee, Member, 2008–2010.
- Computer science candidates day. 2012–2013.
- Departmental benchmarking committee, Member, 2008