Mohannad Alhanahnah – Research Statement

Vision. My perspective on software security goes beyond traditional, standalone analyses, acknowledging software's pervasive role in our lives and devices. This pervasive nature, seen in systems like IoT and Generative AI, complicates security measures and introduces new threats. My primary goal is to enhance software systems' safety for both society and developers. To achieve this vision, my research approach relies on three pillars: (1) developing appropriate abstraction to facilitate understanding the nature of interactions in emergent platforms, (2) defining threat models corresponding to these interactions, and (3) hardening software systems in isolation before exposing them by applying minimization and enforcing the classic principle of least-privilege. Through systems design and program analysis, my research seeks to improve security [5, 13], safety [10], privacy [1], and robustness [15] guarantees in these emergent platforms.

Impact. The practical significance of my research findings is confirmed by their acceptance in top-tier security and software venues such as USENIX Security, EuroS&P, ISSTA, NeurIPS, TIFS, and TSE. Beyond academic recognition, I strongly believe in the importance of technology transfer and commercialization as means to maximize societal benefits. Reflecting this belief, the technology emerging from my research has been approved for technical transfer, and a startup has been initiated with $500K in pre-seed funding. Following, I will delve into the details of my research and briefly outline my future research vision.

1 Research Themes

I will discuss recent research results that focus on two areas: (1) enforcing minimization and normalization concepts and (2) understanding the security and safety consequences of interactions in emergent platforms. The interested reader may refer to the bibliography for pointers to my work in cloud trust [3, 4, 6], formal verification [2], provenance analysis [8], secure API usage [12], and IoT malware detection [7].

1.1 Software and Data Minimization

- **LMCAS (EuroS&P'22 [5])**: a software debloating tool that minimizes applications footprint. It applies the concept of partial evaluation to generate specialized applications based on the supplied inputs by the user (specifies the required functionalities). LMCAS relies on the observation of the existence of a boundary between configuration logic and main logic in a given program. It then executes the program up to the boundary to capture the state of the program based on the supplied inputs. LMCAS finally applies a set of customized compiler optimizations to remove unused components and dependencies.

  Impact: LMCAS has been accepted for the Tech Transfer program funded by the Office of Naval Research (ONR). ONR also invited us to deliver LMCAS demo at the second Software Security School. Finally, a patent was filed for LMCAS technology.

- **minTAP (USENIX'22 [1])**: improves the privacy of TAPs by using language-based data minimization. It generates specialized minimizers that enforce the principle of least privilege by releasing only the necessary user data attributes to TAPs and fending off unrelated API access. The minimizers can be generated statically or dynamically. The integrity of the minimizers is protected by maintaining their digital signatures. We deploy minTAP on IFTTT, showing how to minimize trigger data before they are sent, thus boosting privacy while preserving the functionality.

  Impact. A patent was filed corresponding to minTAP technology.

1.2 Machine Learning Robustness

- **N&P (NeurIPS'22 [15])**: This work is inspired by my previous research [7] where Machine Learning (ML) was utilized to generate concise signatures for detecting IoT malware based on structural, statistical, and string features. We discovered that these ML detection systems can be vulnerable to semantic-preserving attacks designed to evade detection. To counter this, my collaborators and I developed a learning framework that employs input normalization to achieve guaranteed learning robustness against relational adversaries. These adversaries manipulate original test inputs via transformations specified by a logical relation, surpassing ℓ_p-norm bounded attacks as they can apply any sequence of transformations that preserve input semantics. Our approach, called N&P, remedies the challenges of adversarial training against relational adversaries and can be harmoniously integrated with adversarial training to attain the optimal balance between robustness and accuracy.

  Impact. This work has been designated as an ”Oral” at NeurIPS 2022.

2 Future Research Agenda

I would like to direct, but not limit, my future research efforts to the following research problems:

- **Software Supply Chain Security:** The rapid increase in the use of open-source software and third-party libraries has led to growing concerns about potential vulnerabilities and malicious activities, as evidenced by events like the SolarWinds attack and Log4j vulnerability. In response, the US government has mandated the provision of a Software Bill of Materials (SBOM) to identify such libraries, restricting government agencies to work only with software vendors providing SBOMs. Recognizing that about 90%[1] of software developments leverage open-source code, and its emerging status as a public good, my ultimate aim is to enhance the safety of open-source software usage.

  The modern software supply chain, however, includes additional dependencies beyond the applications' code. These dependencies extend to areas like Continuous Integration (CI) and Continuous Development (CD), which involve workflow automation and containerization. Therefore, it is imperative to consider these aspects in any comprehensive solution, calling for the development of a holistic dependency tree that traces the chain of dependencies across all necessary

[1] https://www.infoworld.com/article/3245308/rethinking-your-open-source-use-policy.html

and unnecessary components. I plan to leverage my previous work in software debloating [5, 9] and ongoing collaborations on container debloating [16, 17] to construct this tree, which will prove invaluable for various types of analysis, including threat and security evaluations.

- **LLM Applications Safety:** A number of platforms, such as Langroid [2] and LangChain [3], have been suggested to improve the functionalities of large language models (LLMs) by harnessing the strength of multi-agent systems. These systems are renowned for resolving intricate problem-solving scenarios and incorporating concurrent computation models, such as the Actor Model. This strategy breaks down the problem into smaller, more manageable tasks, with each being tackled by a specific LLM agent. These agents collaborate autonomously to address complex tasks with greater efficiency and effectiveness. Based on my experience while developing Langroid, although this method offers remarkable benefits, the interaction among these agents can yield unpredictable outcomes, which could be detrimental in certain situations.
  Drawing on my prior experience in investigating interactions on Android [13, 14] and IoT platforms [10, 11, 2], this project proposes to identify the safety and security challenges prevalent in the LLM application ecosystem. The initial phase of the research will be dedicated to pinpointing the necessary safety and security characteristics that should be sustained in these systems. Subsequent to this identification process, the proposal aims to devise an enforcement mechanism that mandates the incorporation of these properties, ensuring the effective and secure operation of the LLMs in diverse applications.

- **Green Coding:** As we increasingly digitize sectors and even entire nations, our global society faces the challenge of escalating data usage. To counteract this, large corporations such as Google and Amazon Web Services have initiated environmental sustainability programs aimed at controlling carbon emissions, setting ambitious targets to power their operations with carbon-free and renewable energy by 2030 and 2025 respectively. One approach to achieving these goals is through the adoption of Green Coding practices. These practices involve creating energy-efficient algorithms, a strategy that requires software engineers to write code that minimizes energy consumption.

  This concept encourages, among other principles, the removal of unused software features, a move that improves energy efficiency and ease of maintenance. It's a necessary step as around 90% of software developments utilize open-source code, which often includes redundant sections that consume excess processing power and emit unnecessary carbon. My research in this domain envisions sustainability as (physical and cyber) security problem. Therefore, I will explore the use of software debloating and compiler optimization techniques to enhance the efficacy of such in reducing hardware requirements and, subsequently, energy consumption and carbon emissions. To realize this, I intend to apply for research grants from programs committed to environmental sustainability and addressing climate change issues.

[2] https://github.com/langroid/langroid

[3] https://github.com/hwchase17/langchain

References

[1] Y. Chen, **M. Alhanahnah**, A. Sabelfeld, R. Chatterjee, and E. Fernandes. Practical Data Access Minimization in Trigger-Action Platforms. In *USENIX Security Symposium*, 2022.

[2] C. Stevens, **M. Alhanahnah**, Q. Yan, and H. Bagheri. Comparing Formal Models of IoT App Coordination Analysis. In *SEAD 2020.*

[3] **M. Alhanahnah**, P. Bertok, and Z. Tari. Trusting Cloud Service Providers: Trust Phases and a Taxonomy of Trust Factors. *IEEE Cloud Computing*, 4(1):44–54, Jan 2017.

[4] **M. Alhanahnah**, P. Bertok, Z. Tari, and S. Alouneh. Context-Aware Multifaceted Trust Framework For Evaluating Trustworthiness of Cloud Providers. *Future Generation Computer Systems*, 79:488 – 499, 2018.

[5] **M. Alhanahnah**, R. Jain, V. Rastogi, S. Jha, and T. Reps. Lightweight, multi-stage, compiler-assisted application specialization. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, pages 251–269. IEEE, 2022.

[6] **M. Alhanahnah**, A. Jhumka, and S. Alouneh. A Multidimension Taxonomy of Insider Threats in Cloud Computing. *The Computer Journal*, 59(11):1612–1622, 11 2016.

[7] **M. Alhanahnah**, Q. Lin, Q. Yan, N. Zhang, and Z. Chen. Efficient Signature Generation for Classifying Cross-Architecture IoT Malware. In *2018 IEEE Conference on Communications and Network Security (CNS)*, pages 1–9, May 2018.

[8] **M. Alhanahnah**, S. Ma, A. Gehani, G. Ciocarlie, V. Yegneswaran, S. Jha, and X. Zhang. autoMPI: Automated Multiple Perspective Attack Investigation with Semantics Aware Execution Partitioning. *IEEE Transactions on Software Engineering*, 2022.

[9] **M. Alhanahnah**, P. Schubert, N. Gupta, T. Reps, S. Jha, and E. Bodden. Slash: Static configuration-logic identification. **Under submission**.

[10] **M. Alhanahnah***, C. Stevens*, and H. Bagheri. Scalable Analysis of Interaction Threats in IoT Systems. In *ISSTA 2020*, 2020 (*Equal contribution) (**ACM SIGSOFT Distinguished Paper Award**).

[11] **M. Alhanahnah**, C. Stevens, B. Chen, Q. Yan, and H. Bagheri. IoTCOM: Dissecting Interaction Threats in IoT Systems. *IEEE Transactions on Software Engineering*, 2022.

[12] **M. Alhanahnah** and Q. Yan. Towards best secure coding practice for implementing SSL/TLS. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–6, April 2018.

[13] **M. Alhanahnah**, Q. Yan, H. Bagheri, H. Zhou, Y. Tsutano, W. Srisa-an, and X. Luo. Detecting Vulnerable Android Inter-App Communication in Dynamically Loaded Code. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pages 550–558, April 2019.

[14] **M. Alhanahnah**, Q. Yan, H. Bagheri, H. Zhou, Y. Tsutano, W. Srisa-an, and X. Luo. Dina: Detecting hidden android inter-app communication in dynamic loaded code. *IEEE Transactions on Information Forensics and Security*, 15:2782–2797, 2020.

[15] Y. Wang, **M. Alhanahnah**, X. Meng, K. Wang, M. Christodorescu, and S. Jha. Robust learning against relational adversaries. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[16] H. Zhang, F. Abdulqadir Ahmed, D. Fatih, A. Kitessa, **M. Alhanahnah**, P. Leitner, and A. Ali-Eldin. Machine learning containers are bloated and vulnerable. *arXiv preprint arXiv:2212.09437*, 2022.

[17] H. Zhang, **M. Alhanahnah**, and A. Ali-Eldin. Blafs: A bloat aware file system. **Under submission**.