Securing Consumer IoT Swarms Using Graph Transformers and SRAM for Firmware Attestation

Varun Kohli^a, Bhavya Kohli^b, Muhammad Naveed Aman^c, *Senior Member, IEEE*, Biplab Sikdar^a, *Senior Member, IEEE*

^{a} Department of Electrical and Computer Engineering, National University of Singapore, Singapore.
^{b} Center for Machine Intelligence and Data Science, Indian Institute of Technology Bombay, India.
^{c} School of Computing, University of Nebraska-Lincoln, USA.

***Abstract*—Consumer Internet of Things (IoT) networks have gained widespread popularity due to their convenience, automation, and security provisions in personal and home environments. Ubiquitous resource-constrained devices, however, are plagued with security issues that often arise from firmware-related issues and their propagated effects. While various studies on firmware attestation are available, they require firmware copies, specific hardware, and complex computation on the IoT device. This paper presents a study on the application of Graph Transformer Networks (GTN) in verifying the firmware integrity of consumer IoT swarms using SRAM as an attestation feature. The proposed method achieves an overall 0.99 accuracy on authentic samples from development and physical twin networks, 0.99 on malware, and 0.97 on propagated misbehavior at a $\sim 10^{-4}$ second inference latency on a laptop CPU.**

***Index Terms*—Internet of Things (IoT), Swarm Attestation, Firmware Attestation, Malware Detection, Anomaly Detection, Graph Neural Network (GNN), Security**

I. Introduction

Consumer IoT has emerged as a popular field that includes smart home appliances, wearables, security systems, transportation, healthcare, and utility management. Billions of resource-constrained devices, performing sensing, processing and actuation tasks, have been deployed in collaborative networks worldwide to enhance the quality of life of end-users [1]. However, recent studies highlight that threats such as malware injection attacks are a leading cause of security issues in IoT networks [2]. Firmware attestation has thus emerged as a distinct field of research to address this problem and build the trust users have in their smart IoT applications.

Related works: Various software [3], hardware [4], and hybrid [5] approaches to remote attestation have been proposed for single-node and swarm attestation. Software-based methods typically involve multiple iterations of checksums over the IoT device's program memory and need a copy of the firmware, which may not be available due to the manufacturers' Intellectual Property (IP) rights. Hardware-based methods assume the availability of Trusted Platform Modules (TPM), Trusted Execution Environments (TEE), and other specialized hardware unavailable on most IoT devices. And lastly, hybrid approaches typically overlook roving malware. Aside from [6], no other study has explored using an IoT device's SRAM for attestation. SRAM is much smaller than flash memory and faster to traverse. It captures runtime information and can indicate roving malware. It also eliminates the need for firmware copies. Lastly, the SRAM contents of connected nodes are related in collaborative swarms due to the exchange of information between the IoT devices. The SRAM can, therefore, also help us indicate propagated effects when combined with GNN architectures such as Graph Transformer Networks (GTN) [7], which is among the state-of-the-art for GNNs.

Contributions: This paper presents a preliminary study using lightweight GTN for firmware attestation in consumer IoT swarms using the SRAM contents of IoT devices. It makes a minimal assumption regarding the hardware on IoT devices. The proposed method is tested on a real-world SRAM swarm attestation dataset [8] and its latency and memory evaluation, highlighting practicality in real-world use cases.

II. Network and Threat Model

The IoT network model considered in this paper comprises a verifier (ID_V) and a swarm (ID_S) of IoT device provers ($\{\mathrm{N}_j : j = 1,..,n\}$, where n is the number of IoT devices in ID_S). ID_V is a trusted and secure device that verifies the integrity of firmware in ID_S. It broadcasts attestation requests (C) to all N_j in ID_S and evaluates their asynchronous SRAM response (r_j) collected into the swarm response set, $\mathcal{R} = \{r_j : j \in [1,n]\}$, using a GTN ($\mathcal{G}$). We assume that the verifier knows the details of the firmware loaded on each N_j in ID_S and their corresponding expected response lengths (l_j). It also has sufficient computing power to run $\mathcal{G}$. Each prover, N_j, is a resource-constrained device that responds with its SRAM data section contents (r_j, refer to [9] for sections of the SRAM) upon receiving an attestation request. We assume the adversary can send malicious firmware updates to the IoT devices, which may have node-level and downstream effects in the swarm.

III. Proposed Technique

The proposed attestation technique consists of two phases.

This research was supported in part by the National Research Foundation, Singapore and Infocomm Media Development Authority under its Future Communications Research Development Programme, under grant FCP-NUSRG-2022- 019. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Infocomm Media Development Authority.

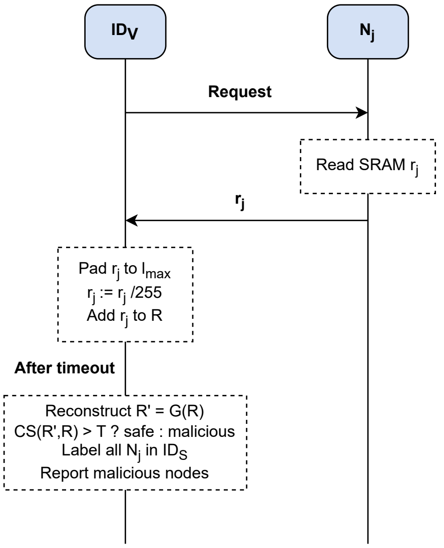

Fig. 1: An overview of the attestation procedure.

A. Training Phase

During the training phase, ID_V securely samples a sufficient number of swarm responses $\mathcal{R}$ from ID_S, creates a training distribution X and a uniformly perturbed distribution $\tilde{X}$ and trains $\mathcal{G}$ to reconstruct the training distribution X from $\tilde{X}$. Upon completion of the optimization process, ID_V uses X to select Cosine Similarity (CS)-based detection thresholds $t_j = f \cdot \min_{x \in X_j} CS(\hat{x}, x)$ where f (=0.99, in this paper) is the threshold scaling factor, X_j is the SRAM response set of N$_j$.

B. Attestation Phase

An overview of the attestation phase is provided in Figure 1. ID_V selects a swarm ID_S and loads the stored parameter set $P = \{\mathcal{G}, \mathcal{T}\}$. It then requests each node in the swarm for its corresponding memory contents r_j, scales the received memory traces by a factor 255 to bring them in a [0,1] range, and pads them with zeros to the maximum response length l_max. After the timeout, ID_V creates the swarm response $\mathcal{R}$ and reconstructs $\mathcal{R}' = \mathcal{G}(\mathcal{R})$. It then evaluates $CS(\hat{r_j}, r_j)\forall$N$_j$ and the CS scores above respective $t_j \in \mathcal{T}$ are labeled safe (0), and otherwise unsafe (1).

IV. Results

The swarms and the number of bytes shared between their nodes are shown in Figure 2. The GTN for each swarm follows an encoder-decoder structure consisting of two PyG TransformerConv layers, activated by the Rectified Linear Unit (ReLU) activation, that reduce the input trace to a latent dimension of 64 and reconstruct it, respectively. The models are optimized using MSE loss and the Adam optimizer at a learning rate of 0.005 for 100 epochs for 800 training samples. Table I compiles the overall accuracy for different behavior types included in the SRAM swarm attestation dataset [8]. As the table shows, the proposed method has a 0.99 accuracy in detecting normal firmware, 0.99 on malware, and 0.97 on propagated anomalies. Each model occupies 1.3 MB of verifier memory and has an inference latency of 10^{-4} seconds on an Intel i7 processor laptop with 16 GB DRAM. In addition, we simulated a GTN of up to 50 nodes (smart homes typically have 20-50 devices), and the model occupied memory of the order 10^2 MB.

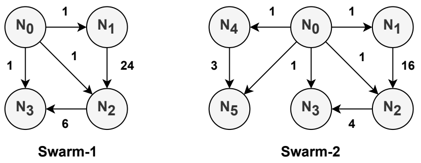

Fig. 2: Directed graph structure and number of bytes exchanged between nodes in the swarms covered by [8].

TABLE I: Behavior types, their corresponding scenarios, and overall detection accuracy.

S.No.	Behavior	Scenarios	Accuracy
1	Normal state	$\forall$D$_i$ P$_i$	0.99
2	Physical twin	$\forall$P$_i$	0.99
3	Malware	$\forall$AN$_j$	0.99
4	Propagated anomaly	AN$_{1,2,4,12,13}$	0.97
5	Tampered data generation	AN$_1$	1.00
6	Tampered processing	AN$_2$	0.99
7	Tampered actuation	AN$_{3,5}$	1.00
8	Tampered functions	AN$_{0,4}$	1.00
9	Added peripherals	AN$_5$	1.00

V. Conclusion

This paper presented a preliminary study on the application of GTN in verifying the integrity of consumer IoT swarms using SRAM as a feature for firmware attestation. The proposed GTN had an overall 0.99 accuracy for all behavior types, is lightweight (occupies 1.3 MB of verifier memory), and had an inference latency of 10^{-4} seconds on a laptop CPU.

References

[1] S. A. Baho and J. Abawajy, “Analysis of consumer iot device vulnerability quantification frameworks,” *Electronics*, vol. 12, no. 5, p. 1176, 2023.

[2] T. Alladi, V. Chamola, B. Sikdar, and K.-K. R. Choo, “Consumer iot: Security vulnerability case studies and solutions,” *IEEE Consumer Electronics Magazine*, vol. 9, no. 2, pp. 17–25, 2020.

[3] S. F. J. J. Ankergård, E. Dushku, and N. Dragoni, “State-of-the-art software-based remote attestation: Opportunities and open issues for internet of things,” *Sensors*, vol. 21, no. 5, p. 1598, 2021.

[4] I. Sfyrakis and T. Gross, “A survey on hardware approaches for remote attestation in network infrastructures,” *arXiv preprint arXiv:2005.12453*, 2020.

[5] W. A. Johnson, S. Ghafoor, and S. Prowell, “A taxonomy and review of remote attestation schemes in embedded systems,” *IEEE Access*, vol. 9, pp. 142 390–142 410, 2021.

[6] M. N. Aman, H. Basheer, J. W. Wong, J. Xu, H. W. Lim, and B. Sikdar, “Machine-learning-based attestation for the internet of things using memory traces,” *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 20 431–20 443, 2022.

[7] Y. Shi, Z. Huang, S. Feng, H. Zhong, W. Wang, and Y. Sun, “Masked label prediction: Unified message passing model for semi-supervised classification,” *arXiv preprint arXiv:2009.03509*, 2020.

[8] V. Kohli, B. Kohli, M. Naveed Aman, and B. Sikdar, “Iot swarm sram dataset for firmware attestation,” 2024. [Online]. Available: https://dx.doi.org/10.21227/gmee-vj41

[9] V. Kohli, M. N. Aman, and B. Sikdar, “An intelligent fingerprinting technique for low-power embedded iot devices,” *IEEE Transactions on Artificial Intelligence*, 2024.