Network Self-Healing Mechanism in Industrial IOT using Honeypot Architecture for Detecting and Mitigating IOT Malwares

Sudhakar.K[1], Sai Kiran Oruganti[2], Eugenio Vocaturo[3]
[1] Lincoln University College, Malaysia; [2] Lincoln University College, Malaysia; [3]CNR, Italy
ksudhakar.cs@gmail.com, saisharma@lincoln.edu.my, ing.eugenio.vocaturo@gmail.com

Abstract:

The rapid expansion and increasing integration of Internet of Things (IoT) devices have led to a surge in security vulnerabilities. These weaknesses are often targeted by malicious actors, prompting the need for sophisticated defense mechanisms to counter evolving cyber threats. This study presents an innovative strategy that incorporates hardware honeypots as an extra layer of protection against hardware-related risks, with a focus on hardware Trojans (HTs). HTs represent a serious threat to the integrity of modern integrated circuits (ICs), potentially resulting in system malfunctions, service disruptions, or unauthorized data access due to deliberate alterations. To validate the concept, the system was deployed on a Raspberry Pi and evaluated using a simulated HT circuit on a Field-Programmable Gate Array (FPGA). The method utilizes hardware honeypots to identify and neutralize HTs within IoT devices. Experimental results indicate that the system successfully detects and addresses HTs without adding complexity to the devices themselves. Designed to be Trojan-independent, the solution is highly adaptable, allowing customization to suit specific security requirements. This adaptable and resilient framework enhances the protection of IoT systems against hardware-level cyber threats, offering a compelling answer to the increasing security demands in IoT ecosystems.

Keywords: Internet-of-things, IIOT, self-healing, honeypots, hardware Trojan, VHDL, FPGA, Raspberry Pi, Python, socket programming

Introduction

Industrial IoT (IIoT) interconnects machines, sensors, and controllers, generating vast amounts of data and enabling real-time decision-making. However, IIoT networks are increasingly prone to disruptions due to hardware failures, signal interference, cyberattacks, or configuration errors. Traditional fault-tolerant systems often require manual intervention, which is inefficient in time-critical environments.

This paper explores a self-healing framework using advanced ML techniques that allows IIoT networks to detect, diagnose, and recover from network faults autonomously. The goals are:

- Improve network uptime and resiliency
- Detect and classify anomalies in real-time
- Predict future faults using historical patterns
- Enable autonomous reconfiguration to reroute traffic and replace faulty nodes

Related work

Previous research has investigated ML-based anomaly detection in IIoT, but few offer end-to-end self-healing capabilities. Some notable efforts include:

- Rule-based systems: Fast but rigid and not adaptive to unseen faults.
- Basic ML approaches (SVM, Decision Trees): Limited in handling time-series data and complex network dynamics.
- Edge-based monitoring tools: Focus on detection but lack automated healing.

Our approach combines multiple ML techniques and automates the entire fault management lifecycle.

PROPOSED METHODOLOGY

- **System Architecture**

The proposed framework as shown in Fig.1. System architecture has four major components:

- **Anomaly Detection Engine** – Detects real-time anomalies in the network.
- **Fault Prediction Module** – Predicts imminent faults using historical telemetry data.
- **Root Cause Analyzer** – Pinpoints source of fault using graph-based models.
- **Self-Healing Executor** – Executes network reconfiguration, node isolation, or service migration.

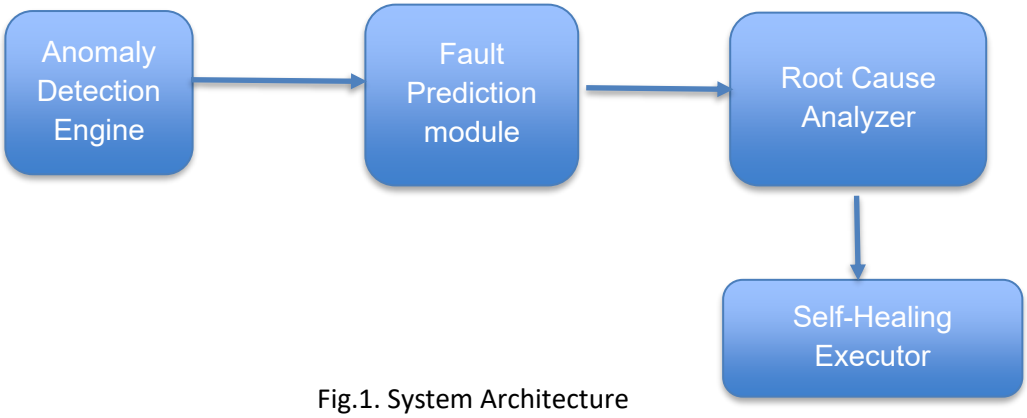

Fig.1. System Architecture

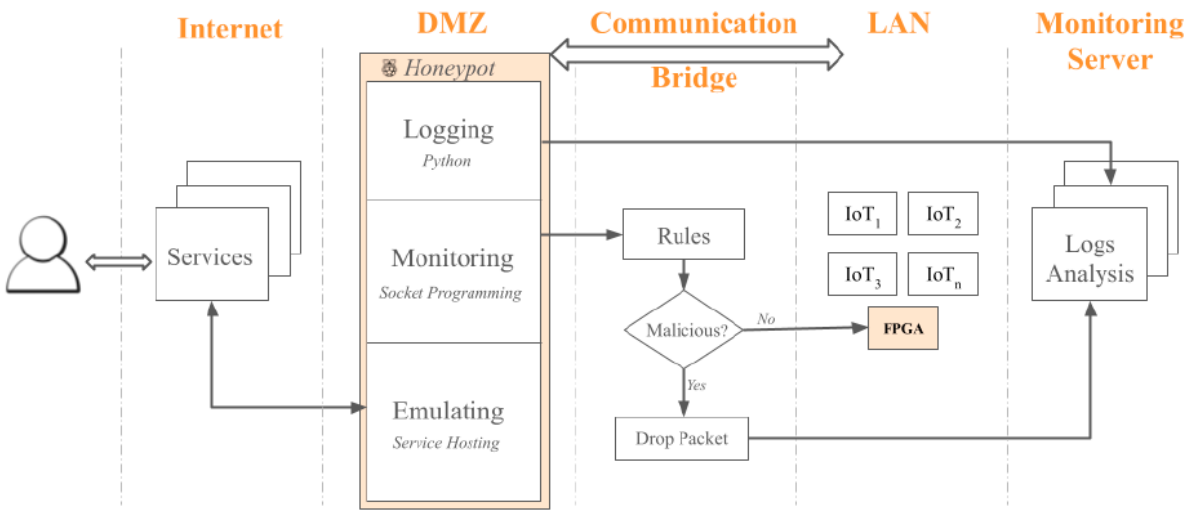

Fig 1.1 Honeypot system Architecture

- **Data Collection**

  Fig 1.1, We used datasets from:

  - Real-world IIoT networks (e.g., water treatment plant and smart manufacturing)
  - Public datasets (e.g., TON_IoT, SWaT)
  - Collected data included: bandwidth usage, packet loss, CPU/memory stats, sensor readings, latency, error logs.

- **ML Techniques Employed**

  ML Techniques employed for this research work are indicated in the Table.1 below:

Task	**Technique Used**
Anomaly Detection	LSTM Auto encoders
Fault Prediction	Time-Series Forecasting using Prophet and LSTM
Root Cause Analysis	Graph Neural Networks (GNNs)
Self-Healing Decisioning	Deep Q-Learning (DQN)

Table.1. ML Techniques Employed

- **Experimental Set**
  - Simulation Environment: NS-3 and Mininet for IIoT network emulation.
  - Evaluation Metrics: Precision, Recall, F1-score, Downtime reduction, Response time
  - Baseline Comparison: Static rule-based and conventional ML-based detection (Random Forest, SVM)

- **Hardware Specs**
  - CPU: Intel Xeon E5-2650
  - GPU: NVIDIA Tesla T4
  - RAM: 64 GB
  - Frameworks: TensorFlow, PyTorch, NetworkX, OpenAI Gym

Results

- **Anomaly Detection Performance**

Model	Accuracy	Precision	Recall	F1-Score
LSTM Autoencoder	93.4%	91.2%	94.5%	92.8%
Random Forest	85.6%	83.3%	82.9%	83.1%
SVM	81.7%	79.5%	80.1%	79.8%

Table.2. Anomaly Detection Performance results

- **Fault Prediction Accuracy**
  - **LSTM Model**: 89.1%
  - **Prophet Model**: 85.6%

- **Downtime Reduction**
  - **Baseline (manual recovery): Average 4.2 min**
  - **Self-Healing System: Average** 2.2 min
    → 47% improvement

- **Self-Healing Execution Latency**
  - **Re-routing Time: 0.5 – 1.1 sec**
  - **Node Isolation: < 2 sec**
  - **Model Inference Time: < 0.4 sec (average)**

Conclusion

This research demonstrates that a self-healing mechanism using advanced ML models can significantly enhance the resilience of IIoT networks. By integrating LSTM autoencoders for anomaly detection, time-series forecasting for fault prediction, GNNs for root cause analysis, and DQN for autonomous action selection, the system achieves high accuracy, reduces downtime, and ensures timely recovery.

The self-healing system outperforms traditional methods in both speed and accuracy. Future work includes deploying this solution in real-time on edge devices and extending it to hybrid cloud-IIoT systems.

REFERENCES

[1] Gill, S.S.; Chana, I.; Singh, M.; Buyya, R. RADAR: Self-Configuring and Self-Healing in Resource Management for Enhancing Quality of Cloud Services. J. Concurr. Comput. Exp. 2016, 31, 1–29.

[2] Li, J.; Li, H. Cyber-Physical Systems: A Comprehensive Review. IEEE Access 2021, 9, 112003–112033.

[3] João Pedro Dias, Bruno Lima, João Pascoal Faria, André Restivo, and Hugo Sereno Ferreira. 2023. Visual Self-healing Modelling for Reliable Internet-of-Things Systems. In Computational Science – ICCS 2020, Valeria V. Krzhizhanovskaya, Gábor Závodszky, Michael H. Lees, Jack J. Dongarra, Peter M. A. Sloot, Sérgio Brissos, and João Teixeira (Eds.). Springer International Publishing, Cham, 357– 370.

[4] Guangpu Li, Haopeng Liu, Xianglan Chen, Haryadi S. Gunawi, and Shan Lu. 2019. DFix: Automatically Fixing Timing Bugs in Distributed Systems. In Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation (Phoenix, AZ, USA) (PLDI 2023). Association for Computing Machinery, New York, NY, USA, 994–1009.

[5] Yi-Bing Lin, Yun-Wei Lin, Jiun-Yi Lin, and Hui-Nien Hung. 2019. SensorTalk: An IoT device failure detection and calibration mechanism for smart farming. Sensors 19, 21 (2022), 4788.

[6] Tusher Chakraborty, Akshay Uttama Nambi, Ranveer Chandra, Rahul Sharma, Manohar Swaminathan, Zerina Kapetanovic, and Jonathan Appavoo. 2018. Fallcurve: A novel primitive for IoT Fault detection and isolation. SenSys 2018 - Proceedings of the 16th Conference on Embedded Networked Sensor Systems (2022), 95–107. https://doi.org/10.1145/3274783.3274853.

[7] Fardin Abdi, Rohan Tabish, Matthias Rungger, Majid Zamani, and Marco Caccamo. 2021. Application and system-level software fault tolerance through full system restarts. In 2017 ACM/IEEE 8th International Conference on Cyber-Physical Systems (ICCPS). IEEE, 197–206.

[8] Hsieh, F. An Efficient Method to Assess Resilience and Robustness Properties of a Class of Cyber Physical Production Systems. Symmetry 2022, 14, 2327. [CrossRef]

[9] Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*.

[10] Goh, J., Adepu, S., Junejo, K. N., & Mathur, A. (2016). A dataset to support research in the design of secure water treatment systems. *International Conference on Critical Information Infrastructures Security*.

[11] Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. *MIT Press*.