An Integrated Testbed for Trojans in Printed Circuit Boards with Fuzzing Capabilities

Prashanth Krishnamurthy, Hammond Pearce, Virinchi Roy Surabhi, Joshua Trujillo, Ramesh Karri, Farshad Khorrami

***Abstract*—This paper showcases an all-in-one testing environment that combines Trojan detection and fuzzing capabilities for printed circuit boards using the OpenPLC "NYU Trojan Edition" and a dedicated Trojan detection framework. The demo system is self-contained and equipped with two OpenPLC-based boards (one with a Trojan and one without), and automated tools for inserting the Trojan and collecting side-channel data. We developed a graphical user interface for interactive Trojan selection, data visualization, and anomaly detection analysis.**

***Index Terms*—Trojan detection, anomaly detection, PCB, timing loopback, golden-free, machine learning**

I. Introduction

Global nature of the Printed Circuit Board (PCB) supply chain poses security risks. Threats posed by untrusted third parties by infiltrating hardware/software supply chains include hardware, firmware, and software implants/Trojans. Hence, studying the possible impacts of Trojans on embedded PCBs in cyber-physical system (CPS) applications and Trojan detection/mitigation methods is crucial. This project builds a demo with a user GUI atop a flexible and reconfigurable Trojan PCB testbed (OpenPLC "NYU Trojan Edition (TE)") in [1].

Embedded PCB testbeds and more generally cyber-physical system (CPS) testbeds are highly useful for developing and testing algorithmic methodologies for various tasks such as CPS control, monitoring and state estimation, and anomaly detection. Hence, testbed development has been studied extensively in the literature in the context of various CPS domains. Electrical Power and Intelligent Control (EPIC) [2] testbed simulates a a small smart grid. The authors discuss attacks such as false data injections, malware, power outages, and physical damage. HAI Testbed (HIL-based Augmented ICS) in [3], [4] is an interconnection of three Industrial Control Systems orchestrated by a Hardware-in-the-Loop (HIL) that models an overall power generation system. BU-Testbed [5] for power generation systems addresses cyber-physical attacks including network attacks such as Machine-In-The-Middle (MITM), DNS poisoning, network congestion and delay, malicious software/firmware in controllers, sensors, and HMI. Korkmaz et al. [6] offered a testbed to study susceptibility to time delay attacks. SWaT models a six-stage water treatment facility [7] controlled by a Programmable Logic Controller (PLC) and studies MITM, eavesdropping, and packet modification attacks. Zhang et al. [8] models a two-loop nuclear power system and considers MITM and DoS attacks. LegoSCADA [9] considers MITM attacks (e.g., replay, injection) based on a testbed including a Lego Mindstorms EV3 that emulates a PLC on a car and a Raspberry Pi that emulates an RTU linked to the vehicle. In [10], LICSTER based on an OpenPLC, a web-based HMI, and a SCADA system communicating via Modbus/TCP was applied to study passive/active sniffing, DoS, and MITM.

In comparison with prior studies outlined above, the novel testbed described in this paper is architected specifically to facilitate insertion of Trojans on a CPS system. The overview and motivation of the testbed are discussed in Section II. The OpenPLC-based PCB developed for study of Trojan insertion and detection is described in Section III. The detailed architecture of the Trojan demo platform and associated Graphical User Interface (GUI) are presented in Section IV.

II. Overview and Motivation

To study embedded Trojans, we developed a flexible and reconfigurable OpenPLC "NYU Trojan Edition" PCB (Figure 1) and integrated testbed (Figure 2). We developed several types of Trojans that could be instantiated on this PCB system. The testbed is based on the open-source OpenPLC platform. The Trojan edition of the OpenPLC developed for the testbed provides Trojan insertion/configuration mechanisms including re-configurable wiring options using jumpers, injection of Trojans via a separate microcontroller representing a MITM Trojan, injection of software Trojans via the OpenPLC's processor, and stealthy communication back-channels between Trojan components and OpenPLC processor. The motivating considerations in the development of the OpenPLC Trojan Edition PCB and the integrated testbed are outlined below.

- The Trojan demo testbed is self-contained with two OpenPLC boards (one in non-Trojan or "clean" configuration and one in Trojaned configuration), automated mechanisms for Trojan deployment and side channel data collection driven by a single-board computer integrated

P. Krishnamurthy, H. Pearce, V. R. Surabhi, R. Karri, and F. Khorrami are with the Department of Electrical and Computer Engineering, NYU Tandon School of Engineering, Brooklyn, NY, 11201 USA. e-mails: {prashanth.krishnamurthy, hammond.pearce, virinchi.roy, rkarri, khorrami}@nyu.edu. J. Trujillo is with the Department of Energy's Kansas City National Security Campus (email: jtrujillo@kcnsc.doe.gov). This work was supported in part by DoE Kansas City. Honeywell Federal Manufacturing & Technologies, LLC operates the Kansas City National Security Campus for the United States Department of Energy / National Nuclear Security Administration under Contract Number DE-NA0002839.

NSC-614-5396 dated 06/2023 Unclassified Unlimited Release

979-8-3503-4135-5/23/$31.00 ©2023 IEEE

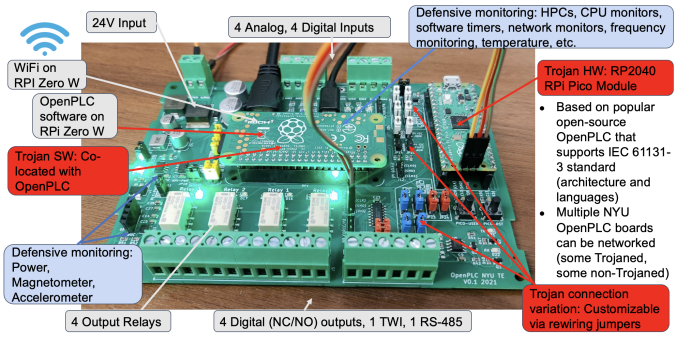

Fig. 1. OpenPLC "NYU Trojan Edition (TE)."

into the demo platform, and a graphical user interface (GUI) for interactive Trojan selection, data visualization, and anomaly detection analysis. Using Trojan insertion mechanisms built into the testbed, Trojans are integrated for "single-click" availability in the demo platform.

- The Trojan detection framework uses digital and analog side channels. The side channels include digital Hardware Performance Counter (HPCs) measurements of code execution, analog measurements such as power, temperature, and magnetic signatures, and timing measurements from communication and analog/digital input/output (I/O). These PCB side channels enable baseline-relative and golden-free Trojan detection analyses. While baseline-relative methods flag deviations from *a priori* known-good side channel signals, golden-free detection disambiguates side channel measurements from Trojan vs. Trojan-free PCBs using design-time information, datasheets, and components absent known-good systems.
- The Trojan detection framework integrated into the testbed is based on monitoring side channels under defender-controlled software-driven excitations and input/output connections (loopbacks) and fusing the collected multi-modal signals using spatio-temporal feature extraction and machine learning (ML) methodologies to flag anomalies by validating probabilistic consistency against expected behaviors. Instances of this approach include comparing delay measurements of I/O loopbacks against design-based baselines, magnetometer-aided monitoring of behaviors of hardware relays under defender-chosen excitations, and observation of digital side channels by applying defender-chosen test codes.

III. OpenPLC NYU TE PCB for Trojan Studies

We use the open-source OpenPLC to develop the Trojan Edition PCB [1] as a representative of real-world embedded systems. The OpenPLC's features for industrial use include a resilient 12-48V power supply input, four relay-isolated outputs to control standard industrial equipment, four opto-isolated 24V digital inputs for equipment sensing, four high-impedance analog inputs, a RS-485 bus for Modbus communication, and an open PLC software stack that supports IEC 61131-3 languages. A Raspberry Pi is the main processor.

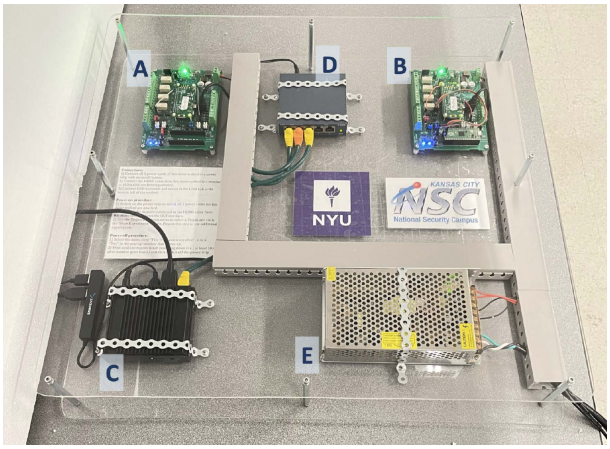

Fig. 2. NYU OpenPLC Trojan Demo Platform. A, B: NYU OpenPLC Trojan Edition boards (A: clean, B: Trojaned); C: single-board computer for sidechannel analysis and GUI; D: Ethernet switch; E: Power supply.

A. Trojan Insertion/Configuration in OpenPLC NYU TE

The OpenPLC NYU TE can instantiate many embedded Trojans on the testbed. The ability of the PCB to insert arbitrary MITM Trojans using a second "Trojan" socketed Raspberry Pi Pico module. This microcontroller can interfere with PLC operation, including I/O and buses. "Jumpers" can support rewiring and rerouting via the Trojan microcontroller. We chose a socketed module for the Trojan microcontroller component for two main reasons. First, the Pico is inexpensive and has an on-PCB RP2040 dual-core ARM-M0+ running at 133 MHz, with 264KB of SRAM, and 2MB of Flash memory. It can interface via USB 1.1. Secondly, as a detachable module, it can demonstrate "addition" and "removal" of Trojans from the design. In addition to firmware Trojans added via the Pico and the wiring changes to reroute communication signals, software Trojans can be instantiated on the OpenPLC's processor (Raspberry Pi Zero).

B. Side Channels for Defensive Monitoring

The testbed enables real-time monitoring of the PLC's performance by measuring side channel signals. These comprise digital side channels from the OpenPLC processor, analog signals like power and magnetic data from on-PCB sensors, and timing information from I/O channels obtained through test codes on the OpenPLC's processor. Current sensors provide readings of the current draws of the overall PCB, Raspberry Pi, and separate microcontroller. The IMU is included at a location near the relays to measure vibrations and magnetic field fluctuations during relay actuation.

C. Trojan Injection and Configuration on OpenPLC NYU TE

The OpenPLC NYU TE testbed is designed such that a wide range of Trojans can be introduced via multiple insertion/configuration mechanisms including:

- **Static wiring changes**: Reconfigurable jumpers on the PCB can re-route signals to select/de-select MITM Trojans. The OpenPLC NYU TE enables reconfigurability of the GPIO, RS-485/UART, and I2C wiring. Besides wiring

changes to introduce MITM Trojans, Trojans can include hardware changes taking advantage of the PCB's modular design (as the Pi and Pico are removable) and modifications of passive components (e.g., added capacitors to introduce faults) to change electrical characteristics.
- **Hardware Trojan emulation**: The Pico can be reprogrammed to act with static (constant) or dynamic (triggered) malicious behavior. When used in conjunction with the reconfigurable jumpers, the Trojans can pick the signals that they read and edit. Trojans can change signal values, reroute or snoop on signals, and insert delays.
- **Software Trojans** introduced on the Pi may leak information, degrade performance and interfere with operation.
- **Back-channel communication** can be used by Trojans. For such "multi-component" Trojans, the PLC includes back-channel I2C and UART communication. With an I2C back-channel, special undocumented addresses can be added to enable and disable (trigger) Trojan features.

IV. Trojan Demo Platform and GUI

The platform is shown in Figure 2 and has two instances of the OpenPLC NYU TE, one clean (Trojan-free) and one Trojaned. The GUI enables interactive deployment of Trojans, side channel data collection, and anomaly detection. Components are indexed based on the letters in Figure 2:

- **A, B:** Two OpenPLC NYU TE systems, denoted as A and B in Figure 2. The left-side board (A) is in a clean configuration while the right-side board (B) has a Pico hardware Trojan. Using the GUI, various specific Trojans can be deployed to the right-side board.
- **C:** Odroid N2+ single-board computer used for the front-end GUI and analysis of side channel data collected from the OpenPLC TE systems.
- **D:** Ethernet switch (all communications between the boards uses wired connections)
- **E:** Power supply.

The demo platform is self-contained and uses the single-board computer to run all the data analysis and the GUI for the human operator. The only connections required for the demo platform are power, HDMI (for display from the single-board computer), and USB keyboard and mouse. The demo platform has been implemented to be automated to the fullest extent possible to facilitate Trojan demonstrations. For example, the GUI is launched automatically on boot-up, all communications between the OpenPLC boards and the demo platform's single-board computer are configured automatically during boot-up, and a single shutdown menu item is included in the GUI to send shutdown commands to the OpenPLC boards and the single-board computer to power-off the testbed.

The GUI for the demo platform is designed with an extensible architecture that can integrate several types of Trojans within a unified easy-to-use interface. As shown in Figure 3, the GUI has two panes. The left-side pane has several elements for Trojan selection, experiment runs, and analysis configuration. After running a Trojan experiment, the results of Trojan detection analysis are shown in the right-side pane. The primary elements in the left-side pane are described below:

- Quick help text and pictures of boards in the block near the top left along with Clean/Trojaned detection annotations after running an experiment.
- Trojan selection dropdown menu and Trojan description text box (of the selected Trojan).
- Start Experiment button: After selecting a Trojan, clicking this button starts the experiment, i.e., deploy the selected Trojan to the right-side board, collect side channel data from both boards, and perform Trojan detection analysis on the collected side channel data to detect anomalies. During the running of an experiment, a progress bar and text updates show the progress of the data collection and analysis. After the analysis, the results are shown as a new tab on the right-side pane as shown in Figure 3.
- Data source configuration (Live/Pre-stored): The data source can be chosen to be either Live or Pre-stored. In the Live mode (which is the default), side channel data is collected from the actual physical OpenPLC TE boards. In the Pre-stored mode, a set of previously collected side channel data (for each Trojan) is used for analysis. The Pre-stored mode is intended for quick tests/demos where one wants to see analysis results without having to run the data collection from the physical boards.
- Configuration options for ignoring some side channels (CPU-based, magnetometer, or both): This is primarily intended as a fallback in case the ambient magnetic environment is too noisy, making it preferable to ignore the magnetometer data. It is recommended however to install the demo platform in a location without too much magnetic noise from the ambient environment (e.g., avoid placing on a table with significant metallic parts or within two feet of large metallic/magnetic items). In addition, the configuration options for ignoring some side channels (CPU-based, magnetometer, or both) can be used to see what the Trojan detection results are when ignoring CPU-based or magnetometer side channel data or both.

The demo platform includes the following Trojans and can be extended to include additional ones:

- **passthrough** is a lightweight Trojan configuration in which the GPIO and UART lines are passed through via the Pico microcontroller. The Trojan is detectable from loopback timing measurements.
- **delay** Trojan injects a 1-second lag between the external I/O pins and the core Pi's I/O. This Trojan is detectable from deviations between observed and expected patterns of magnetometer readings.
- **invert** Trojan inverts the two bits of relay outputs farthest from magnetometer. This Trojan can be detected from deviations between observed and expected patterns of magnetometer readings.
- **replay** Trojan repeatedly transmits a 10-second snippet of relay commands. Trojan is triggered when the last two bits of the relay commands are 1. The Trojan is detectable from deviations between observed/expected patterns of magnetometer readings.
- **acoustic_leak** Trojan snoops on the UART bus and exfiltrates the UART data by "clicking" two of the relays

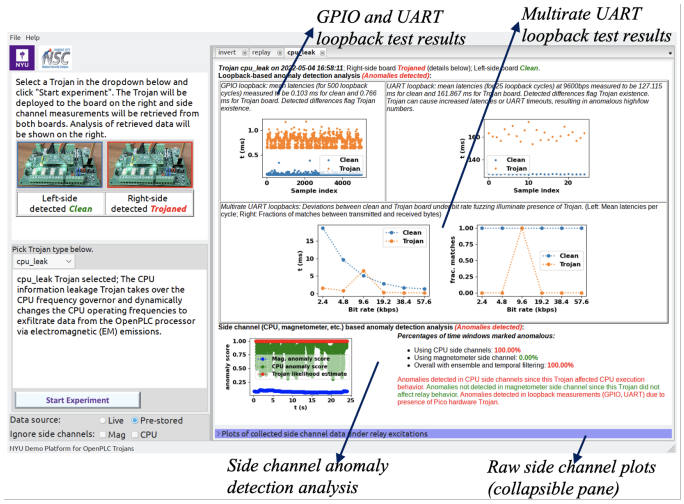

Fig. 3. Screenshot of NYU OpenPLC Trojan Platform GUI: View after running a few Trojan experiments. Analysis results are shown in a separate tab with selected Trojan type shown as the tab name.

(farthest from magnetometer) in an audible pattern. This Trojan is detectable from deviations between observed and expected patterns of magnetometer readings.

- **perf_degrade** Trojan takes over the CPU frequency governor and dynamically changes the CPU operating frequencies to degrade the performance of the OpenPLC processor. For this purpose, the Trojan inverts the normal frequency scaling behavior, i.e., it decreases CPU frequency during high load and increases it otherwise.
- **cpu_leak** Trojan takes over the CPU frequency governor and dynamically changes the CPU operating frequencies to exfiltrate data from the OpenPLC processor via electromagnetic (EM) emissions.

The side channel data collected from the platform include timing measurements during GPIO and UART loopbacks and CPU-based and magnetometer side channel data during relay excitations/actuations. During relay actuations, relay states can be seen on the LEDs on the OpenPLC board. Considering the LEDs for the four relays as a 4-bit integer, the relay commands count sequentially from 0 to 15. Depending on the Trojan, mismatches between the left-side (clean) board and the right-side (Trojan) board can be seen visually. With Trojans that do not affect the relay behavior (i.e., passthrough, perf_degrade, cpu_leak), relay actuations are synchronized between the boards. For Trojans that affect the relay behavior (delay, invert, replay, acoustic_leak), relay actuations of the two boards will differ based on the Trojan's behavior.

Using the side channels collected from the OpenPLC boards, several analyses are performed for anomaly detection based on the methodologies developed in our prior work [1] and the corresponding plots/visualizations are automatically generated and populated into the GUI as shown in Figure 3:

- GPIO loopback timing: differences in mean latencies between the left-side (clean) board and the right-side (Trojaned) board flag Trojan existence.
- UART loopback at 9600bps: similar to GPIO loopback, differences in mean latencies between clean board and the Trojan board (right-side) flag Trojan existence.
- Multirate UART loopback tests (over bit rates from 2.4 kbps to 57.6 kbps): differences in behaviors during multirate loopbacks between the left-side (clean) board and the right-side (Trojaned) board flag Trojan existence.
- CPU and magnetometer side channel anomaly detection analysis: Anomaly detection is performed using ML models over sliding time windows and yields a time series of anomaly scores for CPU and magnetic side channels. From these anomaly scores, ensemble and temporal filtering yields a time series of Trojan likelihood estimates. Percentages of time windows marked as anomalous using CPU side channels, magnetometer side channel, and analysis are included in the summary along with a short description of detected anomalies. Plots of side channel data collected from the boards during relay excitations are included as a pane at the bottom of the tab.

GPIO/UART loopback analysis flags Trojan behavior with all the Trojans listed above since the presence of the Pico on the right-side board results in measurable timing differences even in the passthrough mode. For Trojans that affect the relay behavior (delay, invert, replay, acoustic_leak), magnetometer analysis flags the Trojan behavior. For the Trojans that affect CPU code execution (perf_degrade, cpu_leak), the CPU side channel data flags the Trojan behavior.

REFERENCES

[1] H. Pearce, V. R. Surabhi, P. Krishnamurthy, J. Trujillo, R. Karri, and F. Khorrami, "Detecting hardware Trojans in PCBs using side channel loopbacks," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, no. 7, pp. 926–937, 2022.

[2] S. Adepu, N. K. Kandasamy, and A. Mathur, "EPIC: An electric power testbed for research and training in cyber physical systems security," in *Proceedings of International Workshop on Security and Privacy Requirements Engineering*. Barcelona, Spain: Springer, Sep. 2018, pp. 37–52.

[3] H.-K. Shin, W. Lee, J.-H. Yun, and H.-C. Kim, "Implementation of programmable CPS testbed for anomaly detection," in *Proceedings of USENIX Conference on Cyber Security Experimentation and Test*. Santa Clara, CA: USENIX Association, Aug. 2019, p. 2.

[4] S. Choi, J.-H. Yun, B.-G. Min, and H. Kim, "POSTER: Expanding a programmable CPS testbed for network attack analysis," in *Proceedings of ACM Asia Conference on Computer and Communications Security*, Taipei, Taiwan, Oct. 2020, pp. 928–930.

[5] E. Korkmaz, A. Dolgikh, M. Davis, and V. Skormin, "Industrial control systems security testbed," in *Proceedings of Annual Symposium on Information Assurance*, 2016, pp. 13–18.

[6] E. Korkmaz, A. Dolgikh, M. Davis, and V. Skormin, "ICS security testbed with delay attack case study," in *Proceedings of IEEE Military Communications Conference*, Baltimore, MD, Nov. 2016, pp. 283–288.

[7] A. P. Mathur and N. O. Tippenhauer, "SWaT: a water treatment testbed for research and training on ICS security," in *Proceedings of International Workshop on Cyber-physical Systems for Smart Water Networks*, Vienna, Austria, Apr. 2016, pp. 31–36.

[8] F. Zhang, H. A. D. E. Kodituwakku, J. W. Hines, and J. Coble, "Multilayer data-driven cyber-attack detection system for industrial control systems based on network, system, and process data," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4362–4369, 2019.

[9] J. Rubio-Hernan, J. Rodolfo-Mejias, and J. Garcia-Alfaro, "Security of cyber-physical systems: From theory to testbeds and validation," in *Proceedings of Workshop on the Security of Industrial Control Systems and of Cyber-Physical Systems*. Crete, Greece: Springer, Sep. 2017, pp. 3–18.

[10] F. Sauer, M. Niedermaier, S. Kießling, and D. Merli, "LICSTER–a low-cost ICS security testbed for education and research," *arXiv preprint arXiv:1910.00303*, 2019.