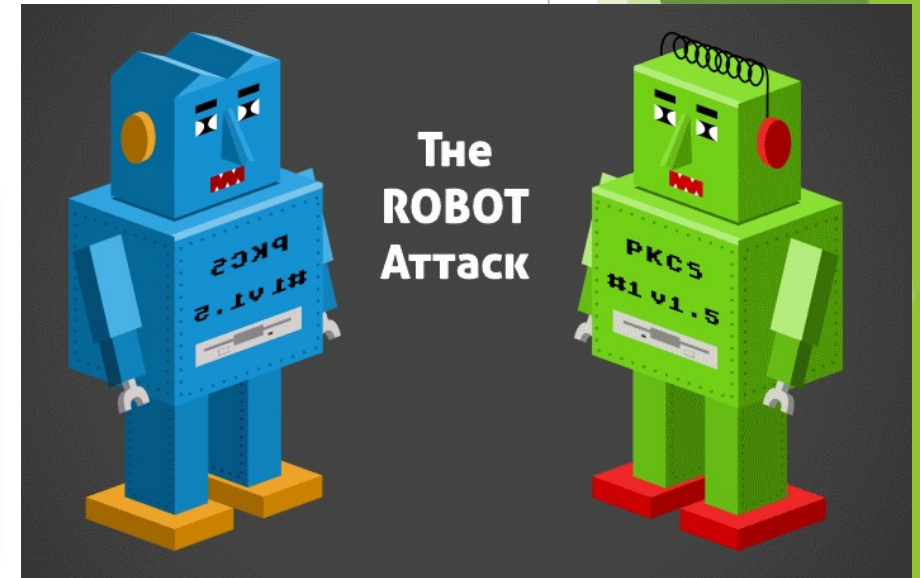
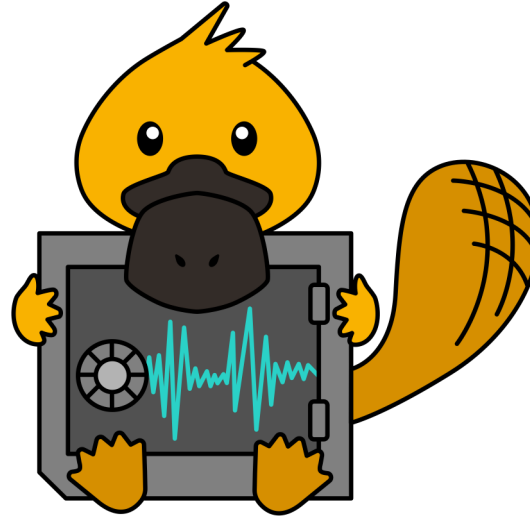


SOVRA Labs Pvt. Ltd.

AI-Powered Security Testing for Crypto Wallets and Financial Systems

Side-channel Attacks: With Great Power Comes Great **LEAKAGE**

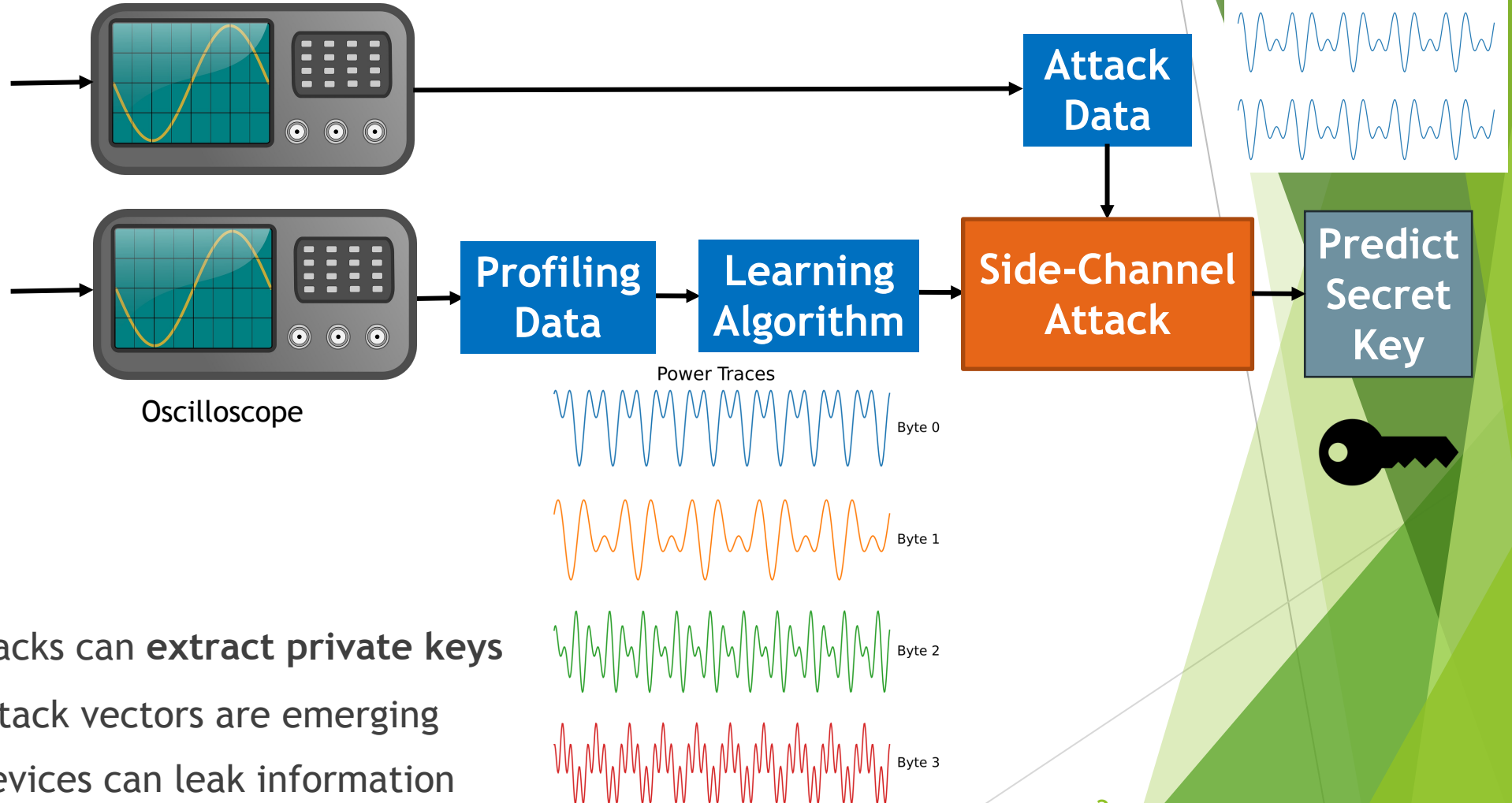


Hardware Side-Channel attacks

Attack Device



Profiling Device



Key Risk

- ▶ Side-channel attacks can **extract private keys**
- ▶ New AI-driven attack vectors are emerging
- ▶ Even “secure” devices can leak information during operation

Hardware Wallets Are Not Fully Secure

⚠️ Current Security Testing is Broken

- Manual audits → **slow and expensive**
- Limited coverage → **unknown attack paths missed**
- No continuous testing → **new threats go undetected**

💰 Impact

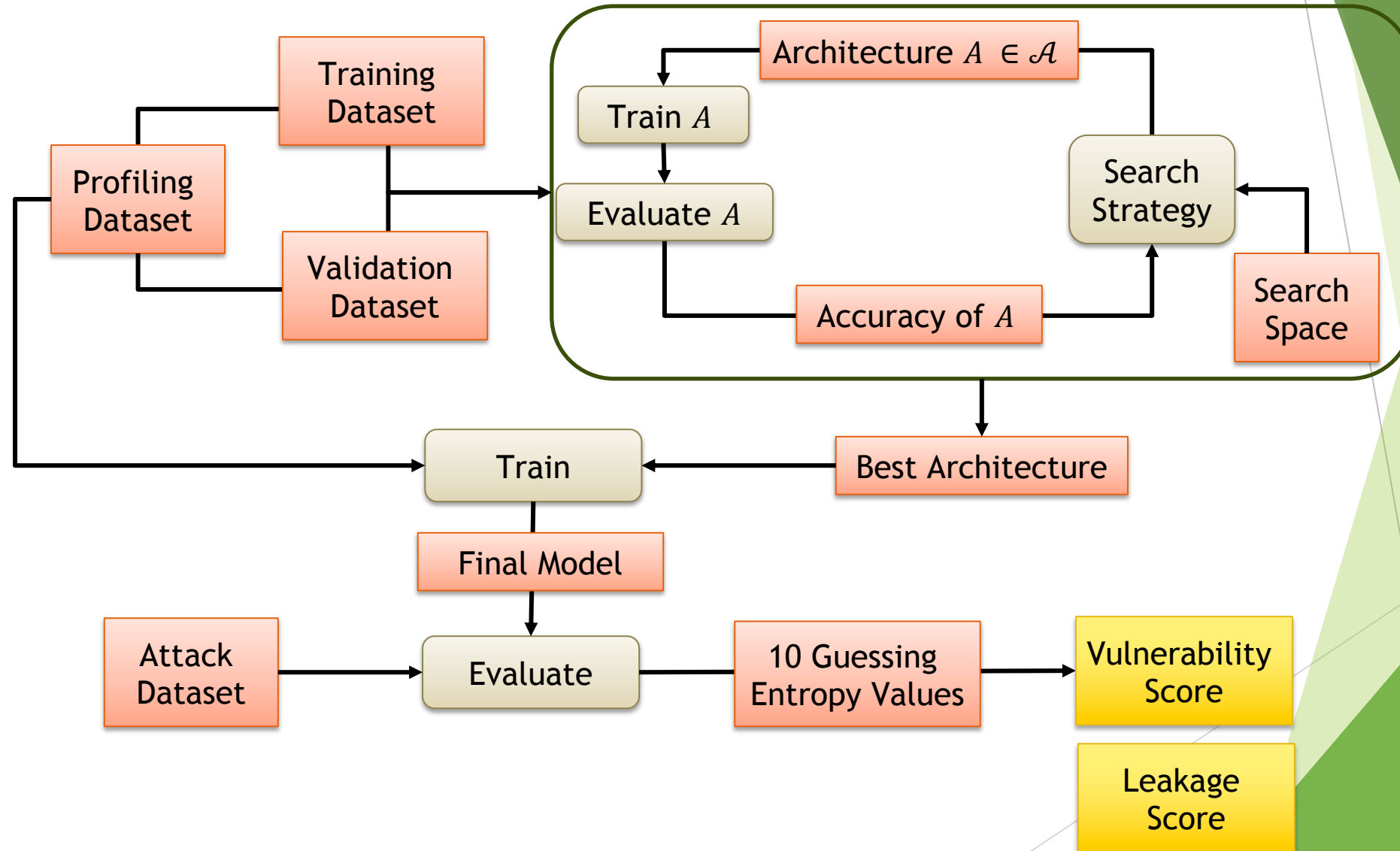
- Loss of funds
- Compromised wallets
- Loss of user trust

Our Proposed Solution

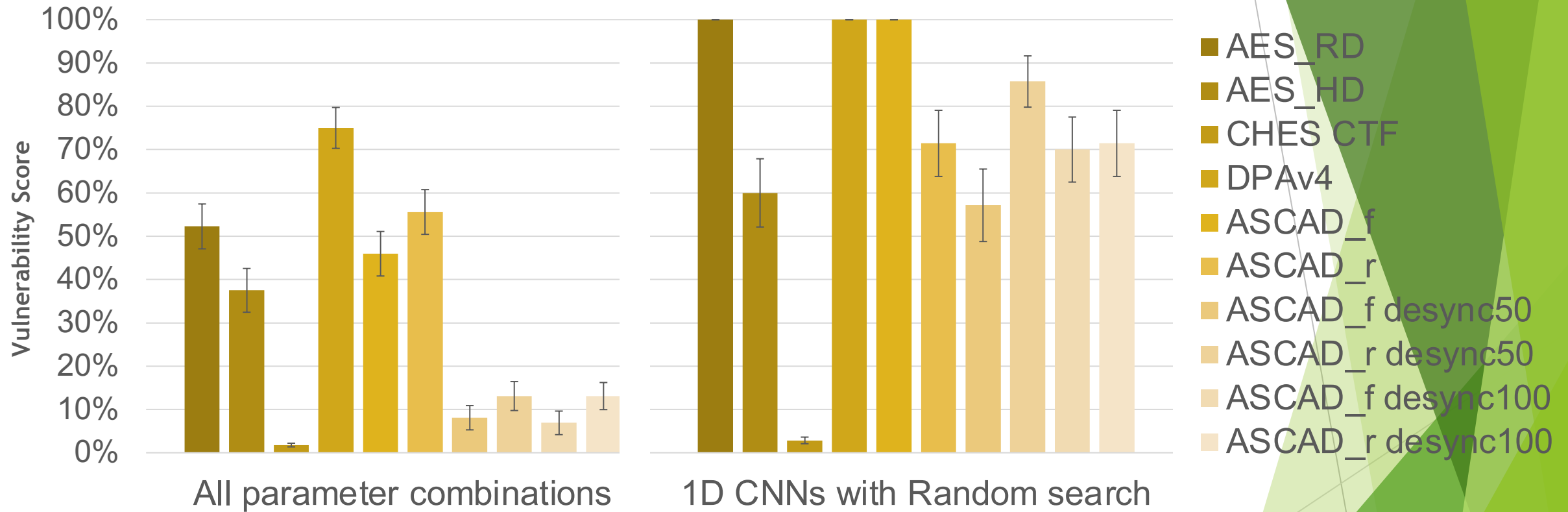
An automated testing platform that simulates real-world attacks on hardware wallets.

- Input: wallet/device/system
- Process:
 - simulate side-channel attacks
 - simulate adversarial inputs
- Output:
 - vulnerability score
 - leakage detection
 - attack traces

Our Neural Architecture Search Black-box Approach



Per-Dataset Performance (ARES-2022)



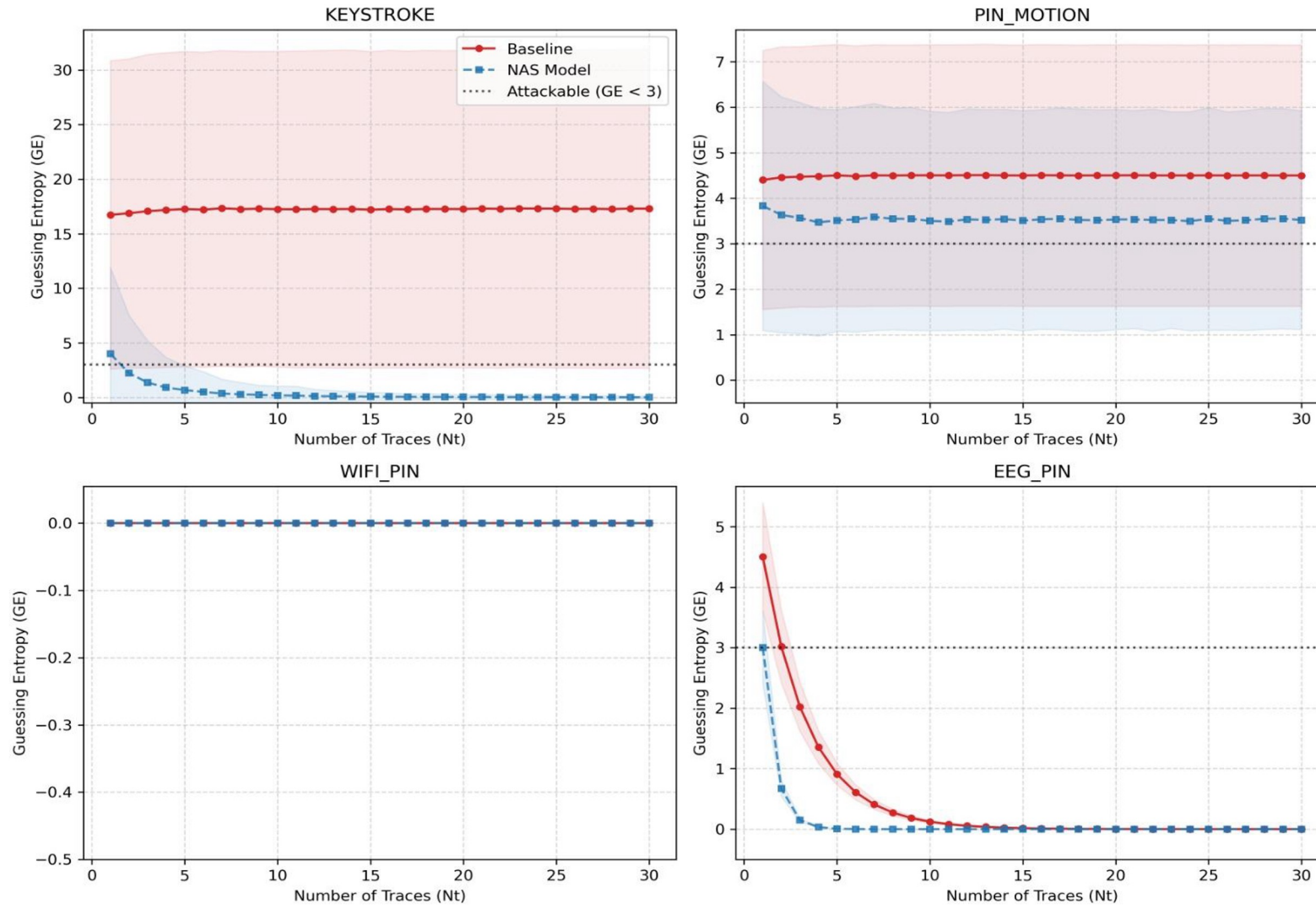
4 Systems: With countermeasures, 6 Systems: Without countermeasures

Cross-Domain NAS Performance Evaluation

The NAS framework was successfully extended from cryptographic side-channel analysis to four diverse side-channel modalities: keystroke dynamics, smartphone motion sensors, Wi-Fi CSI, and EEG signals.

Roobika Thangavel , PSG college of technology , Anna university

Guessing Entropy (GE) vs. Number of Traces by Domain



KEYSTROKE

Dataset: CMU Keystroke Dynamics Benchmark Dataset (Killourhy & Maxion, 2009)

Source: <https://www.cs.cmu.edu/~keystroke/>

Official Data File: DSL-StrongPasswordData.csv

Subjects: 51 users, **Samples:** 400 password entries per subject, **Password Typed:** .tie5Roanl

Features: Keystroke timing measurements including:

- ▶ Hold (Dwell) Time
- ▶ Down-to-Down (DD) Interval
- ▶ Up-to-Down (UD) Interval

Data Format: CSV containing timing features for user authentication research.

Preprocessing: Timing features are converted into temporal sequences of shape (samples, 31, 1) for deep learning models, with **51 target classes** representing individual subjects.

Reference:

Killourhy, K. S., & Maxion, R. A. *Comparing Anomaly-Detection Algorithms for Keystroke Dynamics*. IEEE/IFIP DSN, 2009.

INSIGHT Wi-Fi PIN Inference Dataset

Dataset: INSIGHT Wi-Fi PIN Inference Dataset

Source: *PINSIGHT: A Comprehensive Threat Exploration of Domain-Adaptive Wi-Fi based PIN Code Inference* (Kortz et al., 2026)

Data Type: Wi-Fi Channel State Information (CSI) measurements recorded during PIN entry.

Collection Method: CSI data captured using a robotic typing platform to generate highly repeatable PIN keystrokes across systematically varied environments.

Features: Wi-Fi CSI measurements processed into temporal sequences for deep learning.

Target Classes: 10 digit classes (0-9) for PIN digit recognition.

Application: Domain-adaptive Wi-Fi side-channel attack for PIN code inference and cross-environment evaluation.

► **Reference:**

Kortz, J., Staat, P., Paar, C., & Zenger, C. *PINSIGHT: A Comprehensive Threat Exploration of Domain-Adaptive Wi-Fi based PIN Code Inference*, arXiv:2605.04570, 2026. Link : <https://arxiv.org/abs/2605.04570>

WIFI_PIN (WiFi CSI)

Dataset: Simulated Wi-Fi Channel State Information (CSI) Dataset

Source/Reference: *Keystroke Recognition Using WiFi Signals* (Ali et al., ACM MobiCom 2015),
Reference Link: <https://dl.acm.org/doi/10.1145/2789168.2790109>

Data Type: Simulated Wi-Fi CSI measurements inspired by the WiKey framework.

Simulation Motivation: The original WiKey dataset is not publicly available, and Intel 5300 CSI logs require specialized decoding tools. Therefore, a realistic CSI simulator was developed for model training and evaluation.

Simulation Specifications:

- ▶ 1,000 simulated CSI samples
- ▶ Sequence shape: (100, 30)
- ▶ 100 temporal timesteps, 30 CSI subcarrier channels (modeled after the Intel 5300 NIC)
- ▶ Rayleigh fading channel with localized attenuation patterns representing finger movements

Target Classes: 10 PIN digit classes (0-9)

- ▶ **Reference:** Ali, K., Liu, A. X., Wang, W., & Shahzad, M. *Keystroke Recognition Using WiFi Signals. Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*, 2015. DOI: 10.1145/2789168.2790109.
<https://dl.acm.org/doi/10.1145/2789168.2790109>

EEG_PIN

Dataset: Simulated EEG PIN Dataset

Source/Reference: *Brain Wave Authentication using EEG Signals* (USENIX Security Symposium, 2012)

Data Type: Simulated electroencephalography (EEG) signals for PIN digit recognition.

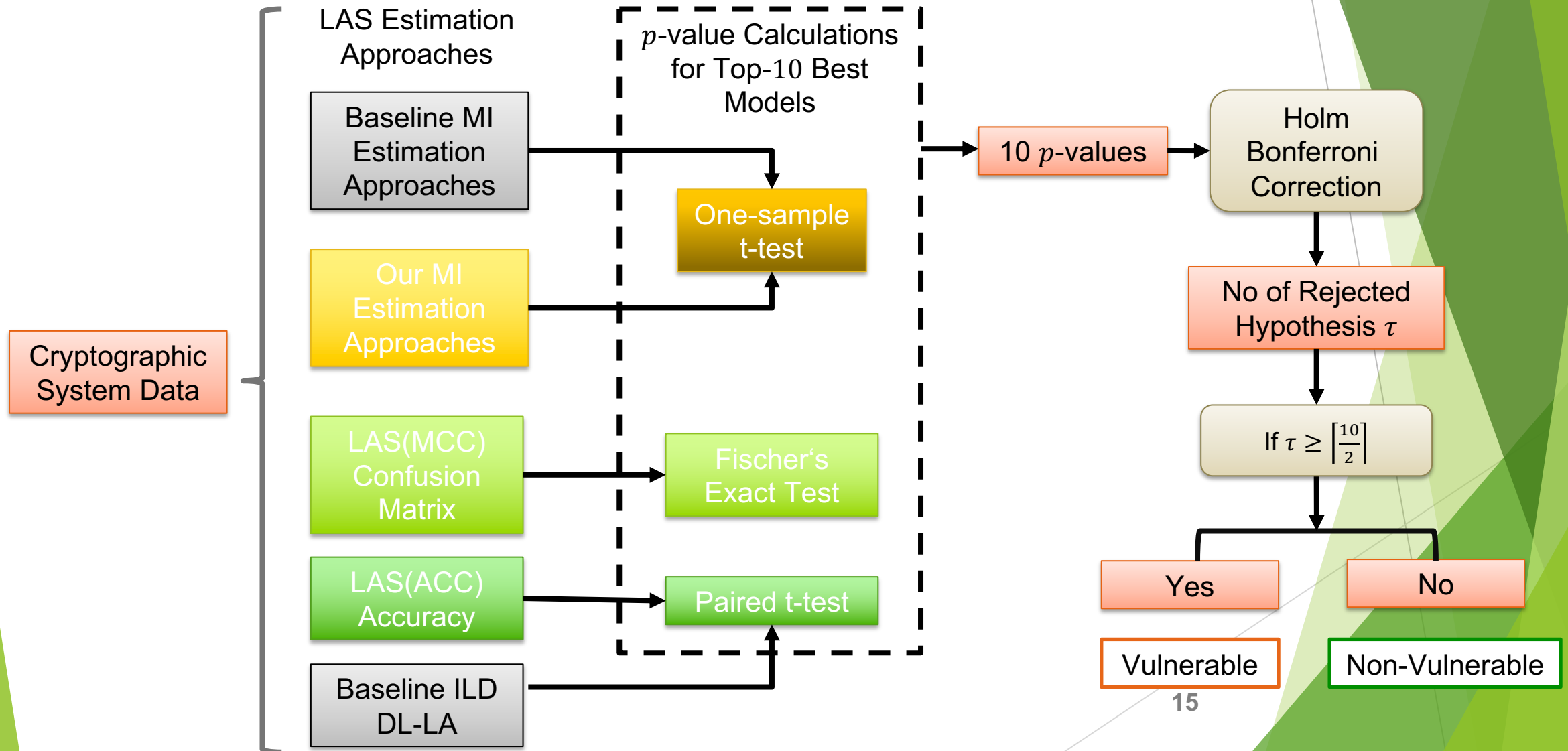
Simulation Motivation: Raw EEG datasets (e.g., PhysioNet .edf recordings) are large and require extensive preprocessing, artifact removal, and manual segmentation. Therefore, a realistic EEG simulator was developed for efficient model training and evaluation.

Simulation Specifications:

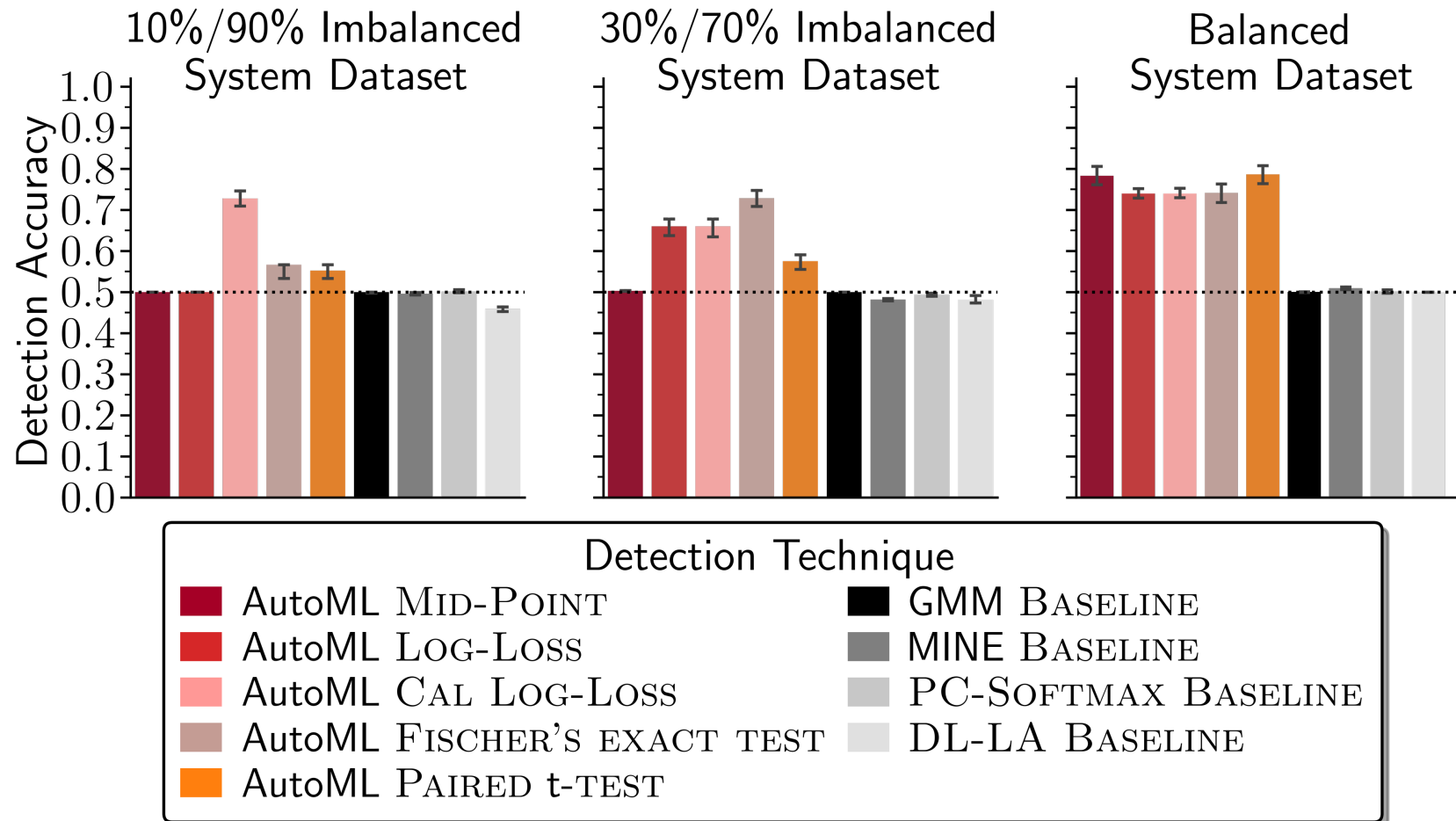
- ▶ 1,000 simulated EEG samples
- ▶ Sequence shape: (250, 64) per sample
- ▶ 250 temporal timesteps
- ▶ 64 EEG electrode channels based on the standard 10-20 EEG cap layout
- ▶ Simulated 1/f pink noise background activity with injected motor-cortex Event-Related Potentials (ERPs), including N200 and P300 waveforms
- ▶ Target Classes: 10 PIN digit classes (0-9)

Reference: Rasmussen, K. B., Roeschlin, M., Martinovic, I., & Tsudik, G. Brain Wave Authentication using EEG Signals. 21st USENIX Security Symposium (USENIX Security '12), 2012. Reference Link: <https://www.usenix.org/system/files/conference/usenixsecurity12/sec12-final56.pdf>

Leakage Detection Process (IS-Journal-2025)

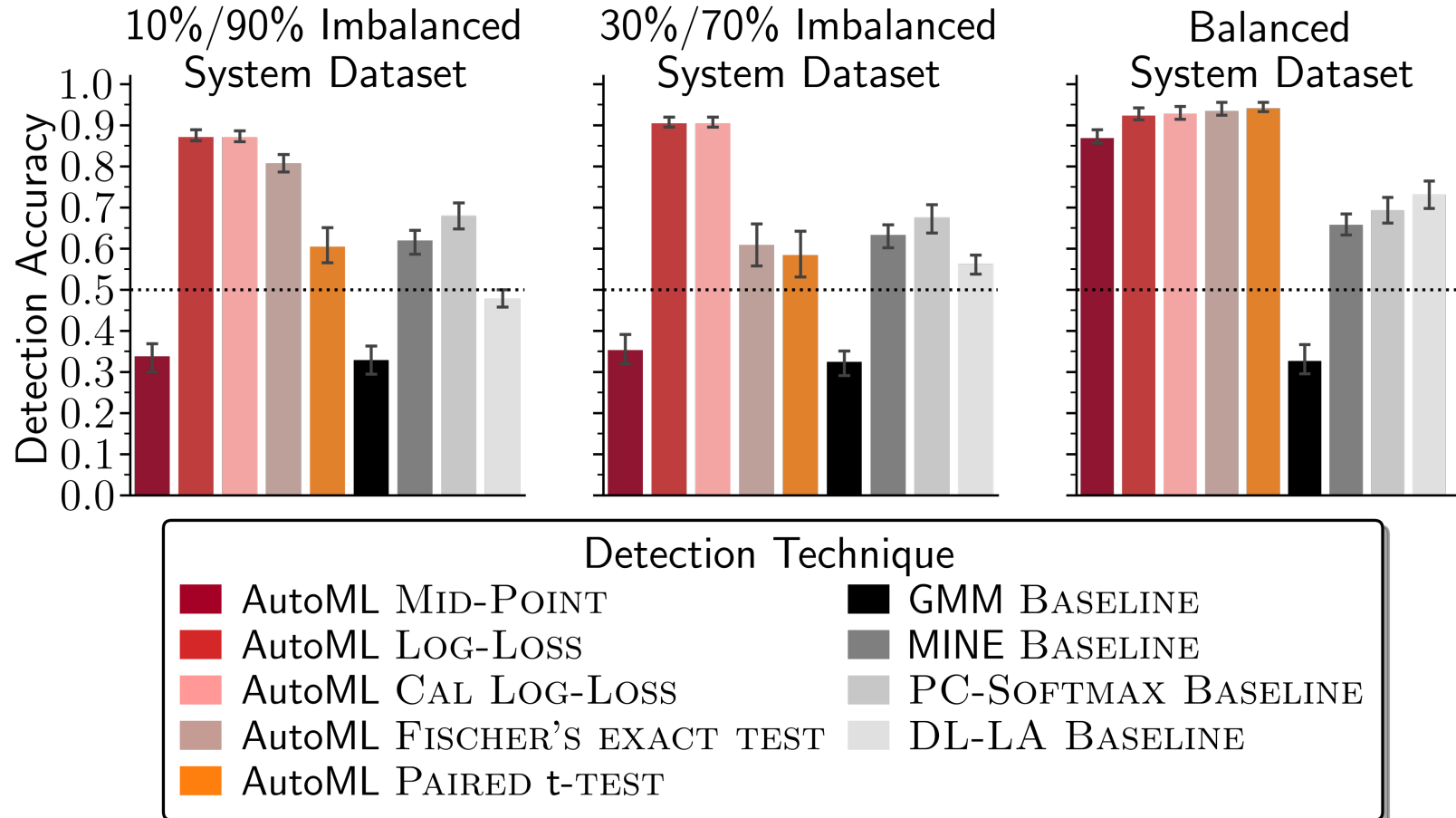


Detection Accuracy on Timing Based Vulnerability



OpenSSL 0.9.7a server: Vulnerable
OpenSSL 0.9.7b server: Non-vulnerable
Difference in processing time:
Range between 1 and 256 μ – seconds

Detection Accuracy on Error-codes Based Vulnerability



OpenSSL TLS
Vulnerable Systems: 5
Non-Vulnerable Systems: 3

Demo: Achelos GmbH

AutoSCA Test Tool: Solution Mapped to test OpenSSL TLS Server

OpenSource Version: <https://github.com/ITSC-Group/autosca-tool>

Full Demo: <https://www.youtube.com/watch?v=-ljLPgrw-DY>

Results: <https://github.com/LeakDetectAI/autosca-hw-wallet-tester>

“To validate this approach, we mapped our solution to an existing industrial-grade tool developed by Achelos GmbH, which performs automated side-channel testing. This demonstrates how our approach works in real-world systems.”

Conclusion

Proposed a model-agnostic LAS measure to quantify information leakage

Our AutoML-based methods

- Provided a black-box automated solution to detect leakage
- Demonstrated scalability to **high-dimensional datasets**
- Provided precise estimation for **imbalanced datasets**

Compared our approaches with baselines using real-world datasets

Our approaches outperform the baseline performance across real-world scenarios in detecting leakages in AES-encrypted systems and OpenSSL TLS servers.

The R&D Pipeline

