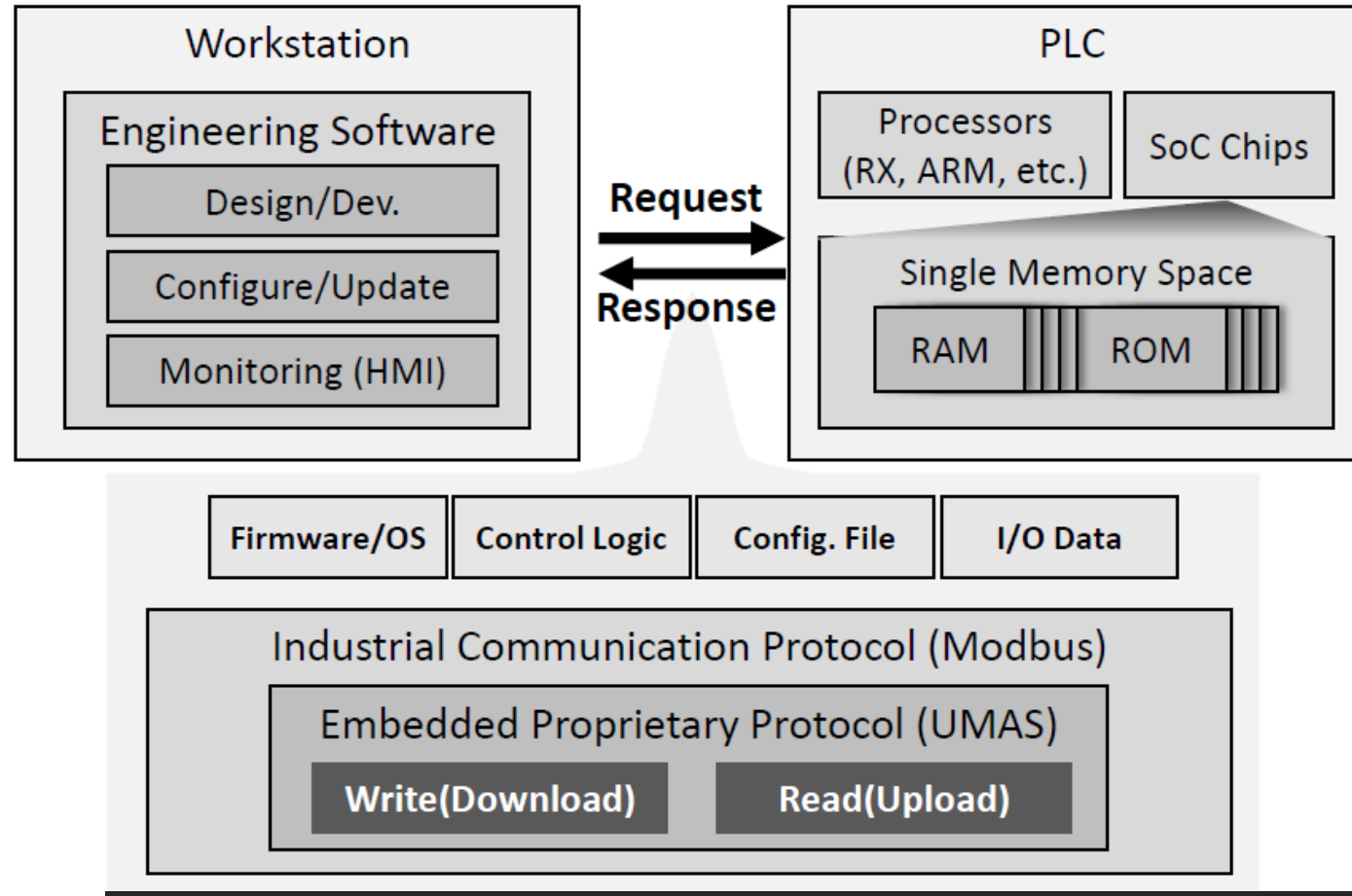
Oops, It Halted Again

Exploiting PLC Memory for Fun and Profit

Wooyeon Jo, Irfan Ahmed

Industrial Control System (ICS) & Programmable Logic Controller (PLC)

ICS are network-driven automation systems that control physical processes through PLCs



- **Industrial Control Systems (ICS)** are used to automate and monitor critical infrastructure (**CI**) based on **PLC**
- **PLC** is running physical process based on Control Logic
- **Control Logic** is written using ladder logic, structured text, or other IEC 61131-3 languages
- **All of these are based on network protocols**

VCU College of Engineering SAFE Lab

ICS is Prime Target

Recent ICS breaches demonstrate attackers shifting towards hardware-level exploitation, increasing sophisticated threats to PLC memory.

- Cyberattacks increasingly targeting Industrial Control Systems (ICS).
- Modern PLCs continuously evolved with advanced functionalities & increased capabilities.
- Existing ICS security research focuses heavily on network side.

VCU College of Engineering

SAFE Lab

The Challenges

- Access PLC Memory
- No Standardization or Ground Truth
- The "Everything is a Pointer" Problem
- The Moving Target Problem
- Finding the True "Root of Execution"
  - Identifying and Verifying RAP regions

Our Approach

- **Access PLC Memory**
  - Building on prior work, we can acquire PLC memory by leveraging proprietary industrial protocols (e.g., UMAS, PCCC) or physical interfaces like JTAG.
- No Standardization or Ground Truth
- The "Everything is a Pointer" Problem
- The Moving Target Problem
- Finding the True "Root of Execution"
  - Identifying and Verifying RAP regions

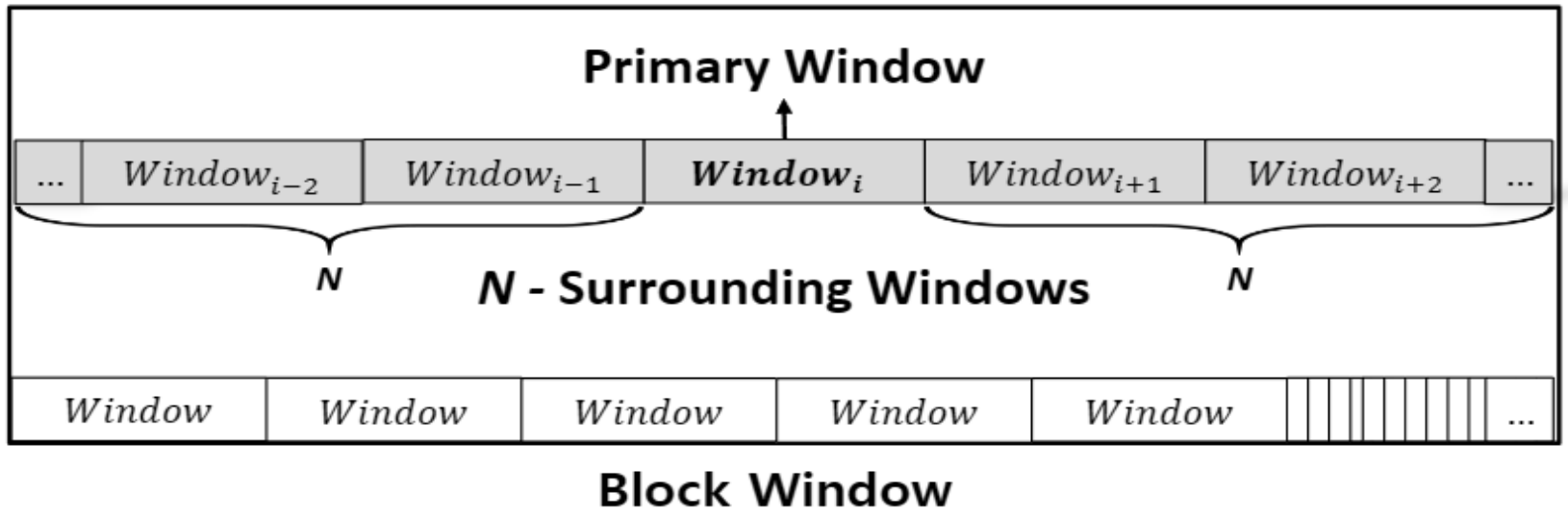
UMAS

Modbus header	Function code (0x5A/90)	Session key	UMAS Function	Data
	1 Byte	1 Byte	1 Byte	1 Byte

Modbus/TCP

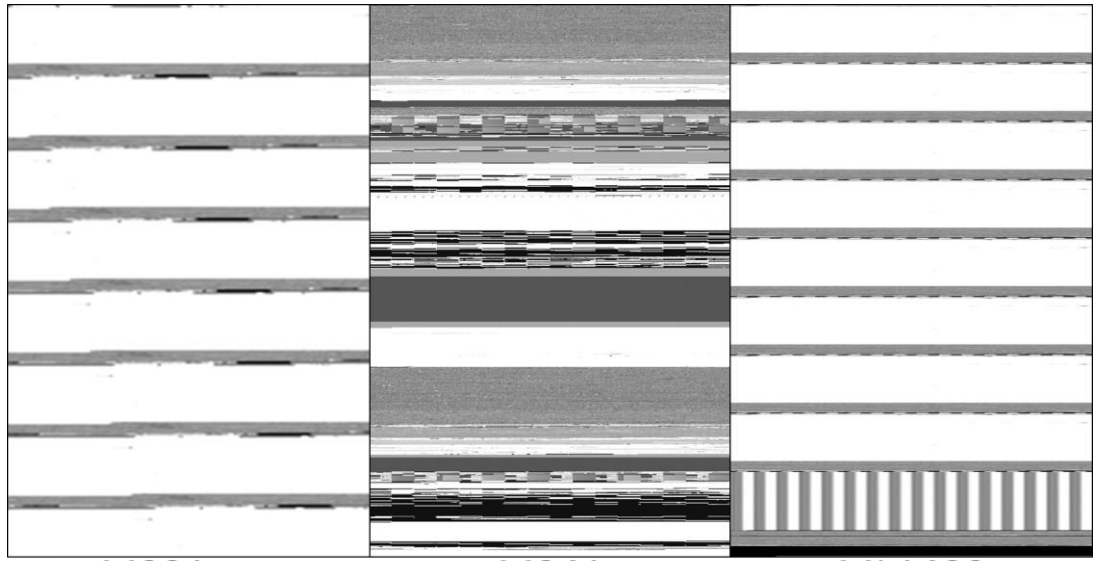
Our Approach

- No Standardization or Ground Truth
- The Moving Target Problem

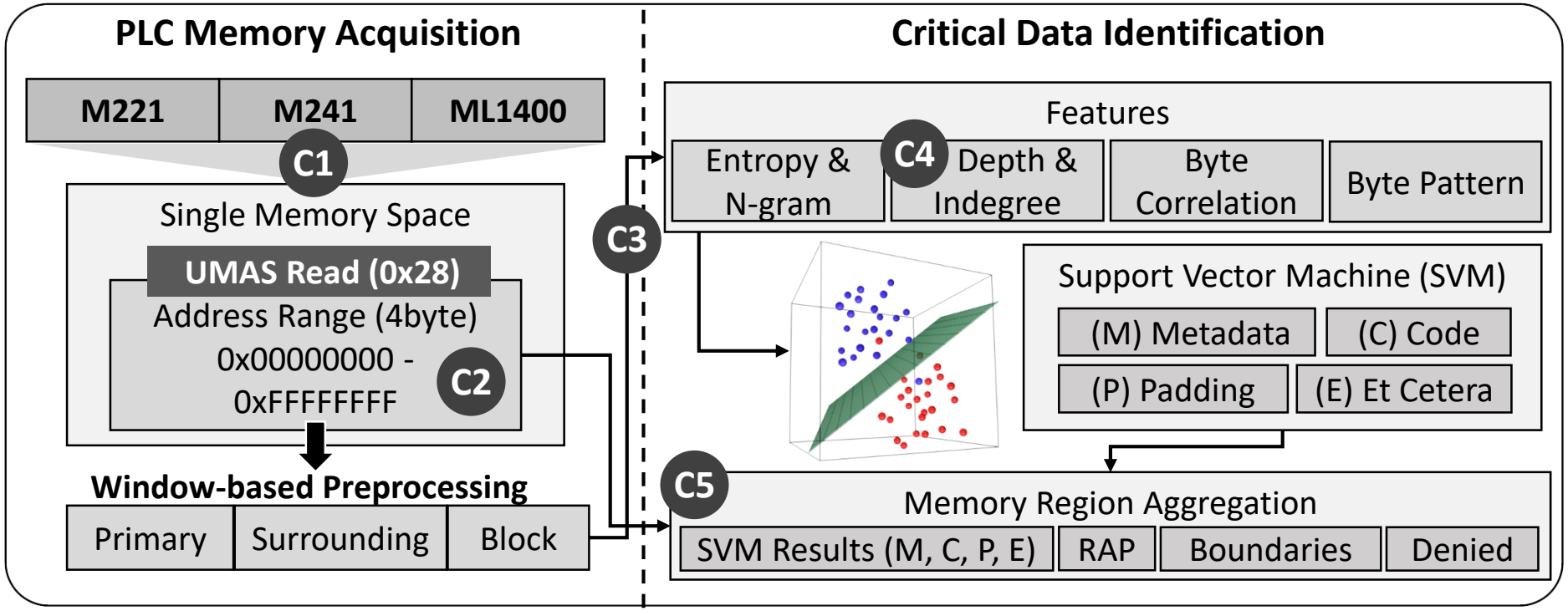


Our Approach

- The "Everything is a Pointer" Problem
- Finding the True "Root of Execution"
  - Identifying and Verifying RAP regions



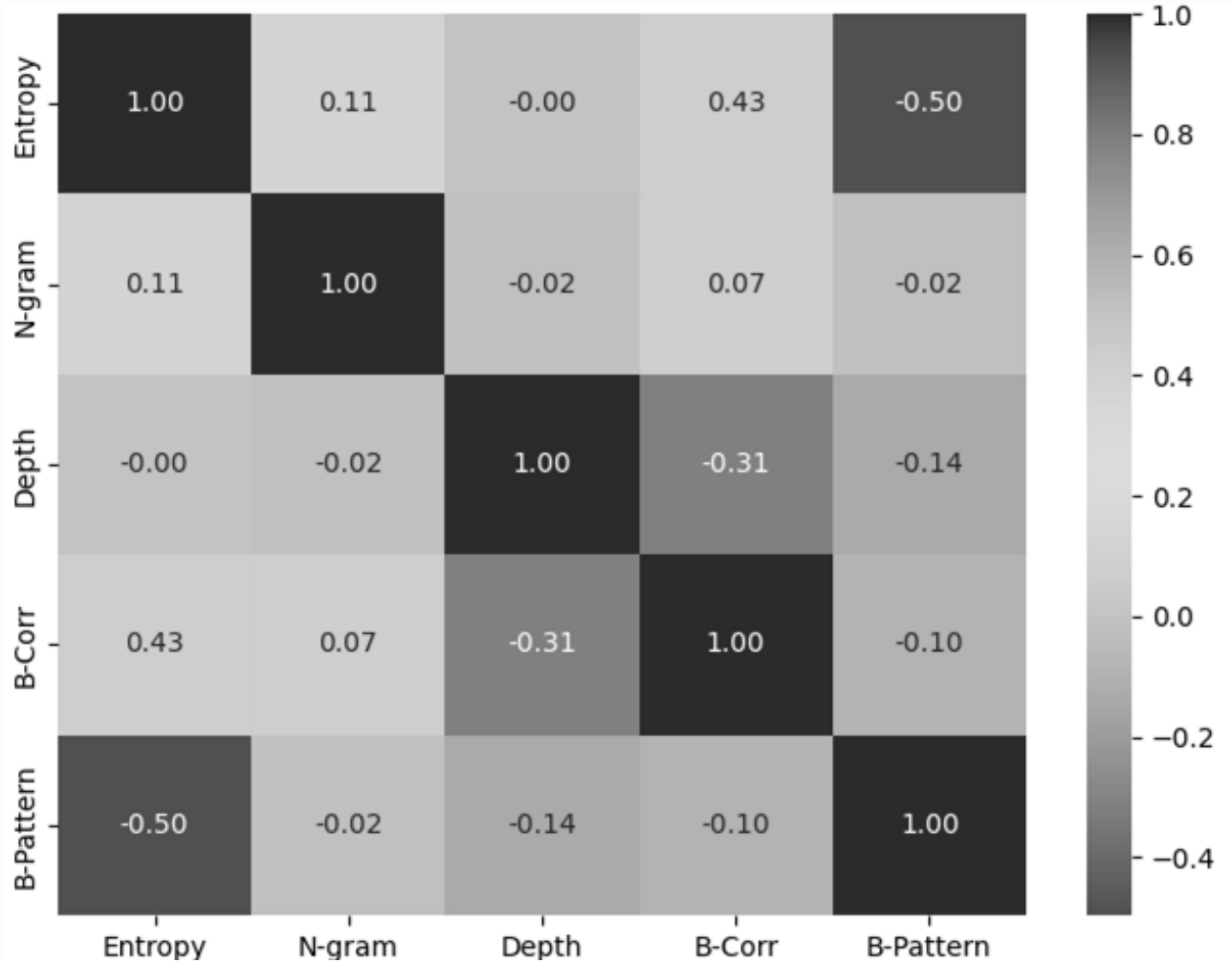
Our Approach



- **Memory Acquisition & Window-based Preprocessing**
- **Feature-driven Classification with SVM**
- **Post-processing and Region Aggregation**

Features

- **Our Features for Memory Classification**
- **Entropy, N-gram, Depth/Indegree, Byte-Correlation, and Byte-Pattern**
- **Pearson correlation analysis** confirms these features are sufficiently **independent**



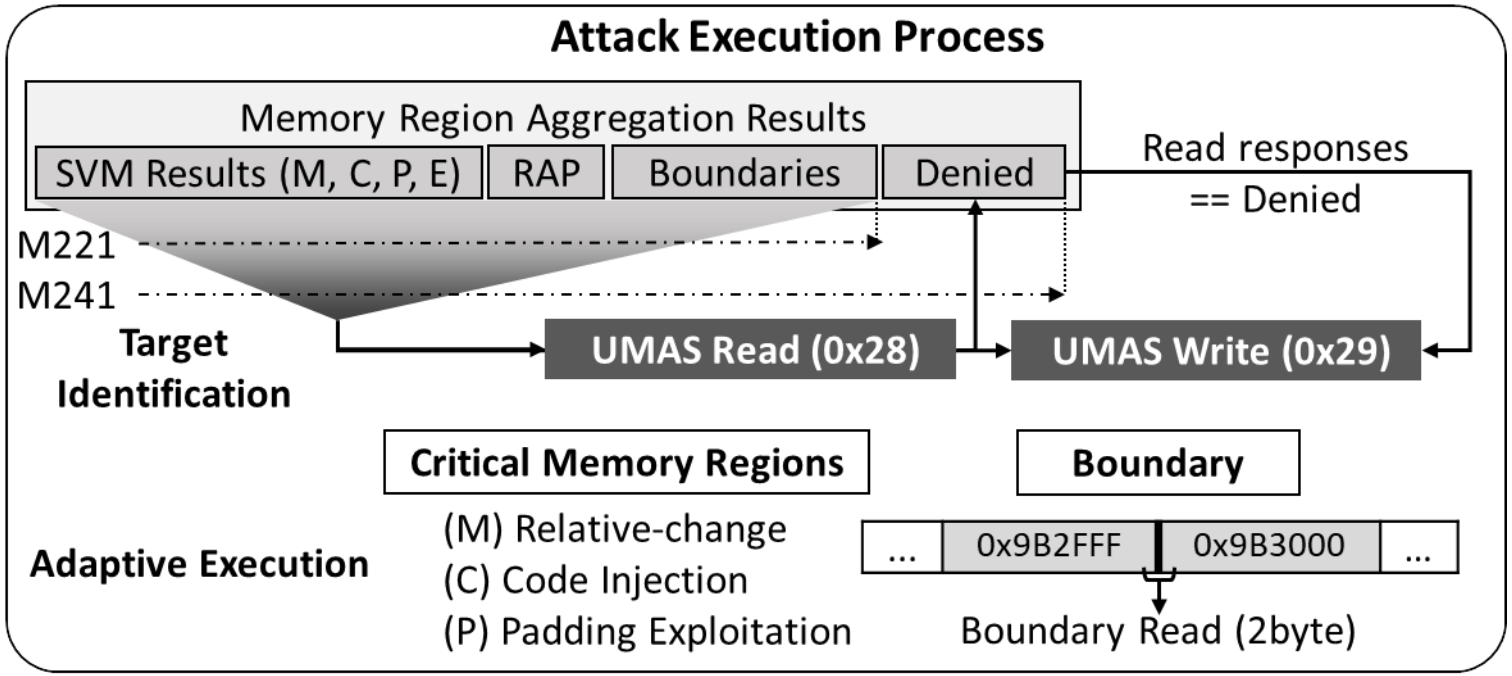
The Targets

PLC Model	Vendor	Series	CPU Architecture	Acquisition	Attack	Access	Detect
M221	Schneider Electric	Modicon	Renesas	Yes	Yes	UMAS Protocol	Yes
M241	Schneider Electric	Modicon	ARM	Yes	Yes	UMAS Protocol	Yes
AB1756	Allen-Bradely	ControlLogix	ARM	Yes	No	JTAG	Yes

- **Testbed:** 3 PLCs, 2 major vendors (Schneider, Allen-Bradley).

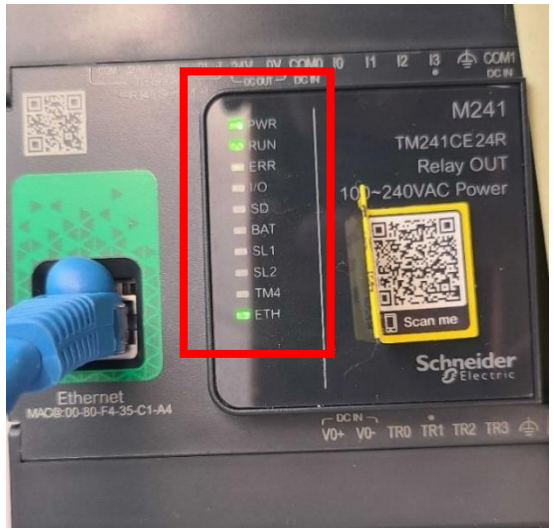
- **Key Test:** Actively attacked two models with different CPUs (Renesas vs. ARM) to prove our method generalizes.

Attack Vectors



- **Our Strategy:** Use ML results to find and attack critical memory.
- **(M) Relative-Change Attack:** A tiny change to a pointer causes a **HALT**.
- **(C) Code Injection:** Simple code duplication causes a **COMPROMISE**.
- **Padding Exploitation:** Use large, empty padding areas for stealthy payload storage.
- **Write-Based Access Violation:** Exploit misconfigured permissions where Read is denied.
- **Boundary Read Probing:** Read at the precise edges between memory regions.

Results



One Request

Halted!

RUN (On → **Off**)

ERR (Off → **On**)

I/O (Off → **On**)

No response

M221: Metadata attacks had a 100% HALT rate.

M241: We found 200+ "kill-switch" memory blocks that cause an instant HALT. The issue has been patched across all relevant product family.

RAP: We found Redundant Address Pin phenomenon across all PLCs

Conclusion

Automated Exploitation Framework	• We introduced a machine learning model that automatically classifies exploitable memory regions (e.g., metadata, code) independent of PLC architecture.
Practical & Severe Threats	• We empirically demonstrated that basic protocol functions, including Read operations alone, can trigger critical system failures, resulting in disclosed CVE
Systemic Weaknesses Uncovered	• Our analysis revealed fundamental weaknesses in PLC security, including the cross-vendor existence of the Redundant Address Pin (RAP) phenomenon and inconsistent memory protection mechanisms
Urgent Need for Defenses	• This work underscores the necessity for vendors to implement stronger memory integrity and access control enforcement to protect against these practical and severe threats

Ethics

Reporting	• All findings were responsibly disclosed to the relevant parties, leading to CVE-2024-11737.
Patch	• While a patch was released in March 2025, the fix was incomplete, as the vulnerability remains accessible at a new location.
Rejection	• Another response team declined to issue a CVE, positing that protocol-level encryption would prevent unauthorized memory access.
Next Steps	• We plan to report on the persistent nature of this vulnerability and disclose additional findings from our ongoing research.

Thank You !

Security through obscurity is not a security

VCU College of Engineering