SoK Paper: Power Side-Channel Malware Detection

Alexander Cathis, Ge Li, Shijia Wei, Michael Orshansky, Mohit Tiwari, Andreas Gerstlauer

Chandra Family Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, Texas, USA

SLAM Lab
System-Level Architecture and Modeling Group

SPARK RESEARCH LAB
Security, Privacy and Computer Architecture

The University of Texas at Austin
Chandra Department of Electrical and Computer Engineering
Cockrell School of Engineering

The International Workshop on Hardware and Architectural Support for Security and Privacy
November 2, 2024

Power Side-Channel

- **Implementation-based medium that leaks information**
  - Electromagnetic, power, timing, etc.
- **Broad and impactful information**
  - Can be used for attack and defense
- **Well suited for defense**
  - Out-of-band implementation
  - No HW/SW overhead

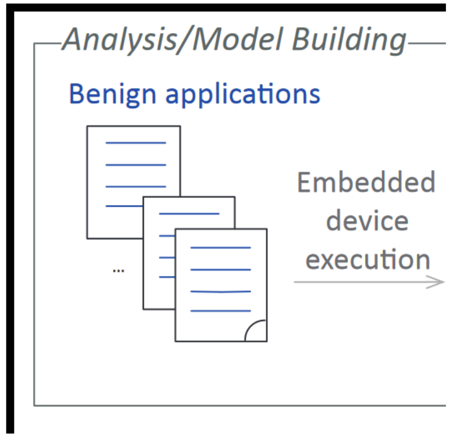
[1] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems, Wei et al. HOST'19

Power Side-Channel

- **Implementation-based medium that leaks information**
  - Electromagnetic, power, timing, etc.

- **Broad and impactful information**
  - Can be used for attack and defense

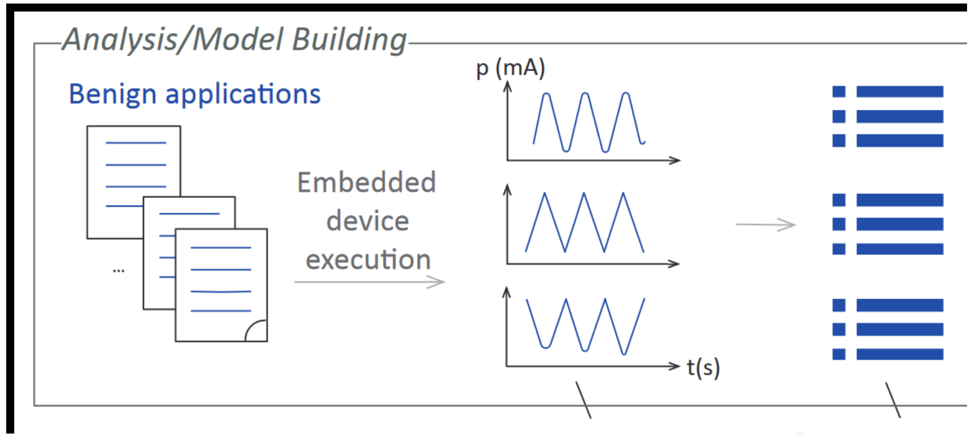
- **Well suited for defense**
  - Out-of-band implementation
  - No HW/SW overhead



[1] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems, Wei et al. HOST'19

Power Side-Channel

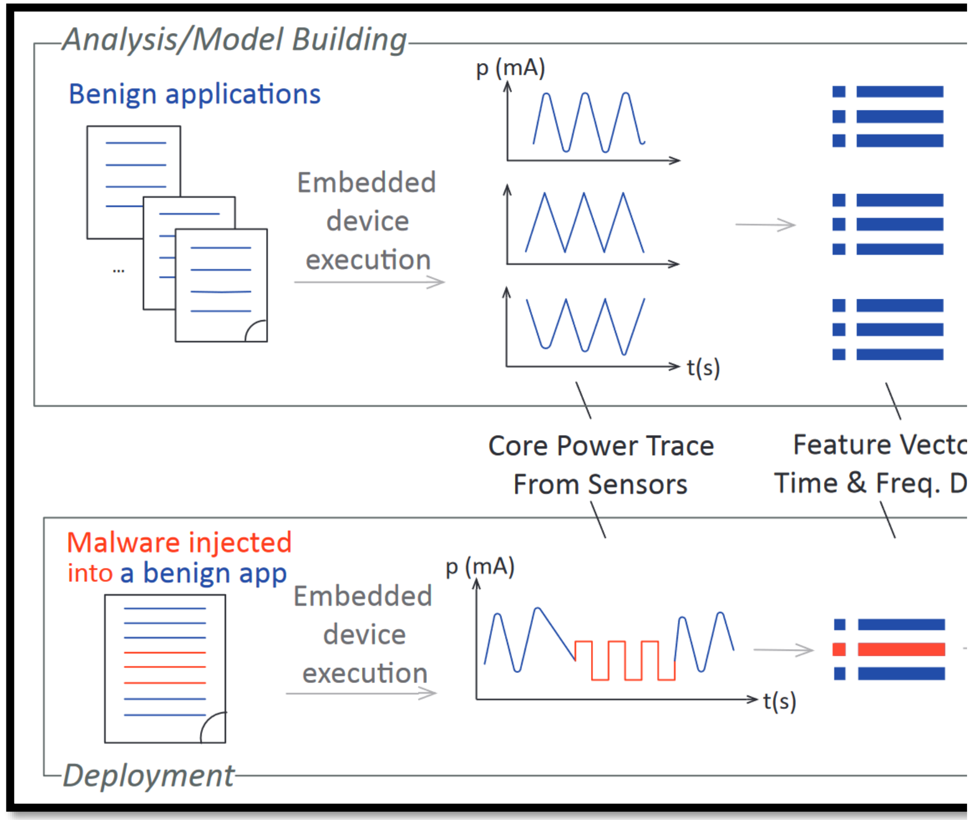
- **Implementation-based medium that leaks information**
  - Electromagnetic, power, timing, etc.
- **Broad and impactful information**
  - Can be used for attack and defense
- **Well suited for defense**
  - Out-of-band implementation
  - No HW/SW overhead



[1] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems, Wei et al. HOST'19

Power Side-Channel

- **Implementation-based medium that leaks information**
  - Electromagnetic, power, timing, etc.
- **Broad and impactful information**
  - Can be used for attack and defense
- **Well suited for defense**
  - Out-of-band implementation
  - No HW/SW overhead

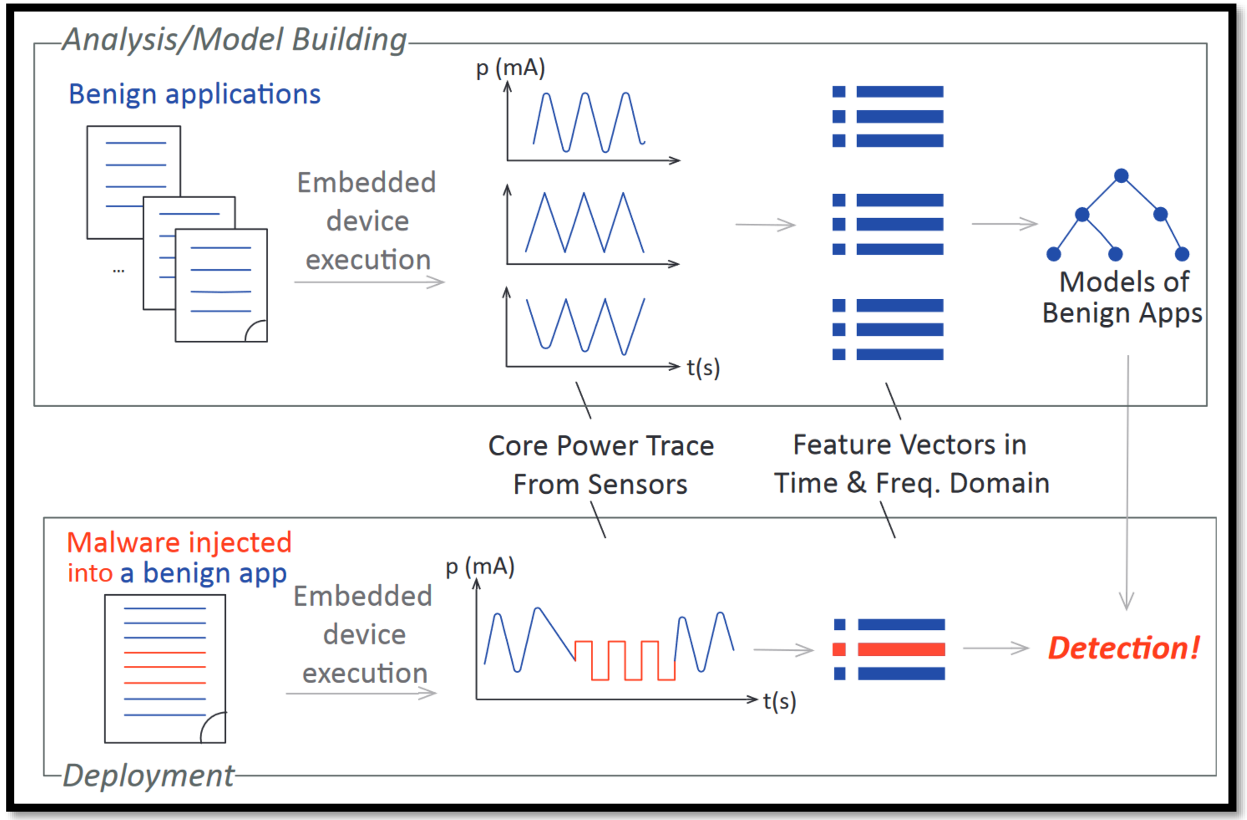


Power-based detector [1]

[1] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems, Wei et al. HOST'19

Power Side-Channel

- **Implementation-based medium that leaks information**
  - Electromagnetic, power, timing, etc.
- **Broad and impactful information**
  - Can be used for attack and defense
- **Well suited for defense**
  - Out-of-band implementation
  - No HW/SW overhead



Power-based detector [1]

[1] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems, Wei et al. HOST'19

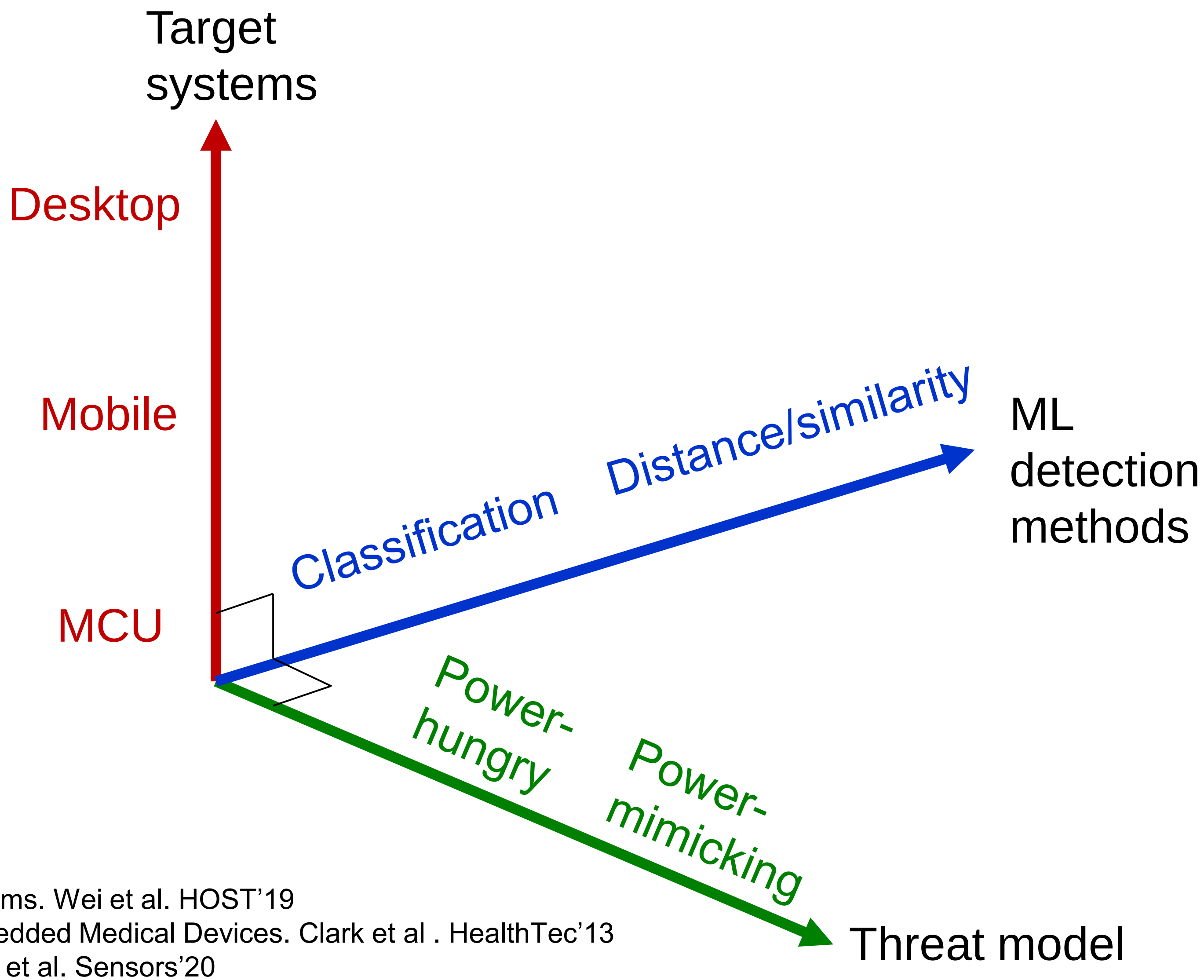
Power-Based Detection Systematization

- Many prior works
- Variety of approaches
- Difficult for new researcher or practitioner to navigate space

[1] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems. Wei et al. HOST'19
[2] Wattsupdoc: Power Side Channels to Nonintrusively Discover Untargeted Malware on Embedded Medical Devices. Clark et al . HealthTec'13
[3] Power-Based Non-Intrusive Condition Monitoring for Terminal Device in Smart Grid. Zhang et al. Sensors'20
[4] Detecting Energy-Greedy Anomalies and Mobile Malware Variants. Kim et al. Mobisys'08
[5] Towards Malware Detection via CPU Power Consumption: Data Collection Design and Analytics. Bridges et al. TrustCom/BigDataSE'18

Power-Based Detection Systematization

- **Many prior works**
- **Variety of approaches**
- **Difficult for new researcher or practitioner to navigate space**
- **Orthogonal (?) variables**
  - Target systems
  - ML detection methods
  - Threat model



[1] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems. Wei et al. HOST'19
[2] Wattsupdoc: Power Side Channels to Nonintrusively Discover Untargeted Malware on Embedded Medical Devices. Clark et al . HealthTec'13
[3] Power-Based Non-Intrusive Condition Monitoring for Terminal Device in Smart Grid. Zhang et al. Sensors'20
[4] Detecting Energy-Greedy Anomalies and Mobile Malware Variants. Kim et al. Mobisys'08
[5] Towards Malware Detection via CPU Power Consumption: Data Collection Design and Analytics. Bridges et al. TrustCom/BigDataSE'18

Outline

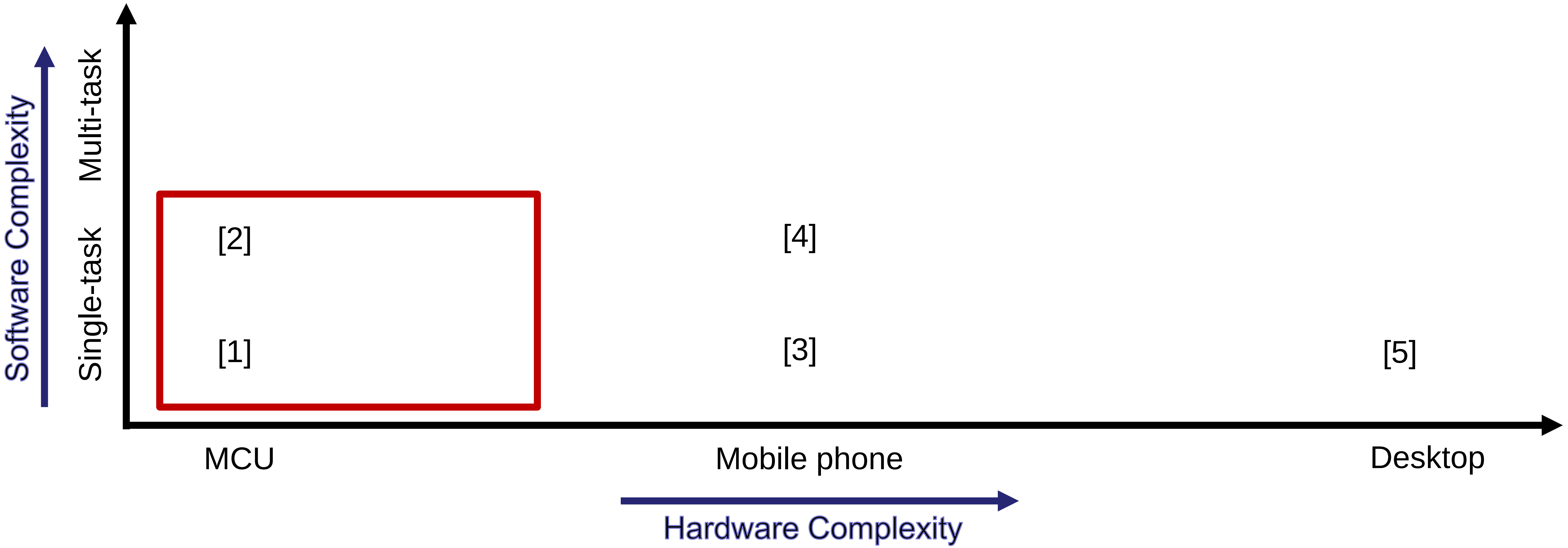
- ✓ **Intro**

- **SoK Taxonomies**
  - Detector context
  - ML pipelines
  - Attacks and datasets

- **Discussion**
  - Research gaps & takeaways

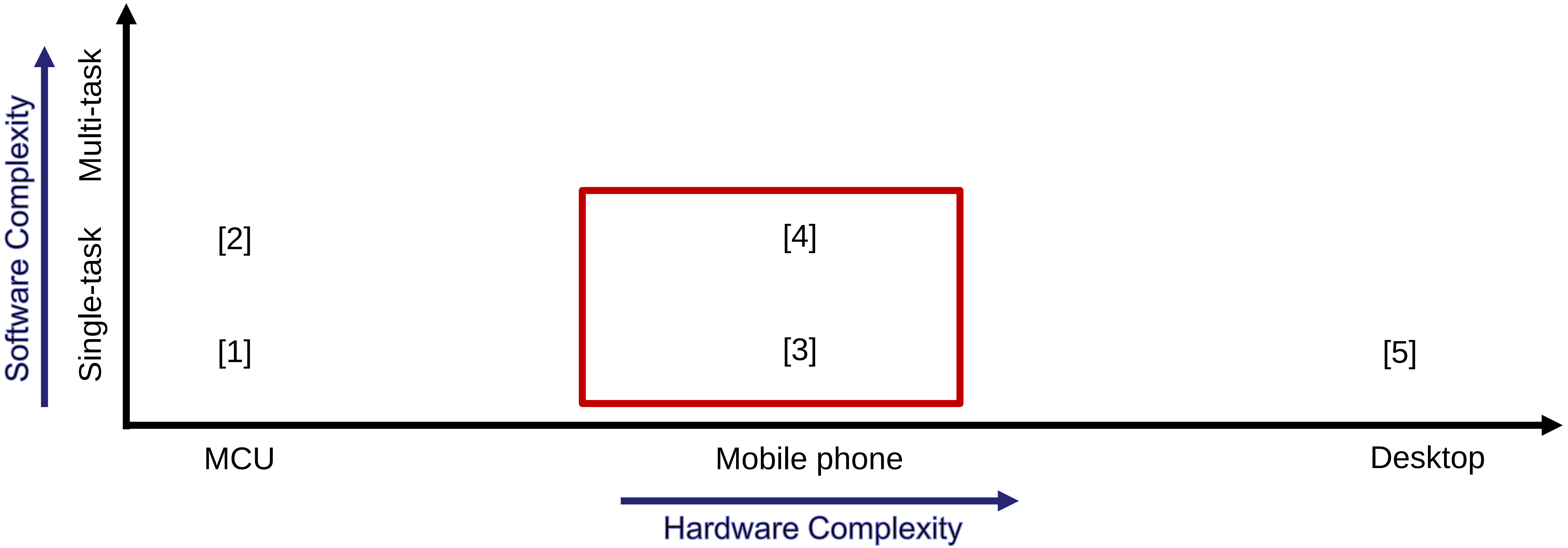
- **Summary, Conclusions and Future Work**

Detector Context



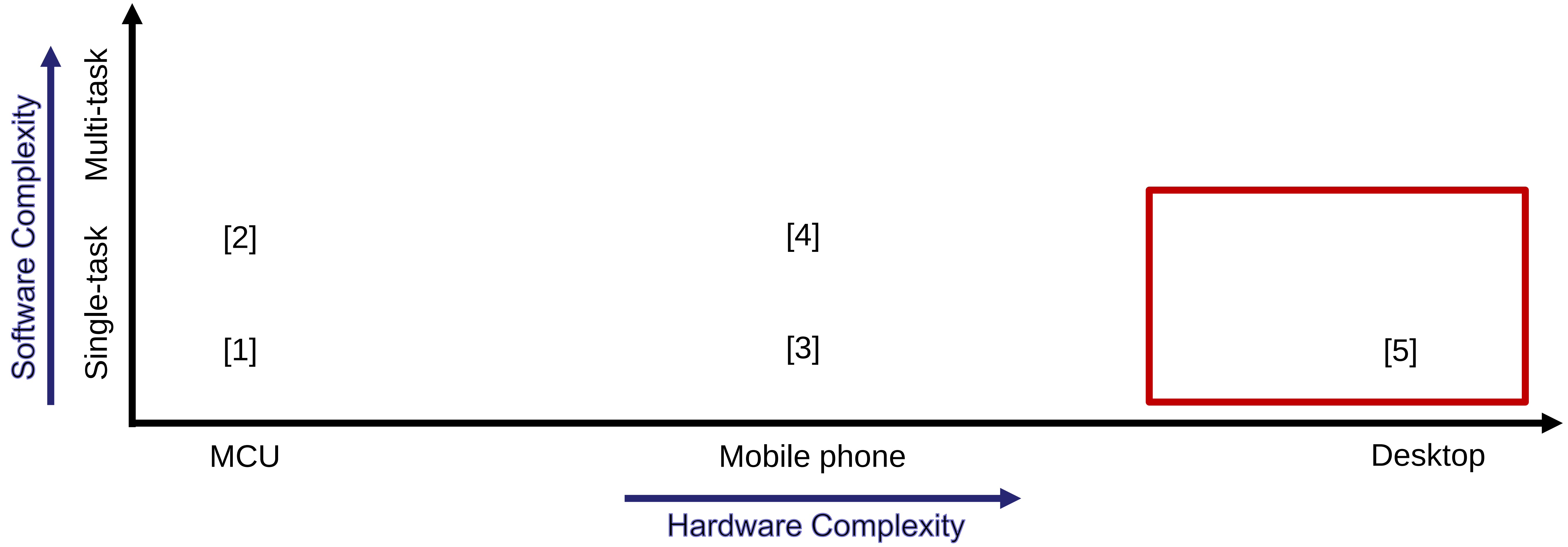
[1] Wattsupdoc: Power Side Channels to Nonintrusively Discover Untargeted Malware on Embedded Medical Devices. Clark et al . HealthTec'13
[2] Power-Based Non-Intrusive Condition Monitoring for Terminal Device in Smart Grid. Zhang et al. Sensors'20
[3] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems. Wei et al. HOST'19
[4] Detecting Energy-Greedy Anomalies and Mobile Malware Variants. Kim et al. Mobisys'08
[5] Towards Malware Detection via CPU Power Consumption: Data Collection Design and Analytics. Bridges et al. TrustCom/BigDataSE'18

Detector Context



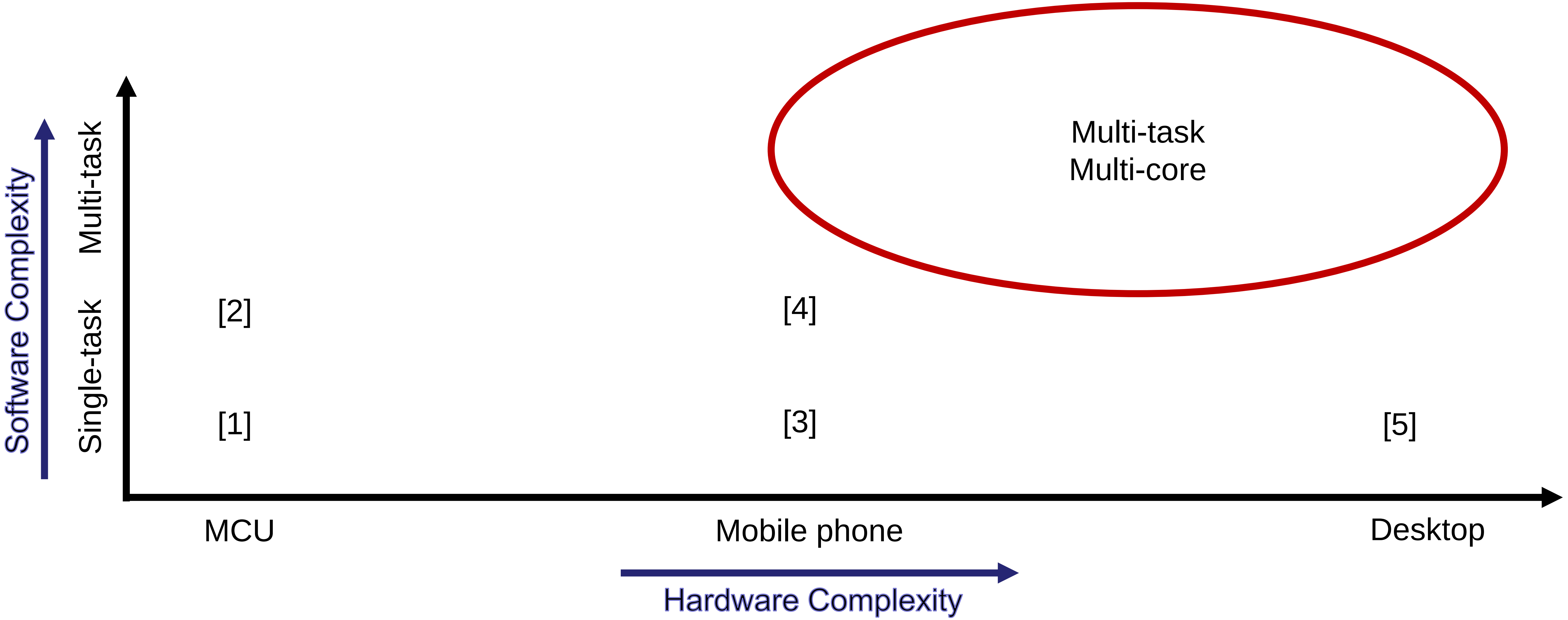
[1] Wattsupdoc: Power Side Channels to Nonintrusively Discover Untargeted Malware on Embedded Medical Devices. Clark et al . HealthTec'13
[2] Power-Based Non-Intrusive Condition Monitoring for Terminal Device in Smart Grid. Zhang et al. Sensors'20
[3] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems. Wei et al. HOST'19
[4] Detecting Energy-Greedy Anomalies and Mobile Malware Variants. Kim et al. Mobisys'08
[5] Towards Malware Detection via CPU Power Consumption: Data Collection Design and Analytics. Bridges et al. TrustCom/BigDataSE'18

Detector Context



[1] Wattsupdoc: Power Side Channels to Nonintrusively Discover Untargeted Malware on Embedded Medical Devices. Clark et al . HealthTec'13
[2] Power-Based Non-Intrusive Condition Monitoring for Terminal Device in Smart Grid. Zhang et al. Sensors'20
[3] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems. Wei et al. HOST'19
[4] Detecting Energy-Greedy Anomalies and Mobile Malware Variants. Kim et al. Mobisys'08
[5] Towards Malware Detection via CPU Power Consumption: Data Collection Design and Analytics. Bridges et al. TrustCom/BigDataSE'18

Detector Context



[1] Wattsupdoc: Power Side Channels to Nonintrusively Discover Untargeted Malware on Embedded Medical Devices. Clark et al . HealthTec'13
[2] Power-Based Non-Intrusive Condition Monitoring for Terminal Device in Smart Grid. Zhang et al. Sensors'20
[3] Using Power-Anomalies to Counter Evasive Micro-Architectural Attacks in Embedded Systems. Wei et al. HOST'19
[4] Detecting Energy-Greedy Anomalies and Mobile Malware Variants. Kim et al. Mobisys'08
[5] Towards Malware Detection via CPU Power Consumption: Data Collection Design and Analytics. Bridges et al. TrustCom/BigDataSE'18

Detector Context Takeaways

- **For multi-core systems, must consider all states**
  - Exponential number of states
  - Malware can execute in parallel to benign tasks
- **Must distinguish all benign from all infected states**
  - Benign state: only benign tasks executing
  - Infected state: at least one malware task

Detector Context Takeaways

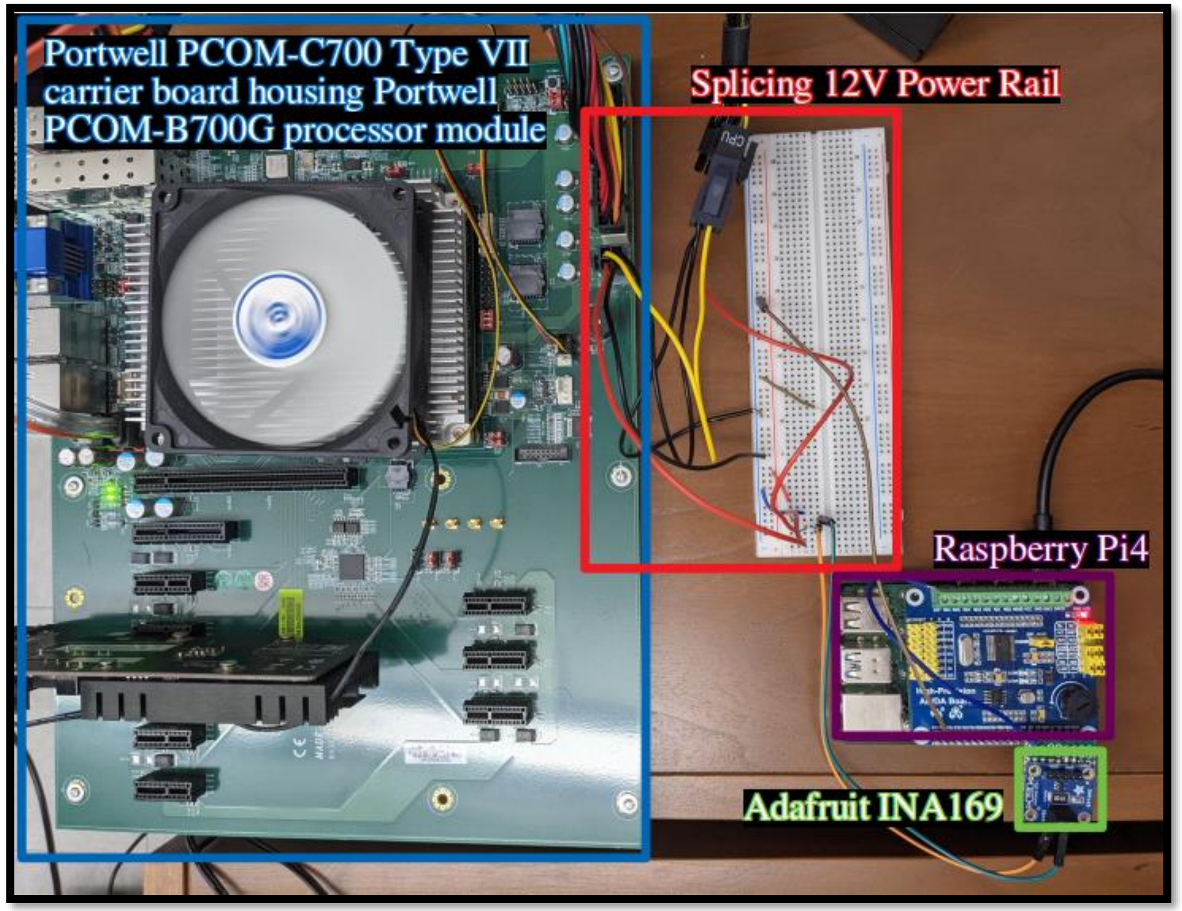
- **For multi-core systems, must consider all states**
  - Exponential number of states
  - Malware can execute in parallel to benign tasks

- **Must distinguish all benign from all infected states**
  - Benign state: only benign tasks executing
  - Infected state: at least one malware task

Research Gap: Lack of evaluation on parallel task sets

Experimental Setup

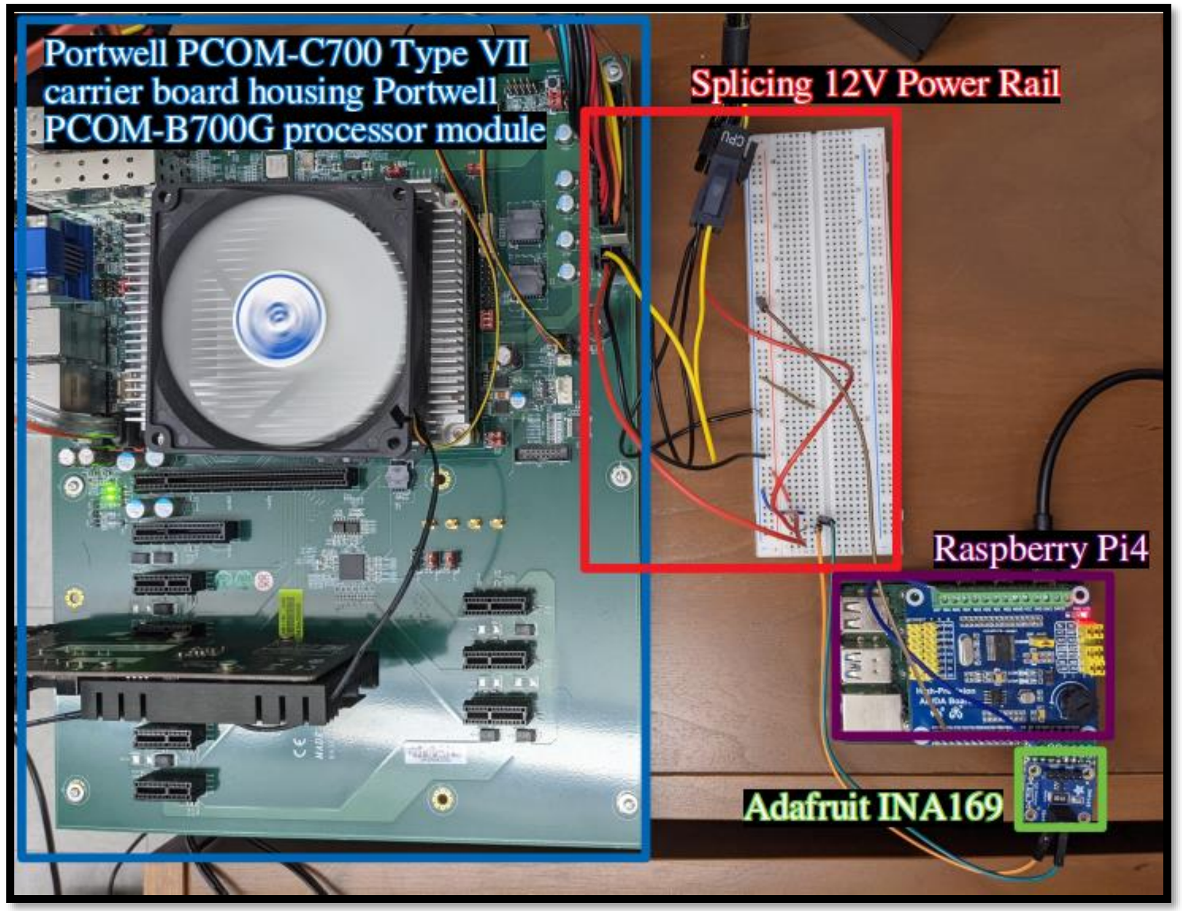
Target Device	Portwell PCOM-C700 Type VII carrier board
	Portwell PCOM-B700G processor module
	8-core Intel Xeon D-1539 embedded class processor
Power Sampling	Spliced 12V CPU power rail, sampled at 2KHz
	Adafruit INA169 analog current sensor
Detector	Deployed on Raspberry Pi4
	Python implementation achieves 27 inferences per second



Experimental Setup

Target Device	Portwell PCOM-C700 Type VII carrier board
	Portwell PCOM-B700G processor module
	8-core Intel Xeon D-1539 embedded class processor
Power Sampling	Spliced 12V CPU power rail, sampled at 2KHz
	Adafruit INA169 analog current sensor
Detector	Deployed on Raspberry Pi4
	Python implementation achieves 27 inferences per second

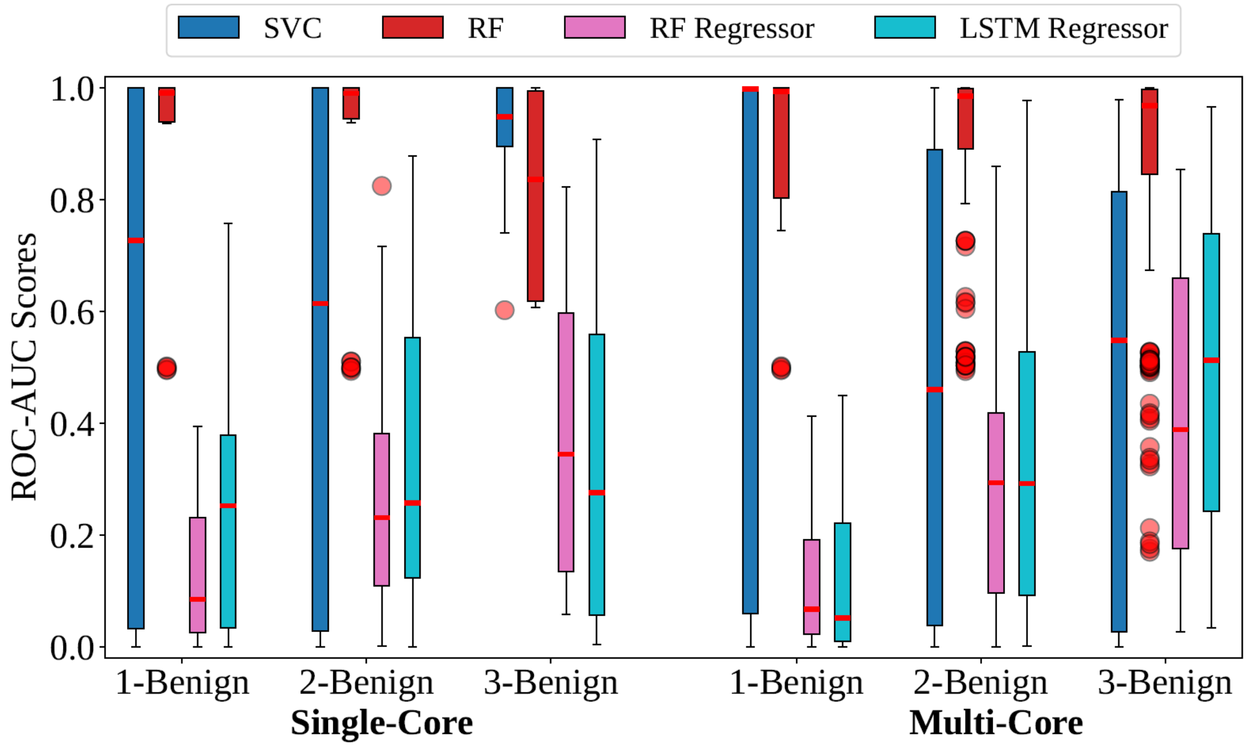
Features	For regression-based detectors, input window was size 1000 and prediction window 3
	For other ML formulations, each sliding window was transformed into a feature vector
	Feature vector consisted of statistical, and bag-of-words features
Prior Works	**Replicated representative works for various ML formulations**
	Non-ensemble formulations include: one-class classification, binary classification, multiclass classification, ensemble of one-class classifiers, regression, statistical tests
	Mix of non-deep and deep methods evaluated
	[Bridges'18, Caviglione'15, Dixon'14, Jiminez'19, Liu'09, Luckett'18, Wang'18, Wei'19]
Benchmarks	**Benign applications representing drone tasks**; SHA-3, face detection, autonomous drone path-finding
	3 Microarchitectural attacks; Meltdown, Spectre, and L1 Cache covert-channel



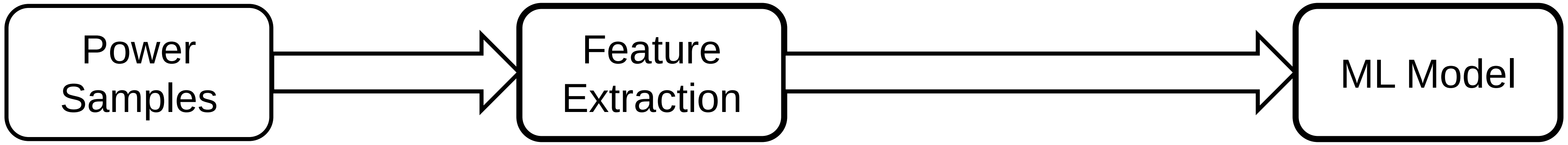
Detector Context Evaluation

- **Characterize operating range**
  - 3 applications
  - 8 benign states
  - 64 comparisons

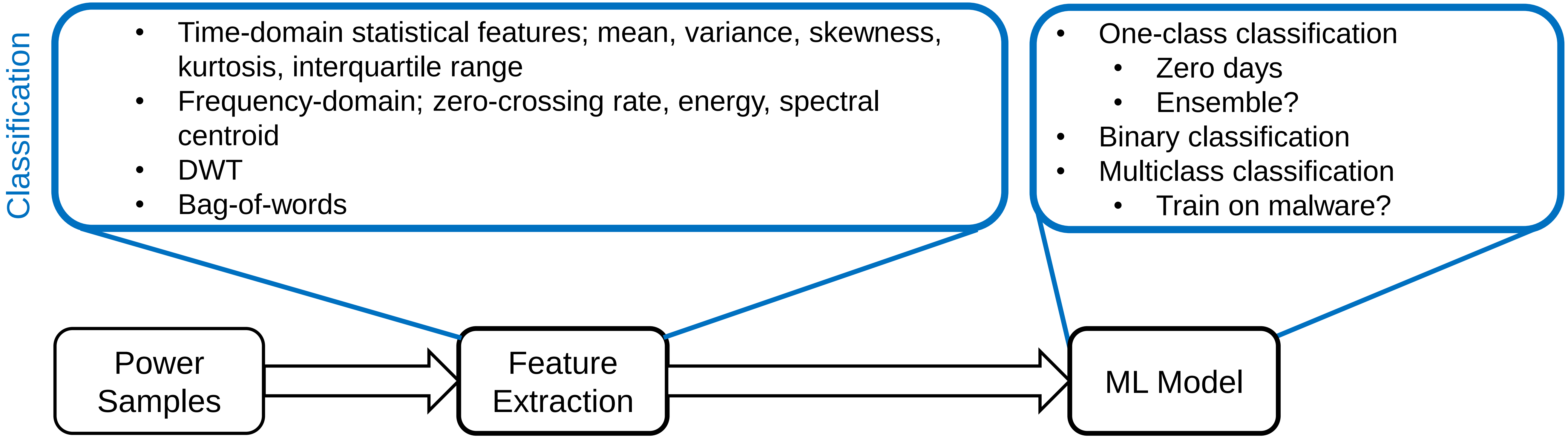
- **Prior work underperforms**
  - Perform poorly in parallel settings
  - Suffer even in single-core context



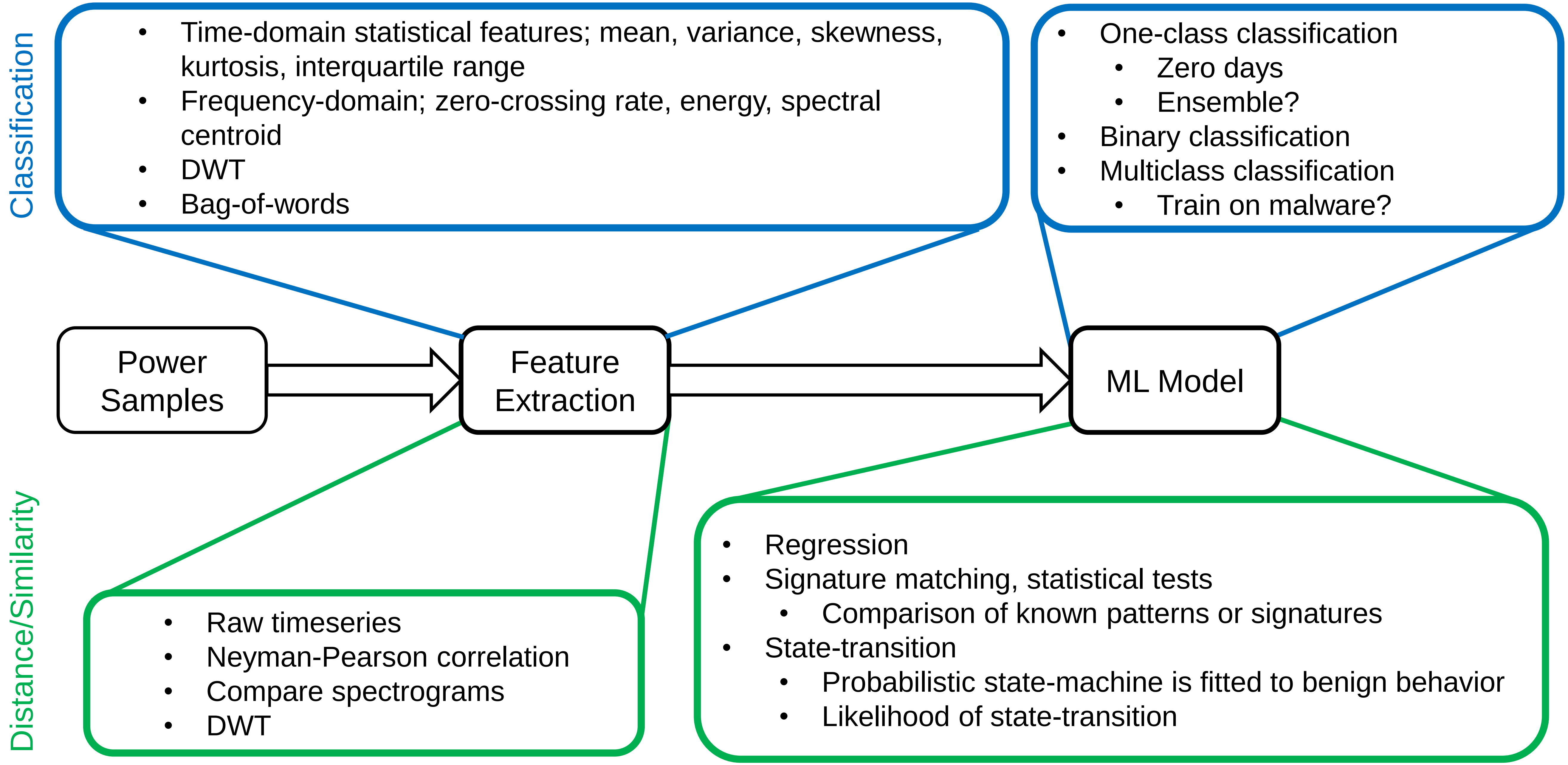
Detector ML Pipelines



Detector ML Pipelines



Detector ML Pipelines



Detector ML Pipelines Takeaways

- **Train on malware with assumption that it is representative**
  - Binary or multi-class classification
- **Regression error as proxy for maliciousness**
  - Time series forecasting
- **Classification confidence as proxy for maliciousness**
  - Multi-class classification

Detector ML Pipelines Takeaways

- **Train on malware with assumption that it is representative**
  - Binary or multi-class classification

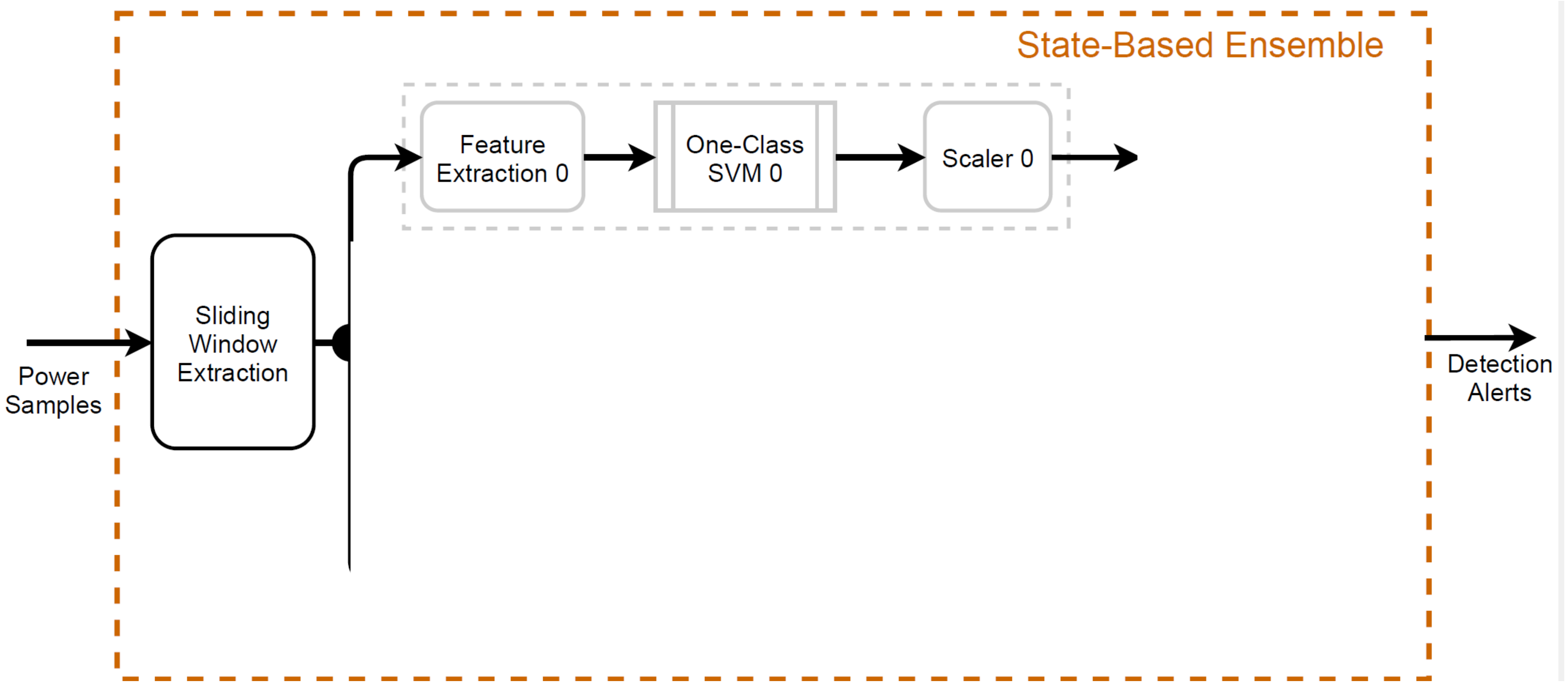
- **Regression error as proxy for maliciousness**
  - Time series forecasting

- **Classification confidence as proxy for maliciousness**
  - Multi-class classification

Research Gap: Inappropriate utilization of ML formulations

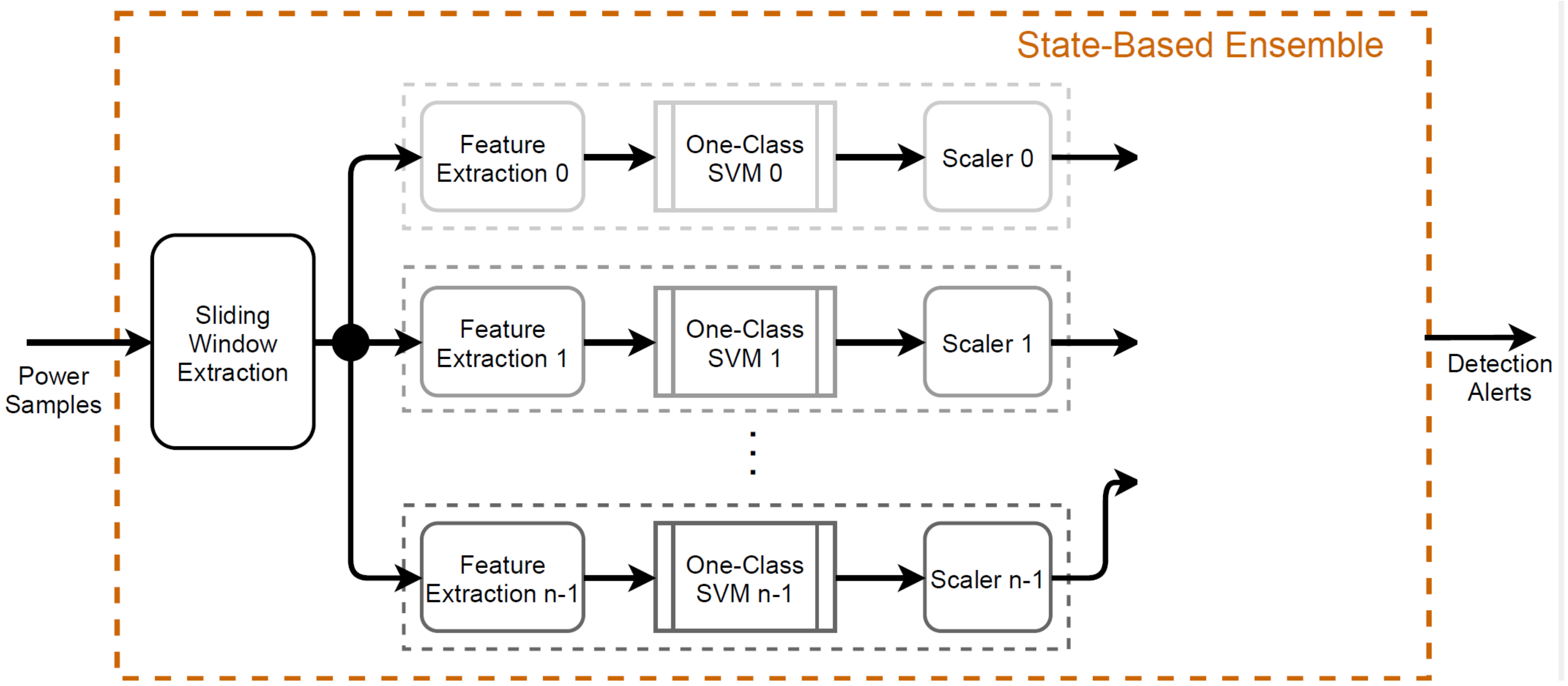
Proposed State-Based One-Class Ensemble

- **State awareness**
  - Any unique combination of executing tasks presents an operating state
  - One-class classifier for each state



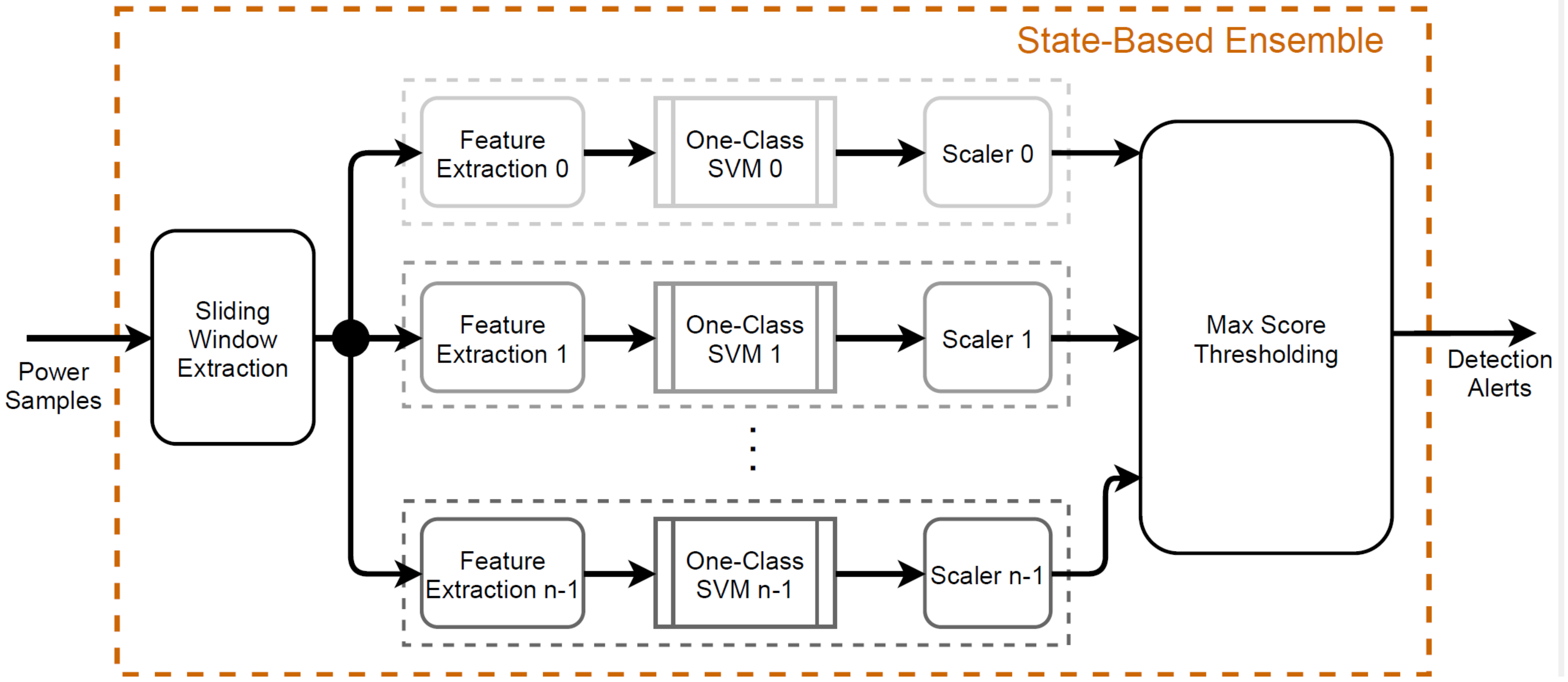
Proposed State-Based One-Class Ensemble

- **Scaling to parallel task sets**
  - With more tasks, add more one-class pipelines
  - Combine one-class detection results (max/or)



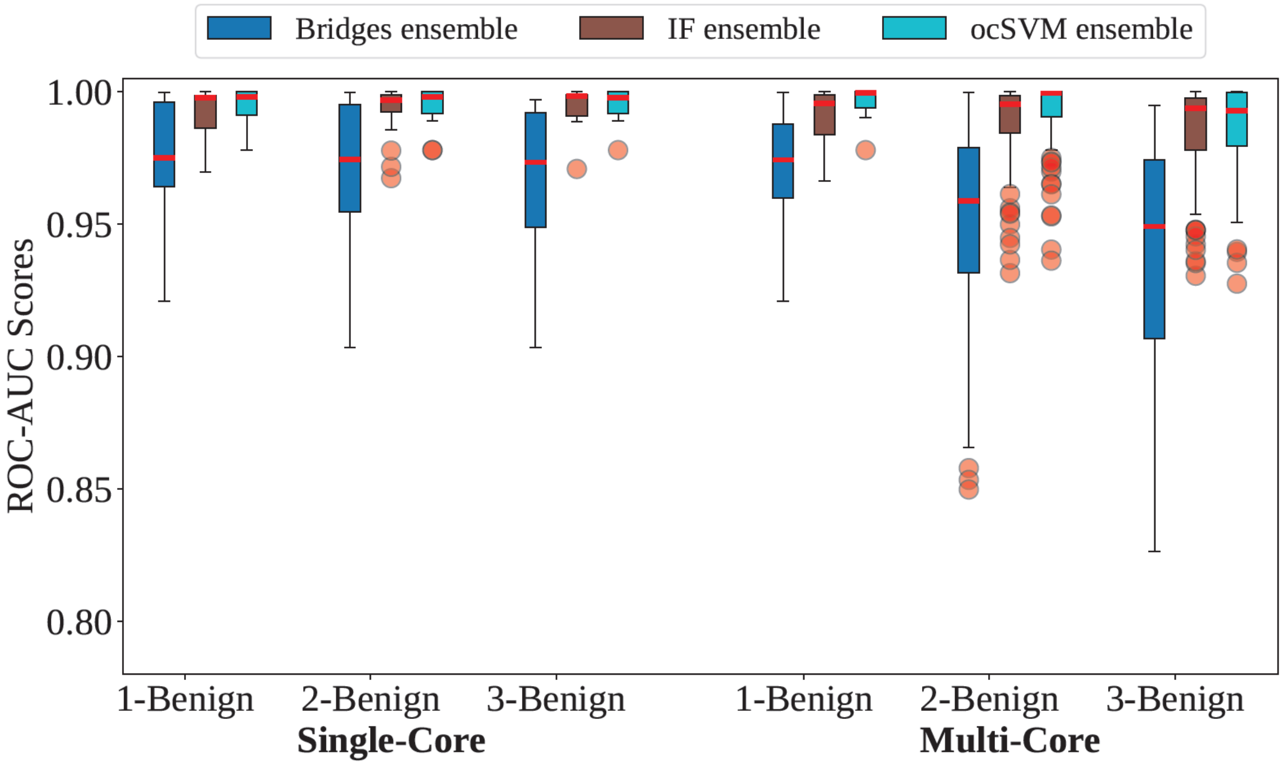
Proposed State-Based One-Class Ensemble

- **Scaling to parallel task sets**
  - With more tasks, add more one-class pipelines
  - Combine one-class detection results (max/or)



Detector ML Pipeline Evaluation

- **Ensemble outperforms prior work**
  - Including prior single-task ensembles
- **Ensemble still has limitations**
  - NOP insert, low-power, power-mimicry
  - Noise
  - Power cannot detect everything



Attacks and Datasets

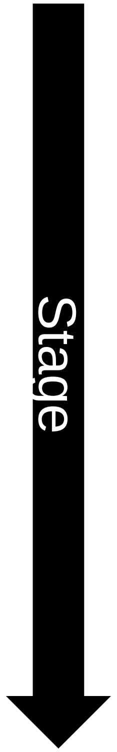
- **MITRE ATT&CK matrix**

Stage →

Reconnaissance (10 techniques)	Resource Development (8 techniques)	Initial Access (10 techniques)	Execution (14 techniques)	Persistence (20 techniques)	Privilege Escalation (14 techniques)	Defense Evasion (43 techniques)	Credential Access (17 techniques)	Discovery (32 techniques)	Lateral Movement (9 techniques)	Collection (17 techniques)	Command and Control (18 techniques)	Exfiltration (9 techniques)	Impact (14 techniques)
Active Scanning (3)	Acquire Access	Content Injection	Cloud Administration Command	Account Manipulation (6)	Abuse Elevation Control Mechanism (6)	Abuse Elevation Control Mechanism (6)	Adversary-in-the-Middle (3)	Account Discovery (4)	Exploitation of Remote Services	Adversary-in-the-Middle (3)	Application Layer Protocol (4)	Automated Exfiltration (1)	Account Access Removal
Gather Victim Host Information (4)	Acquire Infrastructure (8)	Drive-by Compromise	Command and Scripting Interpreter (10)	BITS Jobs	Access Token Manipulation (5)	Access Token Manipulation (5)	Brute Force (4)	Application Window Discovery	Internal Spearphishing	Archive Collected Data (3)	Communication Through Removable Media	Data Transfer Size Limits	Data Destruction
Gather Victim Identity Information (3)	Compromise Accounts (3)	Exploit Public-Facing Application	Container Administration Command	Boot or Logon Autostart Execution (14)	Account Manipulation (6)	BITS Jobs	Credentials from Password Stores (6)	Browser Information Discovery	Lateral Tool Transfer	Audio Capture	Content Injection	Exfiltration Over Alternative Protocol (3)	Data Encrypted for Impact
Gather Victim Network Information (6)	Compromise Infrastructure (8)	External Remote Services	Deploy Container	Boot or Logon Initialization Scripts (5)	Boot or Logon Autostart Execution (14)	Build Image on Host	Exploitation for Credential Access	Cloud Infrastructure Discovery	Remote Service Session Hijacking (2)	Automated Collection	Data Encoding (2)	Exfiltration Over C2 Channel	Data Manipulation (3)
Gather Victim Org Information (4)	Develop Capabilities (4)	Hardware Additions	Exploitation for Client Execution	Browser Extensions	Boot or Logon Initialization Scripts (5)	Debugger Evasion	Forced Authentication	Cloud Service Dashboard	Remote Services (8)	Browser Session Hijacking	Data Obfuscation (3)	Exfiltration Over Other Network Medium (1)	Defacement (2)
Phishing for Information (4)	Establish Accounts (3)	Phishing (4)	Inter-Process Communication (3)	Compromise Host Software Binary	Create or Modify System Process (5)	Deobfuscate/ Decode Files or Information	Forge Web Credentials (2)	Cloud Service Discovery	Replication Through Removable Media	Clipboard Data	Dynamic Resolution (3)	Exfiltration Over Physical Medium (1)	Disk Wipe (2)
Search Closed Sources (2)	Obtain Capabilities (7)	Replication Through Removable Media	Native API	Create Account (3)	Domain or Tenant Policy Modification (2)	Deploy Container	Input Capture (4)	Cloud Storage Object Discovery	Software Deployment Tools	Data from Cloud Storage	Encrypted Channel (2)	Exfiltration Over Web Service (4)	Endpoint Denial of Service (4)
Search Open Technical Databases (5)	Stage Capabilities (6)	Supply Chain Compromise (3)	Scheduled Task/ Job (5)	Create or Modify System Process (5)	Escape to Host	Direct Volume Access	Modify Authentication Process (9)	Container and Resource Discovery	Taint Shared Content	Data from Configuration Repository (2)	Fallback Channels	Scheduled Transfer	Financial Theft
Search Open Websites/ Domains (3)		Trusted Relationship	Serverless Execution	Event Triggered Execution (16)	Event Triggered Execution (16)	Domain or Tenant Policy Modification (2)	Multi-Factor Authentication Interception	Debugger Evasion	Use Alternate Authentication Material (4)	Data from Information Repositories (3)	Hide Infrastructure	Transfer Data to Cloud Account	Firmware Corruption
Search Victim-Owned Websites		Valid Accounts (4)	Shared Modules	External Remote Services	Exploitation for Privilege Escalation	Execution Guardrails (1)	Multi-Factor Authentication Request Generation	Device Driver Discovery		Data from Local System	Ingress Tool Transfer		Inhibit System Recovery
			Software Deployment Tools	Hijack Execution		Exploitation for Defense Evasion				Data from Network Shared Drive	Multi-Stage Channels		Network Denial of Service (2)
			System Services (2)			File and Directory					Non-Application		Resource Hijacking
													Service Stop

Attacks and Datasets

- **Heavy emphasis on execution or impact stage**
  - Easiest to detect
- **Proprietary experimental setup**
  - Reproducibility



Stage	Instance/Family	Papers
Initial Access	Replay Attack	[16]
Discovery/ Resource Development	Botnet	[38]
Execution	Code Modification	[2, 16, 42]
	Control Flow Hijack	[30, 33]
	Cause Spam	[11, 13]
	Virus	[22]
	Microarchitecture Attacks	[39, 43]
	Evasive μ-Arch Attacks	[39]
	Covert-Channels	[9, 39]
Persistence/ Defence Evasion	Rootkit	[8, 13, 42], [12, 22, 31]
	Backdoor	[22]
Lateral Movement	Worm	[22, 24, 29]
Collection/ Exfiltration/ Impact	DDOS	[16]
	Ransomware	[18, 22]
	Spyware	[11, 29]
	Battery Depletion/ Electrical Theft	[6, 24]
	Data Deletion	[18]
Other	Fabricated Virus	[3, 20]

Attacks and Datasets

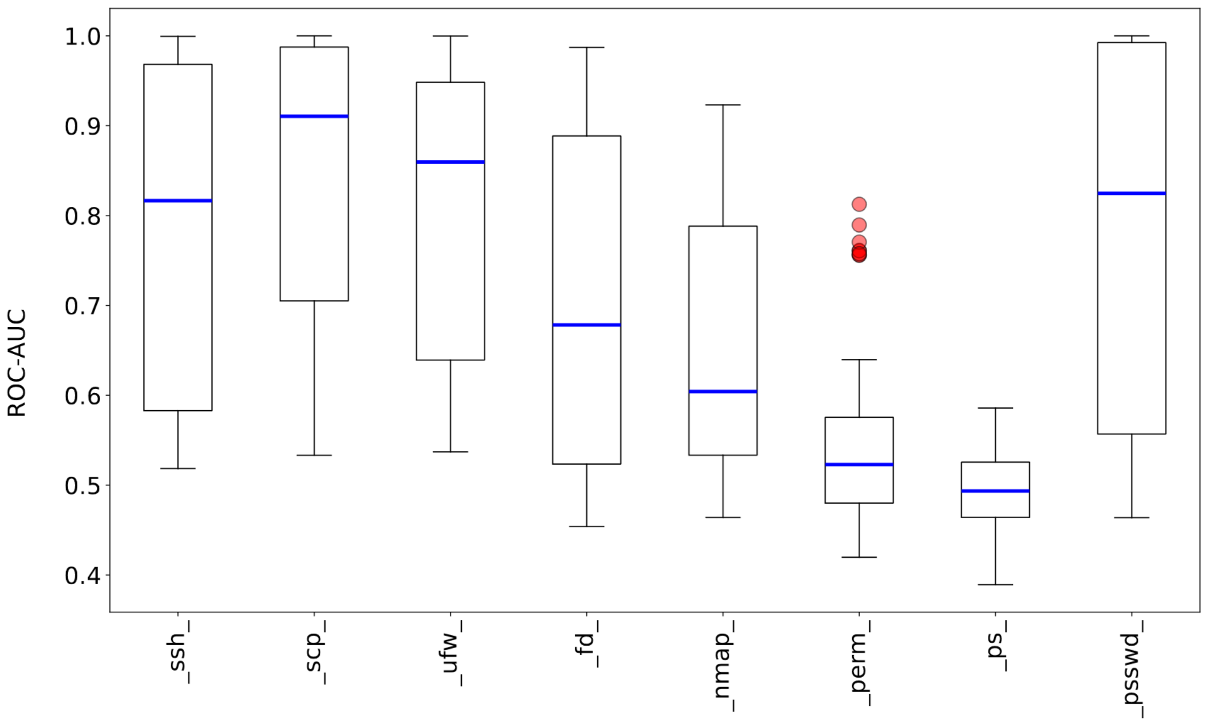
- **Heavy emphasis on execution or impact stage**
  - Easiest to detect
- **Proprietary experimental setup**
  - Reproducibility

Stage

Stage	Instance/Family	Papers
Initial Access	Replay Attack	[16]
Discovery/ Resource Development	Botnet	[38]
Execution	Code Modification	[2, 16, 42]
	Control Flow Hijack	[30, 33]
	Cause Spam	[11, 13]
	Virus	[22]
	Microarchitecture Attacks	[39, 43]
	Evasive μ-Arch Attacks	[39]
	Covert-Channels	[9, 39]
Persistence/ Defence Evasion	Rootkit	[8, 13, 42], [12, 22, 31]
	Backdoor	[22]
Lateral Movement	Worm	[22, 24, 29]
Collection/ Exfiltration/ Impact	DDOS	[16]
	Ransomware	[18, 22]
	Spyware	[11, 29]
	Battery Depletion/ Electrical Theft	[6, 24]
	Data Deletion	[18]
Other	Fabricated Virus	[3, 20]

Attack and Dataset Evaluation

- **Evaluate against other attack stages**
  - Initial access, discovery, lateral movement
  - Cannot expect reliable detections
- **Operating range of detectors**
  - Need to look at worst-case



Attacks and Datasets Takeaways

- **Most focus on easy-to-detect stages of MITRE matrix**
  - Exploitation and impact
- **No established public datasets**
  - No released power traces

Attacks and Datasets Takeaways

- **Most focus on easy-to-detect stages of MITRE matrix**
  - Exploitation and impact

- **No established public datasets**
  - No released power traces

https://github.com/SLAM-Lab/PMD-Dataset

Research Gap: Lack of comprehensive public datasets

Discussion

- **Lack of evaluation on parallel task sets**
  - Multi-core poses new challenges
  - Must evaluate each benign and infected state
- **Inappropriate utilization of ML tools**
  - Detection significantly hinges on formulation
  - Preprocessing is crucial
- **Lack of rigorous public datasets**
  - Understanding detector limits is more important than showing successes

Deployment Suggestions

- **Limit number of benign tasks**
- **Worst case can be much worse than average**
- **Deep model is not a crutch for missing domain expertise**
- **Detector not tested against software-exploiting attacks**

Summary, Conclusions and Future Work

- **Systemization of power side-channel based malware detection**
  - Detector context, ML pipelines, attacks & datasets
- **Identify and address research gaps**
  - Multi-task multi-core evaluation
  - Proposed state-based ensemble detector
  - Public release of dataset

https://github.com/SLAM-Lab/PMD-Dataset

Summary, Conclusions and Future Work

- **Systemization of power side-channel based malware detection**
  - Detector context, ML pipelines, attacks & datasets
- **Identify and address research gaps**
  - Multi-task multi-core evaluation
  - Proposed state-based ensemble detector
  - Public release of dataset
- **Future work**
  - Further characterization of operating range
  - Alternative approaches for more complex detection scenarios
    - Heterogeneous hardware platforms, software-based attacks, power-mimicking malware

https://github.com/SLAM-Lab/PMD-Dataset

Questions

Thank you!

alexander.cathis@utexas.edu

https://github.com/SLAM-Lab/PMD-Dataset