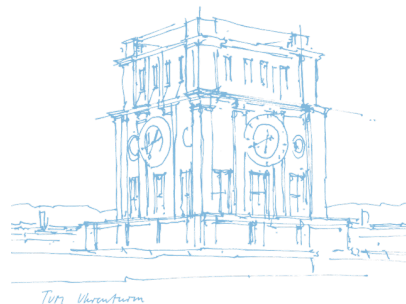
Chair of Network Architectures and Services
School of Computation, Information and Technology
Technical University of Munich

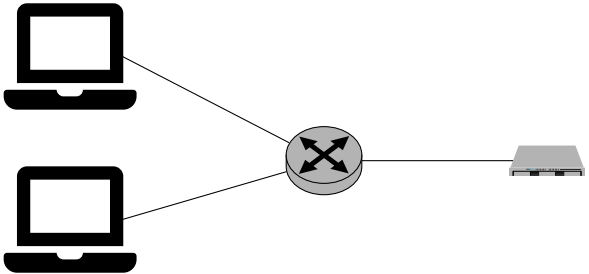
# Securing Networked Systems by Identifying and Monitoring Multi-Layer Dependencies

**Lars Wüstrich**

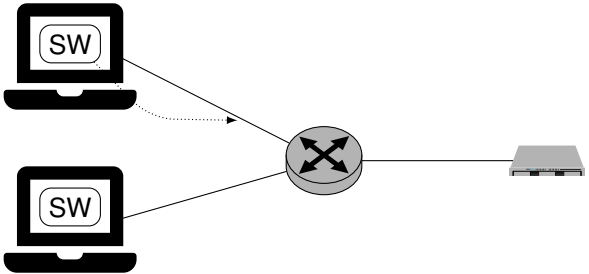
Chair of Network Architectures and Services
School of Computation, Information and Technology
Technical University of Munich



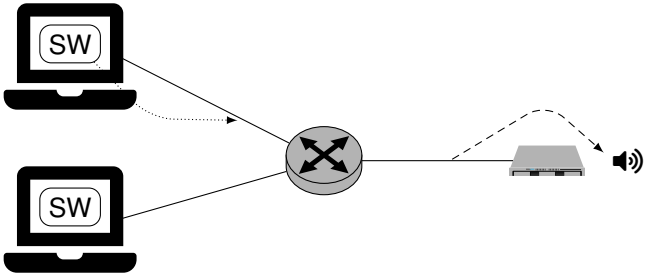
# Motivation



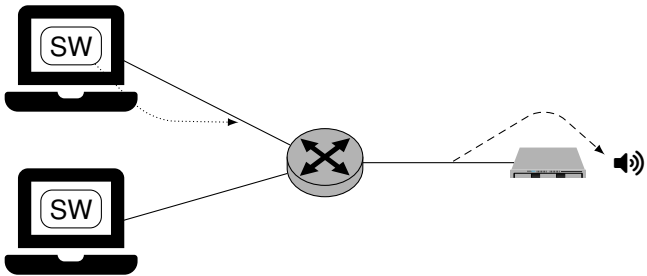
# Motivation



# Motivation



# Motivation




How can we improve the security in distributed systems
by identifying and monitoring multi-layer relationships?

Motivation

Research Questions

Network-Application Relationship

- Associating Application and Network Activity

Cyber-Physical Relationships

- Correlating Network and Physical Activity
- Monitoring Physical Device Behavior in Composite Side Channels

Leveraging Multi-Layer Relationships for Security

Contributions

# Research Questions

How can we improve the security in distributed systems by identifying and monitoring multi-layer relationships?

**Background on multi-layer relationships**

- Causes of multi-layer relationships
- Approaches to leverage multi-layer relationships
- Security implications of multi-layer relationships

**What relationship exists between software and host network activity?**

- Efficiently and reliably associate network activity with applications
- Modeling network activity of applications for anomaly detection

**What relationship exists between control traffic and physical activity?**

- Side-channel monitoring in composite signals
- Modeling cyber-physical relationships
- Identifying device activity in composite signals

# Research Questions

How can we improve the security in distributed systems by identifying and monitoring multi-layer relationships?

**Background on multi-layer relationships**

- Causes of multi-layer relationships
- Approaches to leverage multi-layer relationships
- Security implications of multi-layer relationships

**What relationship exists between software and host network activity?**

- Efficiently and reliably associate network activity with applications
- Modeling network activity of applications for anomaly detection

**What relationship exists between control traffic and physical activity?**

- Side-channel monitoring in composite signals
- Modeling cyber-physical relationships
- Identifying device activity in composite signals

# Research Questions

How can we improve the security in distributed systems by identifying and monitoring multi-layer relationships?

**Background on multi-layer relationships**

- Causes of multi-layer relationships
- Approaches to leverage multi-layer relationships
- Security implications of multi-layer relationships

**What relationship exists between software and host network activity?**

- Efficiently and reliably associate network activity with applications
- Modeling network activity of applications for anomaly detection

**What relationship exists between control traffic and physical activity?**

- Identifying device activity in composite signals
- Modeling cyber-physical relationships
- Side-channel monitoring in composite signals

# Research Questions

How can we improve the security in distributed systems by identifying and monitoring multi-layer relationships?

**Background on multi-layer relationships**

- Causes of multi-layer relationships
- Approaches to leverage multi-layer relationships
- Security implications of multi-layer relationships

Chapter 2
Chapter 3
Chapter 4

**What relationship exists between software and host network activity?**

- Efficiently and reliably associate network activity with applications
- Modeling network activity of applications for anomaly detection

Chapter 5
Chapter 6

**What relationship exists between control traffic and physical activity?**

- Identifying device activity in composite signals
- Modeling cyber-physical relationships
- Side-channel monitoring in composite signals

Chapter 7
Chapter 8
Chapter 9

# Research Questions

How can we improve the security in distributed systems by identifying and monitoring multi-layer relationships?

**Background on multi-layer relationships**

- Causes of multi-layer relationships
- Approaches to leverage multi-layer relationships
- Security implications of multi-layer relationships

Chapter 2
Chapter 3
Chapter 4

**What relationship exists between software and host network activity?**

- Efficiently and reliably associate network activity with applications
- Modeling network activity of applications for anomaly detection

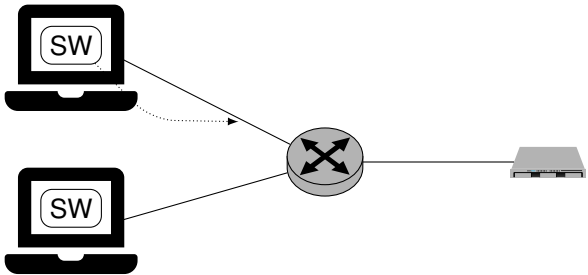
Chapter 5
Chapter 6

**What relationship exists between control traffic and physical activity?**

- Identifying device activity in composite signals
- Modeling cyber-physical relationships
- Side-channel monitoring in composite signals

Chapter 7
Chapter 8
Chapter 9

## Part II: What relationship exists between software and host network activity?



## RQ2.1 How can we reliably attribute network activity to applications?

***Wüstrich**, Schacherbauer, Künßberg, Gallenmüller, Pahl, Carle, "Network Profiles for Detecting Application-Characteristic Behavior Using Linux eBPF", eBPF@SIGCOMM, 2023*

- Overcoming the gap between the application layer and the network layer is challenging
- Existing approaches are often:
  - Incomplete
  - Unreliable
  - Inefficient

# Associating Application and Network Activity

## State of the Art

| | Heuristics | Polling | Logging | eBPF |
|---|---|---|---|---|
| Complete | | | | |
| Reliable | | | | |
| Efficient | | | | |
| Network-Application | | | | |

1
2
3
4
5
6
7
8

# Associating Application and Network Activity

## State of the Art

| | Heuristics | Polling | Logging | eBPF |
|---|---|---|---|---|
| | Asai[1] | | | |
| Complete | ✓ | | | |
| Reliable | ✗ | | | |
| Efficient | ✓ | | | |
| Network-Application | ✗ | | | |

[1] H. Asai, et al. "Network application profiling with traffic causality graphs." International Journal of Network Management, 2014

# Associating Application and Network Activity

## State of the Art

| | Heuristics | Polling | | Logging | eBPF |
|---|---|---|---|---|---|
| | Asai[1] | Haas[2] | Popa[3] | | |
| Complete | ✓ | ✗ | ✗ | | |
| Reliable | ✗ | ✓ | ✓ | | |
| Efficient | ✓ | ✗ | ✗ | | |
| Network-Application | ✗ | ✓ | ✓ | | |

[1] H. Asai, et al. "Network application profiling with traffic causality graphs." International Journal of Network Management, 2014
[2] S. Haas et al. Zeek-osquery: Host-network correlation for advanced monitoring and intrusion detection., IFIP TC 11, 2020
[3] L. Popa et al. Macroscope: End-point approach to networked application dependency discovery. CoNEXT, 2009

# Associating Application and Network Activity

## State of the Art

| | Heuristics | Polling | | Logging | eBPF |
|---|---|---|---|---|---|
| | Asai[1] | Haas[2] | Popa[3] | Ma[4] | |
| Complete | ✓ | ✗ | ✗ | ✓ | |
| Reliable | ✗ | ✓ | ✓ | ✓ | |
| Efficient | ✓ | ✗ | ✗ | ✗ | |
| Network-Application | ✗ | ✓ | ✓ | ✗ | |

[1] H. Asai, et al. "Network application profiling with traffic causality graphs." International Journal of Network Management, 2014
[2] S. Haas et al. Zeek-osquery: Host-network correlation for advanced monitoring and intrusion detection., IFIP TC 11, 2020
[3] L. Popa et al. Macroscope: End-point approach to networked application dependency discovery. CoNEXT, 2009
[4] S. Ma et al. Protracer: Towards practical provenance tracing by alternating between logging and tainting. NDSS, 2016

# Associating Application and Network Activity

## State of the Art

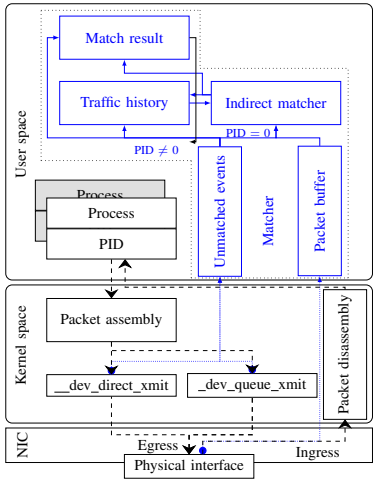
| | Heuristics | Polling | | Logging | eBPF | | | |
|---|---|---|---|---|---|---|---|---|
| | Asai[1] | Haas[2] | Popa[3] | Ma[4] | Sekar[5] | Tetragon[6] | Falco[7] | Opensnitch[8] |
| Complete | ✓ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ |
| Reliable | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Efficient | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ | ✓ | ✓ |
| Network-Application | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ |

[1] H. Asai, et al. "Network application profiling with traffic causality graphs." International Journal of Network Management, 2014
[2] S. Haas et al. Zeek-osquery: Host-network correlation for advanced monitoring and intrusion detection., IFIP TC 11, 2020
[3] L. Popa et al. Macroscope: End-point approach to networked application dependency discovery. CoNEXT, 2009
[4] S. Ma et al. Protracer: Towards practical provenance tracing by alternating between logging and tainting. NDSS, 2016
[5] R. Sekar et al. eAUDIT: A Fast, Scalable and Deployable Audit Data Collection System, IEEE SP, 2024
[6] Cilium Tetragon. https://github.com/cilium/tetragon
[7] Falco. https://falco.org/
[8] Opensnitch. https://github.com/evilsocket/opensnitch

# Associating Application and Network Activity

## Using eBPF to Associate Applications and Packets

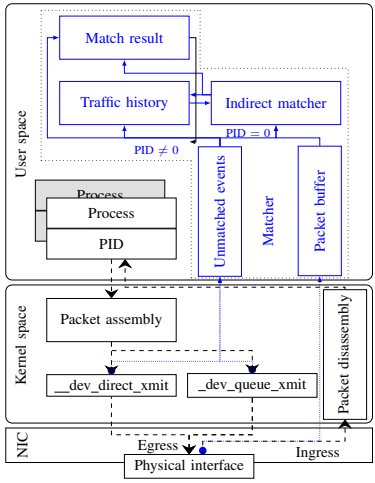
- Analyzed the kernel to select appropriate kernel functions to hook with eBPF
- Tradeoff
  - parsing overhead
  - data completeness



# Associating Application and Network Activity

## Using eBPF to Associate Applications and Packets

- Analyzed the kernel to select appropriate kernel functions to hook with eBPF
- Tradeoff
  - parsing overhead
  - data completeness
- All egress uses one of two kernel functions
  - `__dev_direct_xmit`, or
  - `__dev_queue_xmit`
- Ingress matching via heuristics

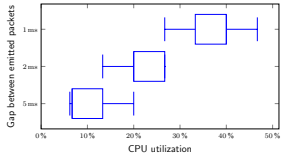


## Contributions

- A tool to associate applications and packets
- Matching rate:
  - 99.9 % at 60 Mbit/s
  - 96.3 % at 120 Mbit/s

# Associating Application and Network Activity

## Contributions

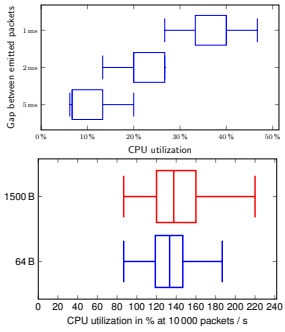



- A tool to associate applications and packets
- Matching rate:
  - 99.9 % at 60 Mbit/s
  - 96.3 % at 120 Mbit/s
- Overhead depends on network activity

# Associating Application and Network Activity

## Contributions

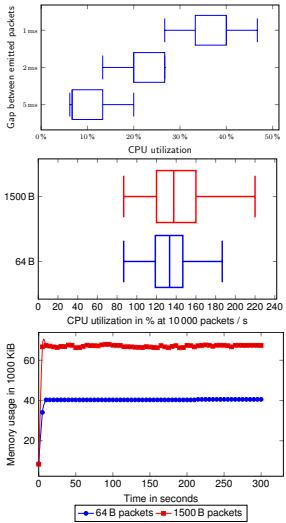
- A tool to associate applications and packets
- Matching rate:
  - 99.9 % at 60 Mbit/s
  - 96.3 % at 120 Mbit/s
- Overhead depends on network activity
- CPU usage is independent of packet size



# Associating Application and Network Activity

## Contributions

- A tool to associate applications and packets
- Matching rate:
  - 99.9 % at 60 Mbit/s
  - 96.3 % at 120 Mbit/s
- Overhead depends on network activity
- CPU usage is independent of packet size
- Memory usage depends on packet size



# Associating Application and Network Activity

## Contributions

| | eBPF | | | | |
|---|---|---|---|---|---|
| | Sekar[5] | Tetragon[6] | Falco[7] | Opensnitch[8] | **This thesis**[9] |
| Complete | ✓ | ✗ | ✗ | ✗ | ✓ |
| Reliable | ✓ | ✓ | ✓ | ✓ | ✓ |
| Efficient | ✓ | ✓ | ✓ | ✓ | ✓ |
| Network-Application | ✗ | ✗ | ✓ | ✓ | ✓ |

[5] R. Sekar et al. eAUDIT: A Fast, Scalable and Deployable Audit Data Collection System, IEEE SP, 2024

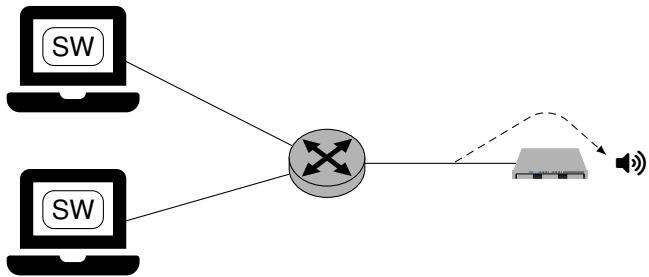
[6] Cilium Tetragon. https://github.com/cilium/tetragon

[7] Falco. https://falco.org/

[8] Opensnitch. https://github.com/evilsocket/opensnitch

[9] **Wüstrich**, Schacherbauer, Künßberg, Gallenmüller, Pahl, Carle, "Network Profiles for Detecting Application-Characteristic Behavior Using Linux eBPF", eBPF@SIGCOMM, 2023

## Part III: What relationship exists between control traffic and physical activity?



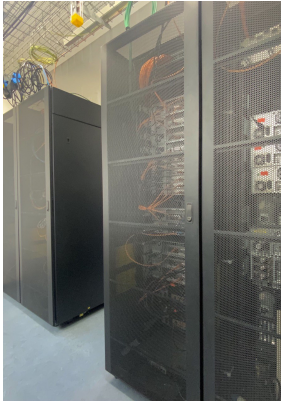
# Correlating Network and Physical Activity

## A Cyber-Physical (DC-)Model

### RQ3.2 What is the relation between network control traffic and physical device activity?

***Wüstrich**, Schröder, Pahl, "Cyber-Physical Anomaly Detection for ICS", manage-IoT@IM 2021*
***Wüstrich**, Gallenmüller, Günther, Carle, Pahl, "Shells Bells: Cyber-Physical Anomaly Detection in Data Centers", NOMS 2024*

- DC monitoring is essential to ensure the health of DC infrastructure
  - Infrastructure monitoring
  - Software monitoring
- Infrastructure and software monitoring typically separate



# Correlating Network and Physical Activity

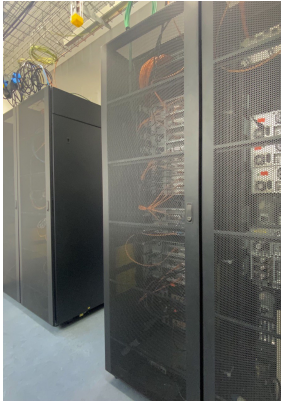
## A Cyber-Physical (DC-)Model

### RQ3.2 What is the relation between network control traffic and physical device activity?

***Wüstrich**, Schröder, Pahl, "Cyber-Physical Anomaly Detection for ICS", manage-IoT@IM 2021*
***Wüstrich**, Gallenmüller, Günther, Carle, Pahl, "Shells Bells: Cyber-Physical Anomaly Detection in Data Centers", NOMS 2024*

- DC monitoring is essential to ensure the health of DC infrastructure
  - Infrastructure monitoring
  - Software monitoring
- Infrastructure and software monitoring typically separate

- Devices often receive instructions over the network
- Protocols for device control often have a request-response design (e. g. IPMI)
  1. Control traffic containing instructions
  2. Physical activity while executing the instructions



# Correlating Network and Physical Activity

## State of the Art

| | Birnbach[10] | Ozmen[11] |
|---|---|---|
| DC environment | ✗ | ✗ |
| Physical signals | ✓ | ✓ |
| Network traffic | ✓ | ✓ |
| Composite signals | ✗ | ✓ |
| Anomaly detection | ✓ | ✓ |
| Audio | ✗ | ✗ |

[10] S. Birnbach, S. Eberz, and I. Martinovic, “Haunted House: Physical Smart Home Event Verification in the Presence of Compromised Sensors”, ACM Transactions on IoT, 2022

[11] M. O. Ozmen, R. Song, H. Farrukh, and Z. B. Celik, “Evasion Attacks and Defenses on Smart Home Physical Event Verification”, NDSS, 2023

12

13

14

# Correlating Network and Physical Activity

## State of the Art

| | Birnbach[10] | Ozmen[11] | Levy[12] | Borghesi[13] | Dayarathna[14] |
|---|---|---|---|---|---|
| DC environment | ✗ | ✗ | ✓ | ✓ | ✓ |
| Physical signals | ✓ | ✓ | ✓ | ✓ | ✓ |
| Network traffic | ✓ | ✓ | ✗ | ✗ | ✗ |
| Composite signals | ✗ | ✓ | ✗ | ✗ | ✗ |
| Anomaly detection | ✓ | ✓ | ✗ | ✓ | ✗ |
| Audio | ✗ | ✗ | ✗ | ✗ | ✗ |

[10] S. Birnbach, S. Eberz, and I. Martinovic, “Haunted House: Physical Smart Home Event Verification in the Presence of Compromised Sensors”, ACM Transactions on IoT, 2022

[11] M. O. Ozmen, R. Song, H. Farrukh, and Z. B. Celik, “Evasion Attacks and Defenses on Smart Home Physical Event Verification”, NDSS, 2023

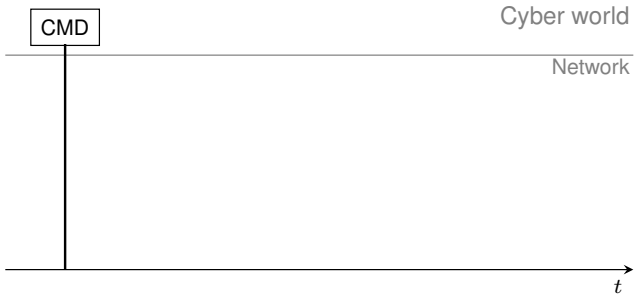
[12] M. Levy and J. O. Hallstrom, “A New Approach to Data Center Infrastructure Monitoring and Management (DCIMM)”, CCWC, 2017

[13] A. Borghesi, A. Libri, L. Benini, and A. Bartolini, “Online Anomaly Detection in HPC Systems”, AICAS, 2019

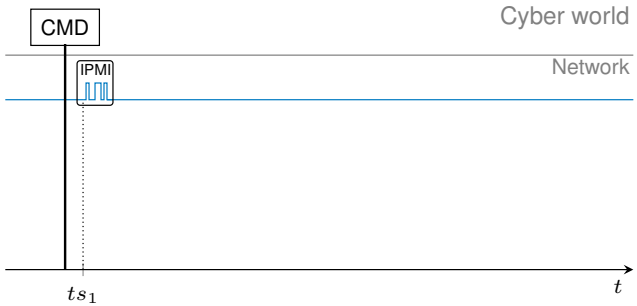
[14] M. Dayarathna, Y. Wen, and R. Fan, “Data Center Energy Consumption Modeling: A Survey", IEEE Communications Surveys & Tutorials, 2015

# Correlating Network and Physical Activity

## A cyber-physical (DC-)Model



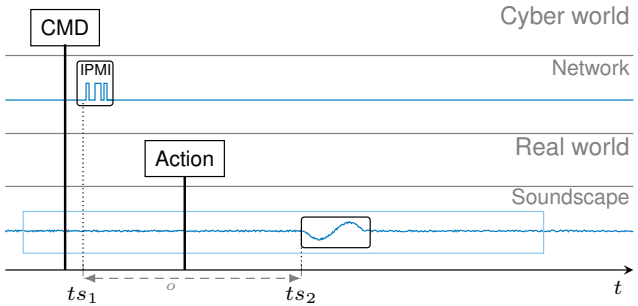
- Network carries commands for devices
- Observe command at time $ts_1$
- Command identifies target devices and instruction



# Correlating Network and Physical Activity

## A cyber-physical (DC-)Model

- Network carries commands for devices
- Observe command at time $ts_1$
- Command identifies target devices and instruction
- Triggers physical device operations (“actions”)
- The physical effects of actions can be measured at some point in time $ts_2$
- This yields an offset $o = ts_2 - ts_1$ to predict physical activity



## How to measure offset $o$?

The offset $o$ depends on a variety of factors:

- Command
- Side-channel
- Guarantees of the system
- Determinism of the system

# Correlating Network and Physical Activity

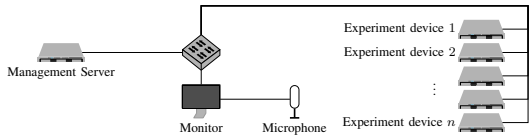
## How to measure offset *o*?

The offset *o* depends on a variety of factors:

- Command
- Side-channel
- Guarantees of the system
- Determinism of the system

Example measurements in a DC environment

- Issue IPMI commands (IPMI is limited to *on* and *off*)
- Use activity detection to identify physical activity and combine results to measure offset *o*



# Correlating Network and Physical Activity

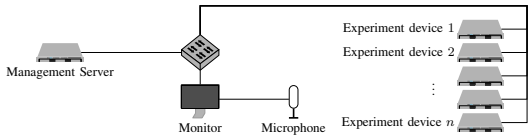
## How to measure offset *o*?

The offset *o* depends on a variety of factors:

- Command
- Side-channel
- Guarantees of the system
- Determinism of the system

Example measurements in a DC environment

- Issue IPMI commands (IPMI is limited to *on* and *off*)
- Use activity detection to identify physical activity and combine results to measure offset *o*




| | Power-On | Power-off |
|---|---|---|
| Average | 3.36 s | 1.57 s |
| Median | 3.29 s | 1.35 s |
| Std. dev. | 1.24 s | 0.53 s |
| Min | 1.42 s | 1.13 s |
| Max | 5.53 s | 2.66 s |

## RQ3.3 How can we use network control traffic to validate physical device activity in composite signals?

***Wüstrich**, Gallenmüller, Günther, Carle, and Pahl, "Shells Bells: Cyber-Physical Anomaly Detection in Data Centers", NOMS 2024*

1. We use the previously described model to dynamically create a reference of expected physical device behavior
2. We compare the real world measurements to the modeled prediction to validate system behavior in a time frame

## RQ3.3 How can we use network control traffic to validate physical device activity in composite signals?

***Wüstrich**, Gallenmüller, Günther, Carle, and Pahl, "Shells Bells: Cyber-Physical Anomaly Detection in Data Centers", NOMS 2024*

1. We use the previously described model to dynamically create a reference of expected physical device behavior
2. We compare the real world measurements to the modeled prediction to validate system behavior in a time frame

   This allows to detect several anomalies and attacks
   - Event spoofing
   - Event masquerading
   - Early action execution
   - Delayed action execution

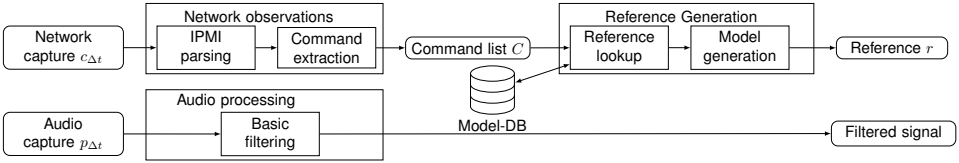
# Monitoring Physical Device Behavior in Composite Side Channels

## Anomaly Detection Phase


- Monitor the network for IPMI commands
- Record the soundscape

# Monitoring Physical Device Behavior in Composite Side Channels

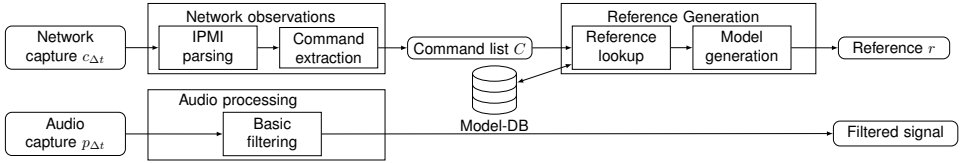
## Anomaly Detection Phase



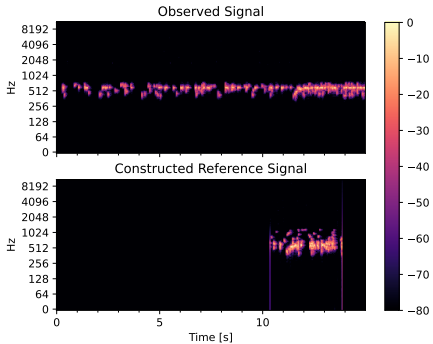

- Monitor the network for IPMI commands
- Record the soundscape
- Construct a reference signal using a Model-DB

# Monitoring Physical Device Behavior in Composite Side Channels

## Anomaly Detection Phase

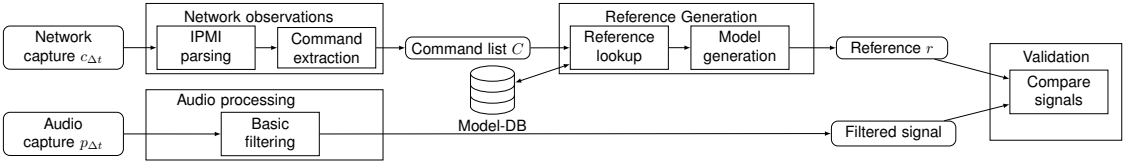



- Monitor the network for IPMI commands
- Record the soundscape
- Construct a reference signal using a Model-DB
  - Generate a silent time frame
  - Per IPMI command calculate relative timestamp
  - Pull reference from Model-DB
  - Add expected change at predicted offset

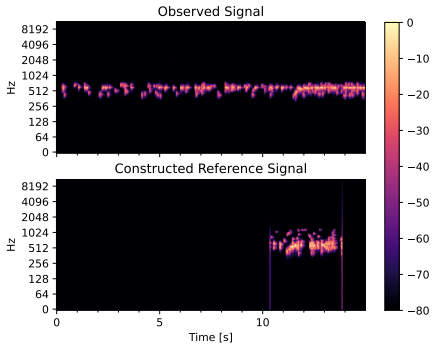


# Monitoring Physical Device Behavior in Composite Side Channels

## Anomaly Detection Phase




- Monitor the network for IPMI commands
- Record the soundscape
- Construct a reference signal using a Model-DB
  - Generate a silent time frame
  - Per IPMI command calculate relative timestamp
  - Pull reference from Model-DB
  - Add expected change at predicted offset



## Anomaly Detection Phase - Validation

- We need to compare the constructed signal to the real observation
- Challenges:
  - Noise
  - Temporal inaccuracies

| Method | Noise | Temporal | Comparison | Reasoning |
|---|---|---|---|---|
| Fingerprinting | ✓ | ✗ | ✓ | % matching hashes |
| Dynamic Time Warping | ✗ | ✓ | ✓ | distance |
| RMS Energy | ✗ | ✓ | ✓ | distance |
| Image recognition | ✓ | ✓ | ✗ | events |
| Machine Learning | ✓ | ✓ | ✓ | events/classes |

## Anomaly Detection Phase - Validation

- We need to compare the constructed signal to the real observation
- Challenges:
  - Noise
  - Temporal inaccuracies

| Method | Noise | Temporal | Comparison | Reasoning |
|---|---|---|---|---|
| Fingerprinting | ✓ | ✗ | ✓ | % matching hashes |
| Dynamic Time Warping | ✗ | ✓ | ✓ | distance |
| RMS Energy | ✗ | ✓ | ✓ | distance |
| Image recognition | ✓ | ✓ | ✗ | events |
| Machine Learning | ✓ | ✓ | ✓ | events/classes |

⇒ Use ML to compare synthetic references with real world measurements

- The approach uses a convolutional neural network (CNN)
- The input is a combined bitmap of the reference and real world signal
- The output is an anomaly class or "normal" classification

Experiments for a data center (DC) environment

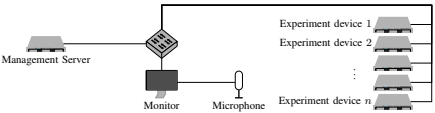
- Around 40 devices, 3 actively cooled switches
- Active air condition
- Devices managed via IPMI
- Curated synthetic dataset from real world recordings

Experiments for a data center (DC) environment

- Around 40 devices, 3 actively cooled switches
- Active air condition
- Devices managed via IPMI
- Curated synthetic dataset from real world recordings

Setup

- Management device controlling DC devices via IPMI
- DC devices executing actions (power-off and power-on)
- Monitoring device




## Results

Use 20 % of synthetic dataset (14520 traces with a length of 15 s, $\geq$ 60 h audio) for testing
Overall accuracy of the CNN: 98.62 %

| Real/Classified | Normal | Early | Delay | Spoof. | Masqu. | Recall |
|---|---|---|---|---|---|---|
| Normal (3000) | 2947 | | | | | |
| Early (1920) | | 1876 | | | | |
| Delay (1920) | | | 1884 | | | |
| Spoofing (3840) | | | | 3811 | | |
| Masquerading (3840) | | | | | 3801 | |
| Precision | | | | | | |

## Results

Use 20 % of synthetic dataset (14520 traces with a length of 15 s, $\geq$ 60 h audio) for testing
Overall accuracy of the CNN: 98.62 %

| Real/Classified | Normal | Early | Delay | Spoof. | Masqu. | Recall |
|---|---|---|---|---|---|---|
| Normal (3000) | 2947 | 23 | 14 | 8 | 8 | 0.9823 |
| Early (1920) | 40 | 1876 | 0 | 0 | 4 | 0.9771 |
| Delay (1920) | 35 | 0 | 1884 | 1 | 0 | 0.9813 |
| Spoofing (3840) | 27 | 0 | 1 | 3811 | 1 | 0.9924 |
| Masquerading (3840) | 38 | 1 | 0 | 0 | 3801 | 0.9898 |
| Precision | 0.9546 | | | | | |

## Results

Use 20 % of synthetic dataset (14520 traces with a length of 15 s, $\geq$ 60 h audio) for testing
Overall accuracy of the CNN: 98.62 %

| Real/Classified | Normal | Early | Delay | Spoof. | Masqu. | Recall |
|---|---|---|---|---|---|---|
| Normal (3000) | 2947 | 23 | 14 | 8 | 8 | 0.9823 |
| Early (1920) | 40 | 1876 | 0 | 0 | 4 | 0.9771 |
| Delay (1920) | 35 | 0 | 1884 | 1 | 0 | 0.9813 |
| Spoofing (3840) | 27 | 0 | 1 | 3811 | 1 | 0.9924 |
| Masquerading (3840) | 38 | 1 | 0 | 0 | 3801 | 0.9898 |
| Precision | 0.9546 | 0.9874 | 0.9921 | 0.9976 | 0.9963 | |

## Contributions

| | Birnbach[10] | Ozmen[11] | Levy[12] | Borghesi[13] | Dayarathna[14] | **This thesis**[15][16] |
|---|---|---|---|---|---|---|
| DC environment | ✗ | ✗ | ✓ | ✓ | ✓ | ✓ |
| Physical signals | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Network traffic | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ |
| Composite signals | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ |
| Anomaly detection | ✓ | ✓ | ✗ | ✓ | ✗ | ✓ |
| Audio | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ |

[10] S. Birnbach, S. Eberz, and I. Martinovic, “Haunted House: Physical Smart Home Event Verification in the Presence of Compromised Sensors”, ACM Transactions on IoT, 2022
[11] M. O. Ozmen, R. Song, H. Farrukh, and Z. B. Celik, “Evasion Attacks and Defenses on Smart Home Physical Event Verification”, NDSS, 2023
[12] M. Levy and J. O. Hallstrom, “A New Approach to Data Center Infrastructure Monitoring and Management (DCIMM)”, CCWC, 2017
[13] A. Borghesi, A. Libri, L. Benini, and A. Bartolini, “Online Anomaly Detection in HPC Systems”, AICAS, 2019
[14] M. Dayarathna, Y. Wen, and R. Fan, “Data Center Energy Consumption Modeling: A Survey", IEEE Communications Surveys & Tutorials, 2015
[15] **Wüstrich**, Schröder, Pahl, “Cyber-Physical Anomaly Detection for ICS”, manage-IoT@IM, 2021
[16] **Wüstrich**, Gallenmüller, Günther, Carle, Pahl, “Shells Bells: Cyber-Physical Anomaly Detection in Data Centers”, NOMS, 2024

# Leveraging Multi-Layer Relationships for Security

| | Application | Network | Physical | Application | Domain |
|---|---|---|---|---|---|
| Haas et al.[2] | ✓ | ✓ | ✗ | AD | Enterprise |
| Popa et al.[3] | ✓ | ✓ | ✗ | Dependency detection | Enterprise |

[2] S. Haas et al. Zeek-osquery: Host-network correlation for advanced monitoring and intrusion detection., IFIP TC 11, 2020

[3] L. Popa et al. Macroscope: End-point approach to networked application dependency discovery. CoNEXT, 2009

# Leveraging Multi-Layer Relationships for Security

| | Application | Network | Physical | Application | Domain |
|---|---|---|---|---|---|
| Haas et al.[2] | ✓ | ✓ | ✗ | AD | Enterprise |
| Popa et al.[3] | ✓ | ✓ | ✗ | Dependency detection | Enterprise |
| Birnbach et al.[10] | ✗ | ✓ | ✓ | AD | IoT |
| Ozmen et al.[11] | ✗ | ✓ | ✓ | AD | IoT |
| Levy et al.[12] | ✗ | ✗ | ✓ | Monitoring | DC |
| Borghesi et al.[13] | ✗ | ✗ | ✓ | Monitoring | DC |
| Dayarathna et al.[14] | ✗ | ✗ | ✓ | AD | DC |
| **This thesis** | ✓ | ✓ | ✓ | **AD** | **Enterprise/DC** |

[2] S. Haas et al. Zeek-osquery: Host-network correlation for advanced monitoring and intrusion detection., IFIP TC 11, 2020

[3] L. Popa et al. Macroscope: End-point approach to networked application dependency discovery. CoNEXT, 2009

[10] S. Birnbach, S. Eberz, and I. Martinovic, “Haunted House: Physical Smart Home Event Verification in the Presence of Compromised Sensors”, ACM Transactions on IoT, 2022

[11] M. O. Ozmen, R. Song, H. Farrukh, and Z. B. Celik, “Evasion Attacks and Defenses on Smart Home Physical Event Verification”, NDSS, 2023

[12] M. Levy and J. O. Hallstrom, “A New Approach to Data Center Infrastructure Monitoring and Management (DCIMM)”, CCWC, 2017

[13] A. Borghesi, A. Libri, L. Benini, and A. Bartolini, “Online Anomaly Detection in HPC Systems”, AICAS, 2019

[14] M. Dayarathna, Y. Wen, and R. Fan, “Data Center Energy Consumption Modeling: A Survey", IEEE Communications Surveys & Tutorials, 2015

# Contributions

## Presented contributions

- Method to efficiently and reliably associate processes and network packets using eBPF
- Cyber-physical model to capture the relationship between control traffic and physical device activity
- Cyber-physical anomaly detection in DC environments

## Additional contributions

- A taxonomy for classifying attacks based on the effects they have on a system[15]
- Network Application Profiles for formalizing typical network application behavior[9]
- Method to identify physical device activity in composite signals[16]

[15] **Wüstrich**, Pahl, and Liebald, “Towards an Extensible IoT Security Taxonomy”, ISCC, 2020
[9] **Wüstrich**, Schacherbauer, Künßberg, Gallenmüller, Pahl, Carle„ “Network Profiles for Detecting Application-Characteristic Behavior Using Linux eBPF”, eBPF@SIGCOMM, 2023
[16] **Wüstrich**, Gallenmüller, Pahl, Carle, "AC/DCIM: Acoustic Channels for Data Center Infrastructure Monitoring" NOMS, 2022