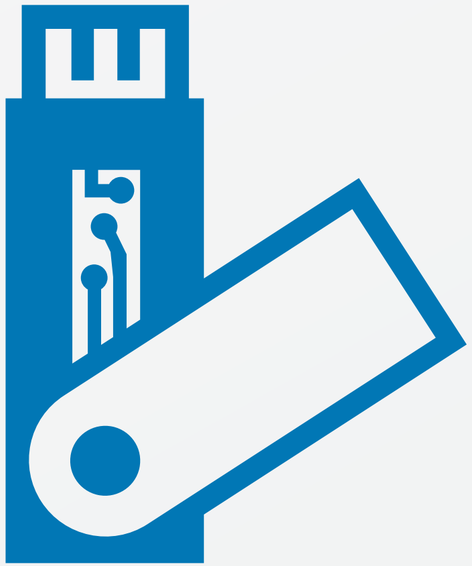
Secure Hardware

EDLs, ECALL, OCALL

intel arm



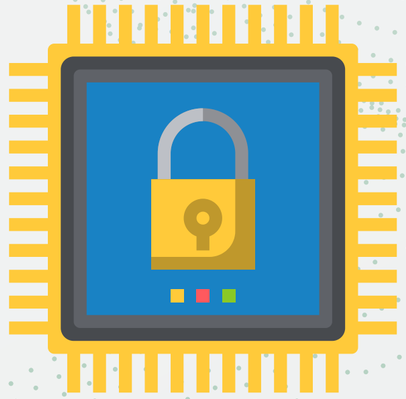
University of Colorado **Boulder**

Publications:

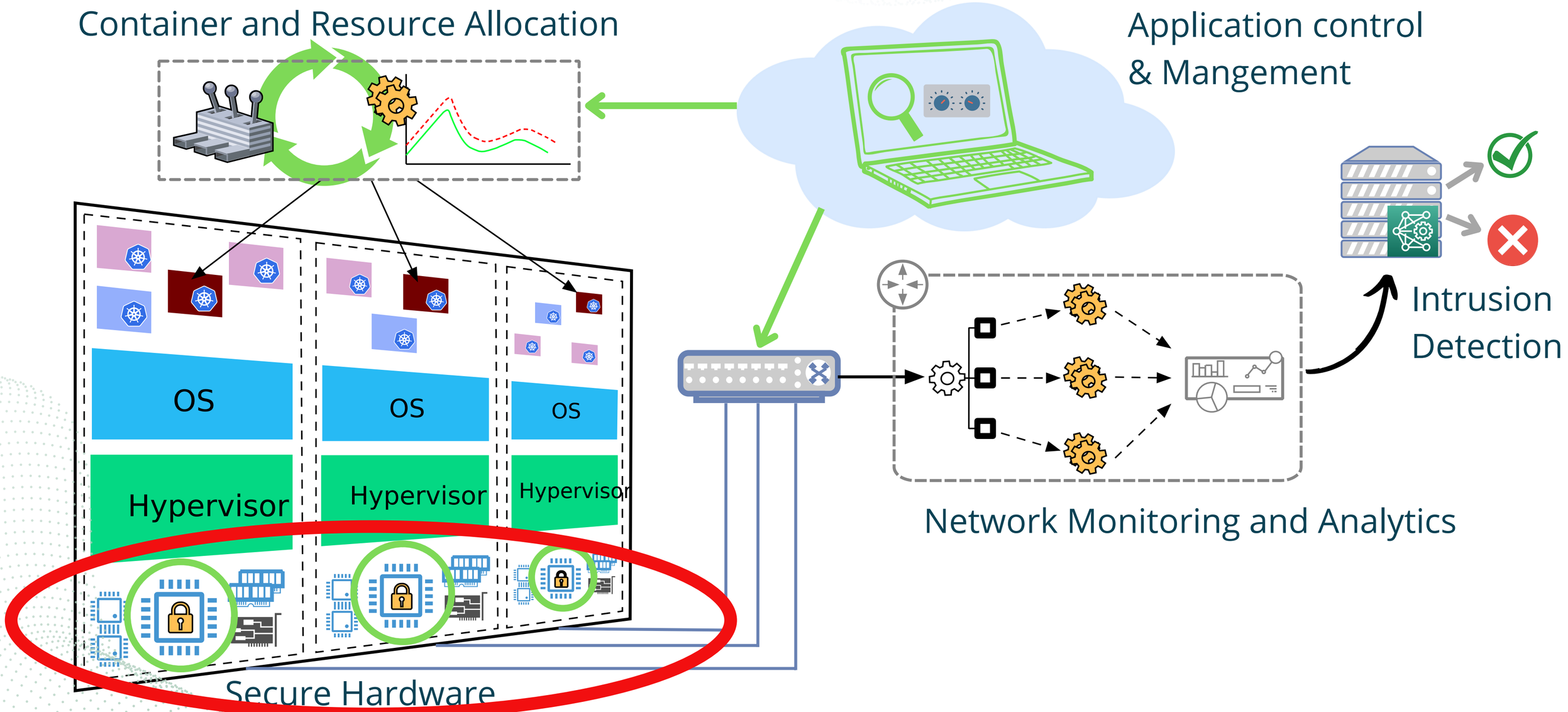
A. Coughlin, G. Cusack, J. Wampler, E. Keller, E. Wustrow.

Breaking The Trust Dependence on Third Party Processes For Reconfigurable Secure Hardware

International Symposium on Field-Programmable Gate Arrays, 2019



Cloud Architecture Overview



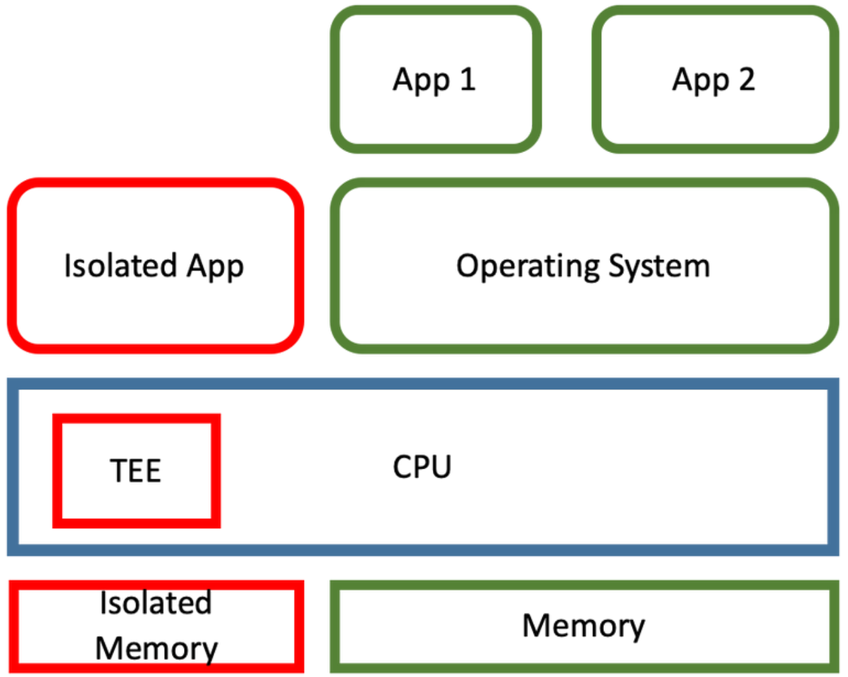
Secure Hardware

Current State

- Hardware that protects against other parts of the system
- Defend against untrusted software
- Defend against some kinds of physical threats
- CPU designers are incorporating secure hardware into their chips

arm TrustZone

intel SGX



Secure Hardware

Problem

Hardware manufacturers make the decisions

Secure Hardware

Problem: Hardware manufacturers make the decisions

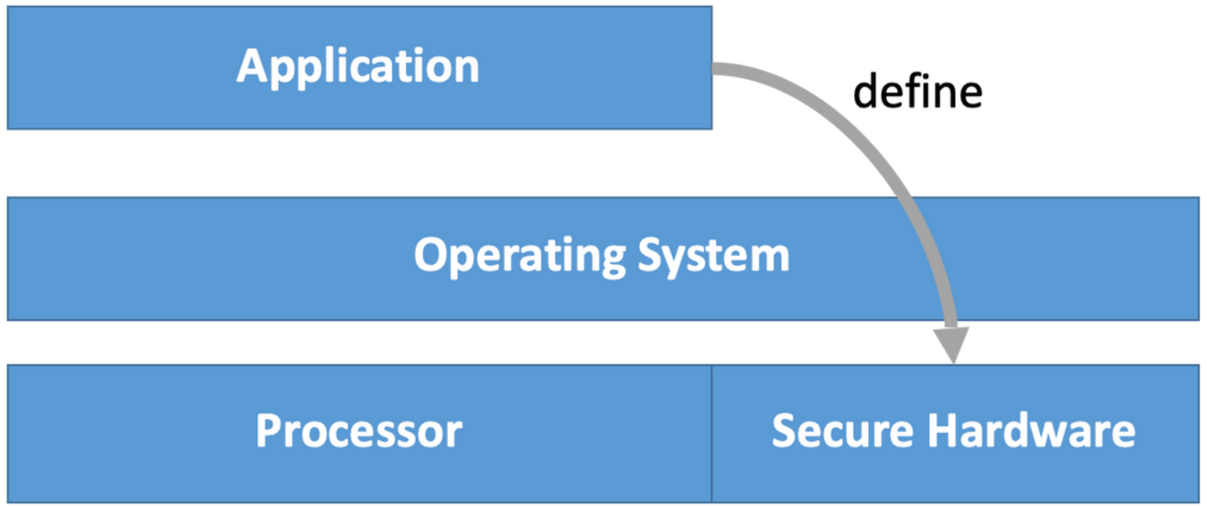
- What features to include
- When updates/bug fixes are available
- Lack of application-specific features for each app
- Rigid!

Feature	TPM	TZ	SGX
Flexible Root of Trust	●	●	○
Trusted Execution Environment	○	●	●
Remote Attestation	●	○	●
Peripheral Access	○	●	○
Trusted Input	○	◐	○
Hardware RNG	●	○	●
Hardware Crypto	●	◐	◐
Secure Storage	●	○	●
Shared Architecture	◐	●	●
Oblivious Memory	○	○	●
Cache Side Channel Defense	●	○	○
TLB Side Channel Defense	○	●	○

Secure Hardware

Our thought

Developers define their own secure hardware applications

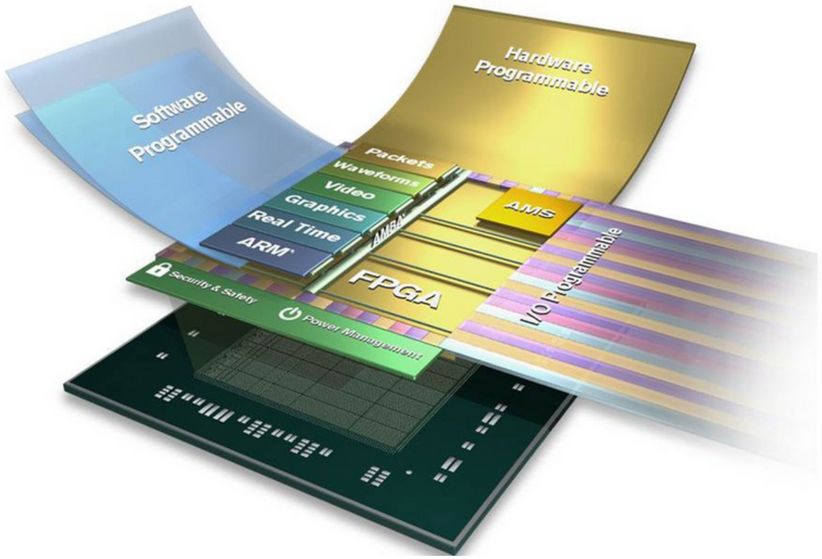


Secure Hardware

Goal: Developers define their own secure hardware applications

Enter FPGAs

- Leverage programmability of FPGAs to enable reconfigurable secure hardware
- Expose programmability to developers
- Lets developers define the secure hardware features they need



Secure Hardware

Goal: Developers define their own secure hardware applications

Example Application needs:

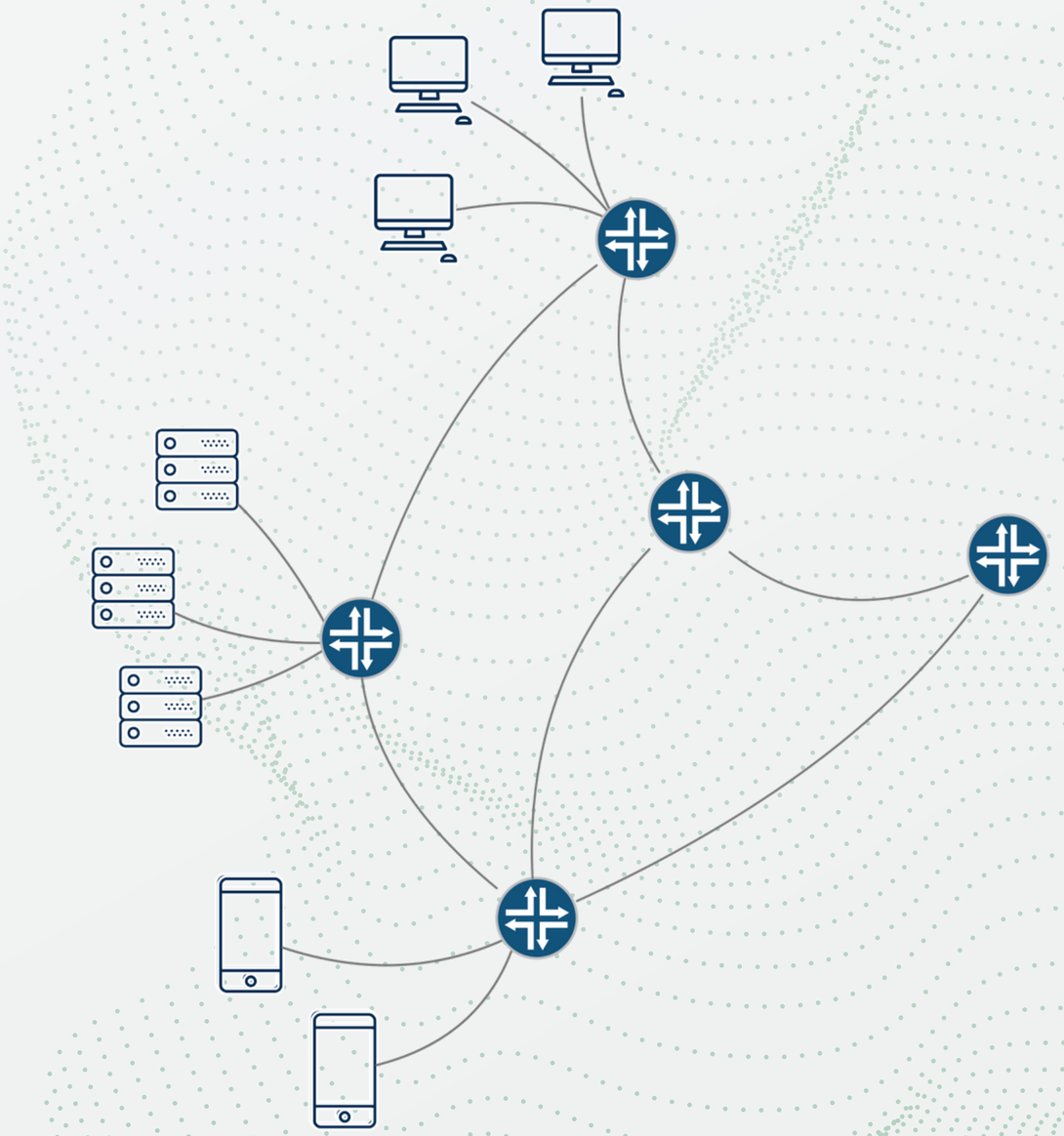
- Flexible Root of Trust
- TEE
- Remote Attestation

Feature	TPM	TZ	SGX
Flexible Root of Trust	◐	◐	○
Trusted Execution Environment	○	◐	◐
Remote Attestation	◐	○	◐
Peripheral Access	○	●	○
Trusted Input	○	◐	○
Hardware RNG	●	○	●
Hardware Crypto	●	◐	◐
Secure Storage	●	○	●
Shared Architecture	◐	●	●
Oblivious Memory	○	○	●
Cache Side Channel Defense	●	○	○
TLB Side Channel Defense	○	●	○

User-defined

University of Colorado **Boulder**

Network Monitoring

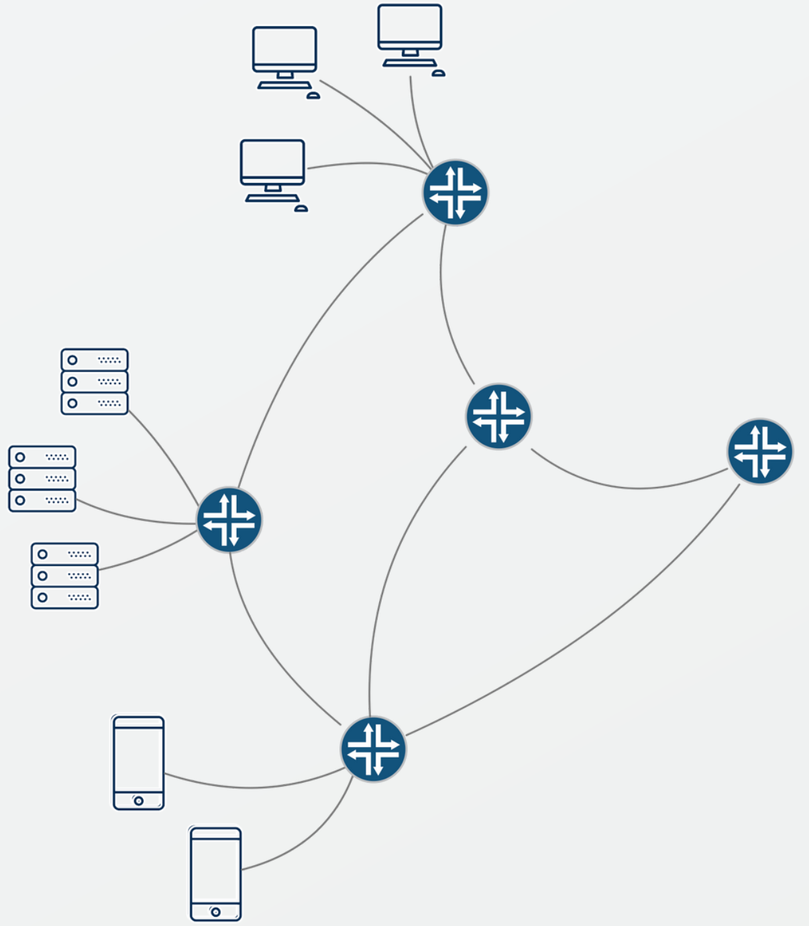


University of Colorado **Boulder**

Publications:

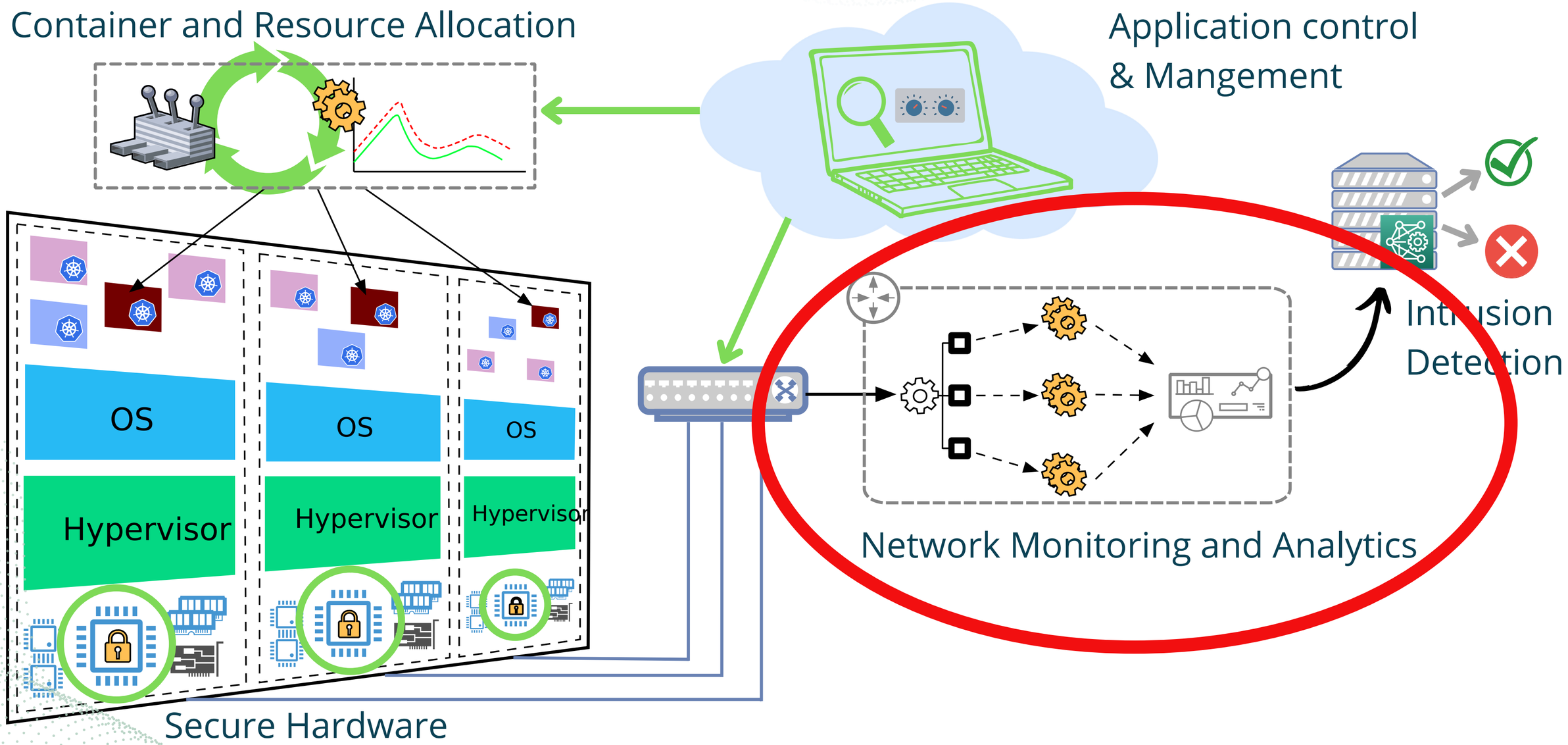
O. Michel, J. Sonchack, G. Cusack, M. Nazari, E. Keller and J. M. Smith.
Software Packet-Level Network Analytics at Cloud Scale
IEEE Transactions on Network and Service Management, vol. 18, no. 1, pp. 597-610, March 2021

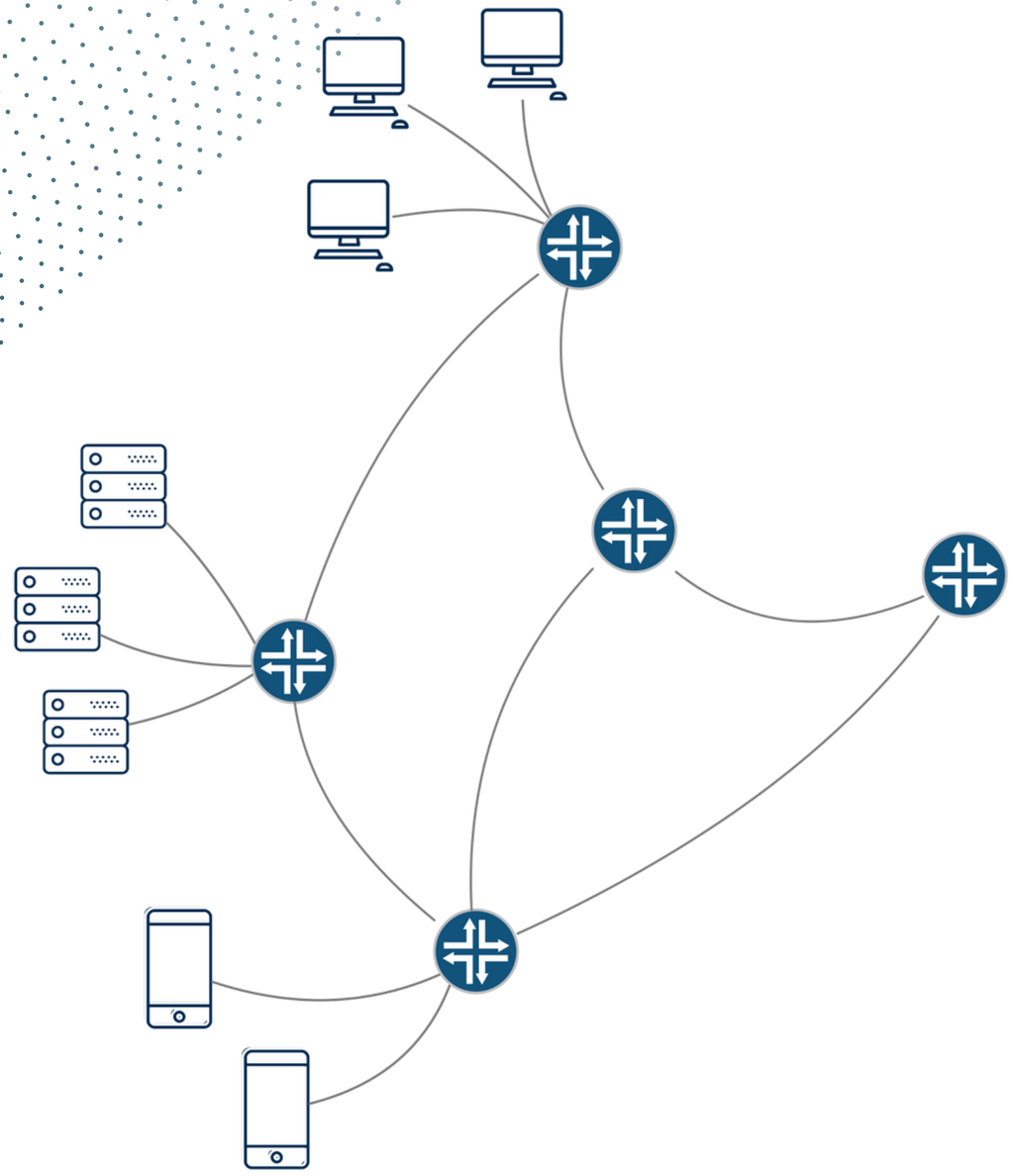
G. Cusack, O. Michel, & E. Keller.
Machine Learning-Based Fingerprinting of Network Traffic Using Programmable Forwarding Engines
Network and Distributed Systems Symposium, 2018 (Poster)



University of Colorado **Boulder**

Cloud Architecture Overview



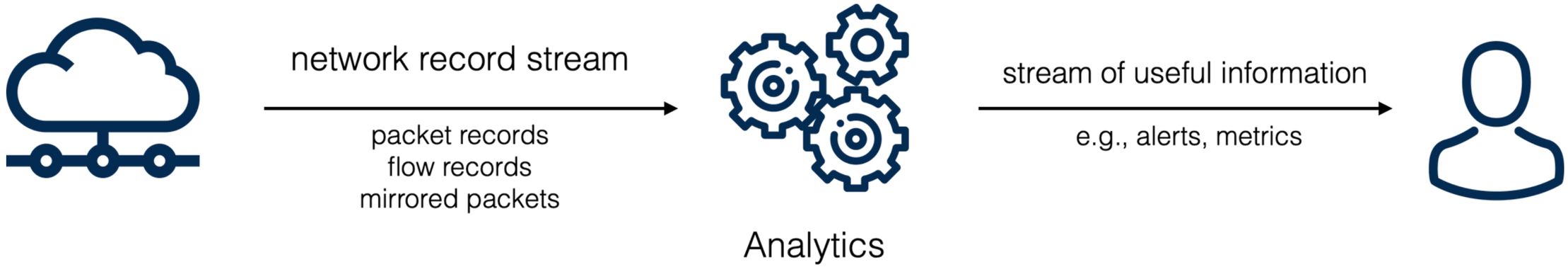


Network Monitoring

- Mine critical information from packet streams
- Trigger actions to guide subsequent decisions
- Troubleshoot, prevent security threats, etc

Network Monitoring

Telemetry-based network monitoring



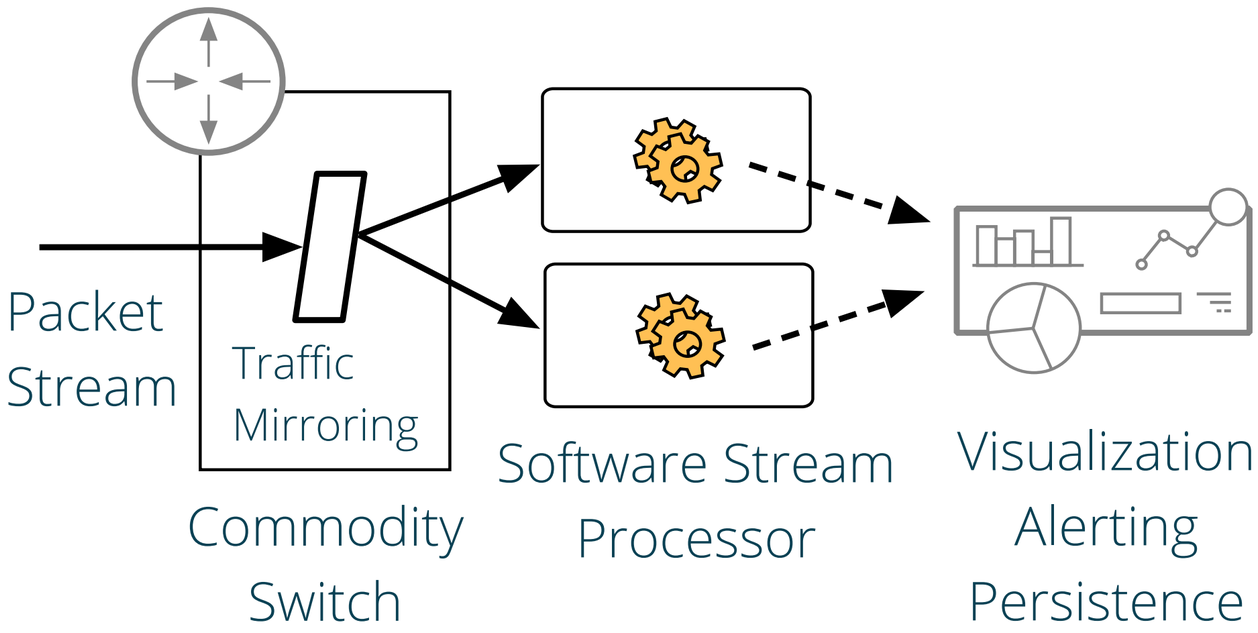
Network Monitoring

Ideal network monitoring system

- Record of every packet
- Full programmability
- Datacenter-scale performance

Recent Work

Analytics in Software

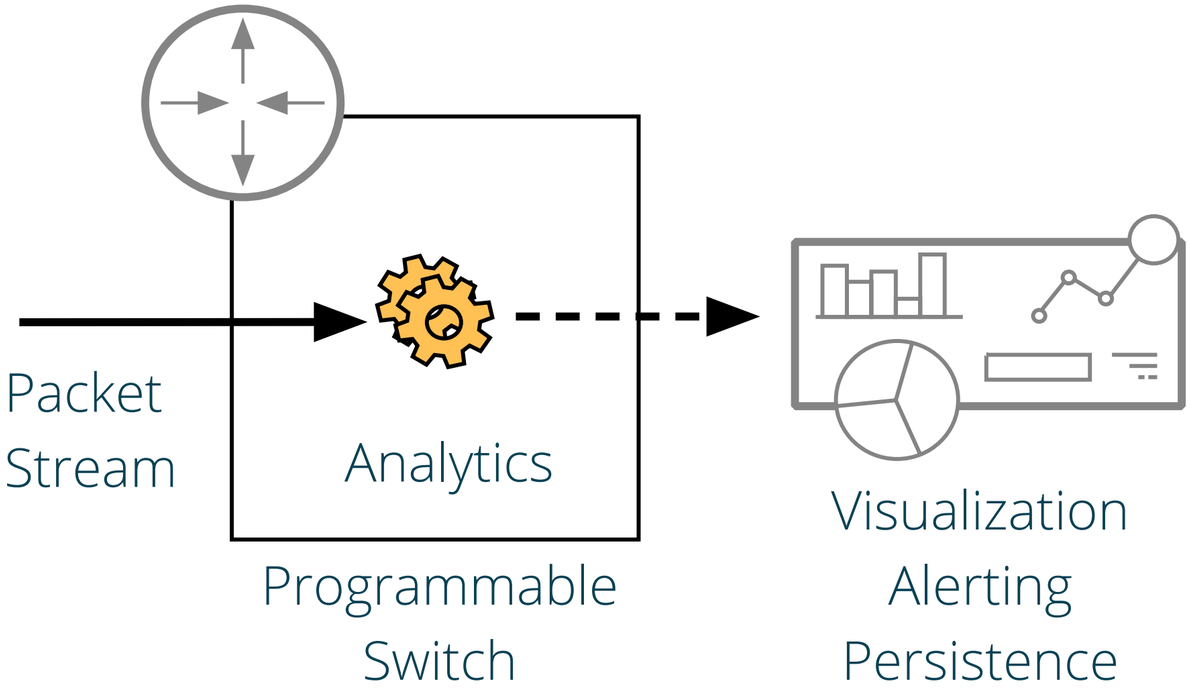


Shortcomings

- Sampling, aggregation, and filtering
- Low Performance
- Low Efficiency
- Typical: 0.1 - 2.5 M packets per core
  - dShark [Yu at al. 2019]
  - Spark [Zaharia et al. 2016]

Recent Work

Analytics in Hardware



Shortcomings

- Limited switch resources
- Low programmability/flexibility
- Sketching can be inaccurate
- Examples
  - Marple [Narayana et al. 2017]
  - OpenSketch [Yu et al. 2013]
  - UnivMon [Liu et al. 2016]
  - Sonata [Gupta et al. 2018]

Network Monitoring

Insight

- Need for compromises is often a product of design
- Software is not inherently incapable

University of Colorado **Boulder**

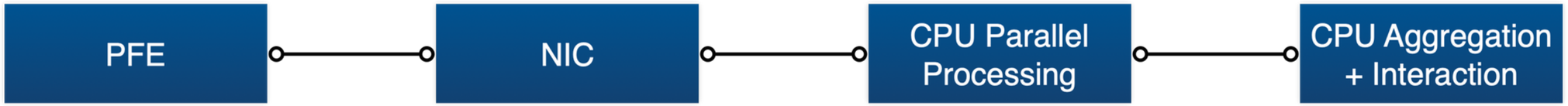
Toccoa

Network Monitoring

Toccoa: Hardware-Software Co-design

Network Monitoring

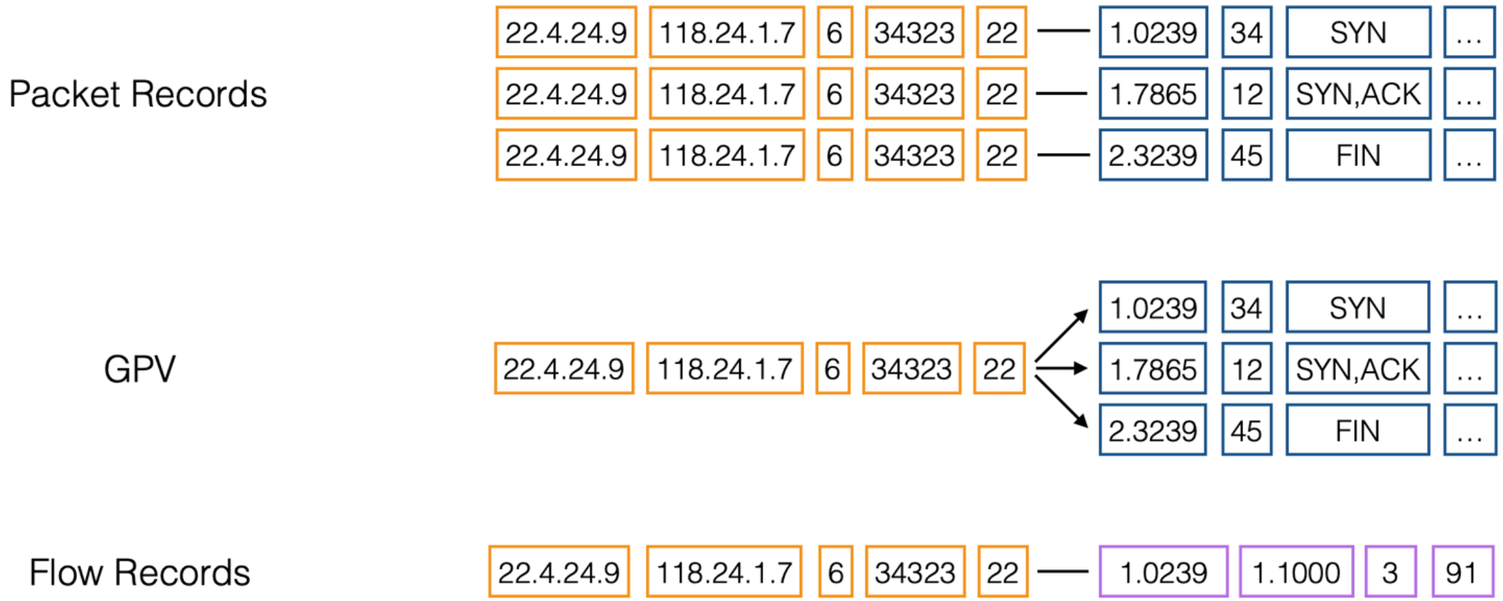
Toccoa: Pipeline



Network Monitoring

Telemetry: Grouped Packet Vectors

Generated on Switch



aggregation keys

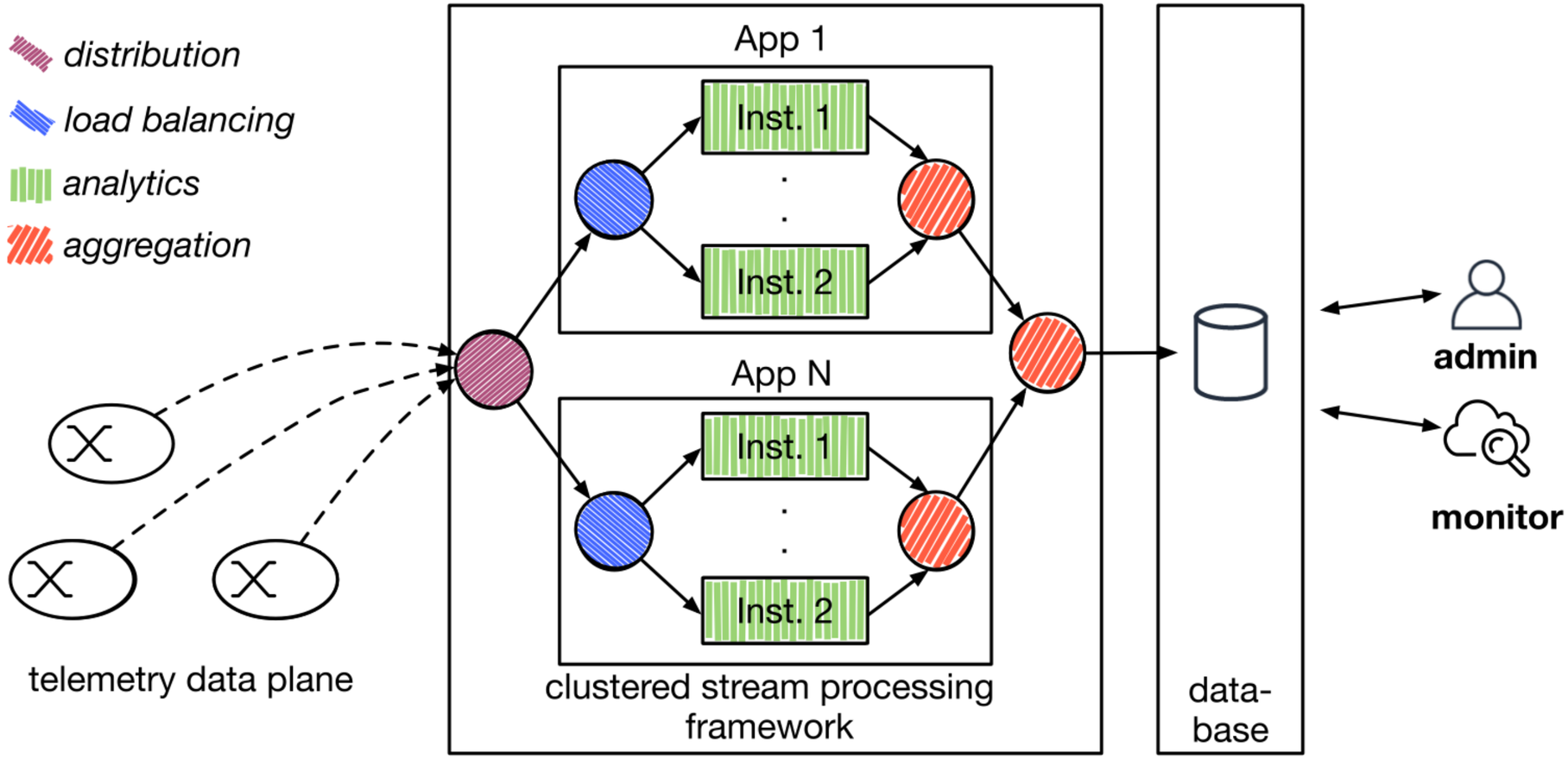
packet features

aggregated features

Scaling Hardware Accelerated Monitoring to Concurrent and Dynamic Queries With *Flow [Sonchack et al. ATC '18]

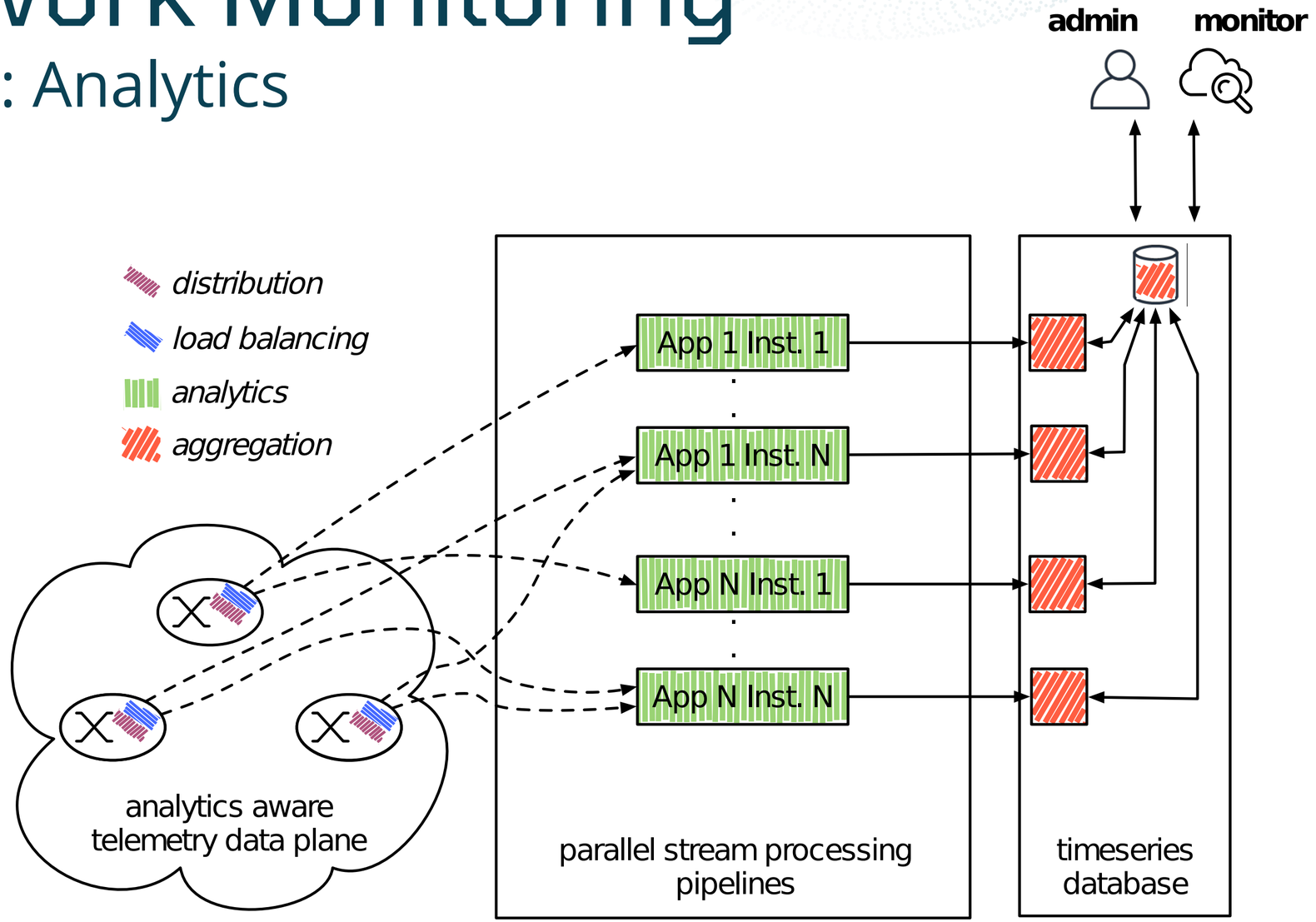
Network Monitoring

Spark: Analytics



Network Monitoring

Toccoa: Analytics



Network Monitoring

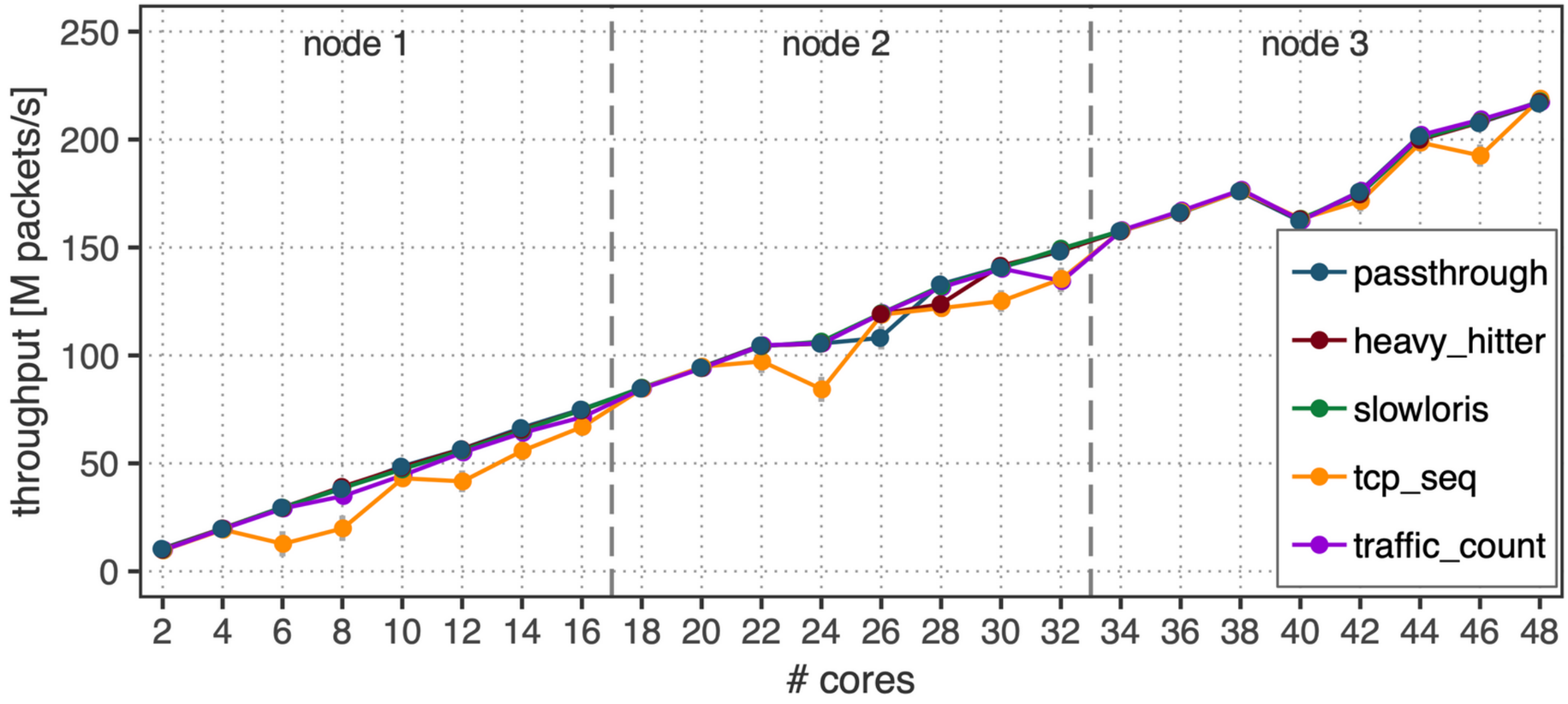
Result: Facebook Traffic Comparison

- Facebook's analytics cluster needs to sustain 961 Mpps to meet peak traffic packet rates

- If we assume 16 cores/server, we would require
  - dShark - ~96 servers
  - Spark - ~480 servers
  - Toccoa - ~4 servers

Network Monitoring

Result: Scalability and Performance



Network Monitoring

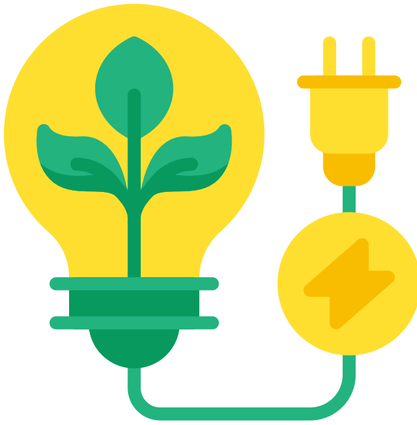
Recap

- Holistically design a telemetry and analytics network monitoring platform
- At scale, network operators now
  - Extract per-packet records from their network at high rates
  - Build high performance analytic applications!

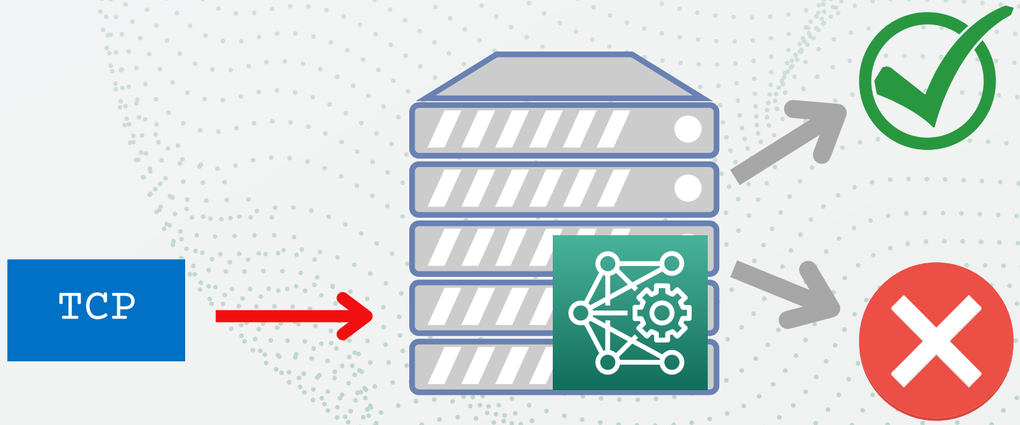
Security

Performance

Efficiency



Network Intrusion Detection



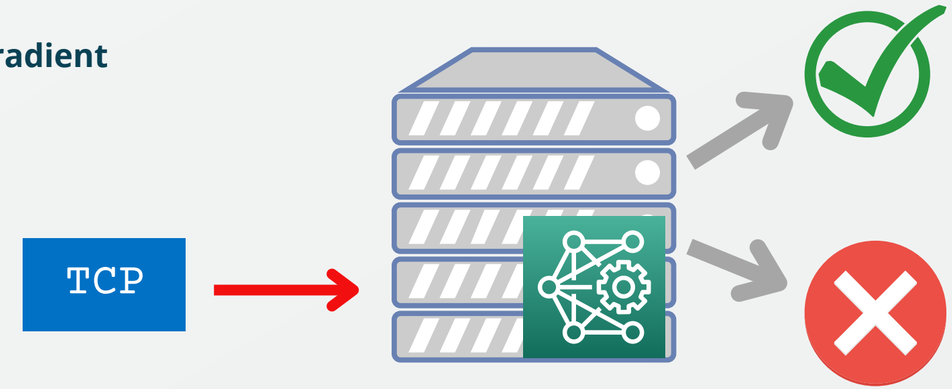
University of Colorado **Boulder**

Publications:

M. Hashemi, G. Cusack, E. Keller.
Towards the Evaluation of NIDSs in an Adversarial Setting
ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks (Big-DAMA), 2019

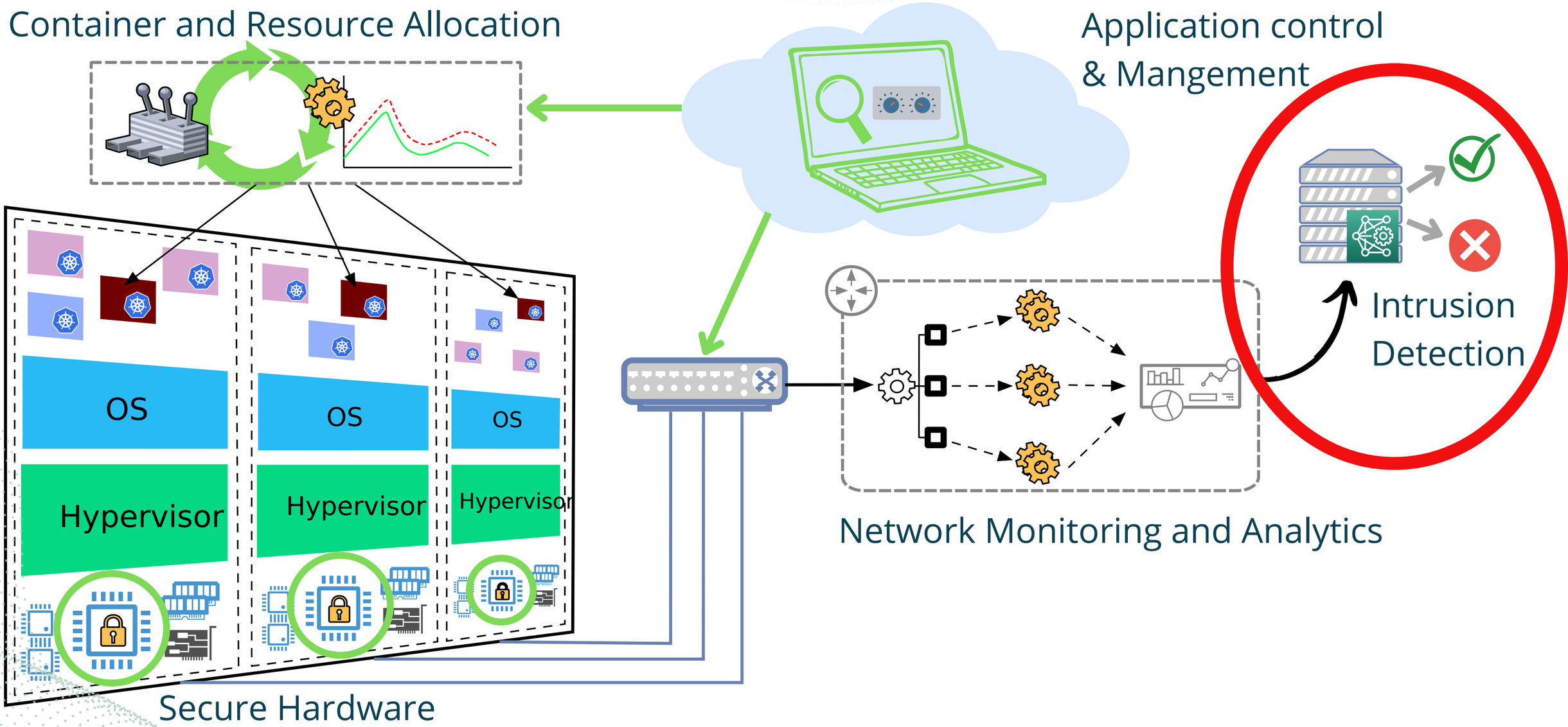
M. Hashemi, G. Cusack, E. Keller.
Stochastic Substitute Training: A Gray-box Approach to Craft Adversarial Examples Against Gradient Obfuscation Defenses
ACM Workshop on Artificial Intelligence and Security (AISec), 2018

G. Cusack, O. Michel, E. Keller.
Machine Learning-Based Detection of Ransomware Using SDN
Workshop on SDN-NFV Security, 2018



University of Colorado **Boulder**

Cloud Architecture Overview

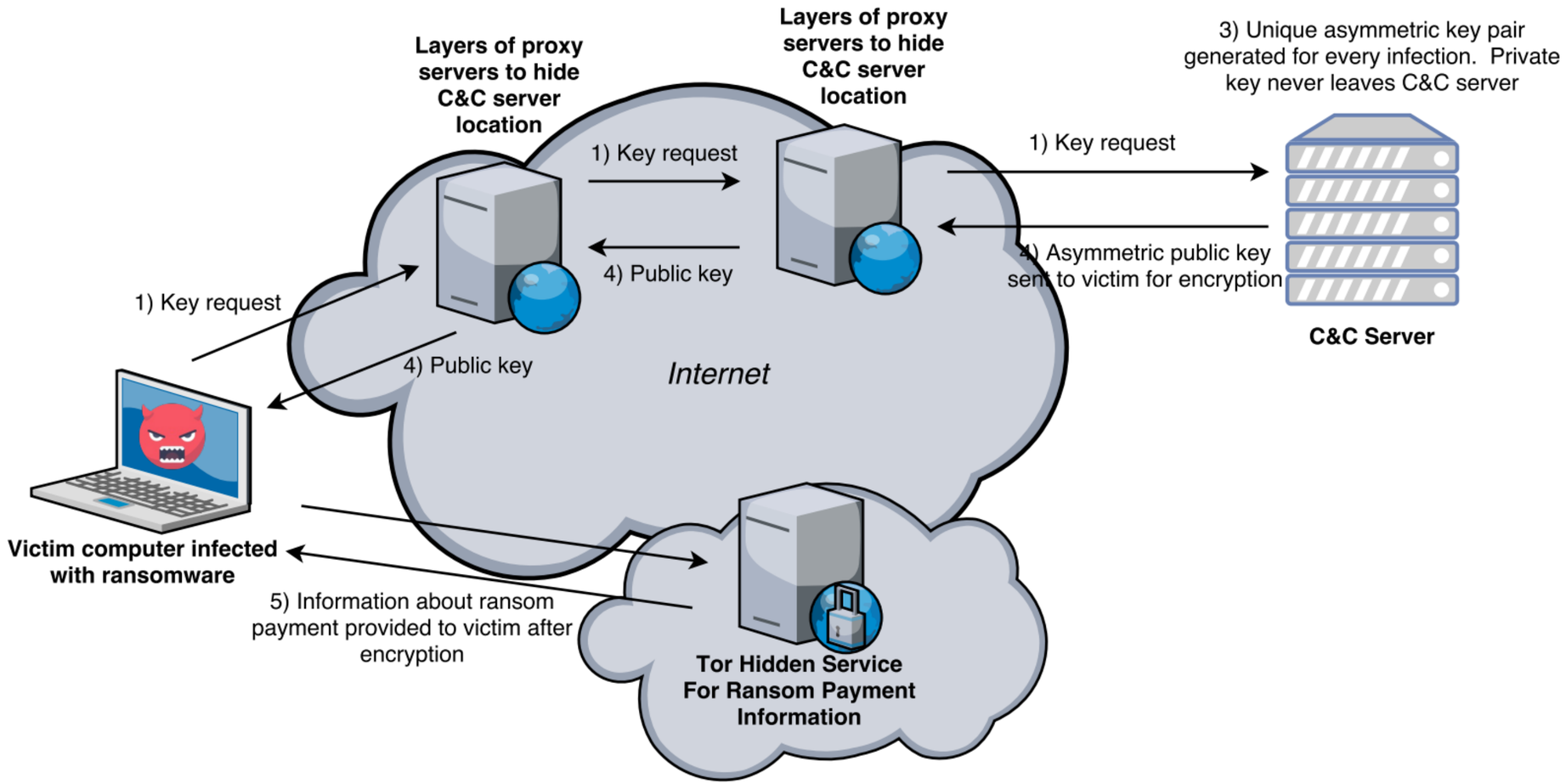


Network Intrusion Detection

Ransomware Detection

Ransomware Detection

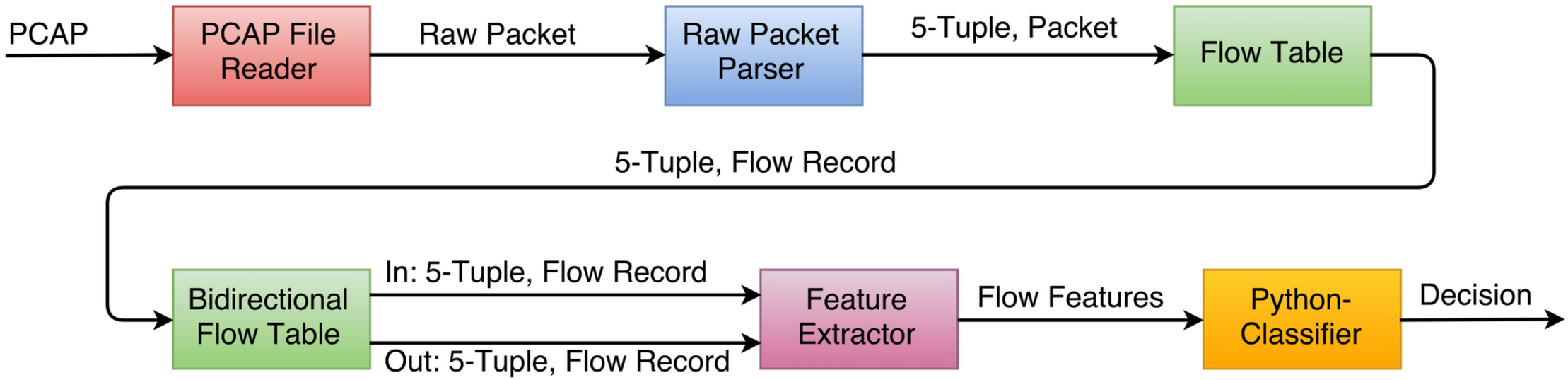
Data Flow



Machine Learning-based Detection of Ransomware using SDN [Cusack et al. SDN/NFV '18]

Ransomware Detection

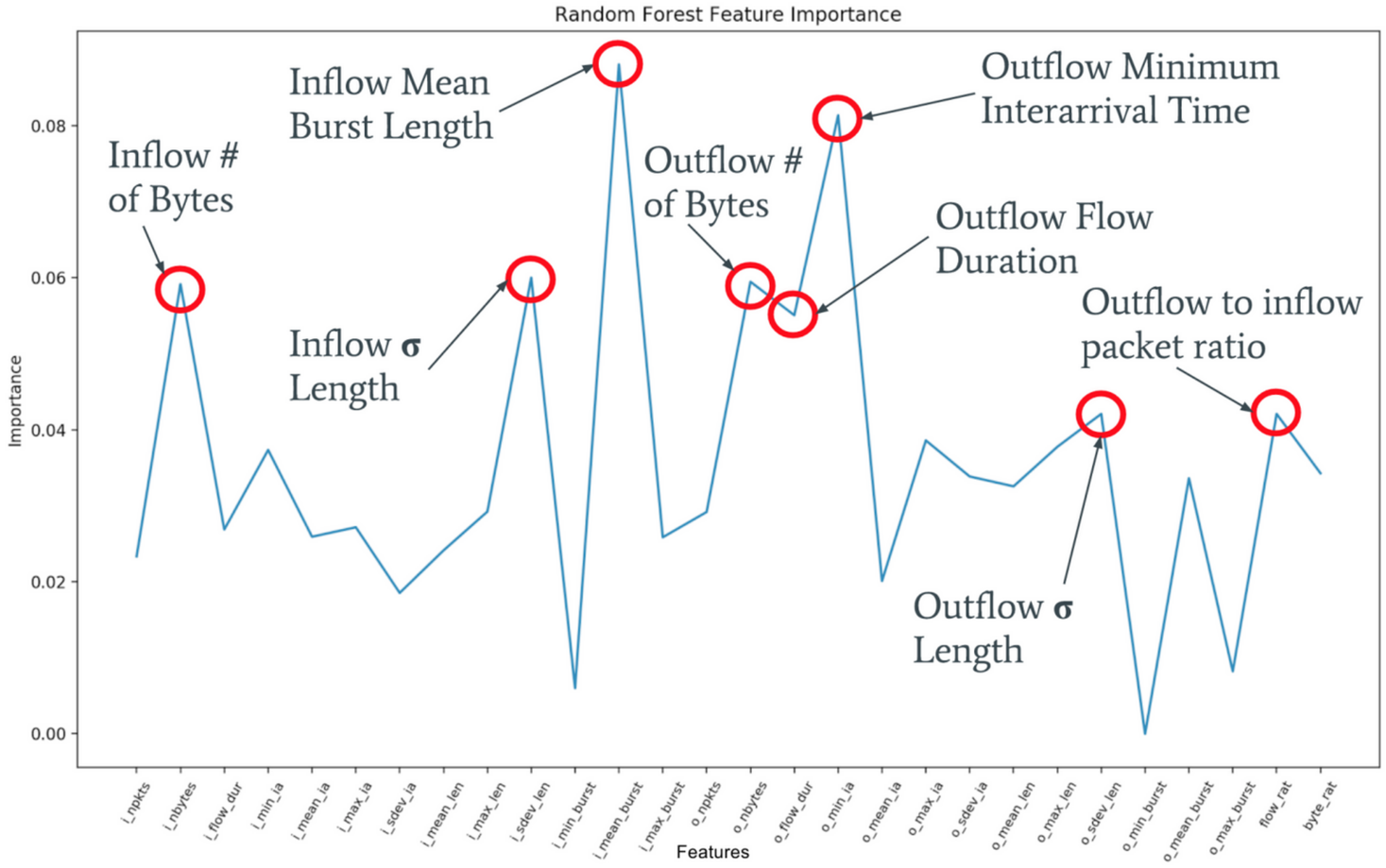
Stream Processor and Classification



Machine Learning-based Detection of Ransomware using SDN [Cusack et al. SDN/NFV '18]

Ransomware Detection

Identifying Key Features



Machine Learning-based Detection of Ransomware using SDN [Cusack et al. SDN/NFV '18]

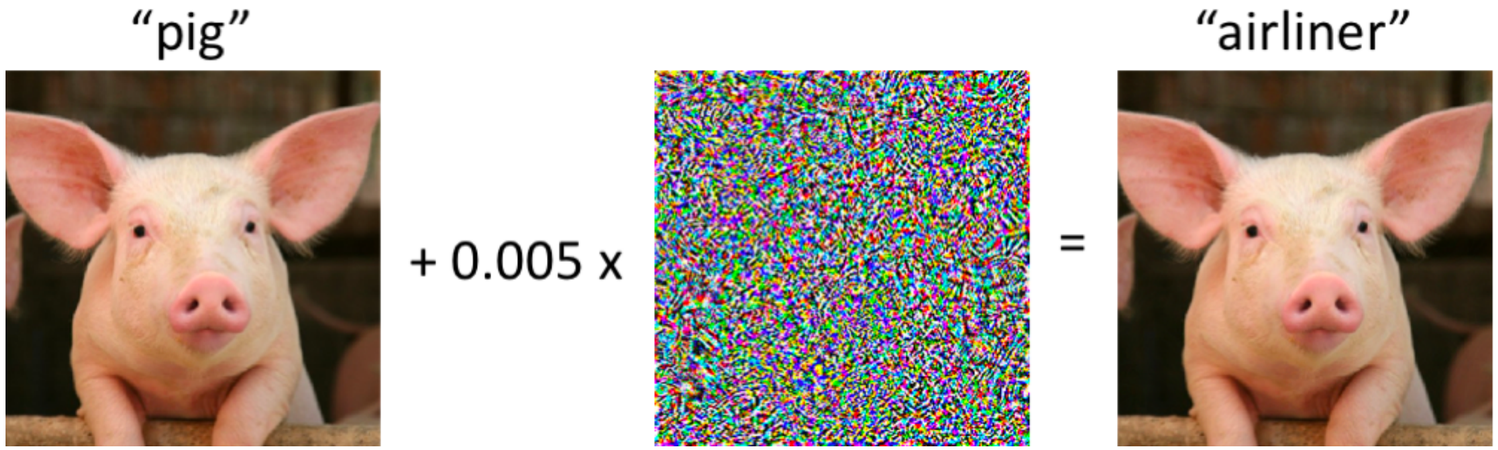
University of Colorado **Boulder**

Network Intrusion Detection

Adversarial Examples in the Network Domain

Network Intrusion Detection

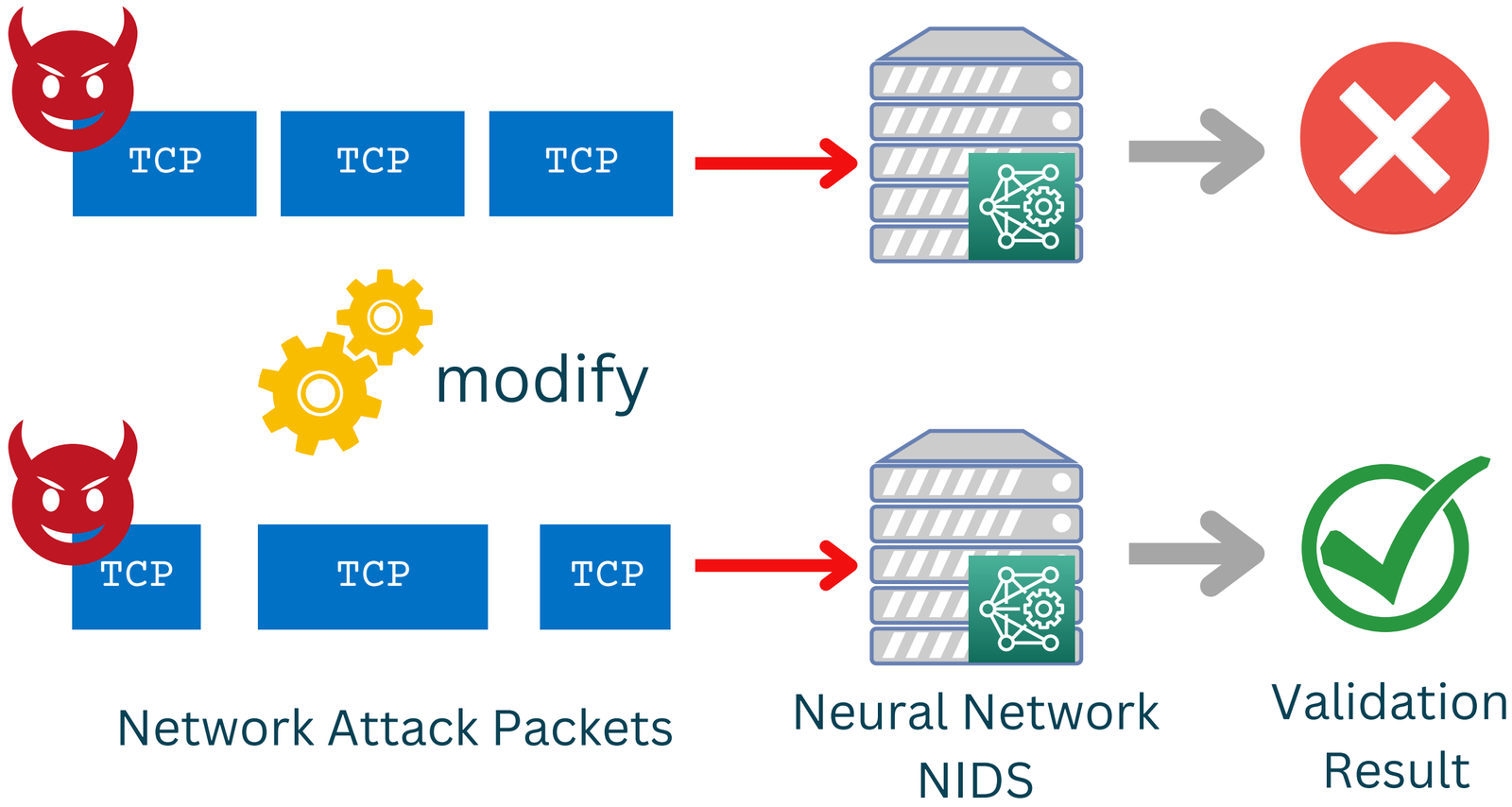
Adversarial Examples in the Image Domain



A Brief Introduction to Adversarial Examples [Mądry & Schmidt, gradientscience.org/intro_adversarial/, 2018]

Network Intrusion Detection

Adversarial Examples in the Network Domain



Network Intrusion Detection

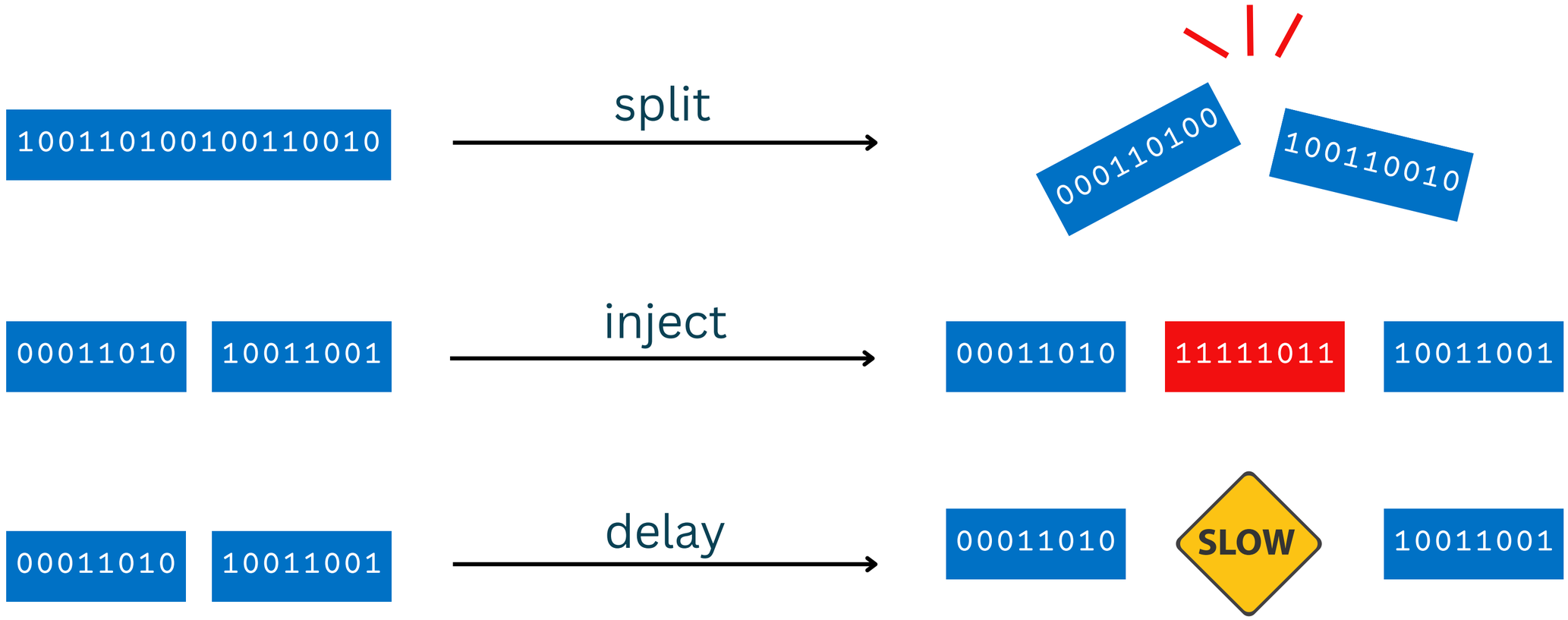
Generating Adversarial Examples in the Network Domain

- Packets must carry out original malicious intent effectively
- Packet transformations must not break the underlying protocols the attack relies on
- The attack must not be flagged as an intrusion by the anomaly-based NIDS

Towards Evaluation of NIDSs in Adversarial Setting [Hashemi et al. Big-DAMA '19]

Network Intrusion Detection

Network Transformations



Towards Evaluation of NIDSs in Adversarial Setting [Hashemi et al. Big-DAMA '19]