University of Colorado **Boulder**

Enabling Application-Specific Programmable Compute Infrastructure

Greg Cusack

Dissertation Defense
December 14, 2022
Boulder, Colorado

Microsoft Azure

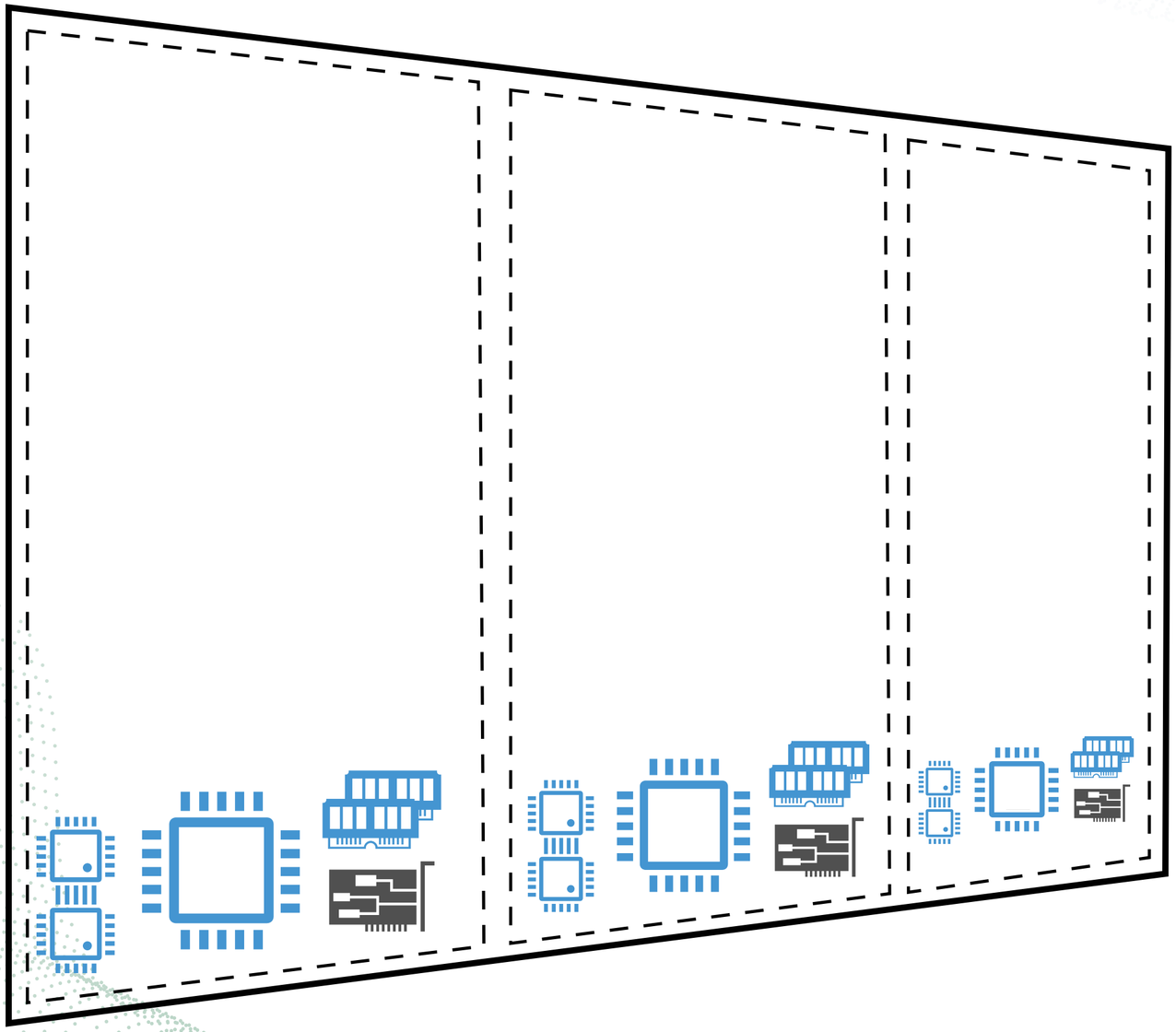
Google Cloud

Cloud Computing

- Demand is increasing
- Designed to support virtually all application types
- Simple and general

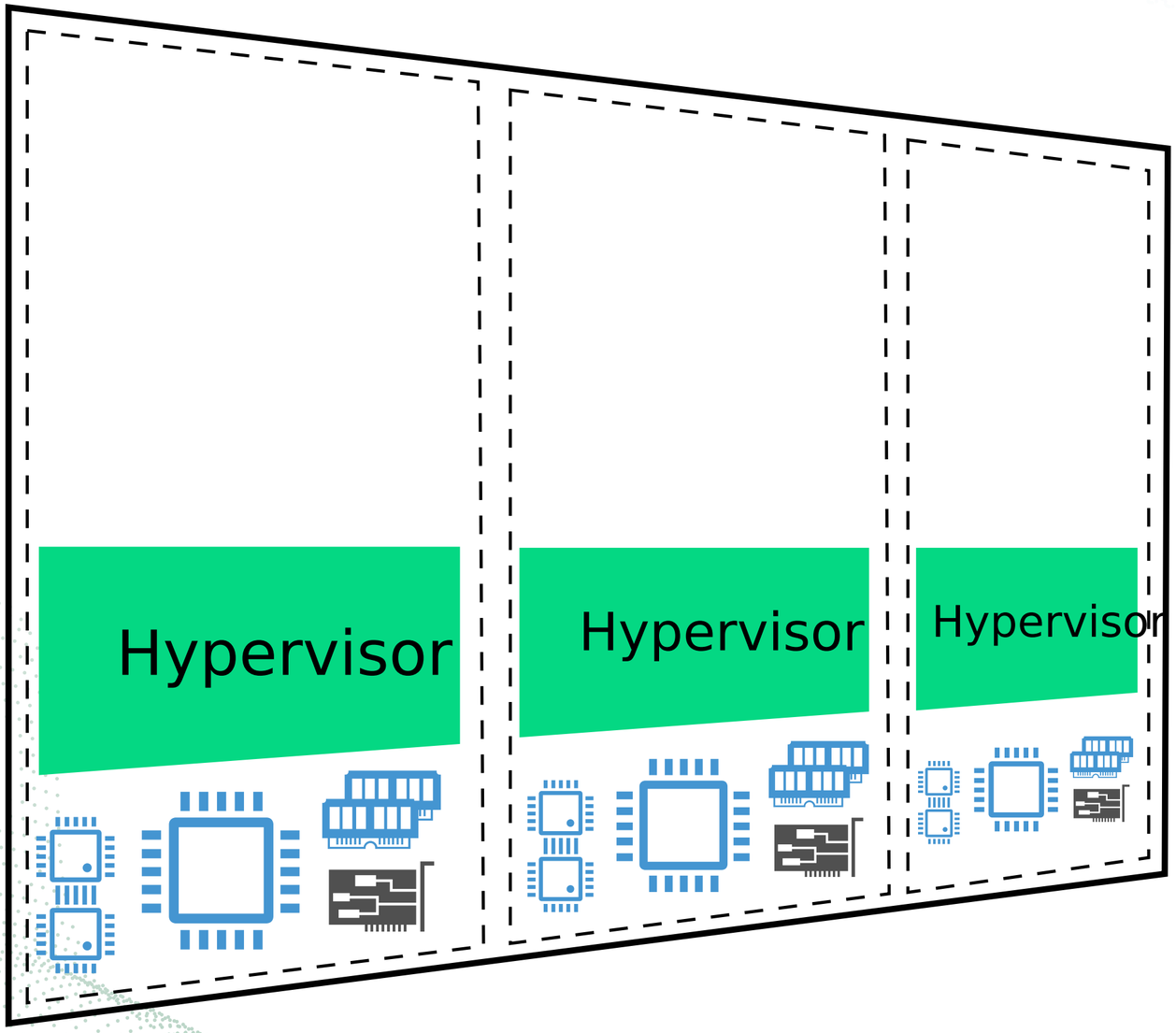
Cloud Architecture Overview

Cloud Architecture Overview



Compute Hardware

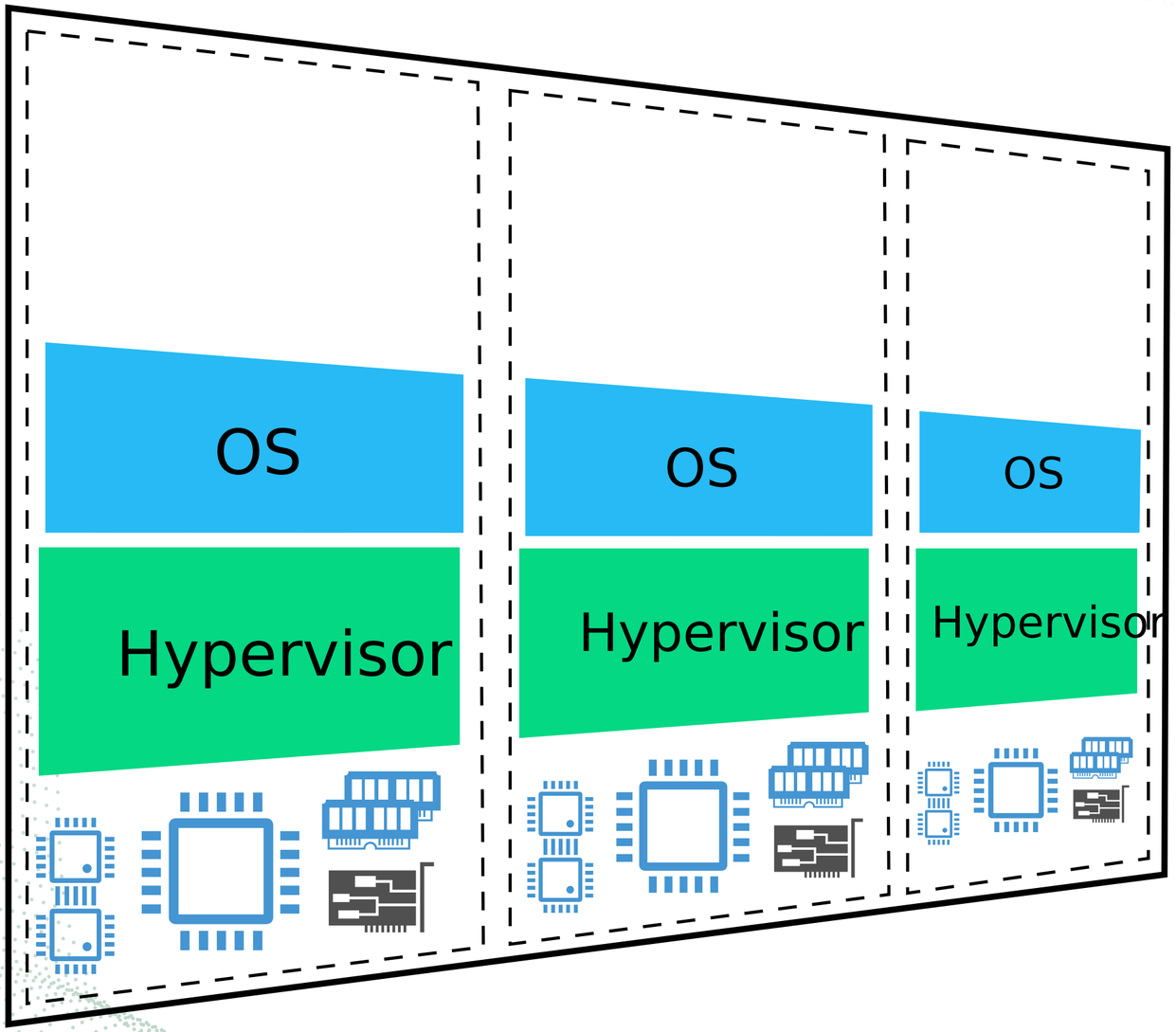
Cloud Architecture Overview



Hypervisor

Compute Hardware
(cpu, mem, NIC, secure hardware, etc)

Cloud Architecture Overview

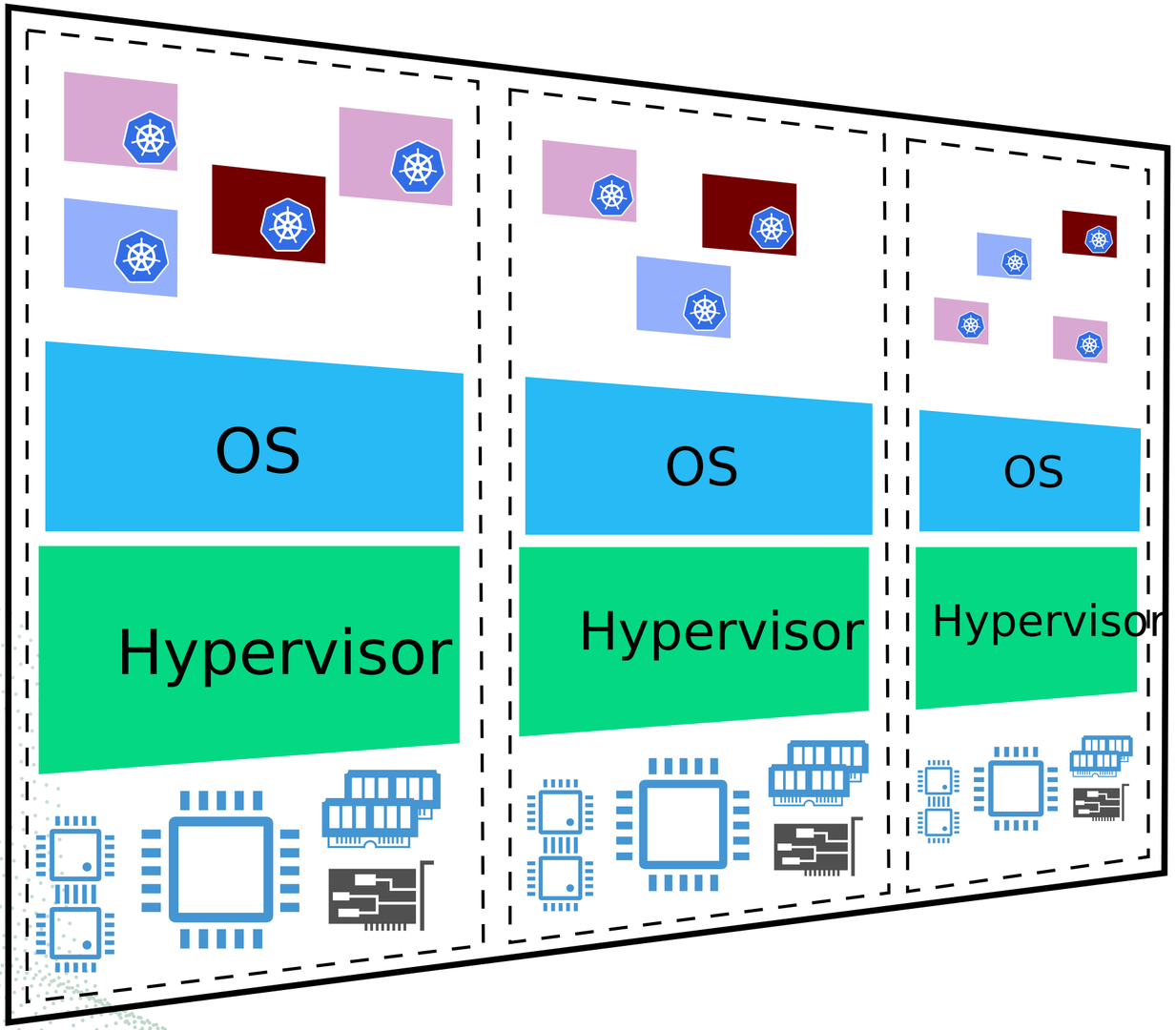


Operating System

Hypervisor

Compute Hardware
(cpu, mem, NIC, secure hardware, etc)

Cloud Architecture Overview

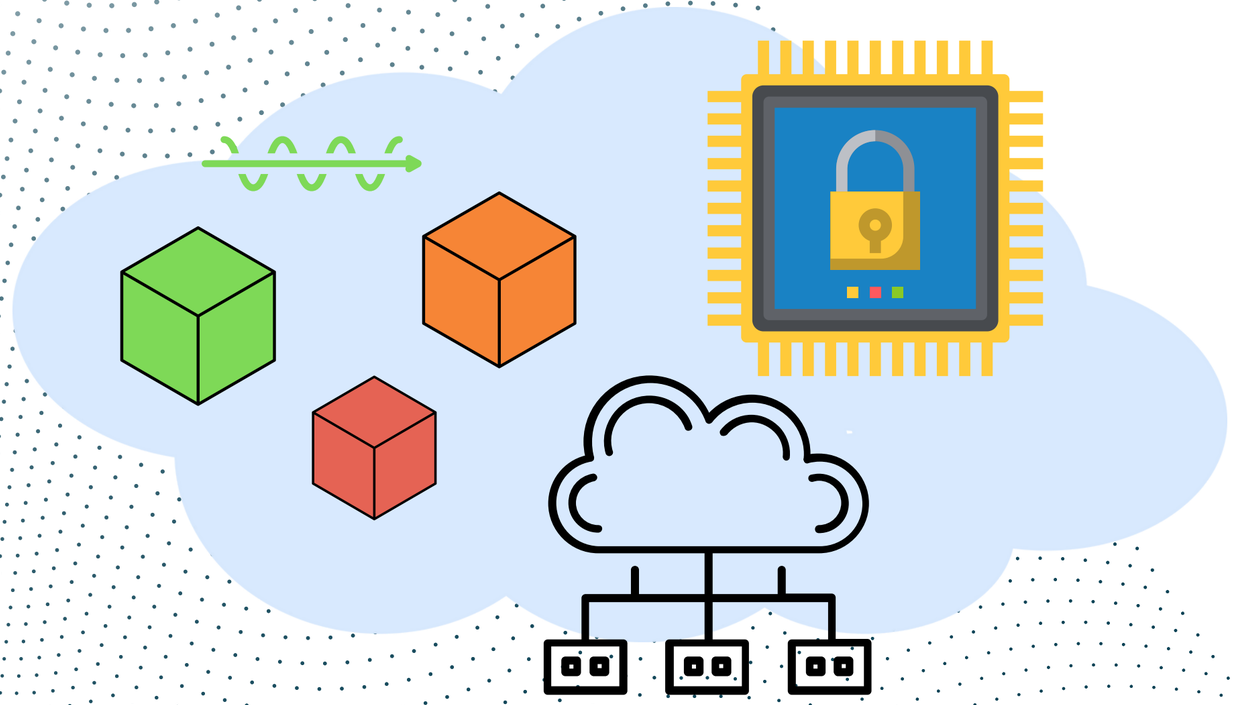


Application Code
(Containers)

Operating System

Hypervisor

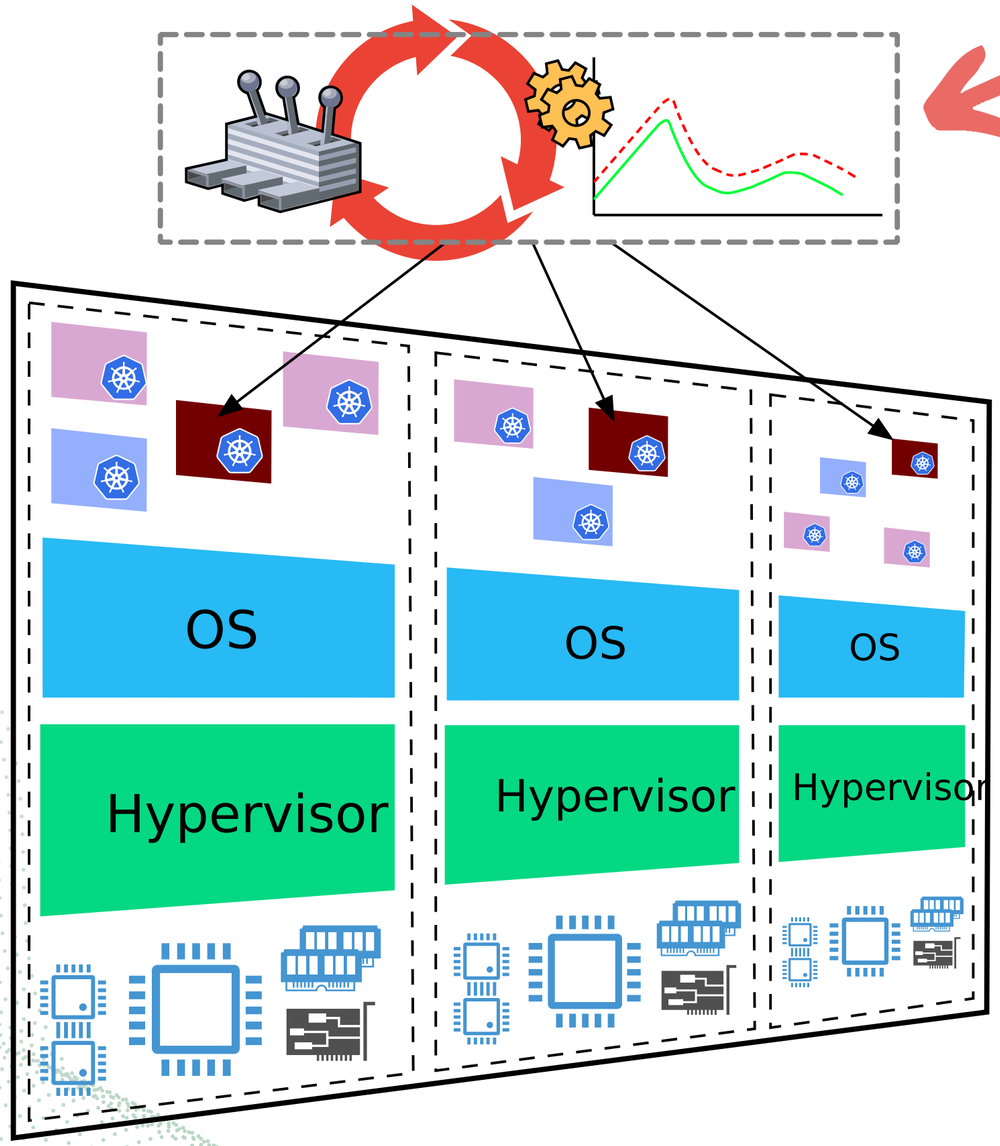
Compute Hardware
(cpu, mem, NIC, secure hardware, etc)



What's in an App?

- Containers
- Networks
- Security
- and more!

Application Management

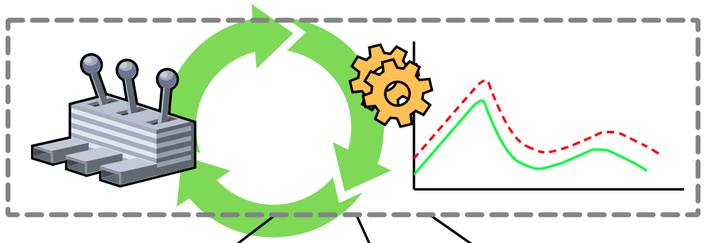


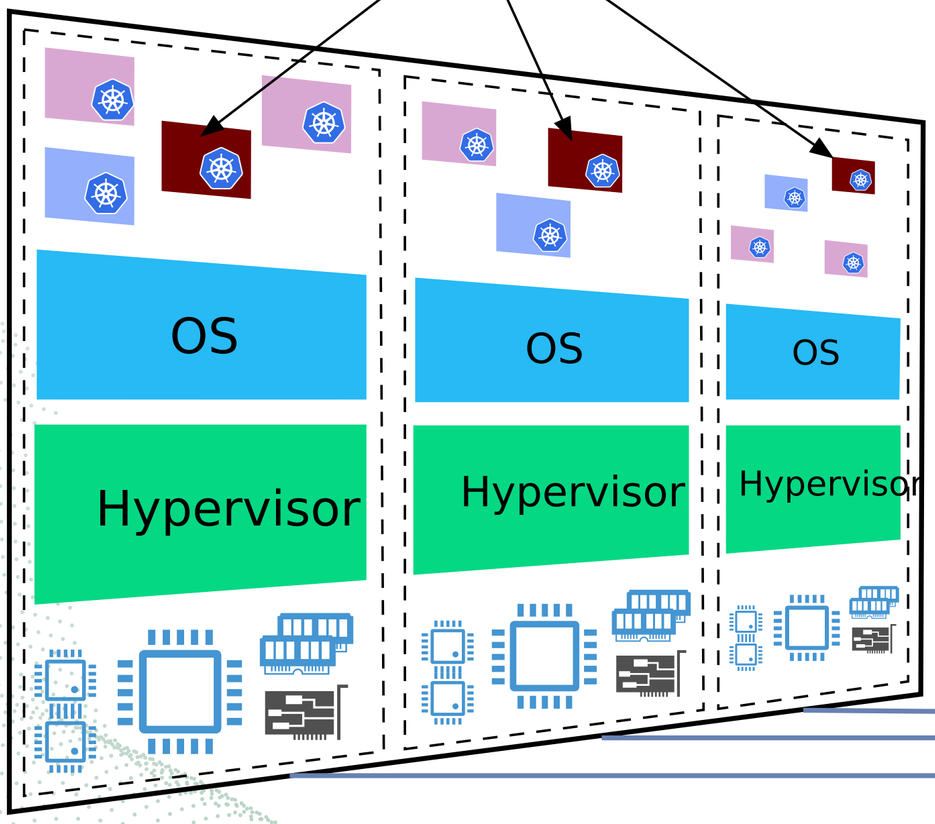
Container and Resource Allocation

University of Colorado **Boulder**

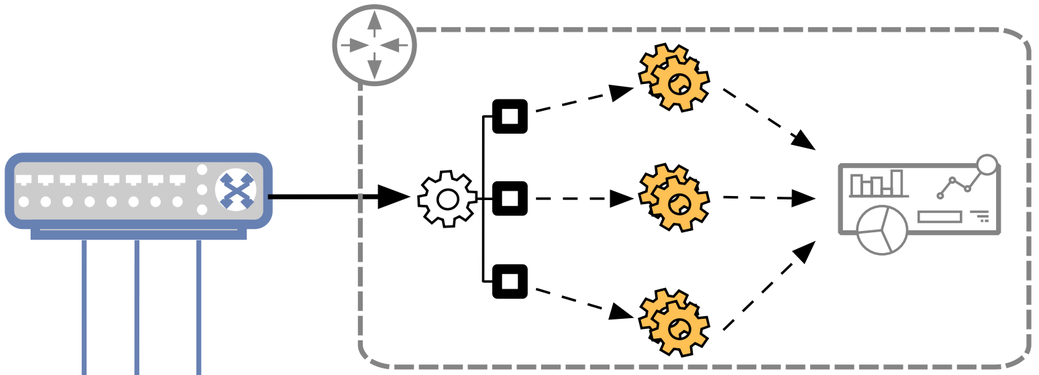
Cloud Architecture Overview

Container and Resource Allocation

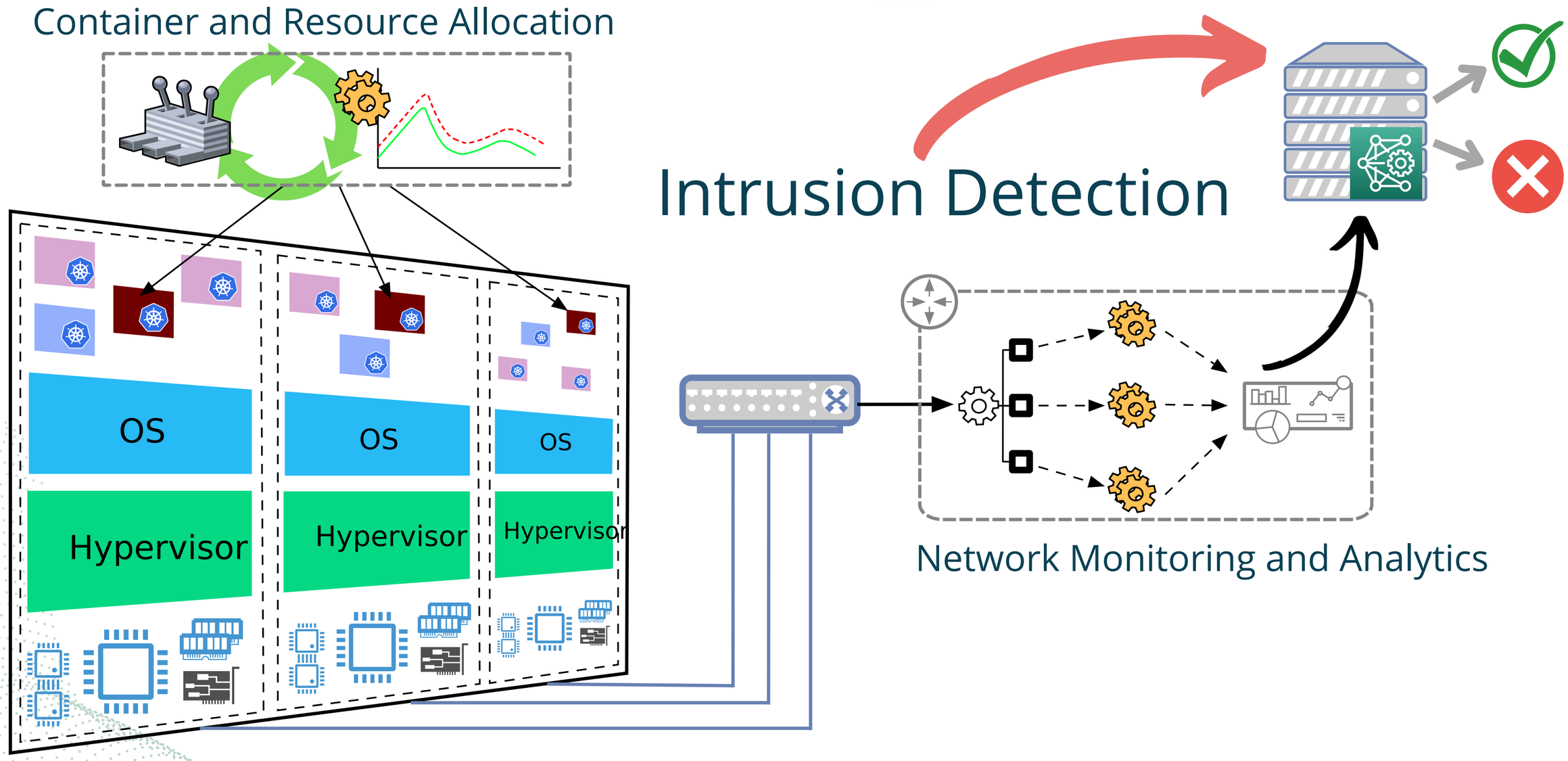




Network Monitoring and Analytics

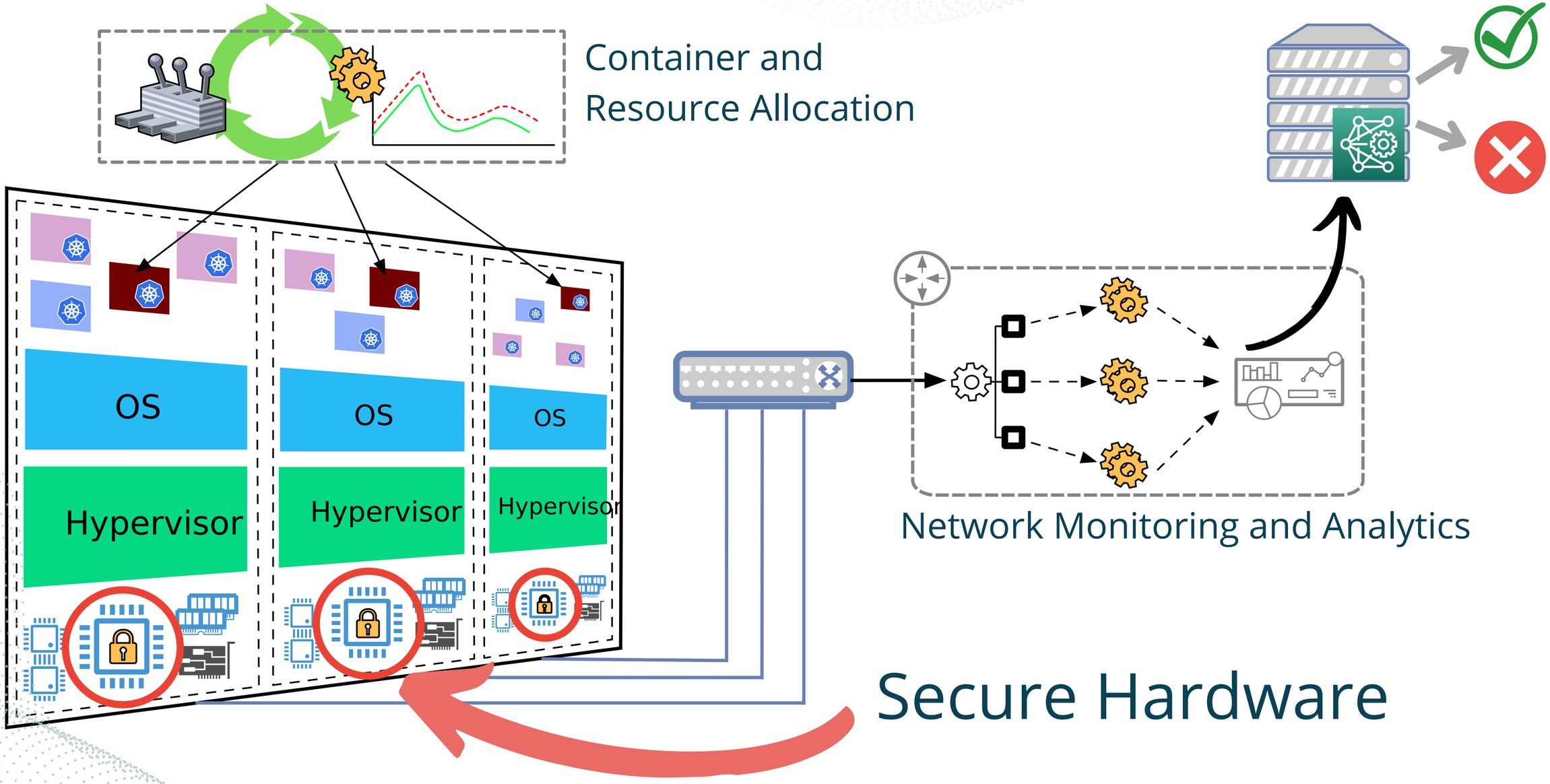


Cloud Architecture Overview



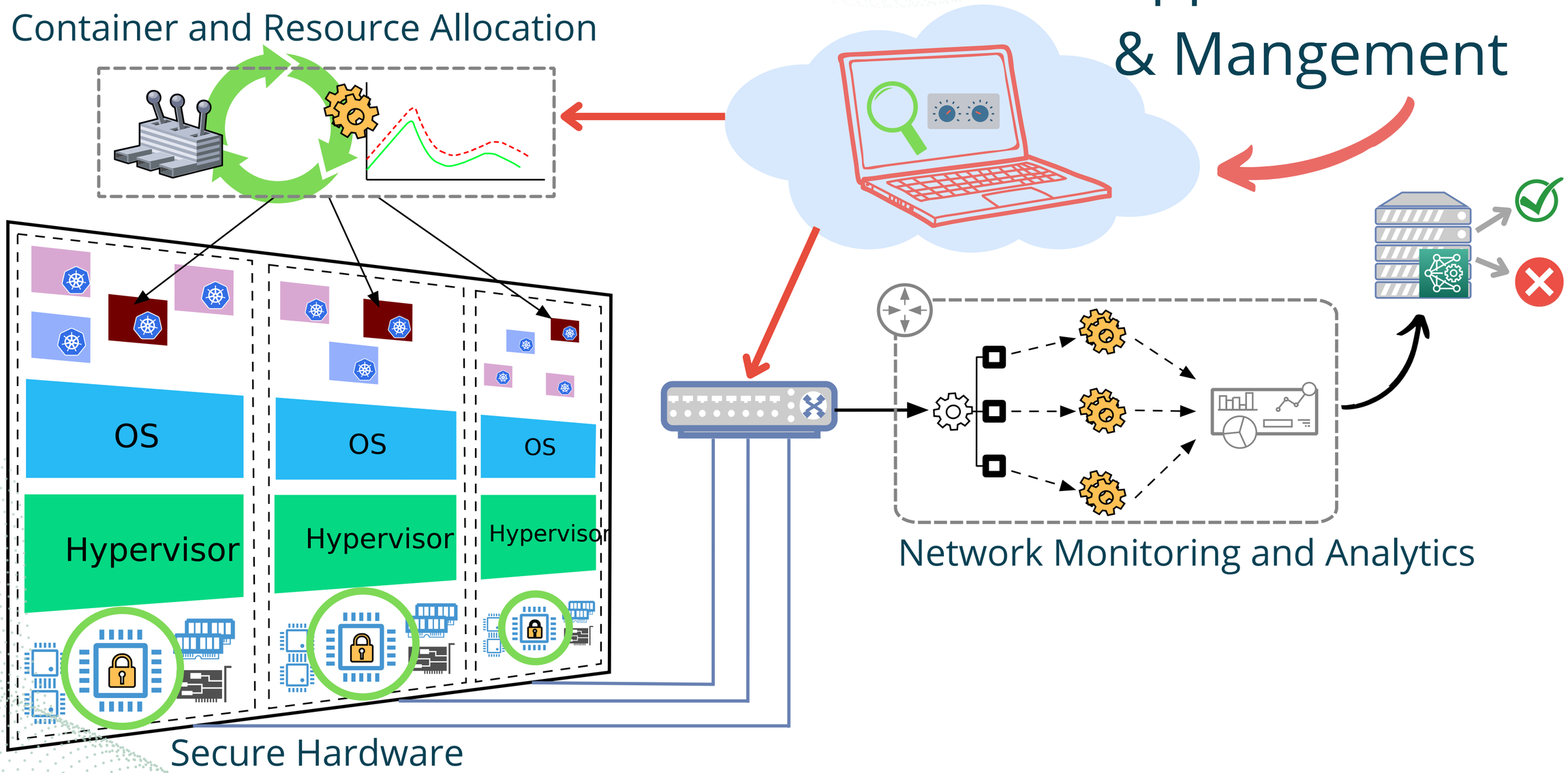
University of Colorado **Boulder**

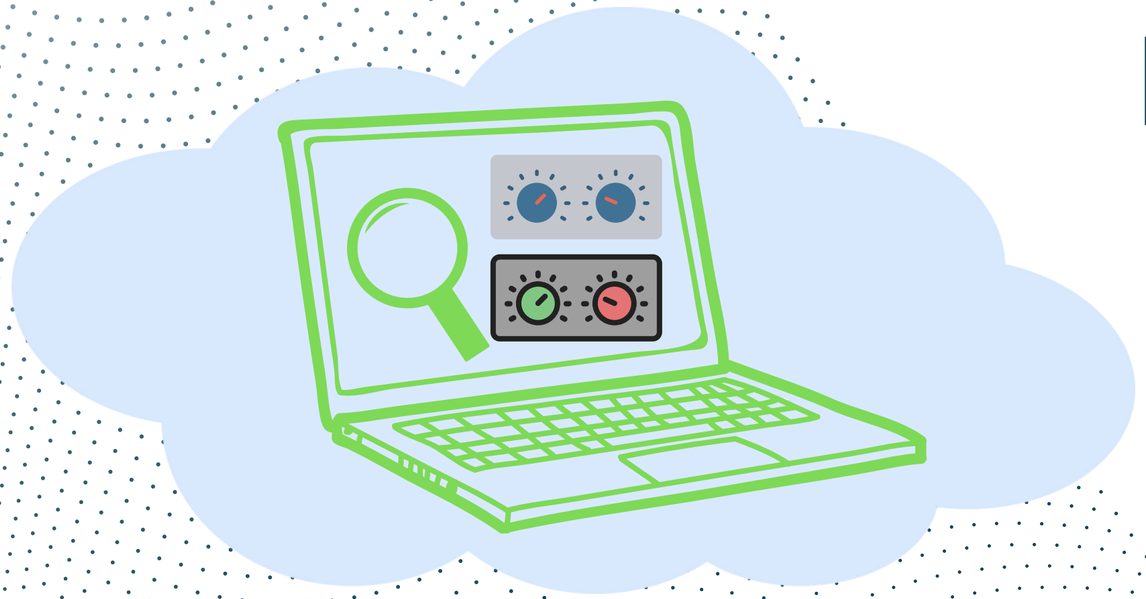
Cloud Architecture Overview



University of Colorado **Boulder**

Cloud Architecture Overview





Application Control & Management

- Entry point to any cloud application
- Collection of tools that are general and designed to be simple to use
- Secure, deploy, and monitor your applications
- Defined by the cloud provider

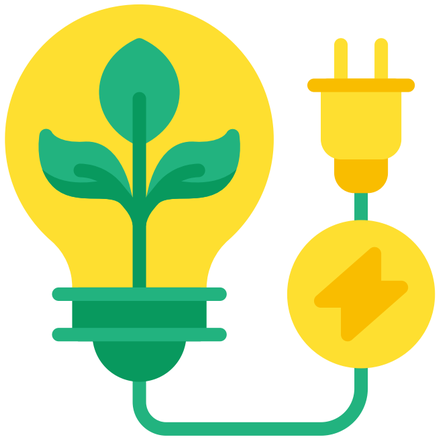
But here's the problem...

Developers *still* lack the controls to optimize their specific applications':

Security

Performance

Efficiency



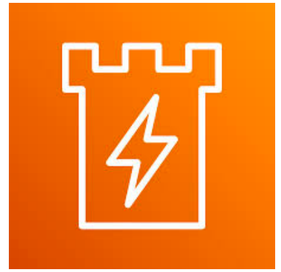
For example...

Let's look at some of AWS' cloud control and management offerings

AWS Nitro Enclaves

AWS' secure hardware offering

- Isolated computing environments
- Protect and securely process highly sensitive data
- Verifiable secure enclave computing



https://docs.aws.amazon.com/enclaves/latest/user/verify-root.html
https://aws.amazon.com/ec2/nitro/nitro-enclaves/

AWS Nitro Enclaves

Problem

- Doesn’t support remote attestation with a flexible root of trust.
- At the whim of Amazon to provide updates, new features, and bug fixes

AWS CloudWatch

AWS' network monitoring system

Available monitoring metrics…

Metric	Description
BytesDropCountBlackhole	The number of bytes dropped because they matched a `blackhole` route.
BytesDropCountNoRoute	The number of bytes dropped because they did not match a route.
BytesIn	The number of bytes received by the transit gateway.
BytesOut	The number of bytes sent from the transit gateway.
PacketsIn	The number of packets received by the transit gateway.
PacketsOut	The number of packets sent by the transit gateway.
PacketDropCountBlackhole	The number of packets dropped because they matched a `blackhole` route.
PacketDropCountNoRoute	The number of packets dropped because they did not match a route.

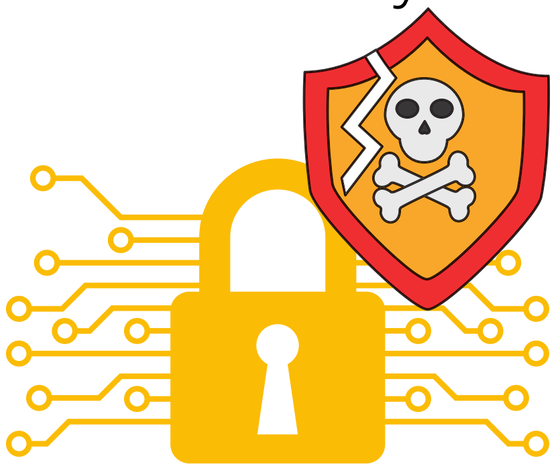
https://docs.aws.amazon.com/vpc/latest/tgw/transit-gateway-cloudwatch-metrics.html

AWS CloudWatch

Problem

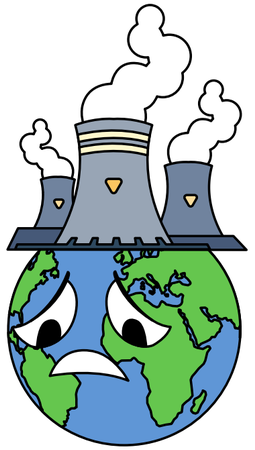
- Can only create simple alarms based on byte and packet counts
- Cannot perform root cause analysis or build security apps based on network traffic

Security



Performance

Efficiency



AWS EKS

AWS' container platform

- Supports vertical autoscaling
- Containers scale as compute demands change over time

https://docs.aws.amazon.com/eks/latest/userguide/autoscaling.html
https://github.com/kubernetes/autoscaler/blob/master/cluster-autoscaler/FAQ.md

AWS EKS

Problem

- Typically need to increase resources manually
- Slow to react, manual, requires container restart

Performance

Efficiency

Amazon EKS

Why are these services rigid and inflexible?

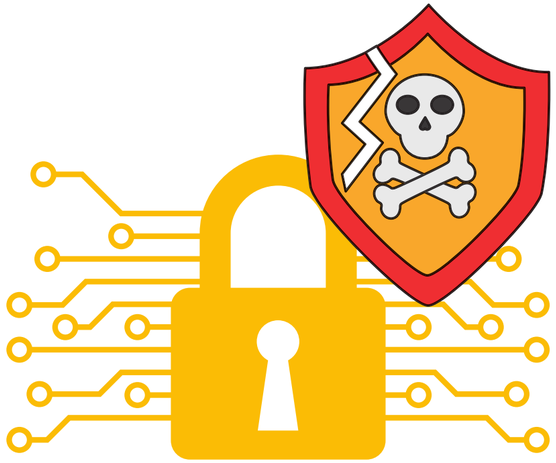
Why are these services rigid and inflexible?

- The underlying platforms simply do not support a high level of flexibility

- Cloud providers cannot provide flexibility if the underlying hardware or software systems do not support it

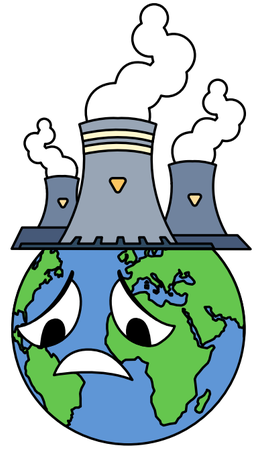
Rigidity results in poorly optimized application:

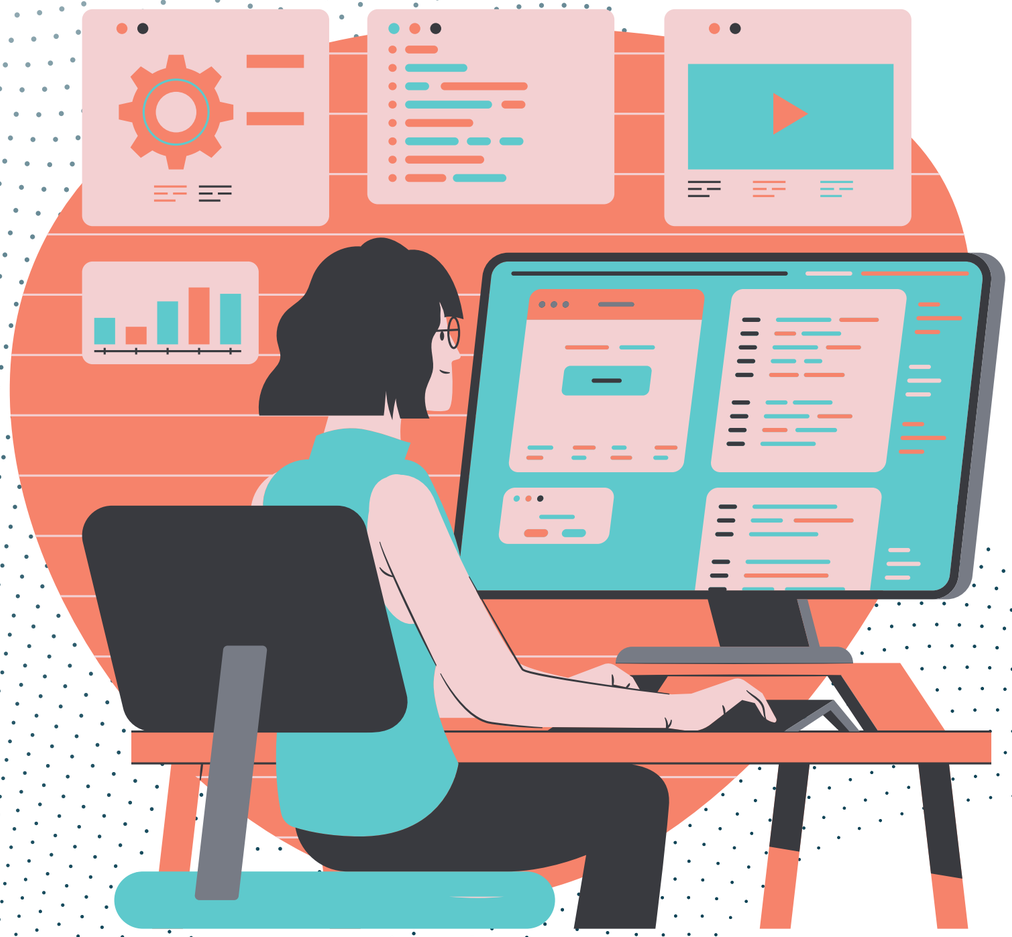
Security



Performance

Efficiency





Our thought

- What if we could create and then expose that flexible underlying platform to developers?

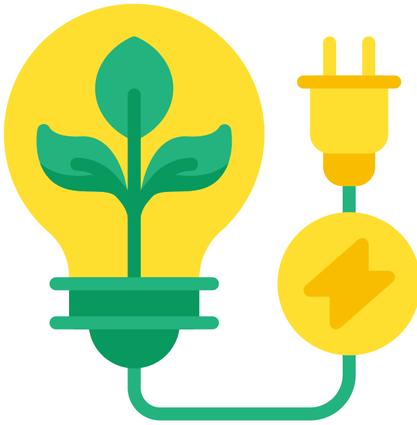
For the developer

- Users could then finely control the underlying compute systems
- Enables them to optimize their specific applications':

Security

Performance

Efficiency



For the cloud provider

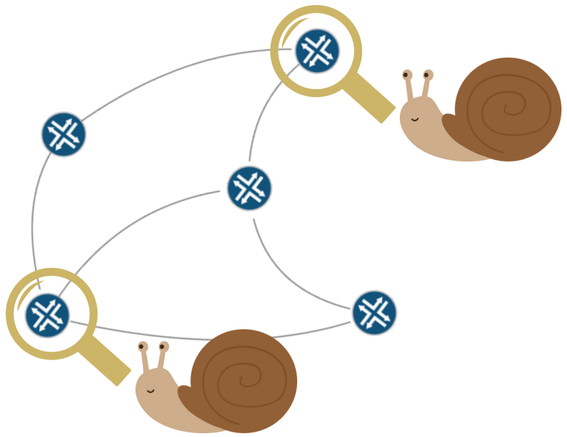
The cloud provider can then build abstractions on top of our implementations

In this dissertation, we…

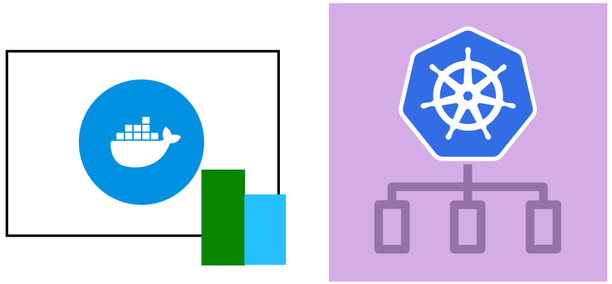
1) Identify the shortcomings and rigidity of the underlying systems that control and manage:

Secure Hardware

Network Monitoring



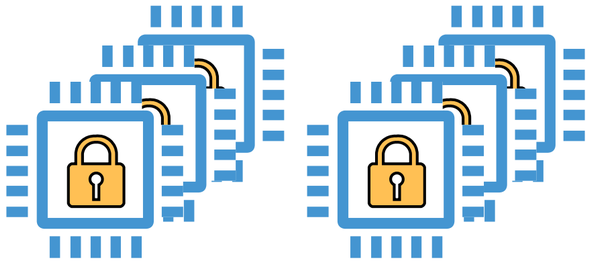
Compute Resources



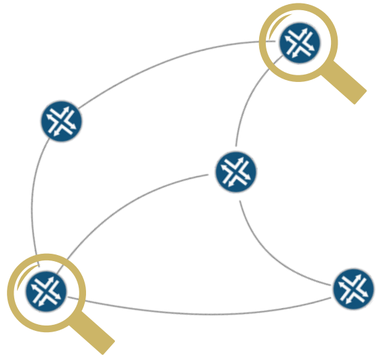
In this dissertation, we…

2) Build new, programmable platforms and systems that allow users to design and implement application-specific:

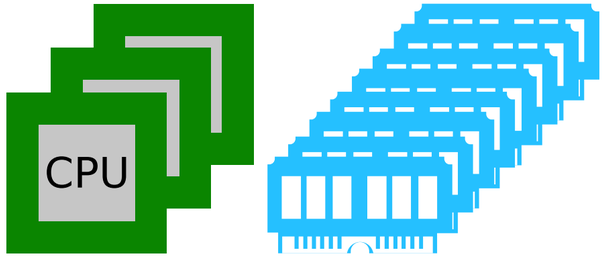
Secure Hardware Features



Network Monitoring Applications

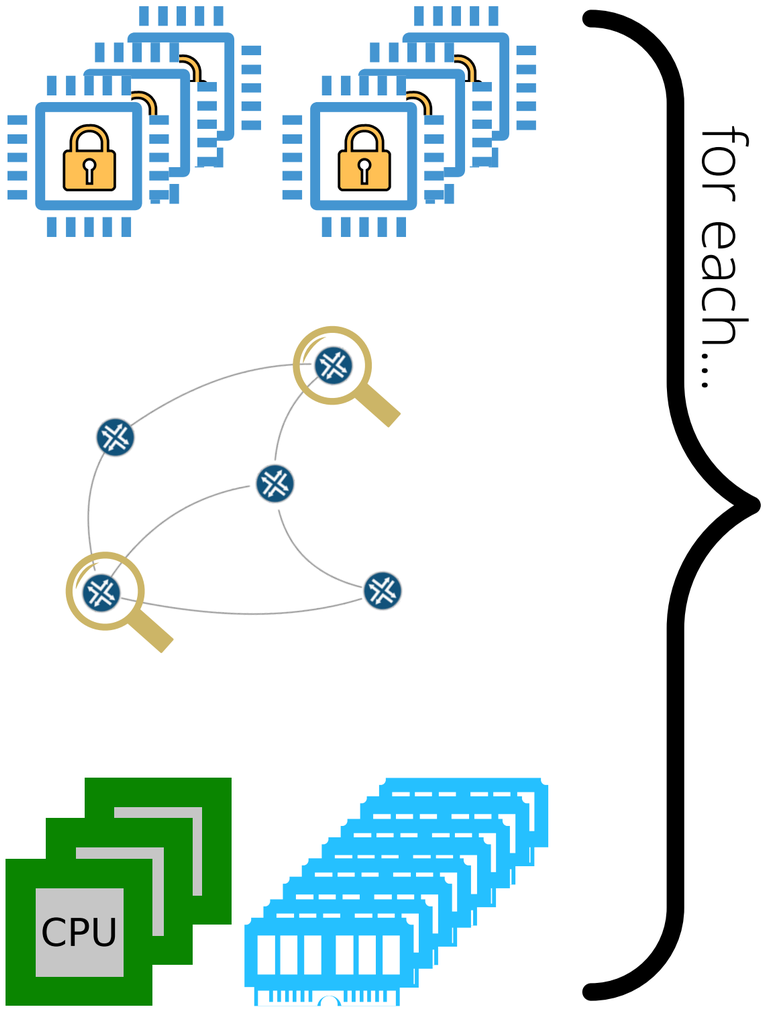


Compute Resource Allocation Decisions

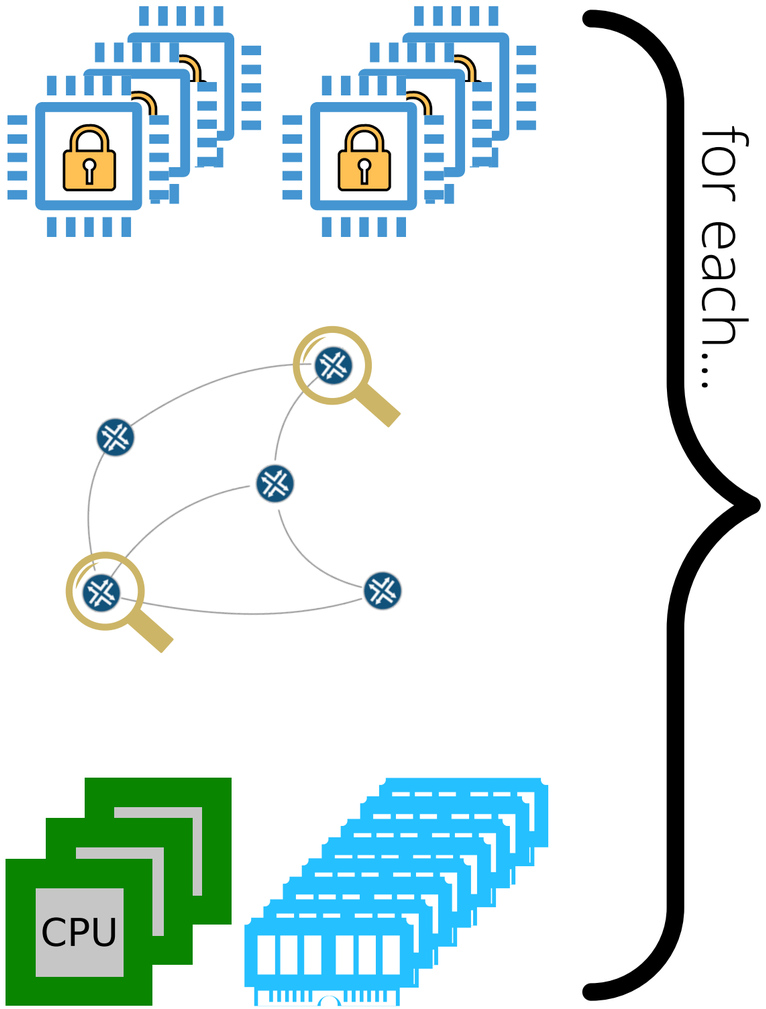


Outline

1) Current research/relevant work



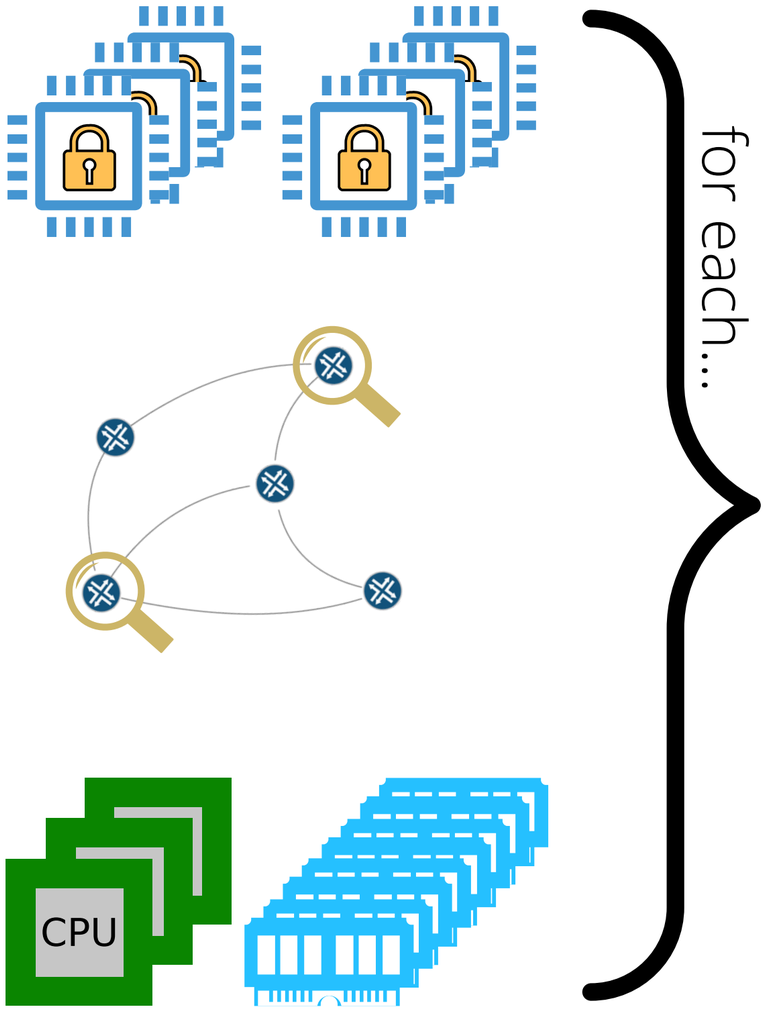
Outline



1) Current research/relevant work

2) How the rigidity of the underlying platform prevents application security, performance, and/or efficiency optimization

Outline



1) Current research/relevant work

2) How the rigidity of the underlying platform prevents application security, performance, and/or efficiency optimization

3) Our solution that enables users to define customized, application-specific systems that optimize their applications' security, performance, and/or efficiency