

Playing with AWS Firecracker VMM (Part 2)

之 抓住夏天的尾巴

之 還是要捲起袖子動手玩

...

Ernest Chiang @ Cloud Native Taiwan User Group

Aug 31, 2020

Give me a place to stand on, and I will move the Earth.

—Archimedes

sli.do

- #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2
- #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2
- #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2 #awsvmm2
- 議程中有任何問題、好奇、疑問，都可以隨時丟進 sli.do
- US\$25 AWS Credits 問券連結，也放在 sli.do 裡頭喔



Ernest Chiang

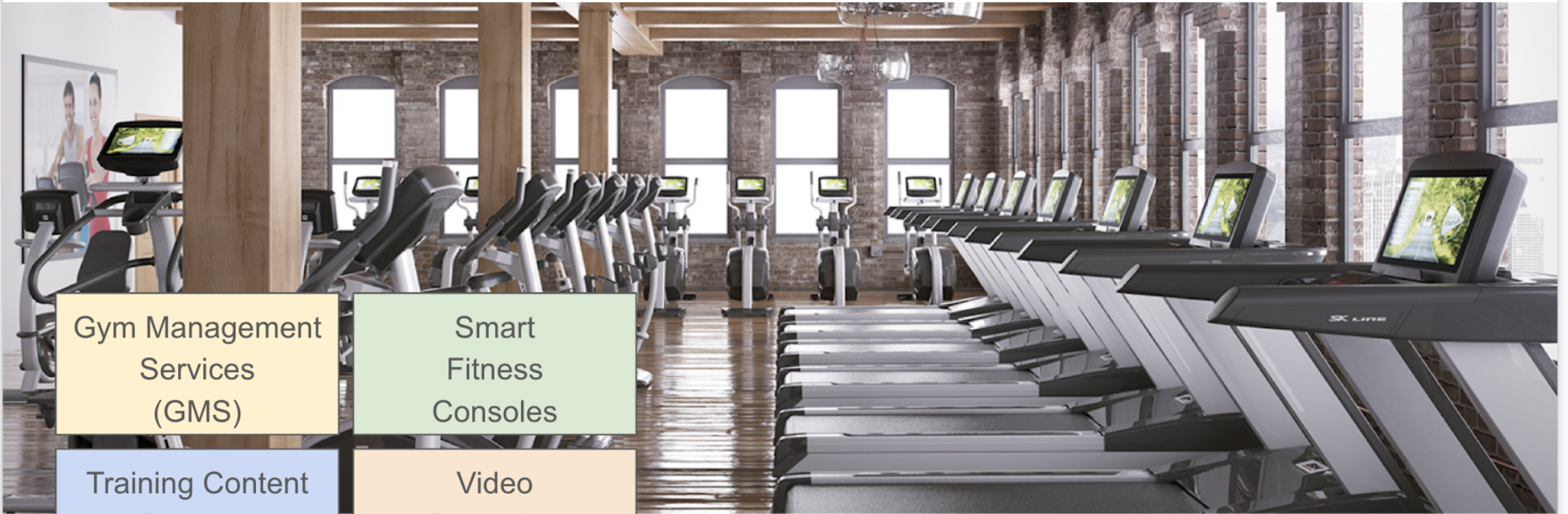
Worked on process integration engineering in semiconductor industry @tsmc.

Doing product and technology integration in fitness industry @pafers.

Off Work TGO Networks Taipei. AWS Community Hero. Mozillian. AIESECer.

PAFERS Fitness Services (PFS)

The integrated products and services for fitness.



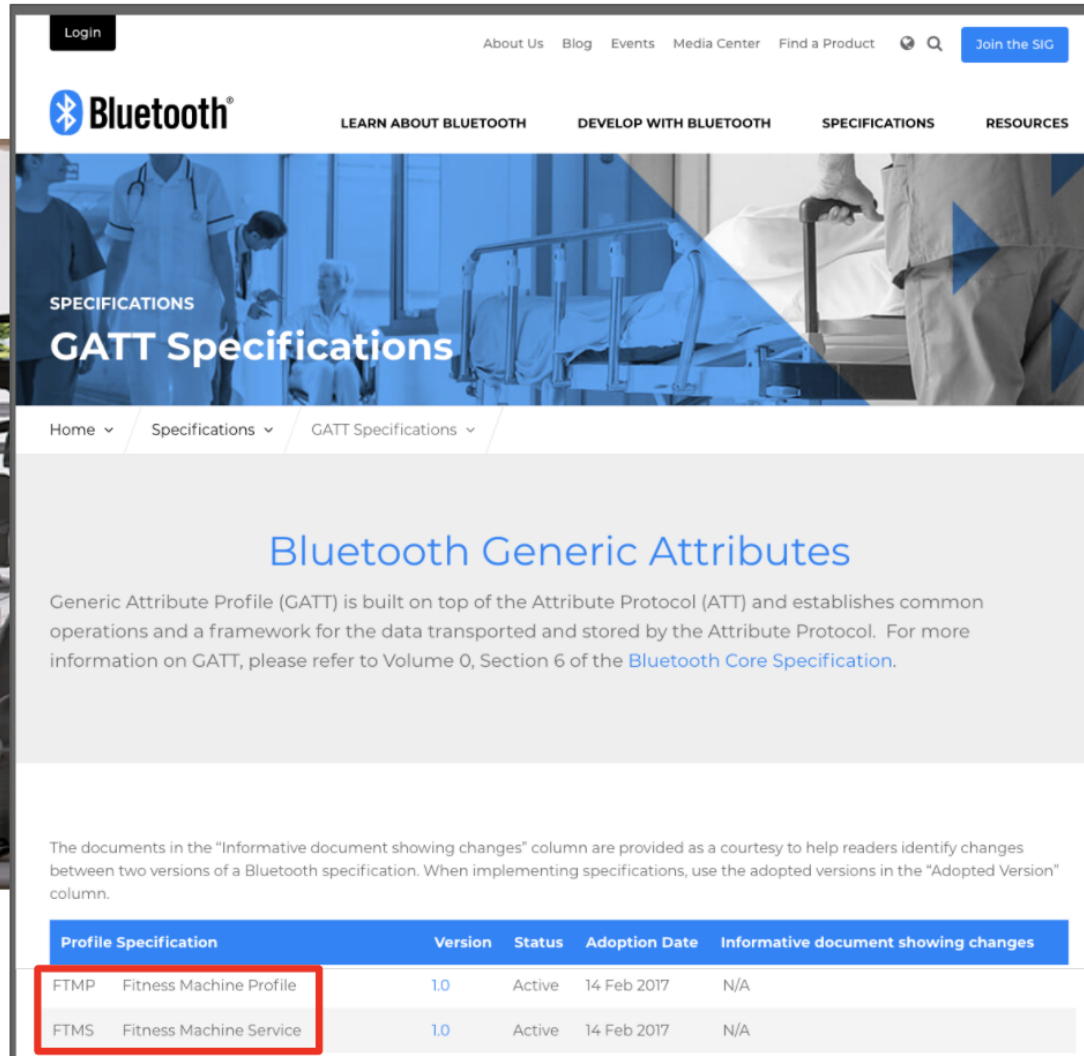
Gym Management
Services
(GMS)

Smart
Fitness
Consoles

Training Content
Services
(TCS)

Video
Streaming
Services

PAFERS Contributes Industry Knowledge to Bluetooth Spec



Bluetooth

LEARN ABOUT BLUETOOTH DEVELOP WITH BLUETOOTH SPECIFICATIONS RESOURCES

SPECIFICATIONS

GATT Specifications

Home Specifications GATT Specifications

Bluetooth Generic Attributes

Generic Attribute Profile (GATT) is built on top of the Attribute Protocol (ATT) and establishes common operations and a framework for the data transported and stored by the Attribute Protocol. For more information on GATT, please refer to Volume 0, Section 6 of the [Bluetooth Core Specification](#).

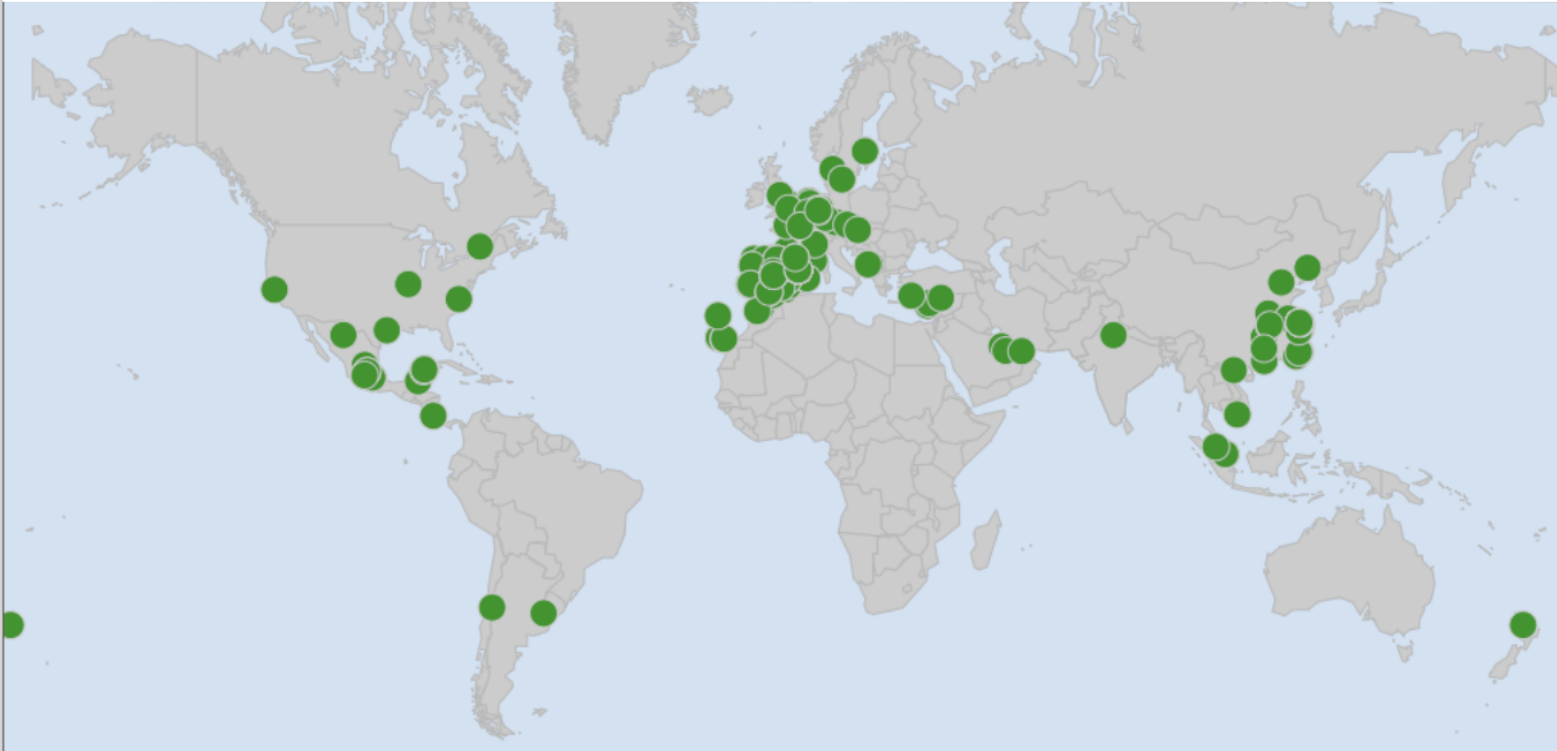
The documents in the "Informative document showing changes" column are provided as a courtesy to help readers identify changes between two versions of a Bluetooth specification. When implementing specifications, use the adopted versions in the "Adopted Version" column.

Profile Specification	Version	Status	Adoption Date	Informative document showing changes
FTMP Fitness Machine Profile	1.0	Active	14 Feb 2017	N/A
FTMS Fitness Machine Service	1.0	Active	14 Feb 2017	N/A



<https://www.bluetooth.com/specifications/gatt/>

Global PAFERS Edges Footprints



Outline

- Part 1: COSCUP 2020
- Part 2: Cloud Native Taiwan User Group
- Firecracker
 - Virtualization & Containerization
- Lambda & Fargate
- Firecracker & container **d**
- Firecracker & Open Source Projects (Weave Ignite)

Part 1: COSCUP 2020

- [Blog](#)
- [Slides](#)
- [Youtube](#)

Part 2: Cloud Native Taiwan User Group

Firecracker



What is Firecracker

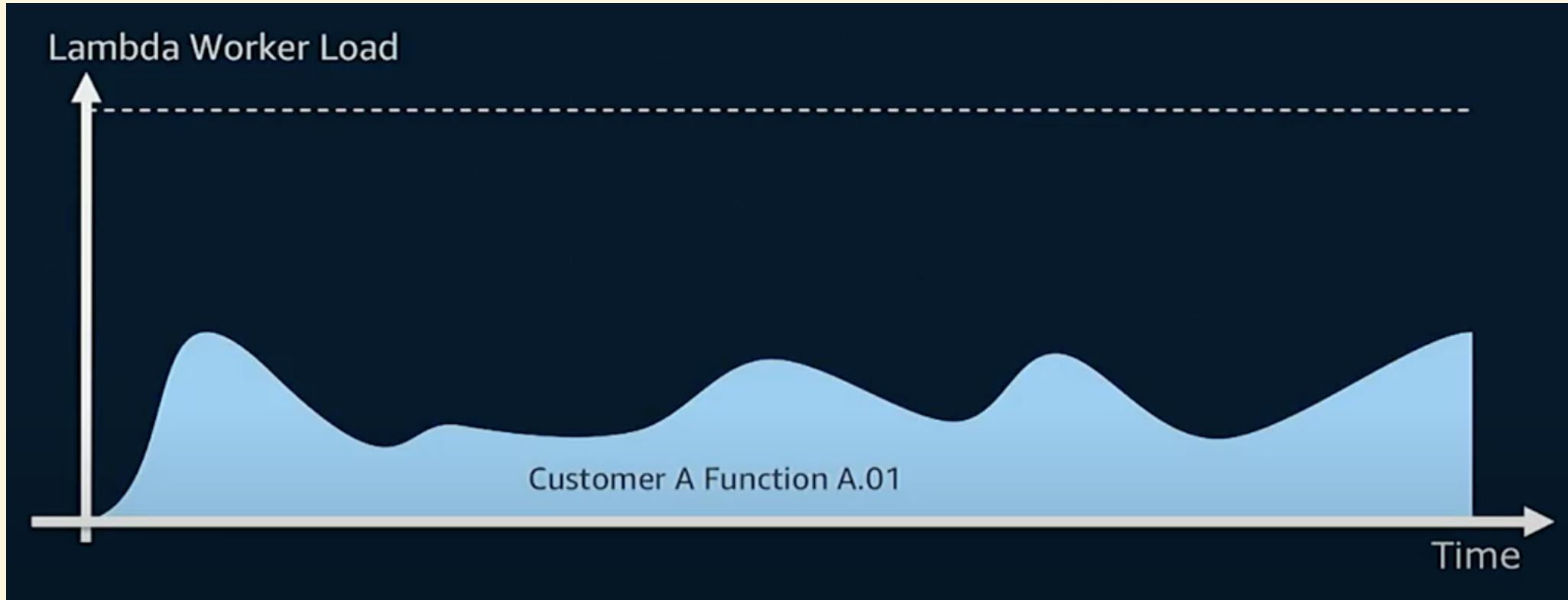
Firecracker is an open source VMM that is purpose-built for creating and managing secure, multi-tenant container and function-based services.



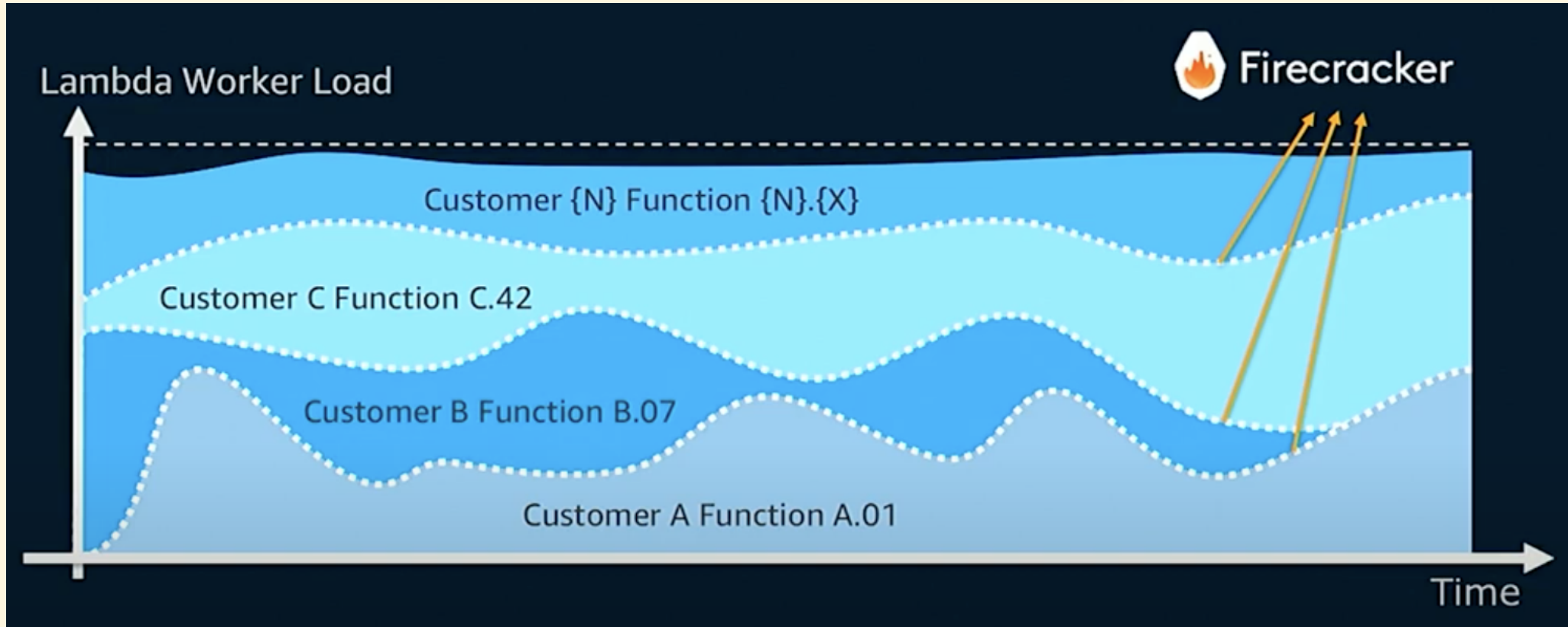
What is Firecracker

Firecracker is an open source **VMM** that is **purpose-built** for creating and managing secure, multi-tenant container and function-based services.

What problem is AWS helping to solve?



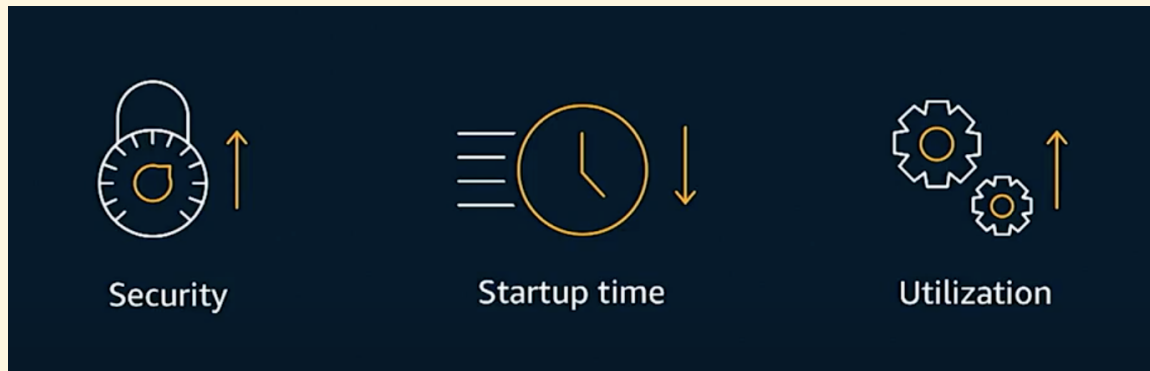
What problem is AWS helping to solve?



What problem is AWS helping to solve?

- Multiple functions
- on multiple environments
- from multiple accounts.

Firecracker



- Open source virtualization technology (microVM)
- Security & isolation of traditional VMs
- Speed & density of containers
- Low resource overhead
- Developed at Amazon

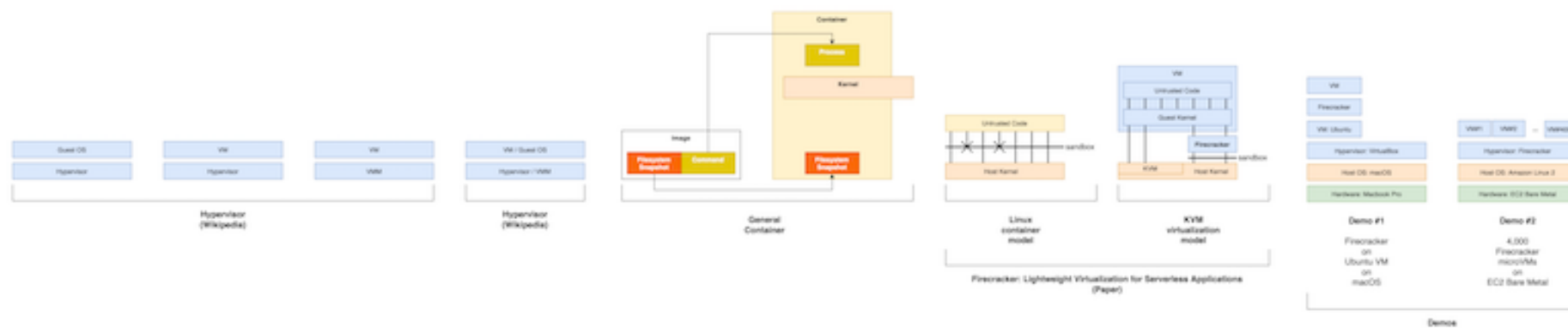
Firecracker Facts

- Started with a branch of [crosvm](#)
 - Removed >50% of the code
- 96% fewer lines of code than QEMU
- Simplified device model
 - no BIOS, no PCI, etc
- Apache 2.0 license

Virtualization

&

Containerization

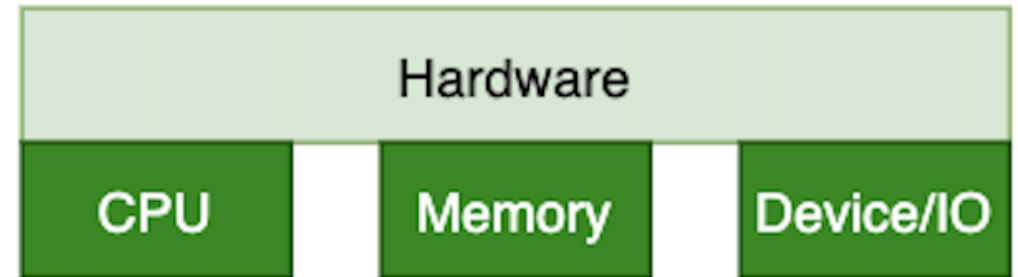


Virtualization
Architecture
Collection
latest updated: 2020/7/31
made by zsmachang.com

1. Virtualization

Creating a **virtual** version of **something**:

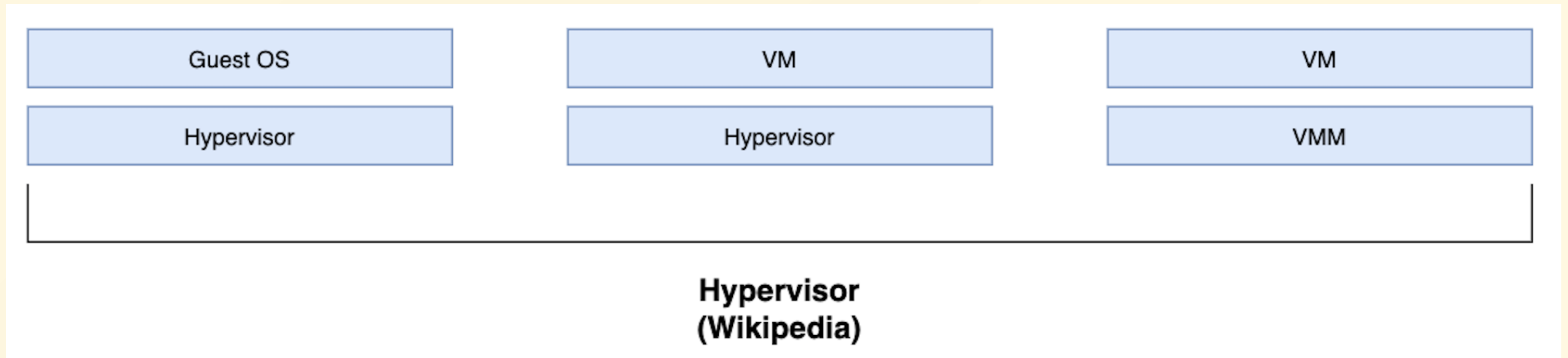
- CPU
- Memory
- Device/IO (Storage, NIC)



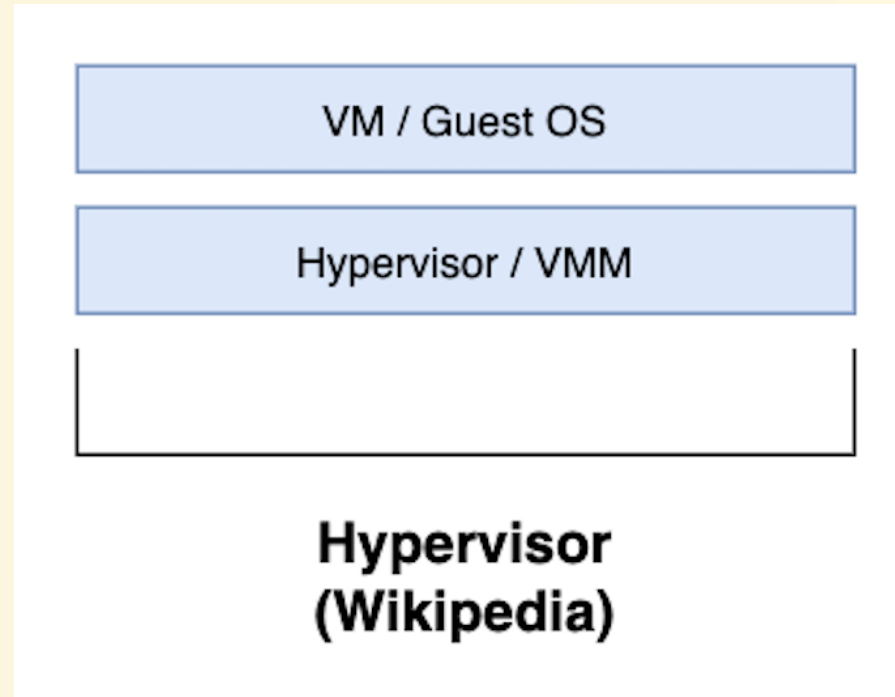
2. Hypervisor (1/3)

A `hypervisor` (or `virtual machine monitor`, `VMM`, virtualizer) is computer software, firmware or hardware that creates and runs virtual machines.

2. Hypervisor (2/3)



2. Hypervisor (3/3)



3. KVM

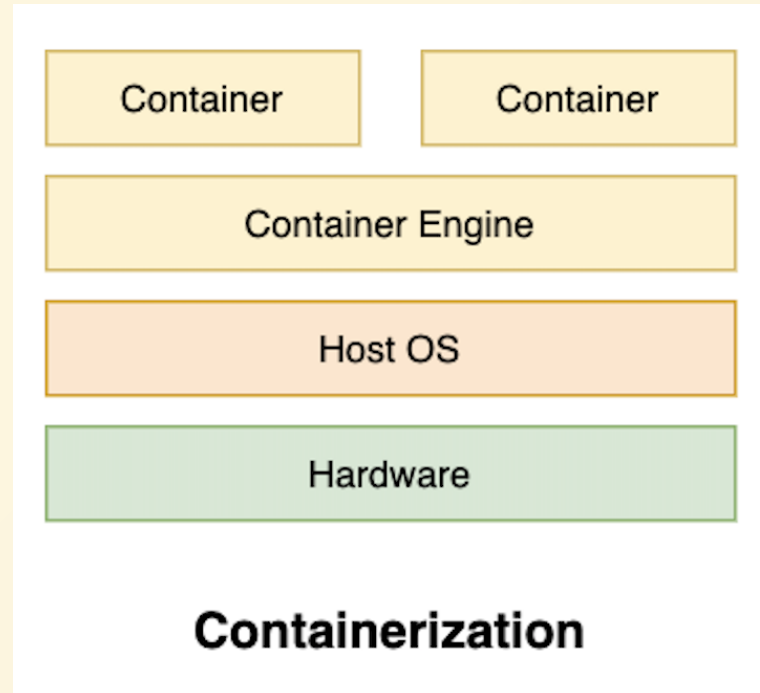
Kernel-based Virtual Machine (KVM) is

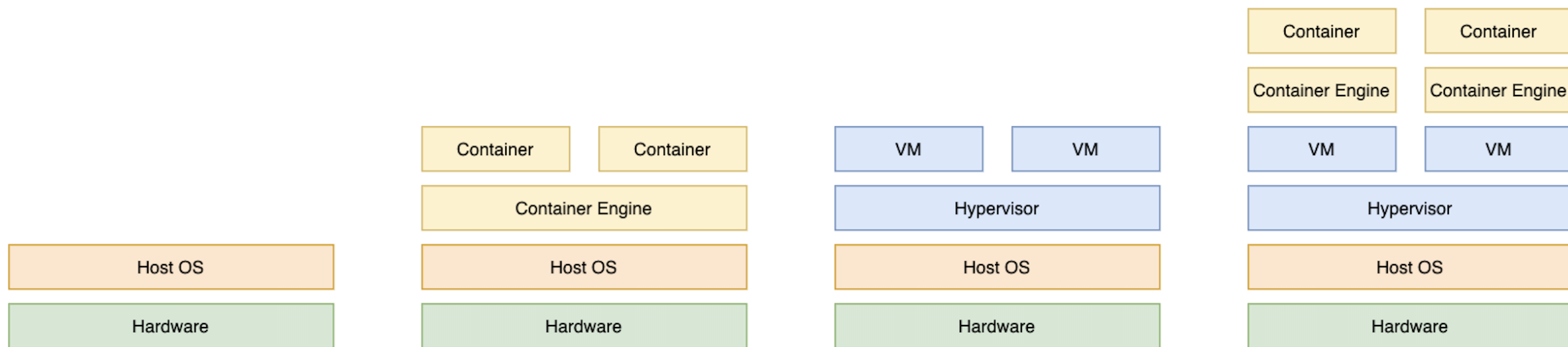
a virtualization module in the Linux kernel
that allows the kernel to function as a hypervisor.

4. Containerization (1/n)

- [OS-level virtualization](#), also known as containerization, refers to an OS feature in which the **kernel** allows the existence of multiple `isolated user-space instances`.
- Such instances, called `containers`, partitions, virtual environments (VEs) or `jails` (**FreeBSD jail** or **chroot jail**), may look like real computers from the point of view of programs running in them.

4. Containerization





Bare Metal

**Container on
Bare Metal**

**VM on
Bare Metal**

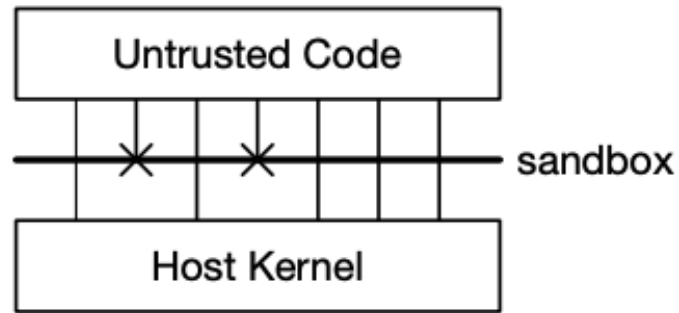
**Container on
VM**

**Common
Combinations**

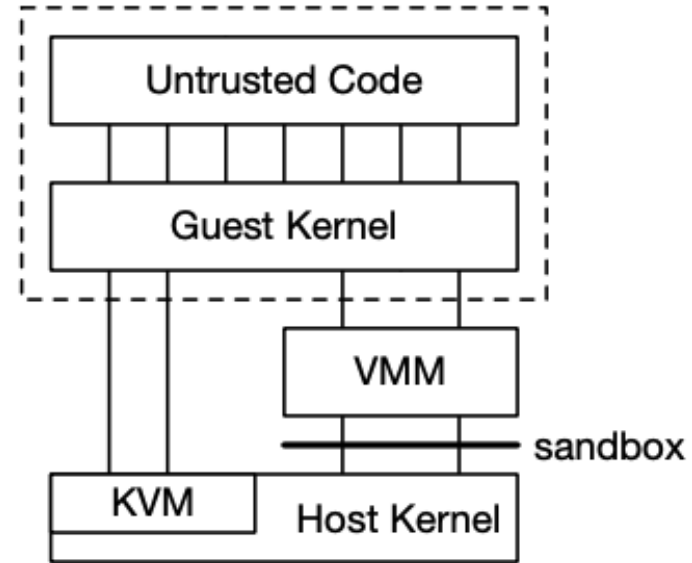
Back to Firecracker

A purpose-built open source VMM

Firecracker Security Models (1/2)

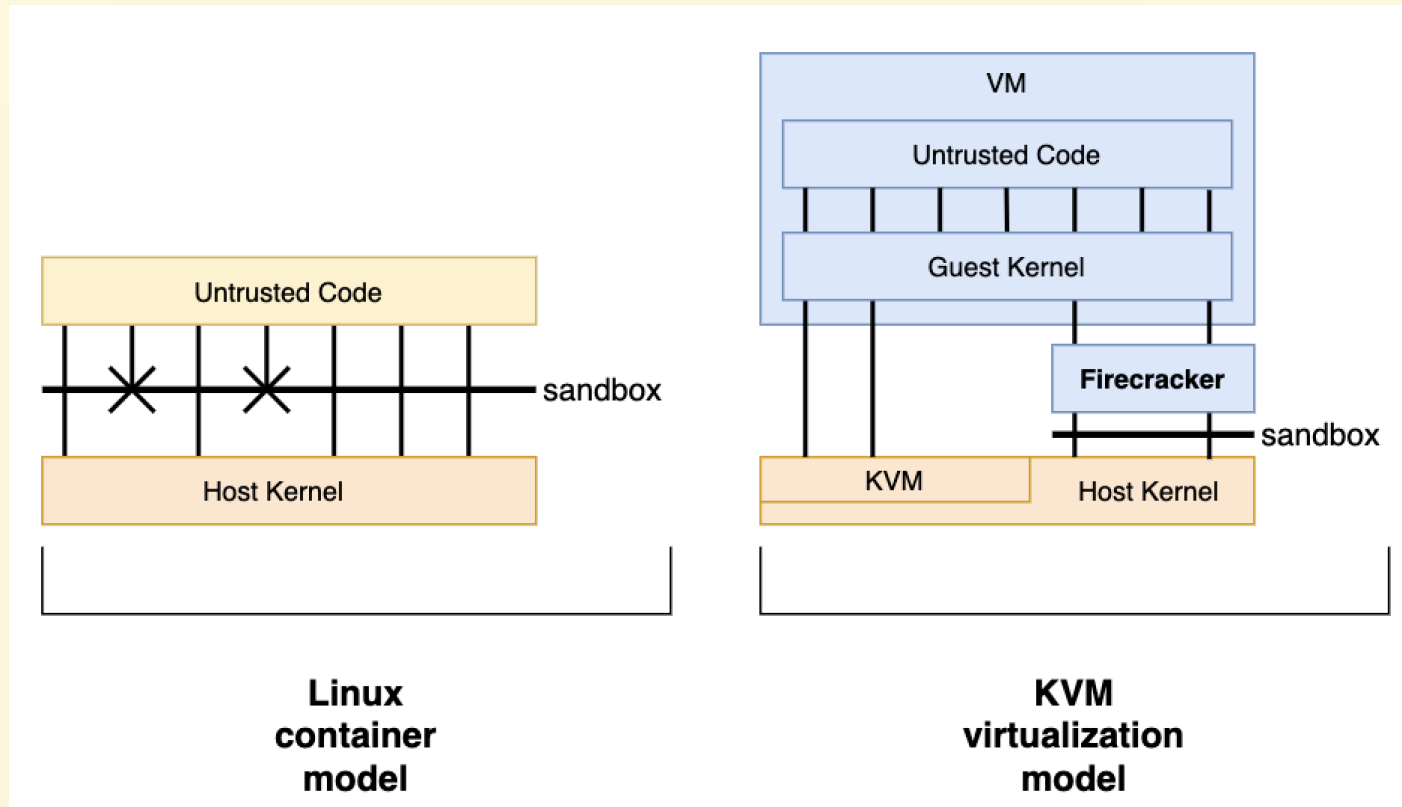


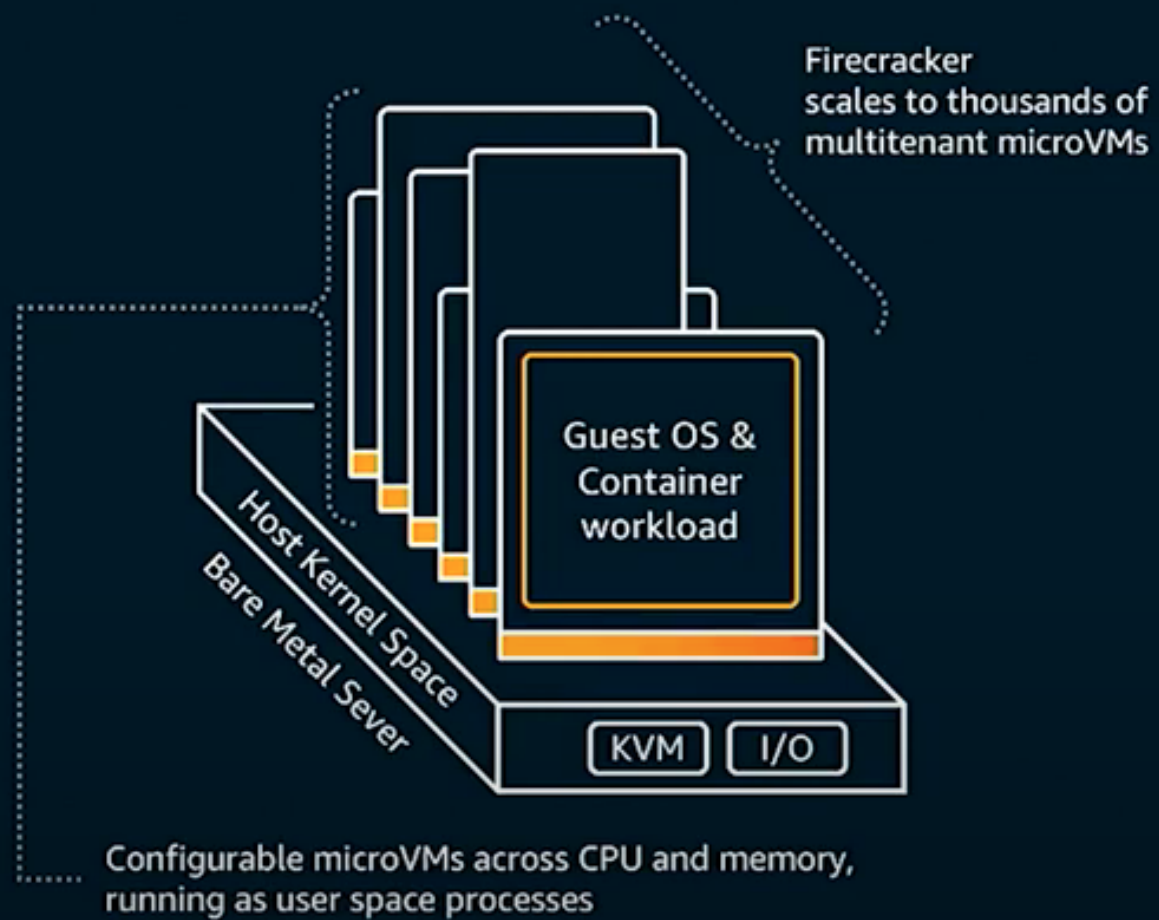
(a) Linux container model

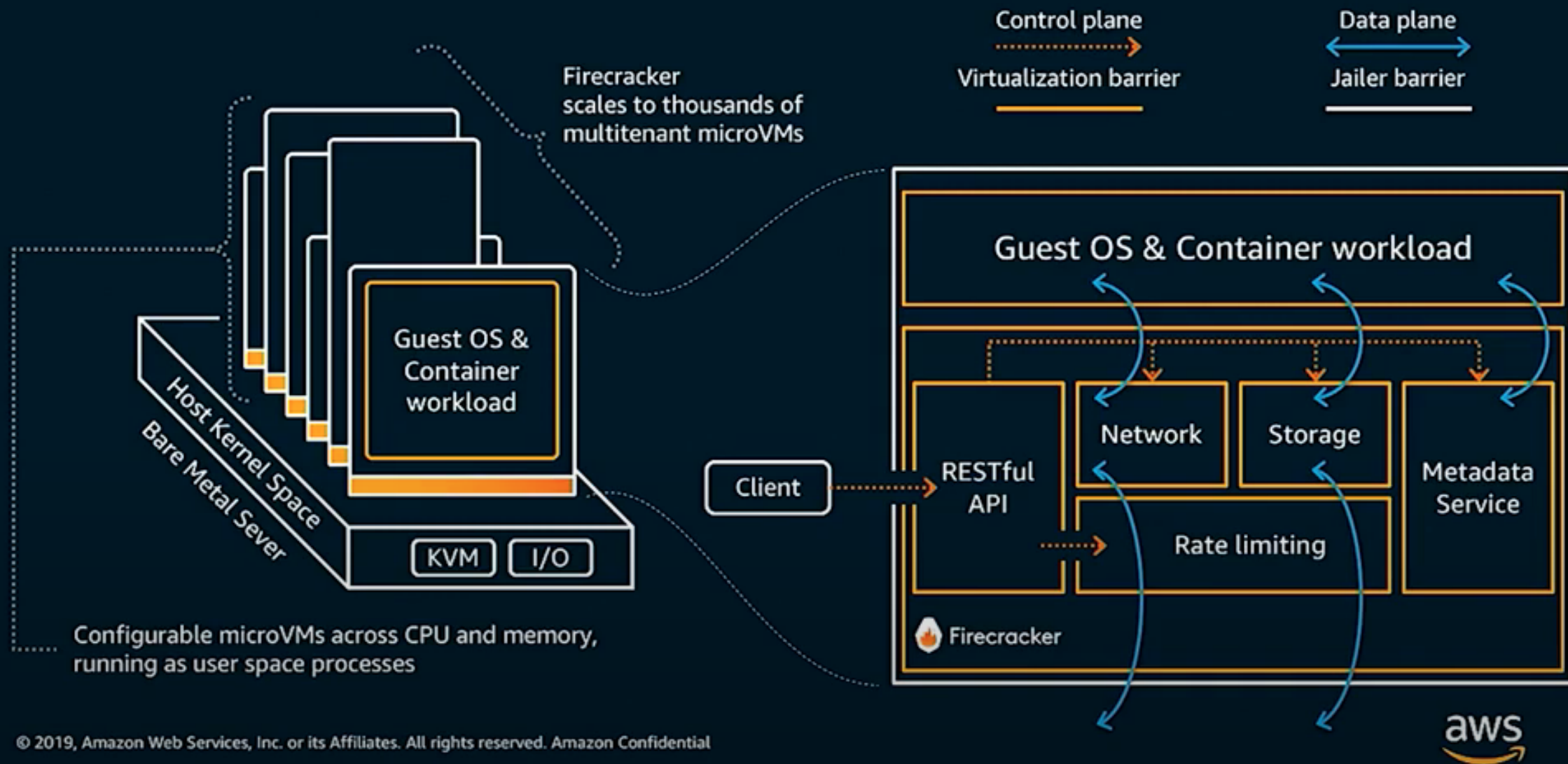


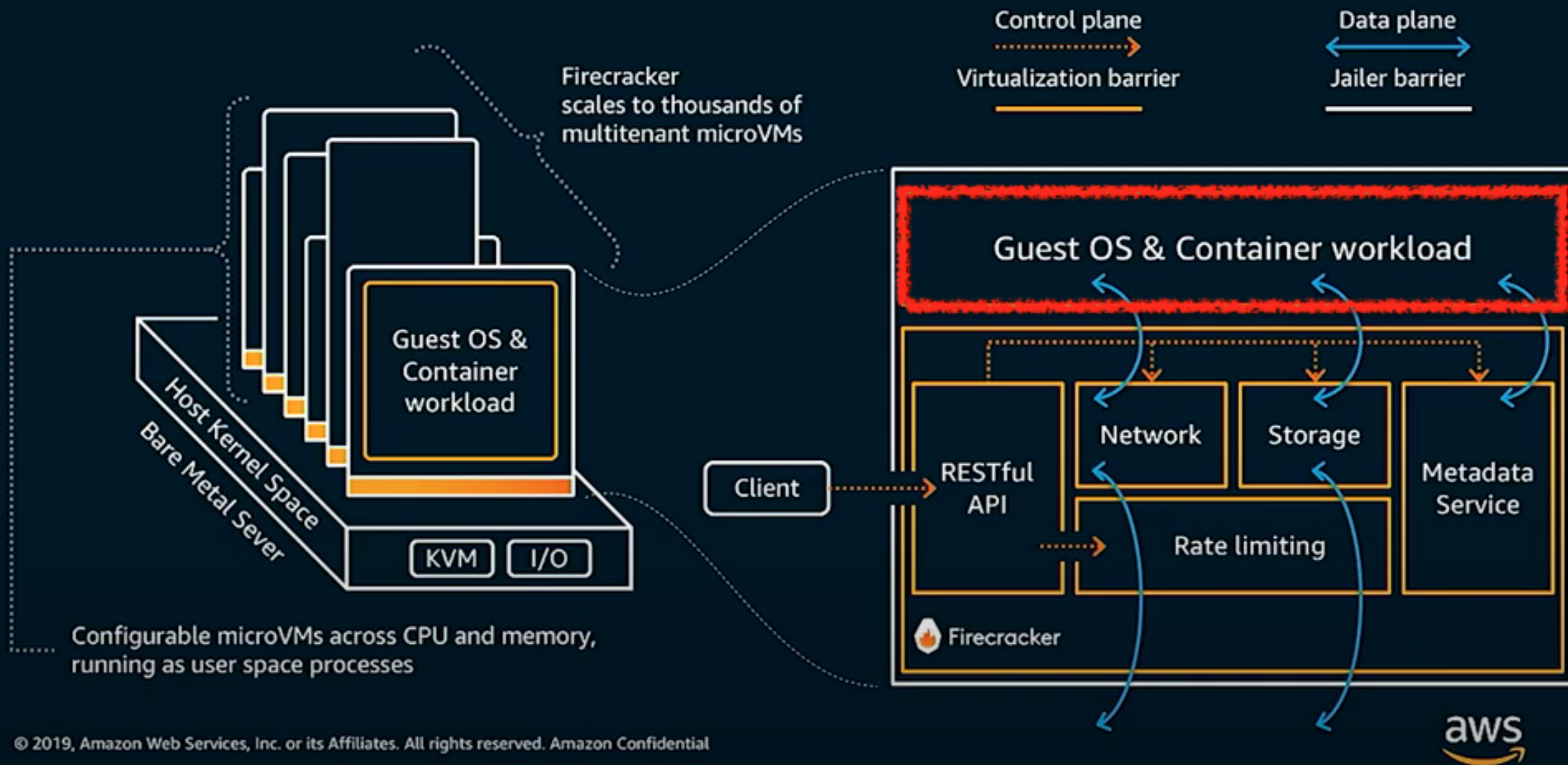
(b) KVM virtualization model

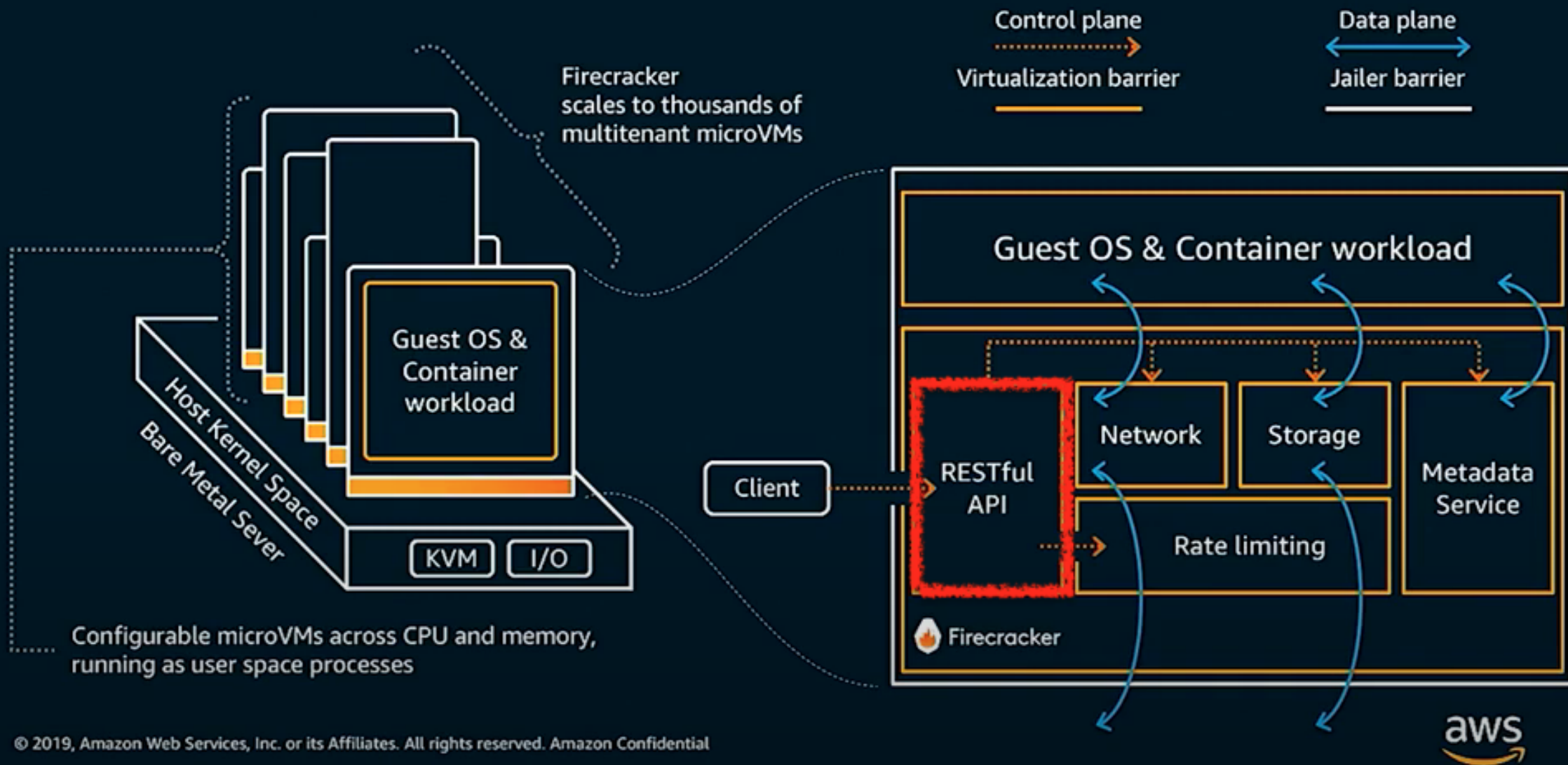
Firecracker Security Models (2/2)











Host-facing REST API

GET / Returns general information about an instance.	GET / machine-config Gets the machine configuration of the VM.
PUT / actions Creates a synchronous action.	PUT / machine-config Updates the Machine Configuration of the VM.
PUT / boot-source Creates or updates the boot source.	PUT / mmds Creates a MMDS (Microvm Metadata Service) data store.
PUT / drives/{drive_id} Creates or updates a drive.	PATCH / mmds Updates the MMDS data store.
PATCH / drives/{drive_id} Updates the properties of a drive.	GET / mmds Get the MMDS data store.
PUT / logger Initializes the logger by specifying two named pipes (i.e. for the logs and metrics output).	PUT / network-interfaces/{iface_id} Creates a network interface.

Demo #1

Getting Started with Firecracker in 2 Minutes

Getting started with Firecracker

- **Firecracker on AWS bare metal**
- Firecracker on other clouds with bare metal (e.g., Packet)
- Firecracker on GCP nested-virt
- Firecracker on Azure nested-virt
- **Firecracker on your dev machine (physical/nested-virt)**

VM

Firecracker

VM: Ubuntu

Hypervisor: VirtualBox

Host OS: macOS

Hardware: Macbook Pro

Demo #1

Demo #1

Getting Started with
Firecracker in 2 Minutes:

Firecracker on VirtualBox on
macOS on Macbook Pro

<https://github.com/dwchiang/firecracker-workshops/tree/master/01-getting-started>

Let's Go!

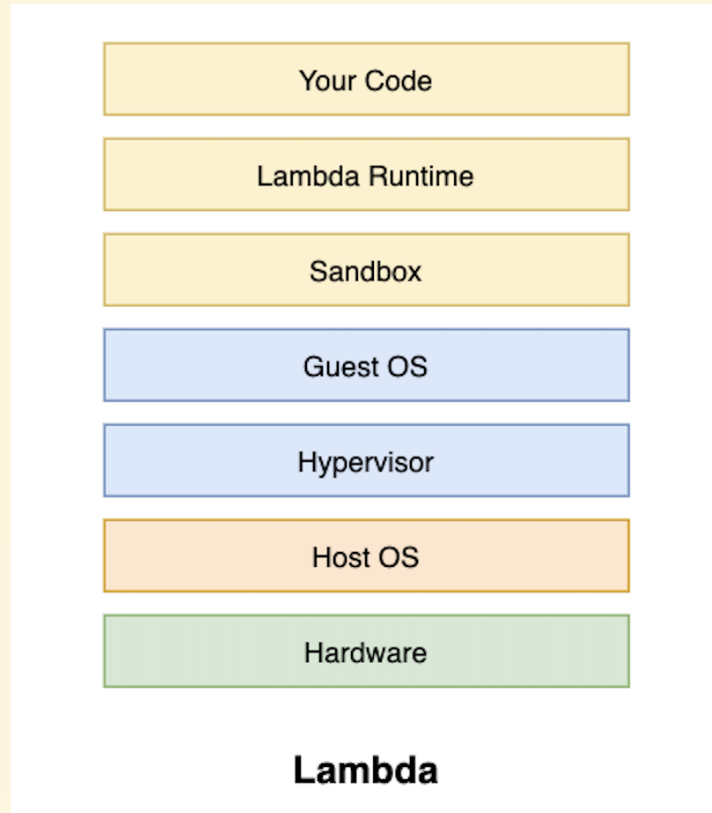
Getting Started with Firecracker in 2 Minutes

Firecracker & AWS Lambda

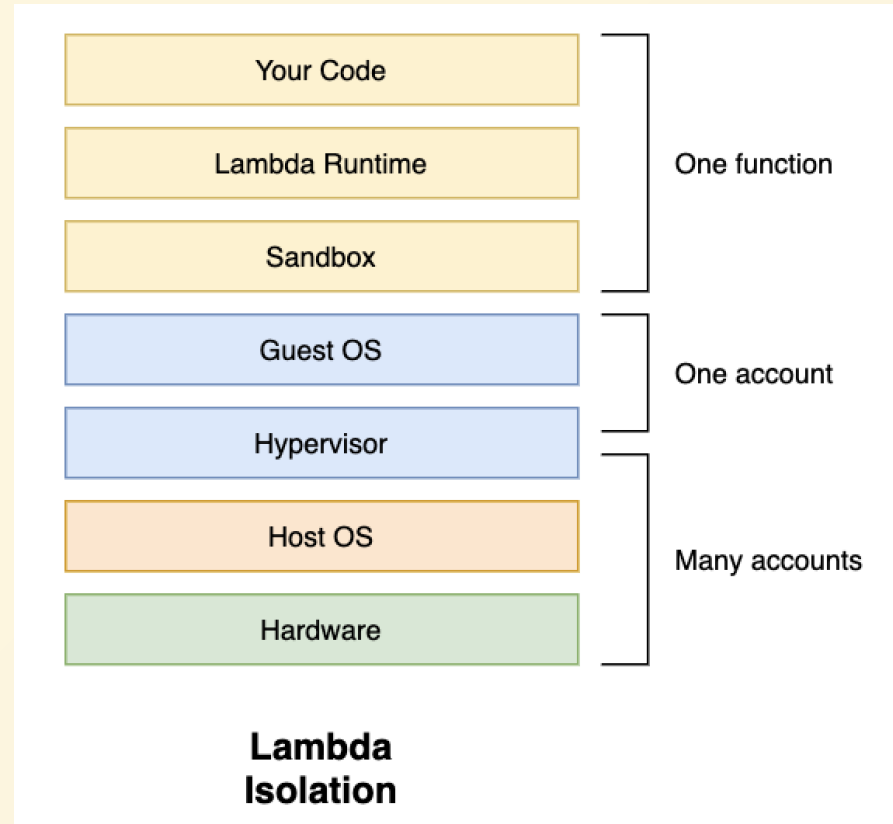
Firecracker & AWS Lambda

Problems & Solutions

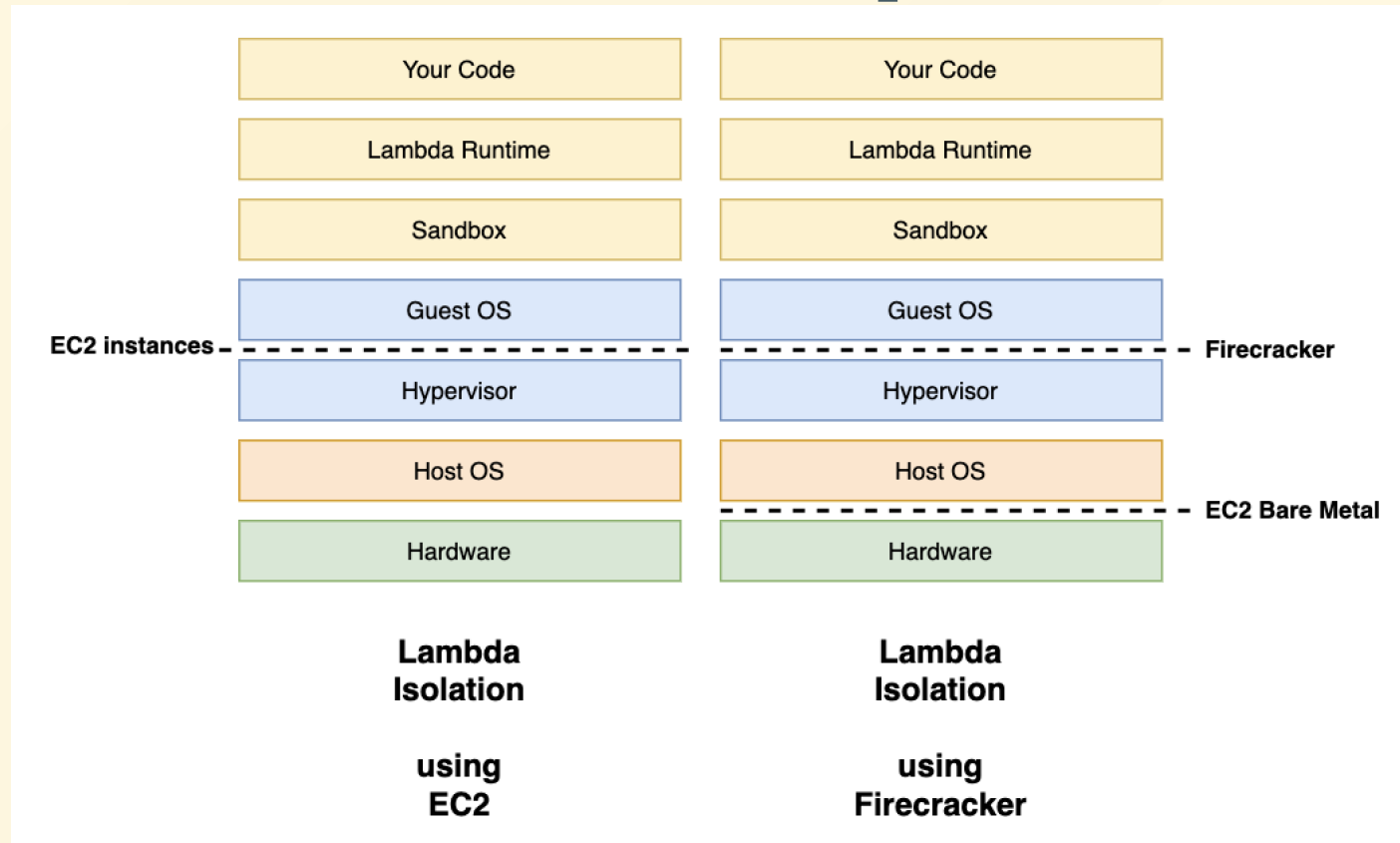
Lambda worker architecture



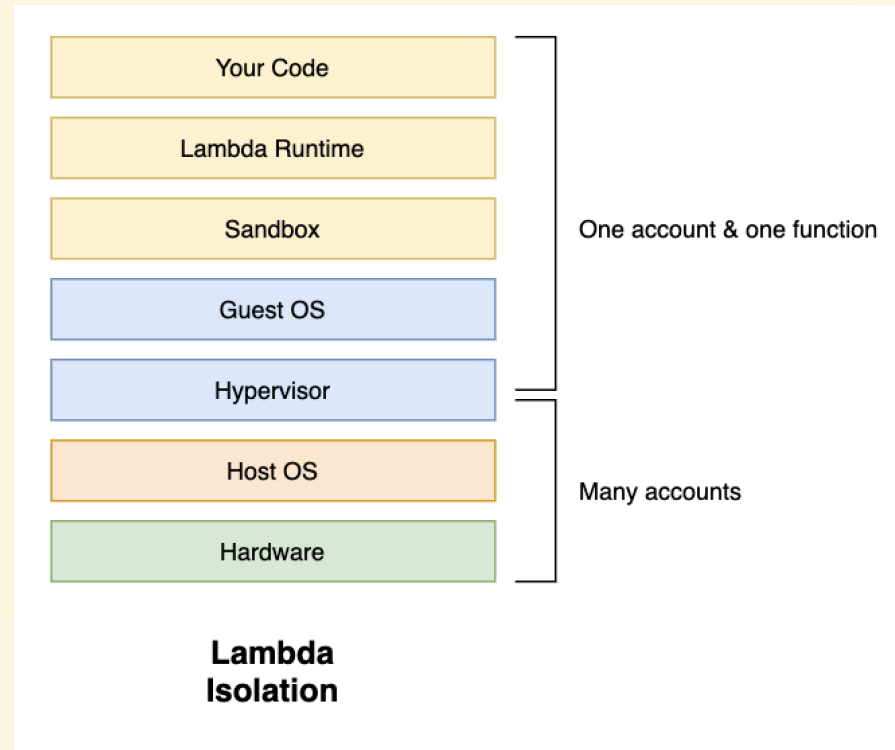
Lambda worker isolation



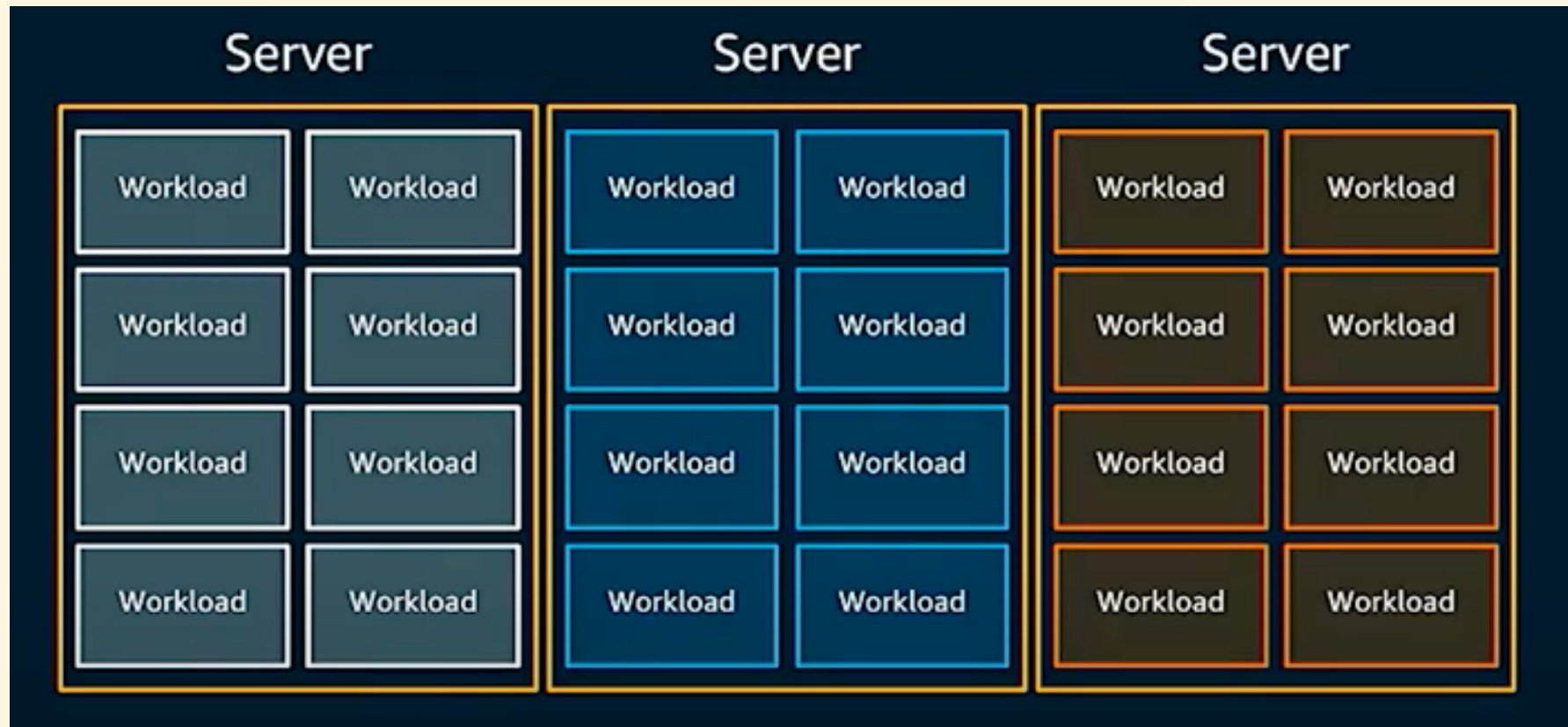
Lambda isolation comparison



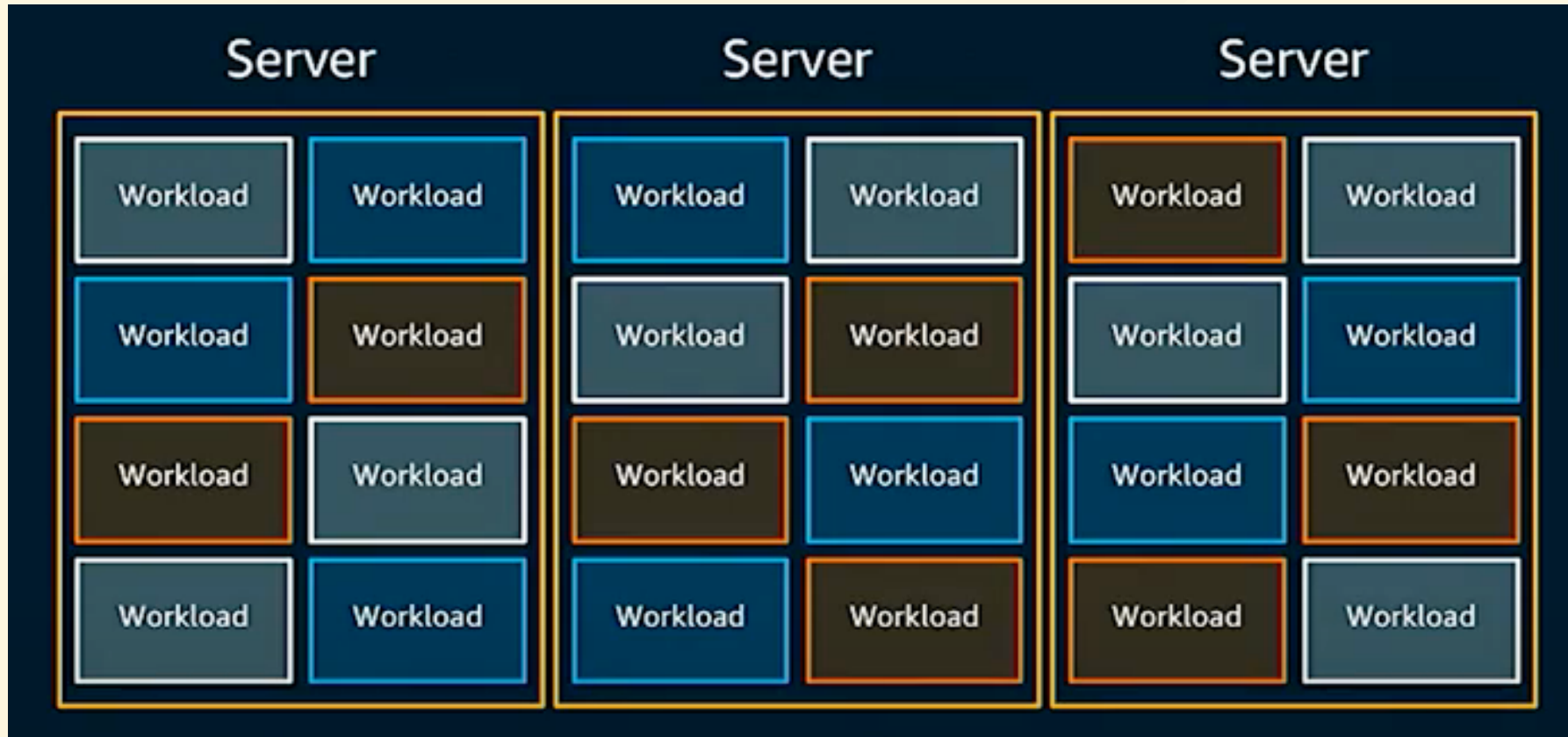
Lambda isolation using Firecracker



Allocate Workloads:



More efficient:



Firecracker



Security



Startup time



Utilization

AWS Container Services landscape

Containers run on `Amazon EC2` and/or `AWS Fargate`

Management

Deployment, scheduling,
scaling & management of
containerized applications



**Amazon Elastic
Container Service**



**Amazon Elastic
Container Service
for Kubernetes**

Hosting

Where the containers run



Amazon EC2



AWS Fargate

Image Registry

Container image repository



**Amazon Elastic
Container Registry**

Firecracker & AWS Fargate

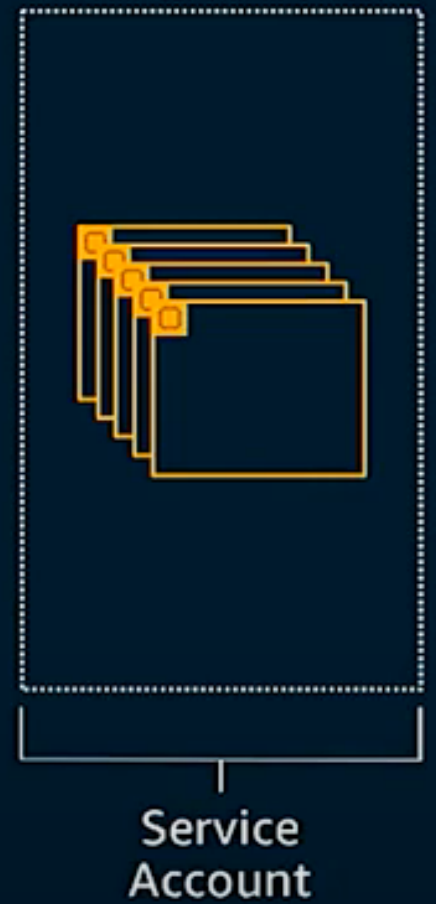
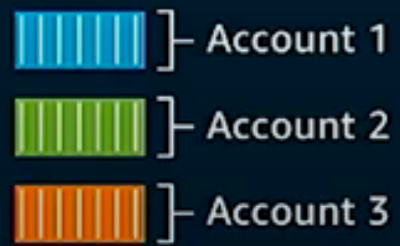
Fargate configurations

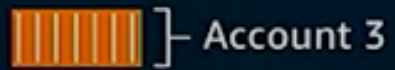
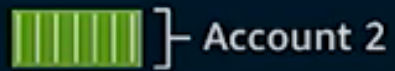
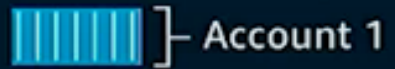
CPU (vCPU)	Memory Values (GB)
0.25	0.5, 1, 2
0.5	Min 1GB, max 4GB, in 1GB increments
1	Min 2GB, max 8GB, in 1GB increments
2	Min 4GB, max 16GB, in 1GB increments
4	Min 8GB, max 30GB, in 1GB increments

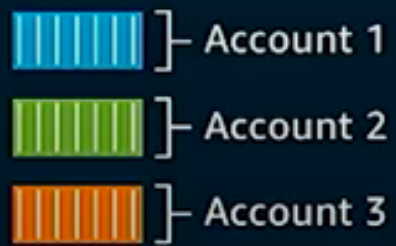
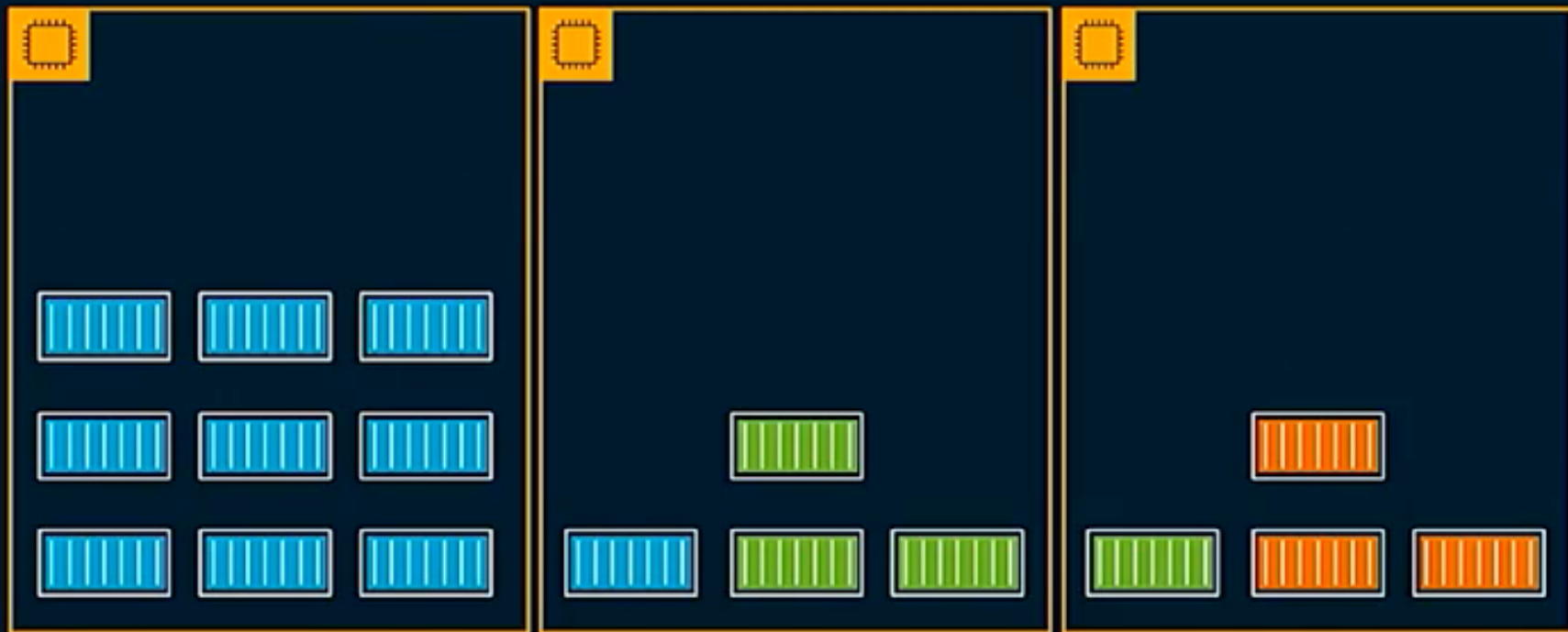


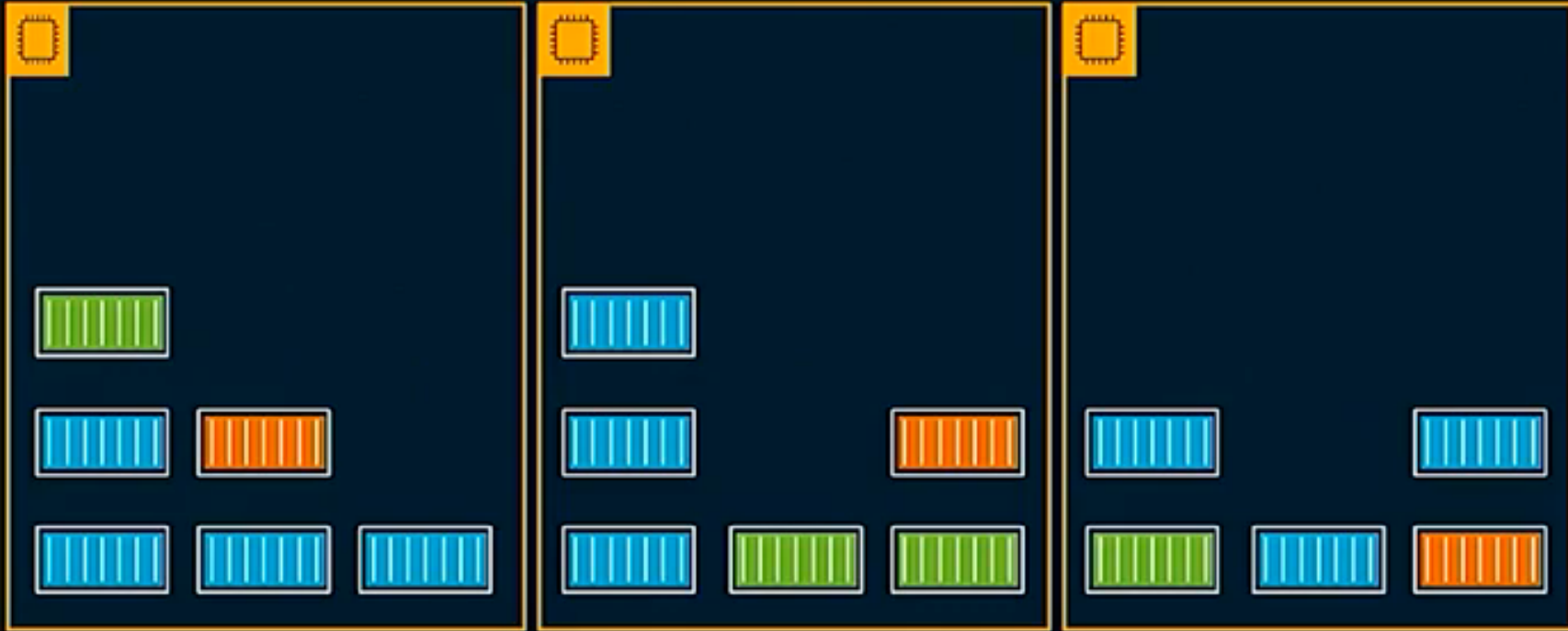
-  } Account 1
-  } Account 2
-  } Account 3





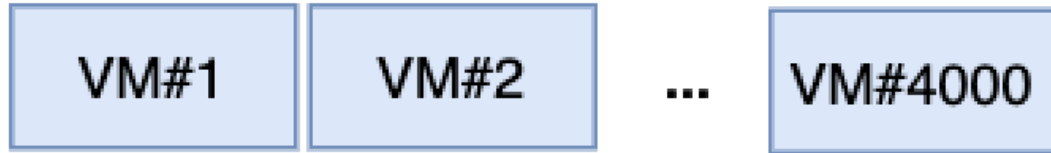






Demo #2

Creating 4,000 microVMs in 90 Seconds



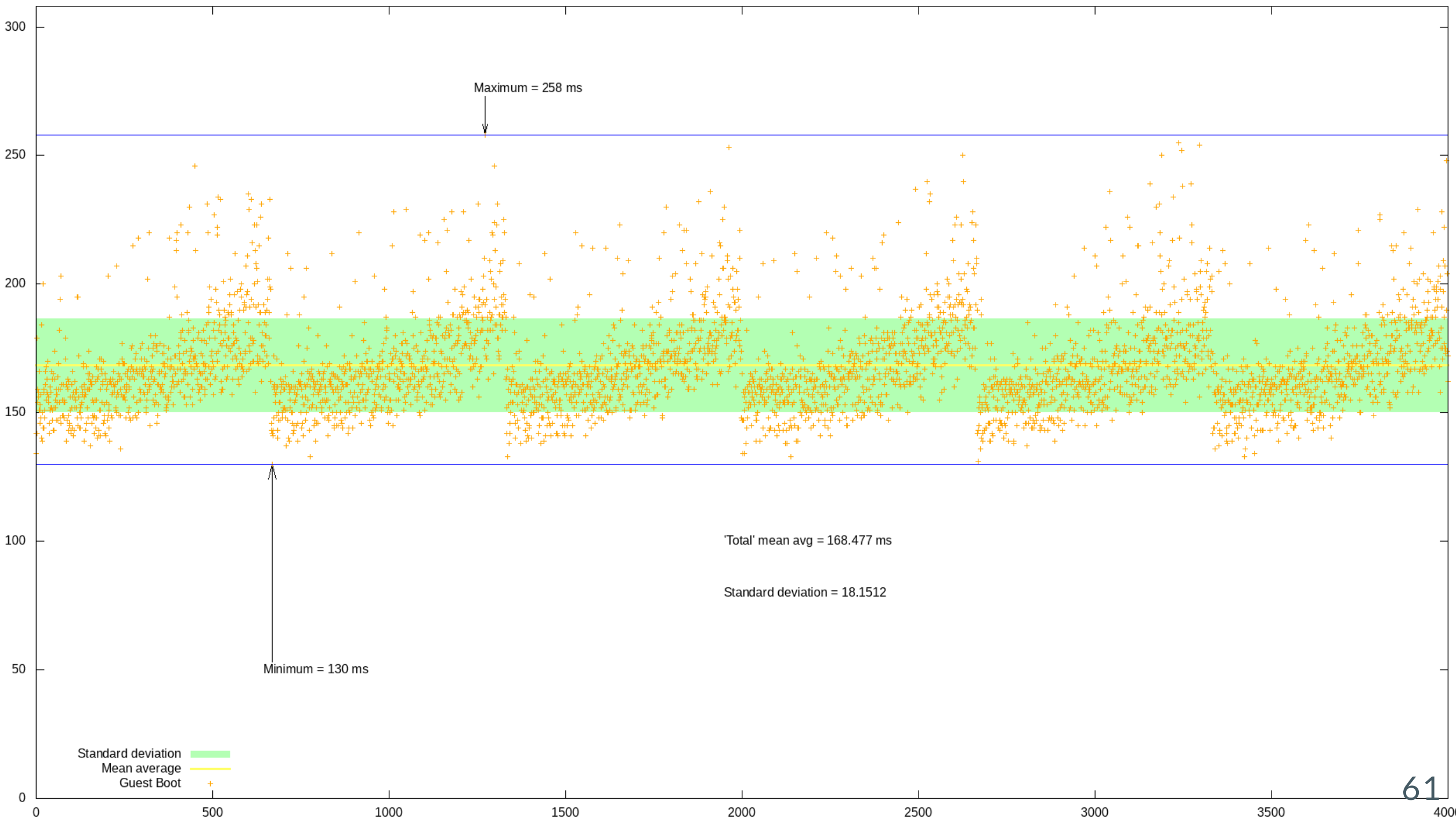
Demo #2

Demo #2

Creating 4,000 microVMs in
90 Seconds:

Firecracker on EC2 Bare
Metal instance

<https://github.com/dwchiang/firecracker-workshops/tree/master/02-4000-microVMs>



Type Name	vCPU	ECU	Memory	Instance Storage	Cost per hour
i3.metal	64	208	512 GiB	8 x 1900 NVMe SSD	\$4.992
m5.metal	96	345	384 GiB	EBS Only	\$4.608
m5d.metal	96	345	384 GiB	4 x 900 NVMe SSD	\$5.424
c5.metal	96	375	192 GiB	EBS Only	\$4.08
c5d.metal	96	375	192 GiB	4 x 900 NVMe SSD	\$4.608

Savings on Spot Instance

Savings

Last hour ▼

A high-level summary of your savings across all of your running and recently terminated Spot Instances.
For detailed reporting on your account-level Spot usage, visit [Cost Explorer](#)

Spot usage and savings

1	72	512	\$4.99	\$1.50	70%
Spot Instances	vCPU-hours	Mem(GiB)-hours	On-Demand total	Spot total	Savings
				\$0.0208	\$0.0029
				Average cost per VCPU-hour	Average cost per mem(GiB)-hour

Details

i3.metal (1)	72 vCPU hours	512 mem(GiB)-hours	\$1.50 total	70% savings
--------------	---------------	--------------------	--------------	-------------

* Spot savings are estimated savings and may differ from actual savings. This is because the savings shown on this page do not include the billing adjustments for your usage.

63

Let's Go!

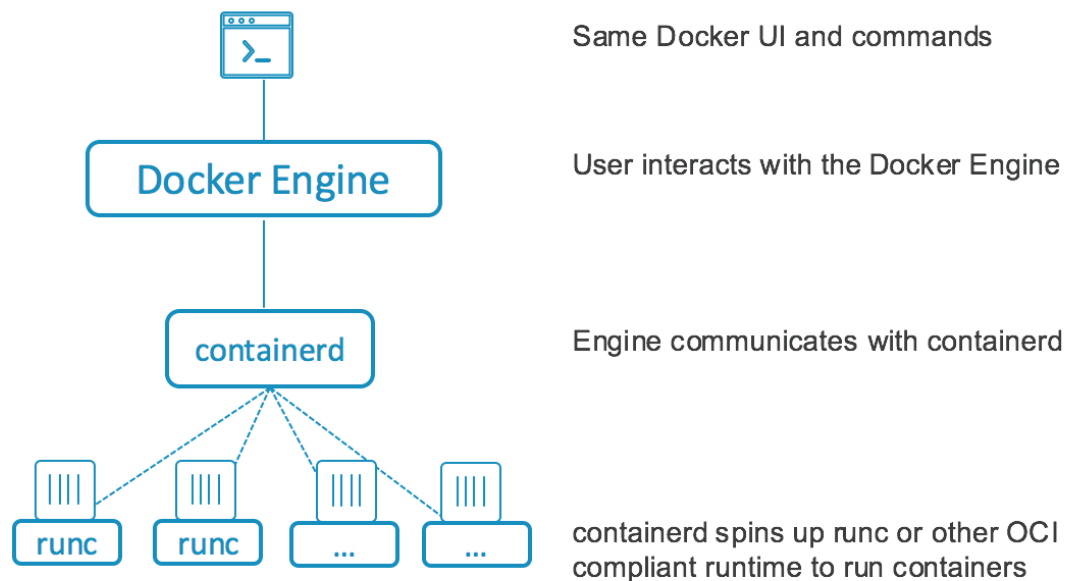
Creating 4,000 microVMs in 90 Seconds

Firecracker & containerd

Firecracker & container **d**

- <https://github.com/firecracker-microvm/firecracker-containerd>
- container **d** to manage **containers** as **Firecracker microVMs**.
- Multi-tenant hosts
- OCI image format
- Work with popular orchestration frameworks
 - Kubernetes and Amazon ECS
- Define a future: **light as container, secure as VM**

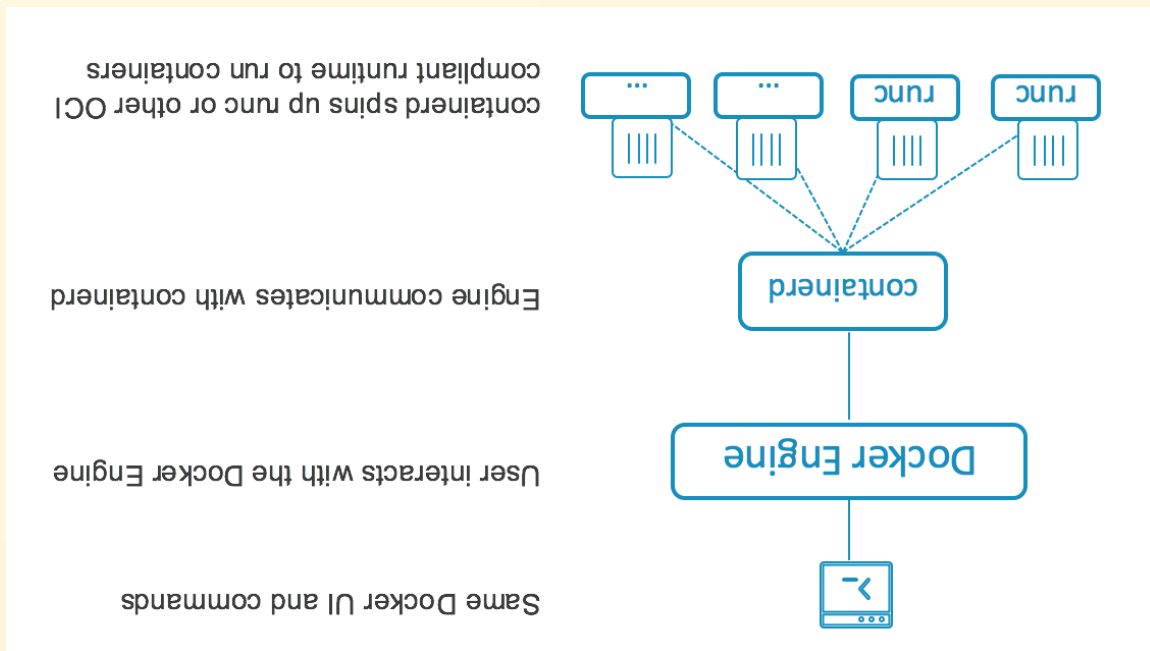
OCI Image & OCI Runtime



- container **d**
- runc
 - is a CLI tool for spawning and running containers according to the OCI specification.

OCI Image & OCI Runtime

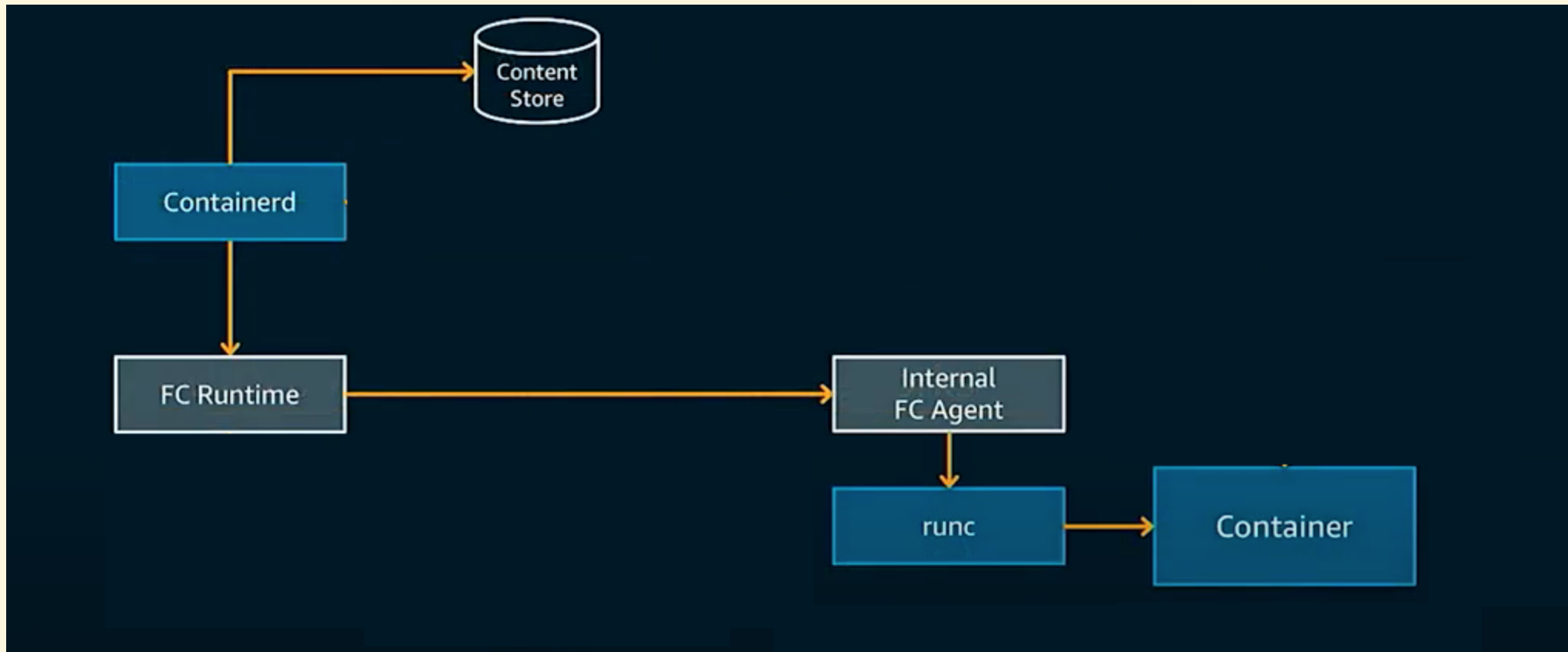
- Let's rotate the chart.
- containerd --> runc --> container



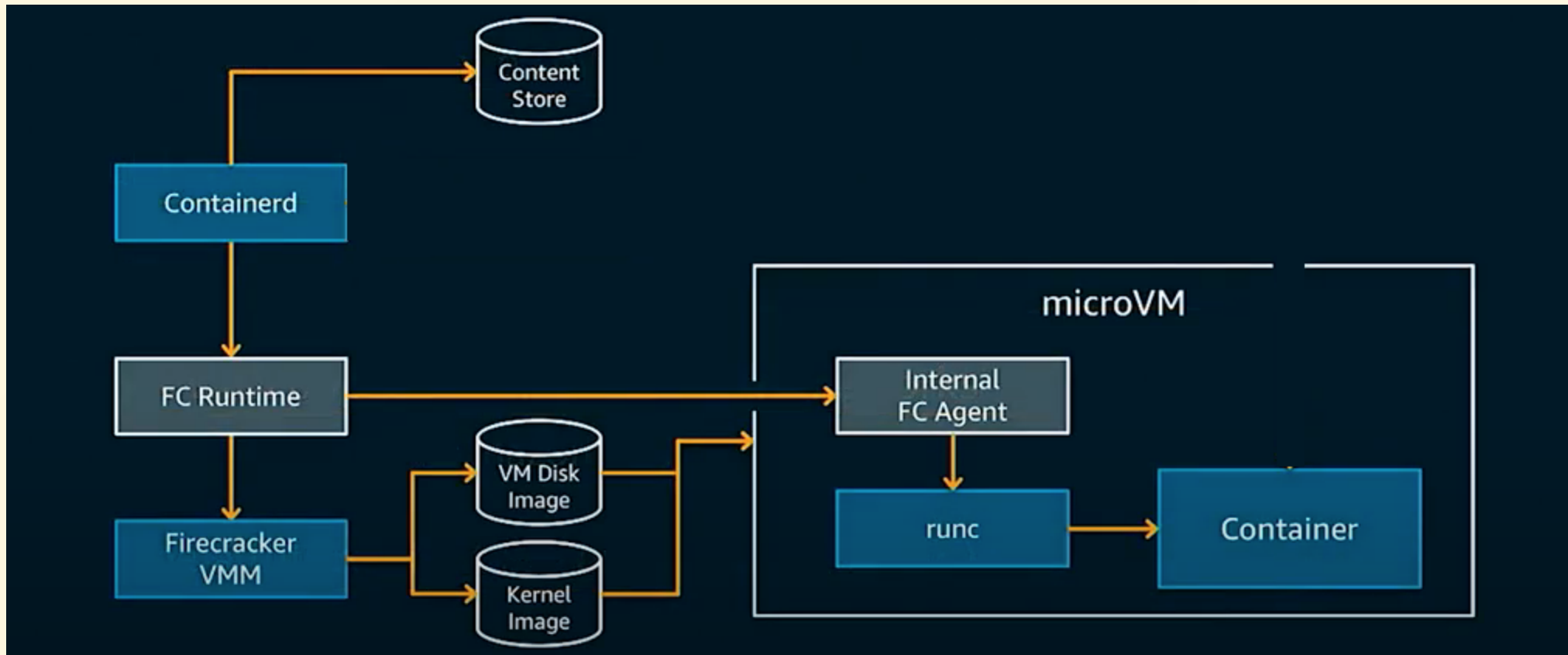
Firecracker & containerd

Part of stack	Example components
Container runtime	runc, Firecracker
Local management	Docker, containerd
Cluster orchestrator	Amazon ECS, Kubernetes, Mesos

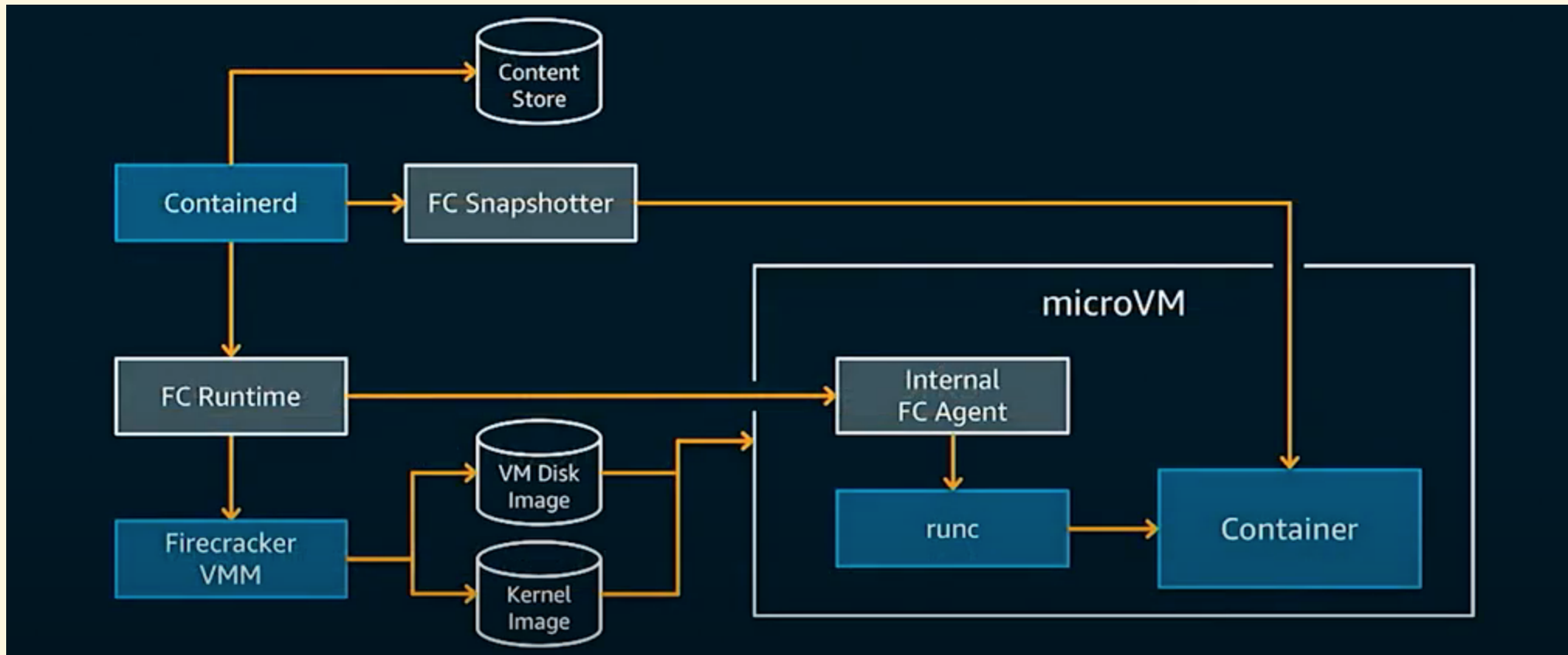
Firecracker & containerd Architecture (1/3)



Firecracker & containerd Architecture (2/3)



Firecracker & containerd Architecture (3/3)



How to run containers with firecracker-containerd

- Deep Dive into firecracker-containerd (re:Invent 2019, CON408)
 - Jump to the [sequence diagram](#) (page 36-52)
- [Demo](#) of Extending containerd - Samuel Karp & Maksym Pavlenko, Amazon

Firecracker & Open Source Projects

Firecracker Integration with Open Source Projects

- Kata Containers
- UniK
- OSv
- **Weave Ignite**

Weave Ignite

- Open source VMM with a container UX
- Combines Firecracker microVMs with OCI images
- Works using **GitOps**
 - `ignite gitops <repo>`
 - <[Fire Up Your VMs with Weave Ignite](#)>

Weave Ignite

- `ignite ps -a`
- `ignite logs <vm>`
- `ignite stop <vm>... [flags]`
- `ignite rm <vm>... [flags]`
- `ignite stop <vm>... [flags]`

Who would use Firecracker?

- Teams building **compute services**
- Teams integrating **Firecracker** with **container** stacks
- Developers & security engineers who want to **contribute**

Takeaways

安全隔離好

啟動時間短

產能效率高

#像極了愛情

-- AWS Firecracker VMM

Q&A & Thank you

Blog <https://www.ernestchiang.com>

Twitter [@dwchiang](https://twitter.com/dwchiang)

#CrossFieldIntegration

#TechnicalManagement

#Bluetooth #AWS

sli.do

- #awsymm2 #awsymm2 #awsymm2 #awsymm2 #awsymm2 #awsymm2
- #awsymm2 #awsymm2 #awsymm2 #awsymm2 #awsymm2 #awsymm2
- #awsymm2 #awsymm2 #awsymm2 #awsymm2 #awsymm2 #awsymm2
- 議程中有任何問題、好奇、疑問，都可以隨時丟進 sli.do
- US\$25 AWS Credits 問券連結，也放在 sli.do 裡頭喔

Community

Community

- Cloud Native Taiwan User Group
 - Facebook : <https://www.facebook.com/groups/cloudnative.tw>
- AWS User Group Taiwan
 - Facebook : <https://www.facebook.com/groups/awsugtw>
- Taiwan CDK Meetup
 - Facebook : <https://www.facebook.com/groups/cdkmeetuptw>

Reference

Reference: Firecracker

- Project Homepage : <https://firecracker-microvm.github.io/>
- Project GitHub : <https://github.com/firecracker-microvm/firecracker>
- Project Roadmap : <https://github.com/firecracker-microvm/firecracker/projects/13>

Reference: Firecracker

- Youtube : [Firecracker: A Secure and Fast microVM for Serverless Computing](#), 2019-0717, by Meena Gowdar (@meejamb) & Arun Gupta (@arungupta)
- Youtube : [NSDI '20 - Firecracker: Lightweight Virtualization for Serverless Applications](#), 2020-02, by Marc Brooker at [NSDI 20](#)
 - Paper (PDF) : [Firecracker: Lightweight Virtualization for Serverless Applications](#)

Reference: Firecracker

- **Blog**: [深度解析 AWS Firecracker 原理篇 - 虚拟化与容器运行时技术](#) by 莫梓元.
- **Blog**: [深度解析 AWS Firecracker 实战篇 - 一起动手点炮竹](#) by 莫梓元.
- **Workshop**: [IGNITE YOUR FIRECRACKER WORKSHOP - AWS TKO 2020](#)
- **Workshop**: [Firecracker Workshop Collections](#)
- **Slide**: [Deep Dive into Firecracker Using Lightweight Virtual Machines to Enhance the Container Security Boundary - AWS Summit Sydney, 2019](#)

Reference: Firecracker

- **Demo** : [A demo running 4000 Firecracker microVMs](#)
- **Docs** : [Firecracker Design](#) (firecracker-microvm/firecracker)
- **Docs** : [Getting started](#) (firecracker-microvm/firecracker)
- **Youtube** : [Running AWS Firecracker in your local machine](#), by Abhijith PK, 2018.

Reference: ecosystems

- `firecracker-containerd`: [Deep Dive into firecracker-containerd](#)
(re:Invent 2019, CON408)
 - [demo script](#)

Reference: ecosystems

- **Weave Ignite** is an open source Virtual Machine (VM) manager with a container UX and built-in GitOps management.
 - <https://github.com/weaveworks/ignite>
 - Docs: <https://ignite.readthedocs.io/en/stable/>
- **OSv** is an open-source versatile modular unikernel designed to run single unmodified Linux application securely as microVM on top of a hypervisor, when compared to traditional operating systems which were designed for a vast range of physical machines.
 - <https://github.com/cloudius-systems/osv>

Reference: ecosystems

- **Kata Containers** is an open source project and community working to build a standard implementation of lightweight Virtual Machines (VMs) that feel and perform like containers, but provide the workload isolation and security advantages of VMs.
 - <https://github.com/kata-containers/kata-containers>

Reference: ecosystems

- [crosvm](#)
- [rust-vmm](#)
- ...
- [Cloud Hypervisor](#)

Reference: Virtualization

- Youtube : [Linux 核心設計 發展動態回顧 \(2020-05-23\)](#) by jserv
- Slide : [Embedded Virtualization applied in Mobile Devices](#) by jserv, 2012.

Reference: AWS Nitro System

- [AWS Nitro System](#) by James Hamilton (VP and Distinguished Engineer at AWS), 2019.
 - The Nitro hypervisor is built on a minimized and modified Linux kernel, including the KVM subsystem that is responsible for programming hardware virtualization features of the processor.

Open Source at AWS

- <https://aws.amazon.com/opensource/>

Firecracker design principles

- Multitenant
- Any vCPU and memory combination
- Oversubscription permissible
- Steady mutation rate: 100+ microVMs/host/sec
- Limited only by hardware resources
- Host-facing REST API
- Minimalist guest device model