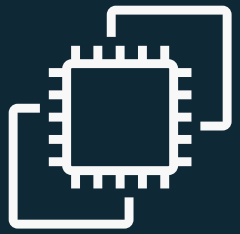
Compute

Choices for Compute



Amazon EC2

Virtual server instances in the cloud

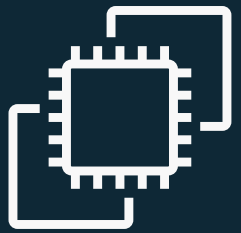
Amazon ECS, EKS, and Fargate

Container management service for running Docker on a managed cluster of EC2

AWS Lambda

Serverless compute for stateless code execution in response to triggers

Amazon EC2



Amazon EC2

Linux | Windows

Arm and x86 architectures

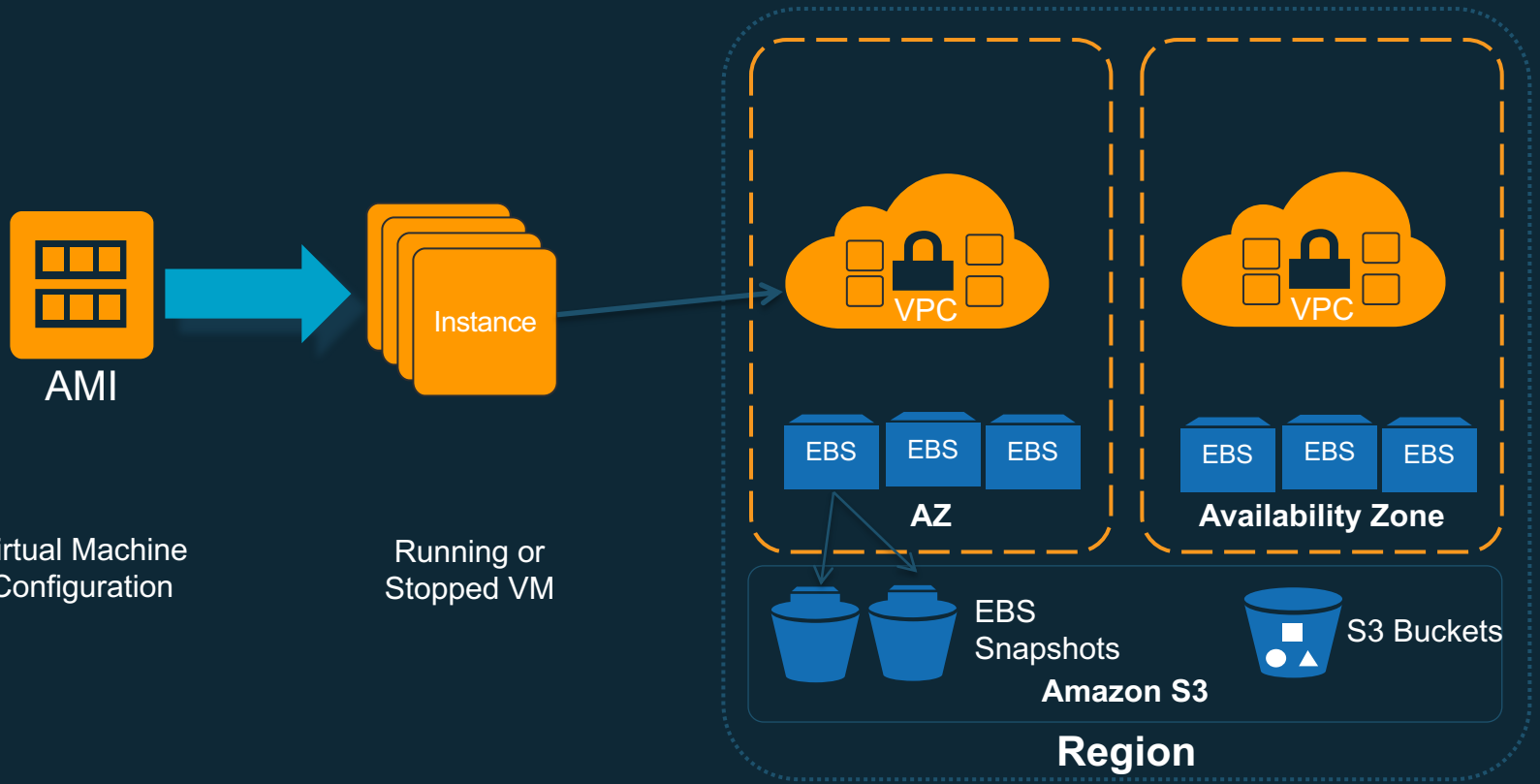
General purpose and workload optimized

Bare metal, disk, networking capabilities

Packaged | Custom | Community AMIs

Multiple purchase options: On-demand, RI, Spot

EC2 Terminology

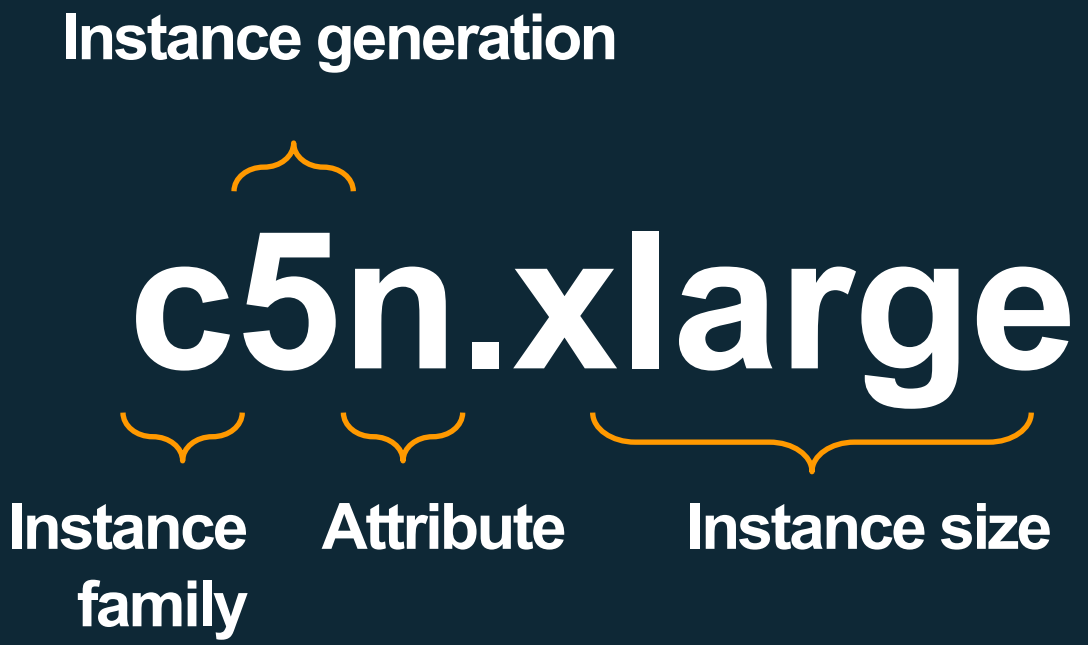


What's a virtual CPU? (vCPU)

- A vCPU is typically a hyper-threaded physical core*
- Divide vCPU count by 2 to get core count
- On Linux, "A" threads enumerated before "B" threads
- On Windows, threads are interleaved

- Cores by Amazon EC2 & RDS DB Instance type: https://aws.amazon.com/ec2/virtualcores/

EC2 Naming Explained



EC2 Operating Systems Supported

Windows 2003R2/2008/2008R2/2012/2012R2/2016
Amazon Linux
Debian
Suse
CentOS
Red Hat Enterprise Linux
Ubuntu

for more OSes see: https://aws.amazon.com/marketplace/b/2649367011

Choose your processor and architecture

Intel® Xeon® Scalable (Skylake) processor

NVIDIA V100 Tensor Core GPUs

AMD EPYC processor

Amazon ARM based Cloud Processor

Right compute for the right application and workload

aws

What is an Amazon Machine Image (AMI)?

Provides the information required to launch an instance

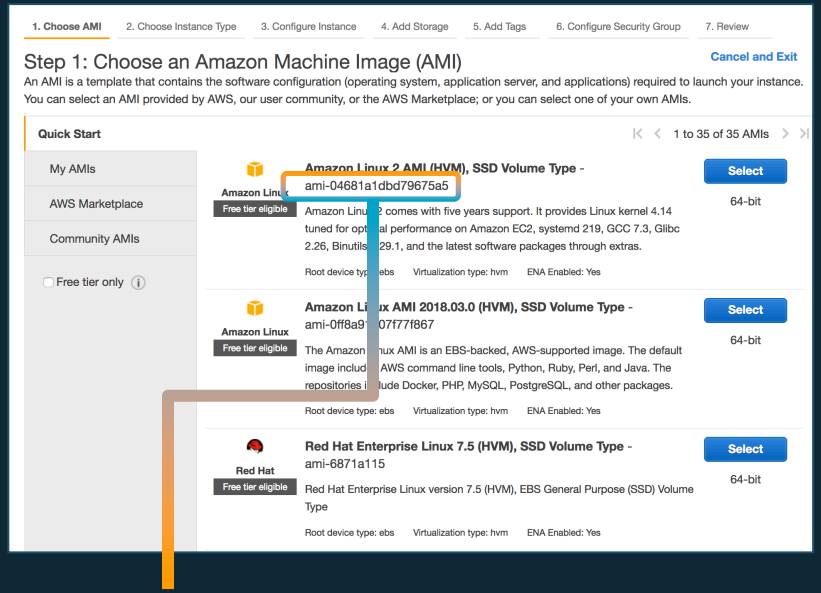
Launch multiple instances from a single AMI

An AMI includes the following

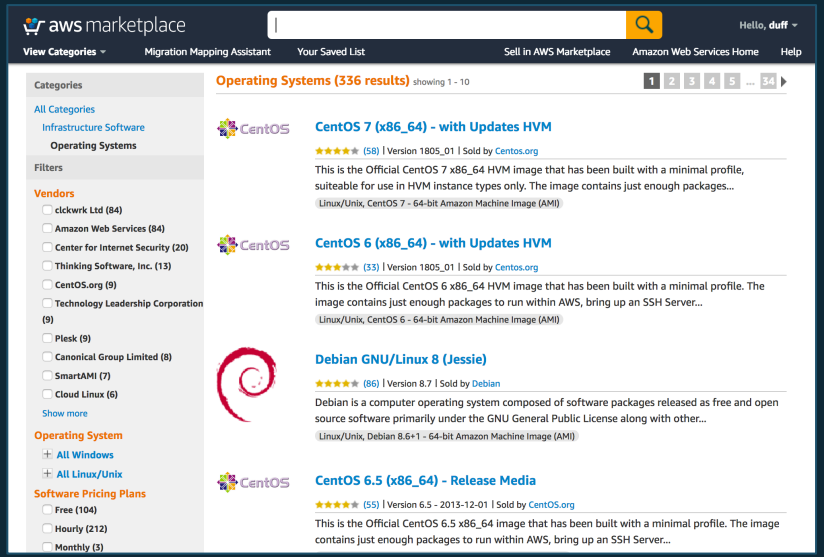
- A template for the root volume (for example, operating system, applications)
- Launch permissions that control which AWS accounts can use the AMI
- Block device mapping that specifies volumes to attach to the instance

Choosing an AMI

AWS Console



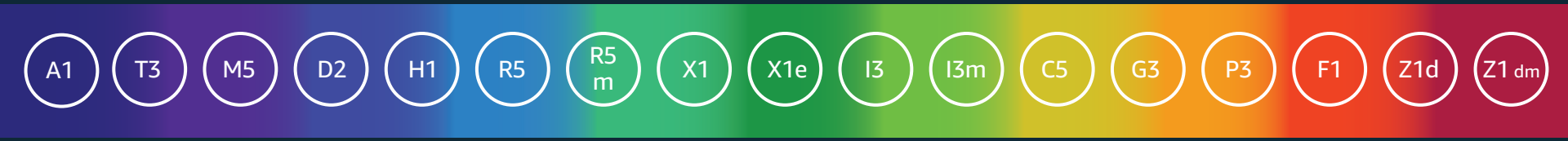
AWS Marketplace



Use the AMI ID to launch through the API or AWS Command Line Interface (AWS CLI)

```
aws ec2 run-instances --image-id ami-04681a1dbd79675a5 --instance-type c4.8xlarge --count 10 --key-name MyKey
```

Instance Types



ARM based	Burstable	General Purpose	Dense Storage	Big Data Optimized	Memory Optimized	In-memory	Memory Intensive	High I/O	Bare Metal High I/O	Compute Intensive	Graphics Intensive	General Purpose GPU	FPGA	Compute and Memory Intensive

EC2 Elastic GPUs

- Graphics acceleration for EC2 instances

EC2 Fleet

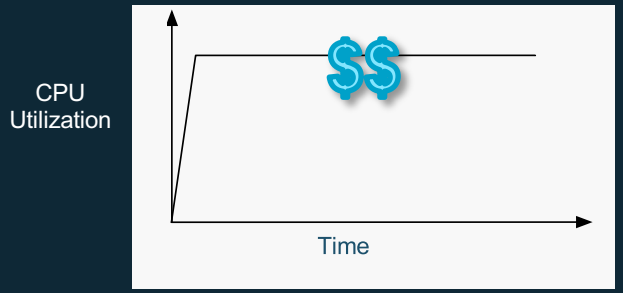
- Simplified provisioning
- Massive scale
- Flexible capacity allocation

Amazon Lightsail

What are Amazon Elastic Compute Cloud (Amazon EC2) burstable instances?

General Purpose Applications…

Continuous Fixed CPU usage

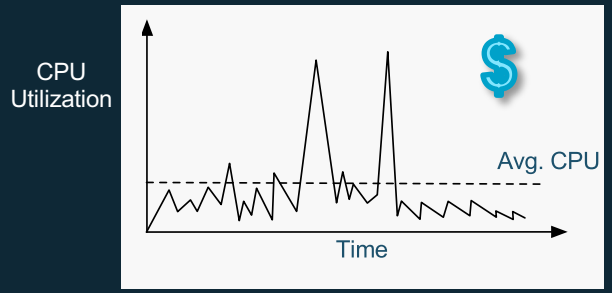


Is it like this?

Why pay more?

or

Variable Peaks CPU usage



Like this?

When you can pay less for the same performance

Opportunity: Most instances aren’t very busy

Burstable instances

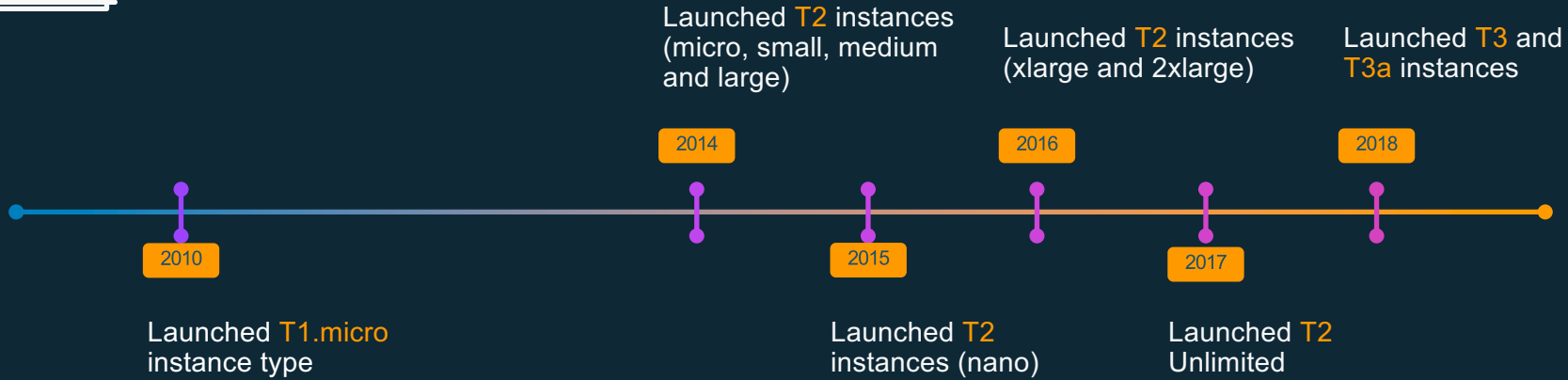
Optimized for most general purpose workloads with occasional high CPU use

Ability to sustain high levels of CPU use indefinitely, if needed

Moderate CPU usage is included in the offering

For sustained high CPU usage, additional CPU usage is “pay as you go”

Burstable instances family



Use burstable instances for…

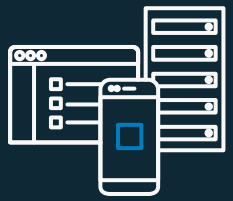
Most general purpose instances that don’t need fixed CPU resources

Applications that occasionally need quick access to high CPU

Idle or need moderate CPU for majority of the time

Need to burst CPU whenever and for as long as needed

… for a variety of general purpose workloads



- Microservices
- Interactive Applications
- Small and Medium Databases
- Code Repositories
- Virtual Desktops
- Dev & Test Environment

CPU credits by instance

Instance size	vCPU count	Optimized for CPU usage	CPU credits per hour	Bucket size (Max CPU credits)
t3.nano	2	≤ 5%	6	144
t3.micro	2	≤ 10%	12	288
t3.small	2	≤ 20%	24	576
t3.medium	2	≤ 20%	24	576
t3.large	2	≤ 30%	36	864
t3.xlarge	4	≤ 40%	96	2,304
t3.2xlarge	8	≤ 40%	192	4,608

Token bucket examples

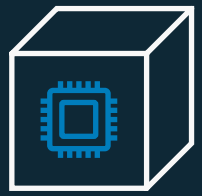
If an instance uses CPU below the baseline rate, the bucket will fill up to the bucket size.

If an instance uses CPU above the baseline rate, the bucket will start emptying.

If the instance CPU use drops back below the baseline rate before the bucket is empty, the bucket starts filling back up.

If the bucket is empty and the instance is still using CPU above the baseline rate, additional CPU usage above the baseline rate is billed at the end of the month

Standard mode



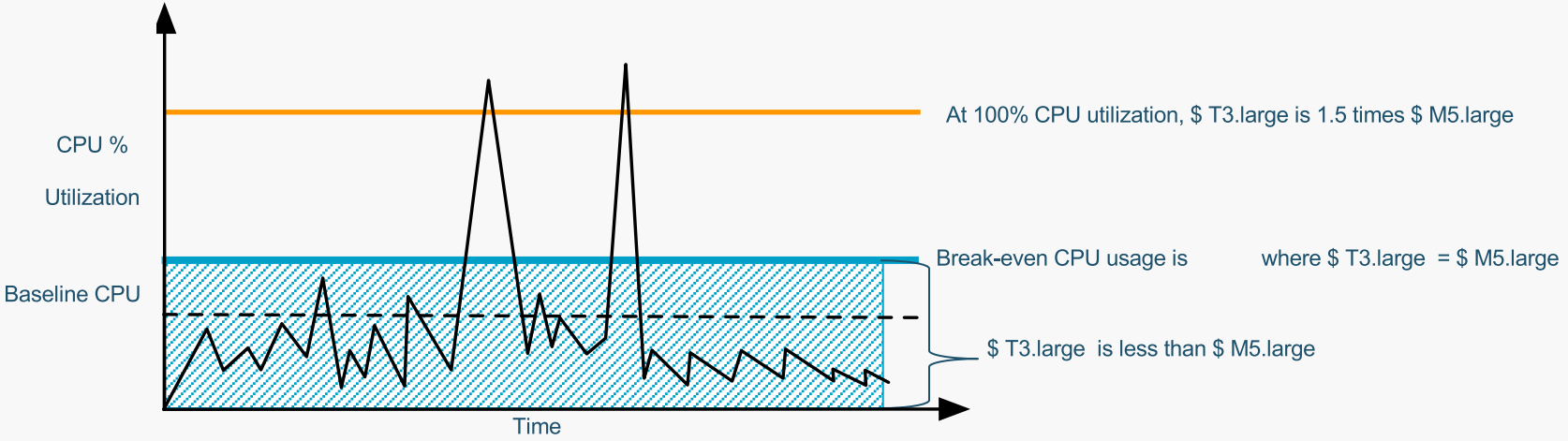
Standard mode is still available for T3 and T2, and is still the default on T2

Guarantees not paying for additional CPU usage

When token bucket is empty, instance cannot exceed baseline level of CPU usage.

Can flip between standard and unlimited mode at any time

For sizes where T3 and M5 overlap, when is T3 cheaper?



For sizes where T3 and M5 overlap, when is T3 cheaper?

instance size	vCPU	T3 price*/hour	M5 price /hour	Diff	T3 Base CPU %	Charge per vCPU hour	Charge per vCPU minute	Addt'l mins avail.	Addt'l CPU % avail.	Break-even CPU %
A	B	C	D	E = D - C	F	G	H = G / 60	I = E / H	J = (I /60) / A	K = F + J
t3.large	2	$0.0835	$0.096	$0.0125	30%	$0.05	$0.000833	15	12.5%	42.5%

t3 instance size	Break-even CPU %
t3.large	42.5%
t3.xlarge	52.5%
t3.2xlarge	52.5%

* prices based on us-east-1 and Linux OS

Purchasing options at a glance

On-Demand Instances

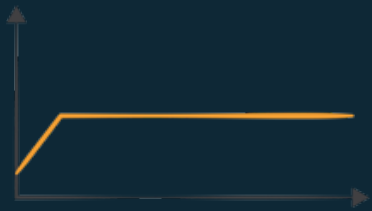
Pay for compute capacity by the hour with no long-term commitments

For Spiky workloads or to define needs

Reserved Instances

Make a low, one-time payment and receive a significant discount on the hourly charge

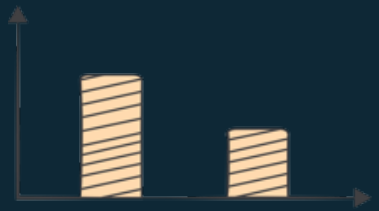
For committed utilization



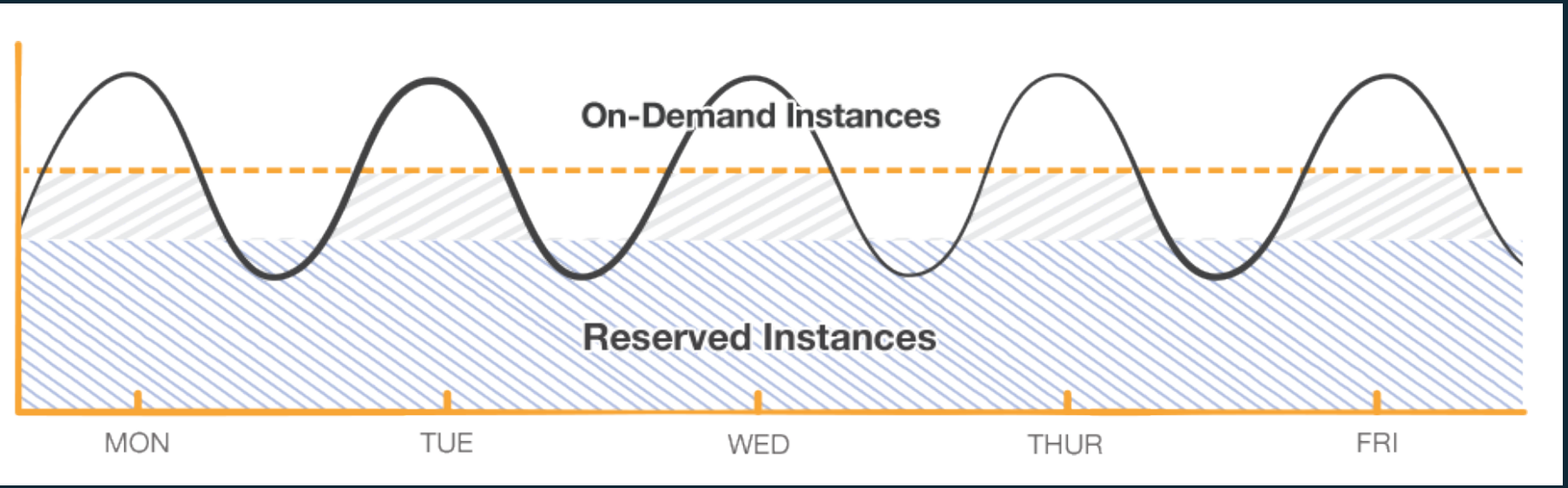
Spot Instances

Bid for unused capacity, charged at a spot price which fluctuates based on supply and demand

For time-insensitive or transient workloads



Layer your options



EC2 Options

Categories

General purpose

Burstable

Compute intensive

Memory intensive

Storage (High I/O)

Dense storage

GPU compute

Graphics intensive

Capabilities

NEW! Choice of processor
(AWS, Intel, AMD)

Fast processors
(up to 4.0 GHz)

High memory footprint
(up to 12 TiB)

Instance storage
(HDD and NVMe)

Accelerated computing
(GPUs and FPGA)

NEW! Networking
(up to 100 Gbps)

Bare Metal

Size
(Nano to 32xlarge)

Options

Elastic Block Store

Elastic Graphics

NEW! Elastic Inference

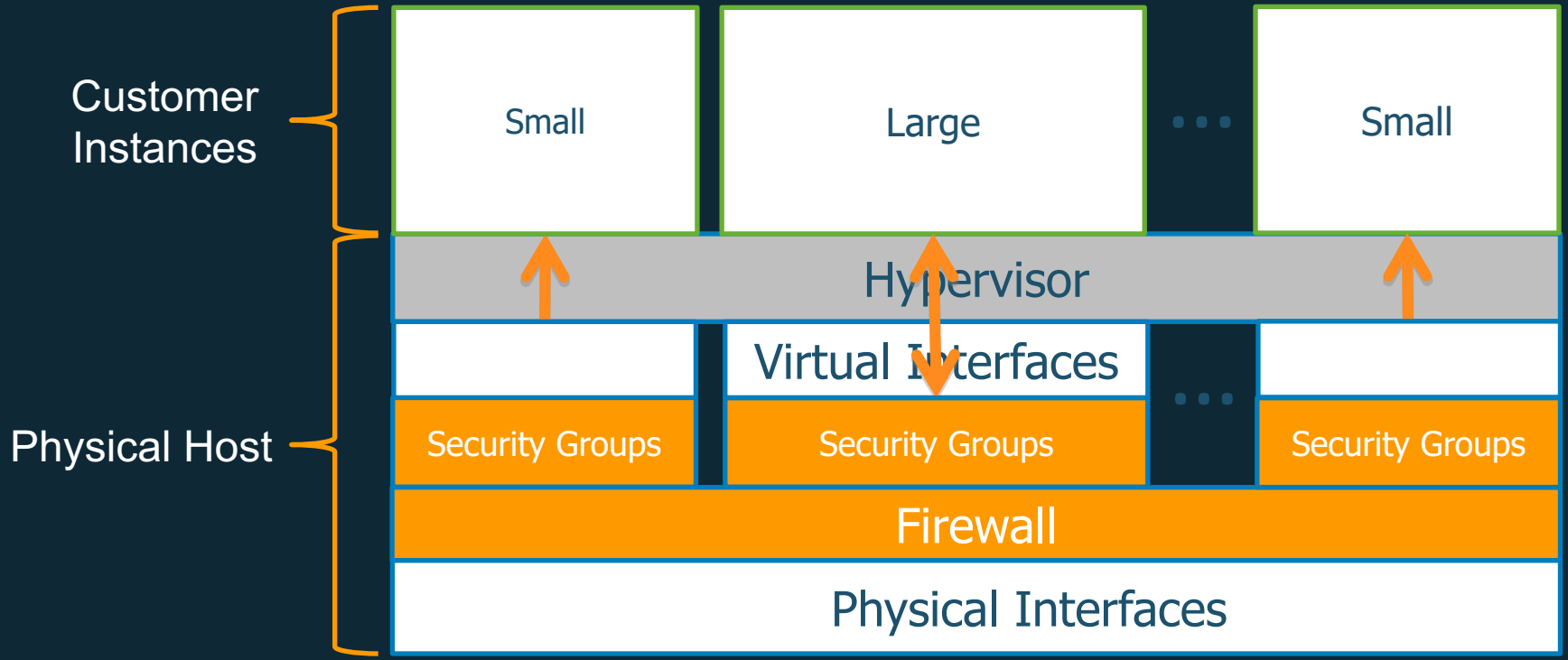
175

instance types

for virtually every workload and business need

EC2 Design

EC2 Host Virtualization



Which hypervisor do we use?

Original: **Xen**

- Original hypervisor
- Consumed excessive resources
- Limited optimization

New (Nov/2017): **Custom KVM based hypervisor**

- Nitro instances
- Less server resources used, more resources for the customer
- AWS optimized

Bare metal: **No AWS provided hypervisor**

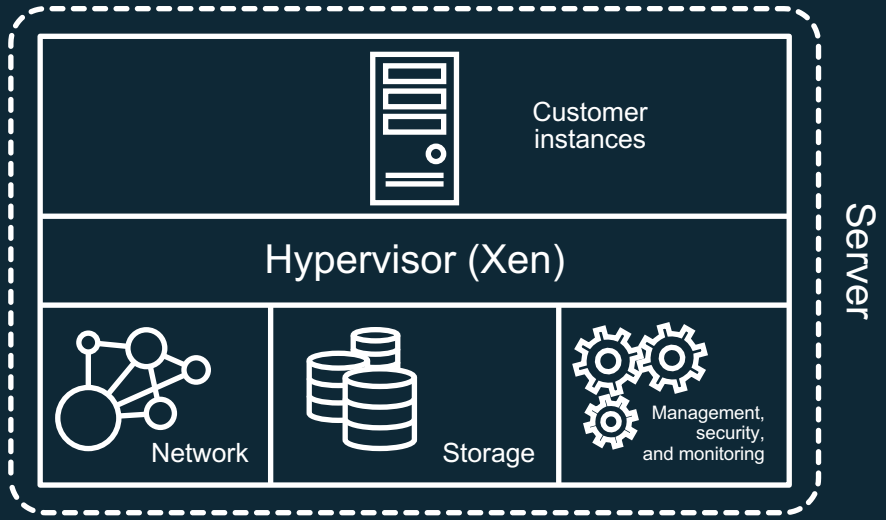
Hypervisor update

Original EC2 host architecture

All resources were on the server

Instance goals

- Security
- Performance
- Familiarity

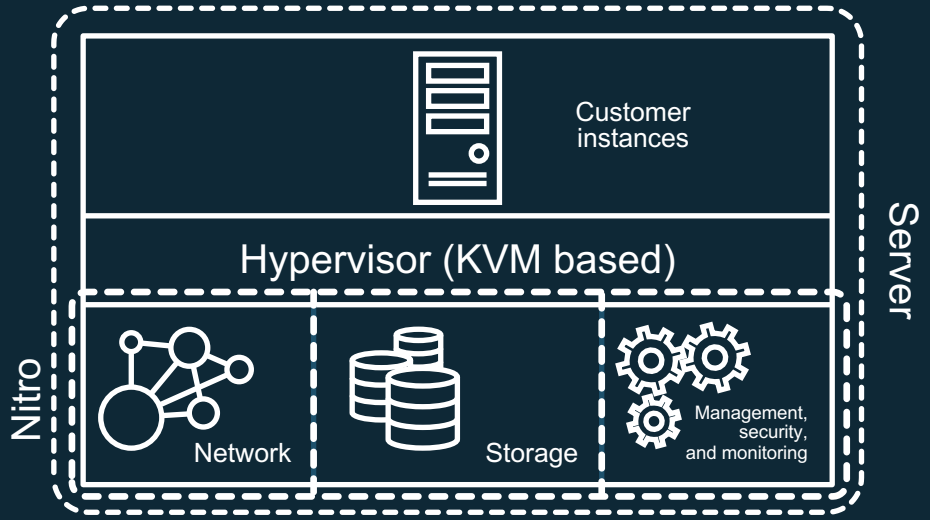


EC2 instance built on AWS Nitro System

Nearly 100% of available compute resources available to customers' workload

Improved security

powered by annapurnalabs an amazon company

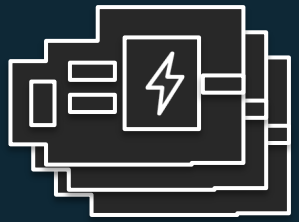


AWS Nitro System

Nitro Hypervisor

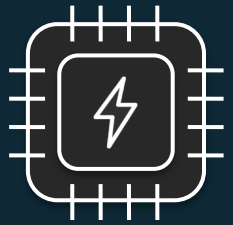
- Lightweight hypervisor
- Memory and CPU allocation
- Bare Metal-like performance

Nitro Card



- VPC Networking
- Amazon EBS
- Local Instance
- System Controller

Nitro Security Chip



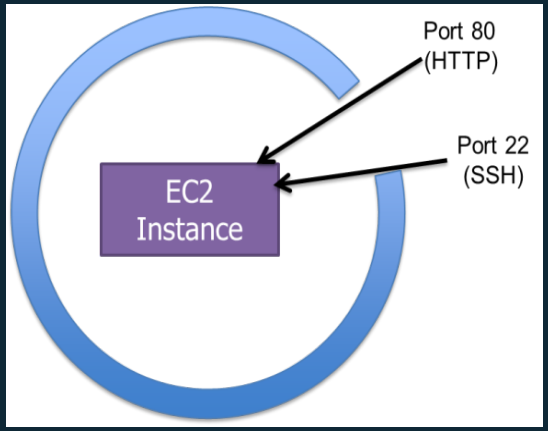
- Integrated into motherboard
- Protects hardware resources
- Hardware Root of Trust

Modular Building Blocks for rapid design and delivery of EC2 instances

EC2 Security Groups

Security Group Rules

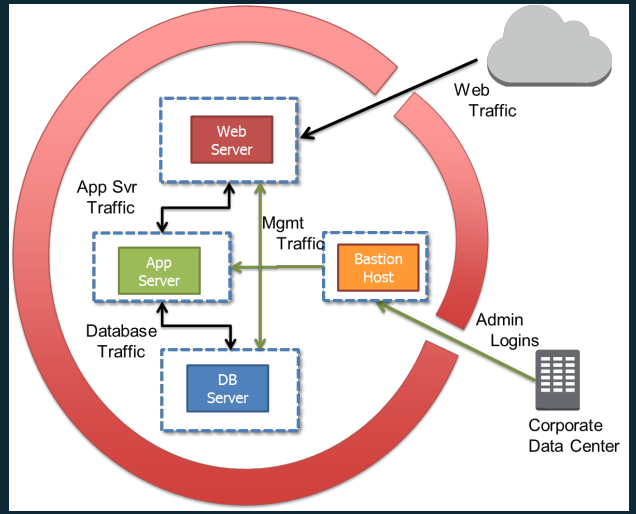
- Name
- Description
- Protocol
- Port range
- IP address, IP range, Security Group name



Tiered EC2 Security Groups

Hierarchical Security Group Rules

- Dynamically created rules
- Based on Security Group membership
- Create tiered network architectures

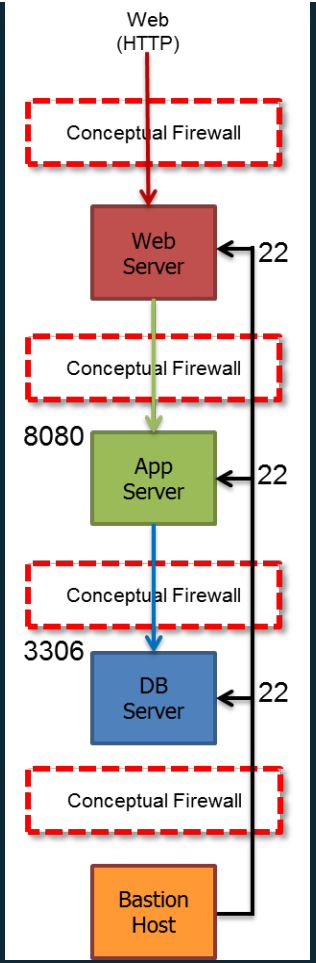


"Web" Security Group:
TCP 80 0.0.0.0/0
TCP 22 "Mgmt"

"App" Security Group:
TCP 8080 "Web"
TCP 22 "Mgmt"

"DB" Security Group:
TCP 3306 "App"
TCP 22 "Mgmt"

"Mgmt" Security Group:
TCP 22 163.128.25.32/32



Instance Metadata

http://169.254.169.254/latest/meta-data/ contains a wealth of info

- ami-id
- ami-launch-index
- ami-manifest-path
- block-device-mapping/
- hostname
- instance-action
- ★ **instance-id**
- instance-type
- kernel-id
- local-hostname
- local-ipv4
- mac
- network/
- ★ **placement/availability-zone**
- profile
- public-hostname
- public-ipv4
- public-keys/

Any Questions?