UNIVERSITY OF ŽILINA
Faculty of Management Science
and Informatics

Presentation 4 - Compute

AWS M6 - Compute

Vytvorené v rámci projektu **KEGA 026TUKE-4/2021**

aws academy

Outline

- Compute services overview
- Amazon EC2
- Container services
- Introduction to AWS Lambda
- Introduction to AWS Elastic Beanstalk

Ak chcete pridať obrázok, kliknite na ikonu

Compute services overview

Some of Amazon AWS compute services

Amazon EC2

AWS Lambda

Amazon Elastic Container Registry (Amazon ECR)

AWS Elastic Beanstalk

Amazon Elastic Container Service (Amazon ECS)

Amazon Elastic Kubernetes Service (Amazon EKS)

Brief description of compute services

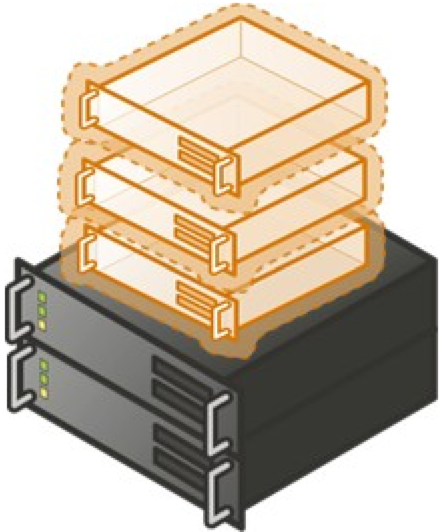
Services	Key Concepts	Characteristics	Ease of Use
• Amazon EC2	• Infrastructure as a service (IaaS) • Instance-based • **Virtual machines**	• Provision virtual machines that you can manage as you choose	A familiar concept to many IT professionals.
• AWS Lambda	• **Serverless** computing • Function-based • Low-cost	• Write and deploy code that runs on a schedule or that can be triggered by events • Use when possible (architect for the cloud)	A relatively new concept for many IT staff members, but easy to use after you learn how.
• Amazon ECS • Amazon EKS • AWS Fargate • Amazon ECR	• **Container-based** computing • Instance-based	• Spin up and run jobs more quickly	AWS Fargate reduces administrative overhead, but you can use options that give you more control.
• AWS Elastic Beanstalk	• Platform as a service (PaaS) • For **web applications**	• Focus on your code (building your application) • Can easily tie into other services—databases, Domain Name System (DNS), etc.	Fast and easy to get started.

Amazon Elastic Compute Cloud (EC2)

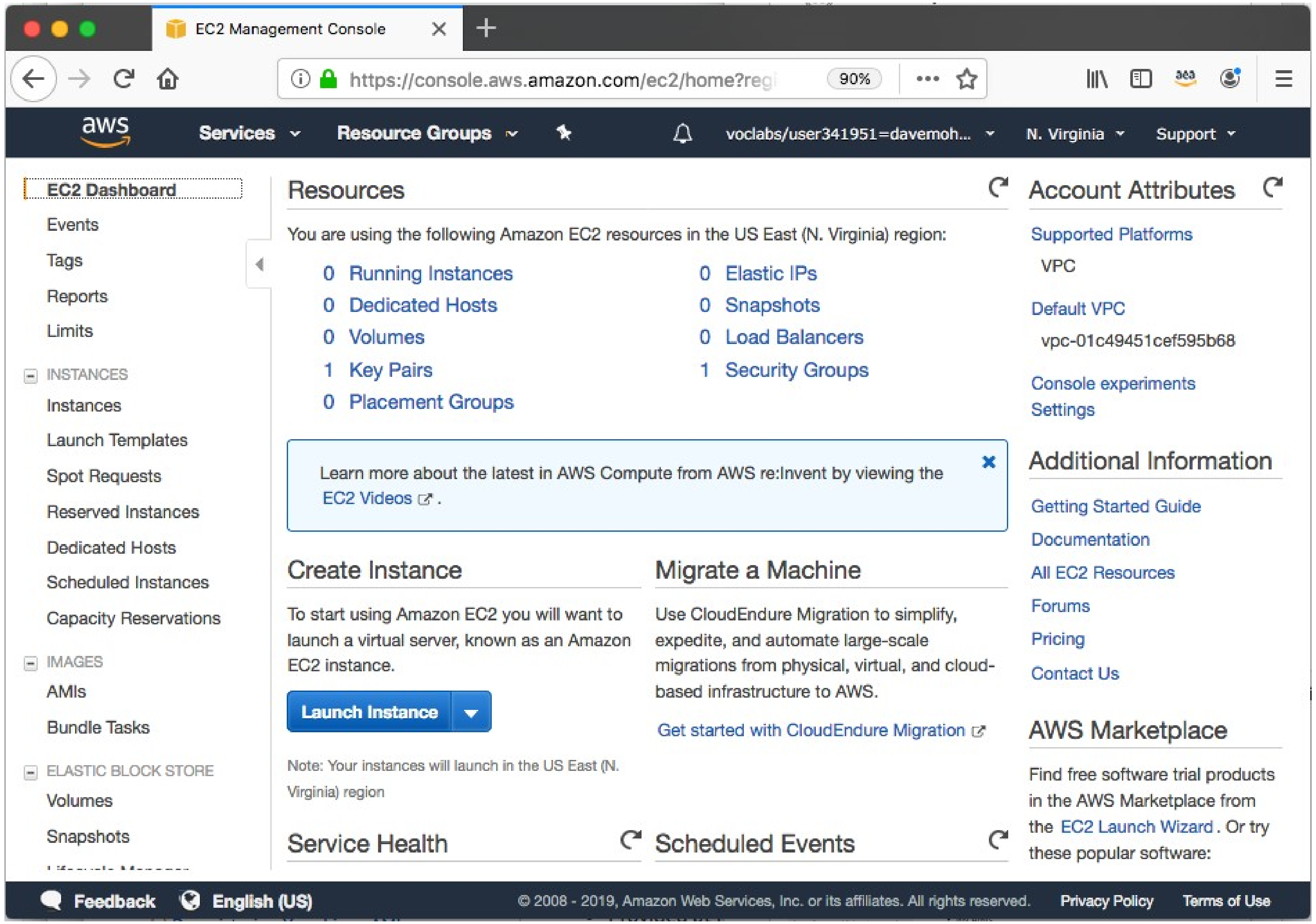
What is EC2 instance?

- EC2 instance = complete virtual server = VM in VirtualBox
  - Including virtual HW
    - vCPU, vRAM, vHDD, vNIC, vGPU, ...
  - Including all software
    - Operating system
    - Libraries
    - Application software
- Same as on-premise server, but has several advantages:
  - You don‘t need electric power
  - You don‘t need cooling
  - You don‘t need housing/space for server
  - You don‘t need server

Amazon EC2



Launching EC2 instance



Resources

You are using the following Amazon EC2 resources in the US East (N. Virginia) region:

0	Running Instances	0	Elastic IPs
0	Dedicated Hosts	0	Snapshots
0	Volumes	0	Load Balancers
1	Key Pairs	1	Security Groups
0	Placement Groups		

Learn more about the latest in AWS Compute from AWS re:Invent by viewing the EC2 Videos.

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Note: Your instances will launch in the US East (N. Virginia) region

Migrate a Machine

Use CloudEndure Migration to simplify, expedite, and automate large-scale migrations from physical, virtual, and cloud-based infrastructure to AWS.

Get started with CloudEndure Migration

Service Health

Scheduled Events

Account Attributes

Supported Platforms
VPC

Default VPC
vpc-01c49451cef595b68

Console experiments
Settings

Additional Information

Getting Started Guide
Documentation
All EC2 Resources
Forums
Pricing
Contact Us

AWS Marketplace

Find free software trial products in the AWS Marketplace from the EC2 Launch Wizard. Or try these popular software:

- When launching EC2 instance, you need to answer to 9 questions

1. Select an AMI

- AMI (Amazon Machine Image) is template from which you will clone instance
- There are 4 types of AMIs
  1. Quick Start – Windowses and Linuxes provided by Amazon
  2. My AMIs – AMIs that you have created
  3. AWS Marketplace – Preconfigurd templates from third parties
  4. Community AMIs – AMIs shared by other users

- You can create AMI from your EC2 instances
  - Save/cature them as AMI in region, where yu want to use them

2. Select an instance type

- The instance type that you choose determines:
  - Memory (RAM)
  - Processing power (CPU)
  - Disk space and disk type (Storage)
  - Network performance

- Instance type categories:
  - General purpose
  - Compute optimized
  - Memory optimized
  - Storage optimized
  - Accelerated computing

- Instance types offer family, generation, and size

EC2 instance type naming and sizes

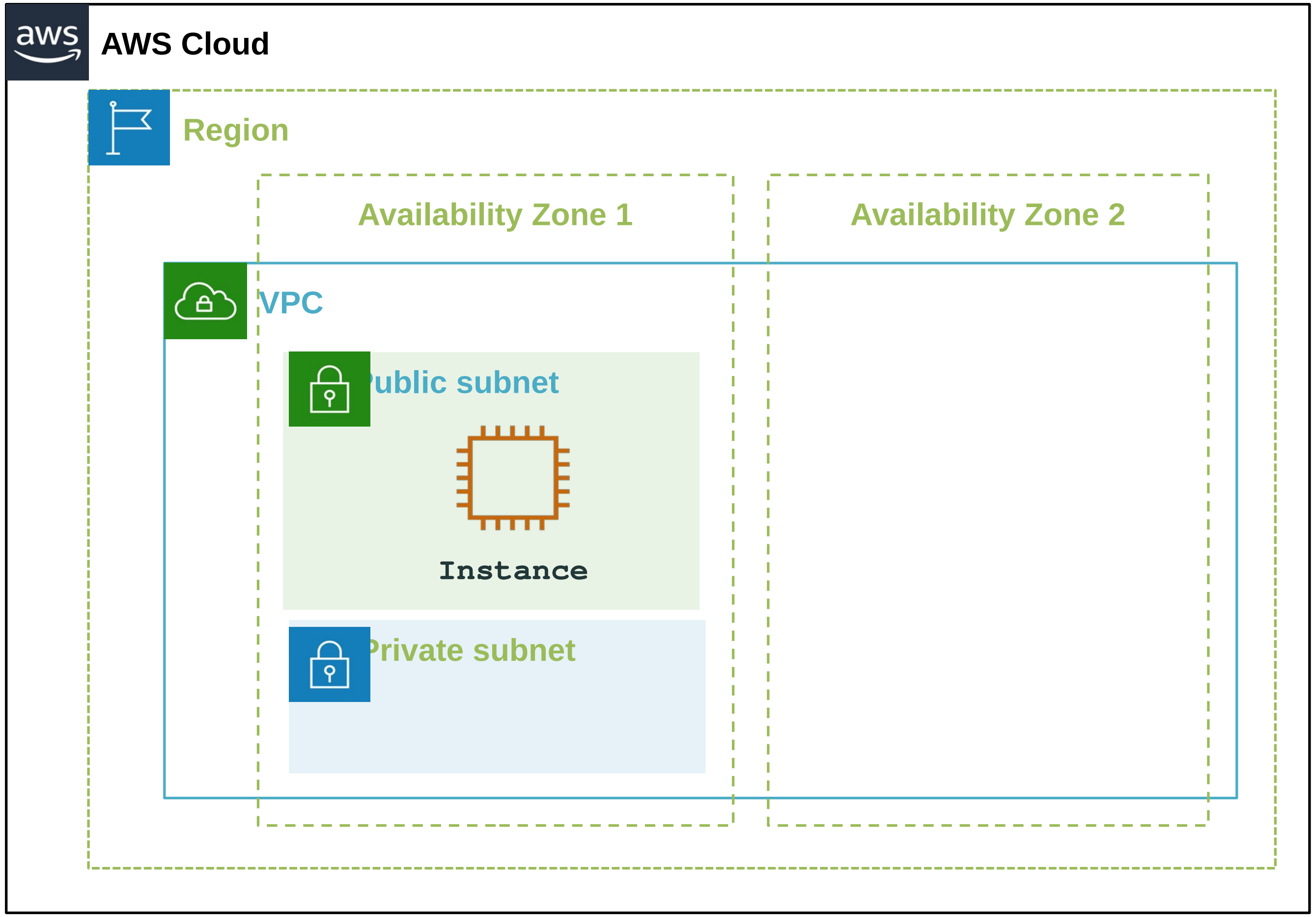
- Example: t3.large
  - T is the family name
  - 3 is the generation number
  - Large is the size

Example instance sizes

Instance Name	vCPU	Memory (GB)	Storage
t3.nano	2	0.5	EBS-Only
t3.micro	2	1	EBS-Only
t3.small	2	2	EBS-Only
t3.medium	2	4	EBS-Only
t3.large	2	8	EBS-Only
t3.xlarge	4	16	EBS-Only
t3.2xlarge	8	32	EBS-Only

3. Specify network settings

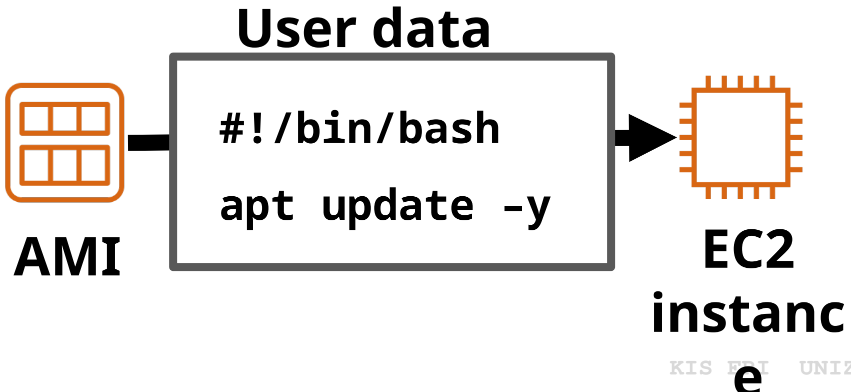
- Where should the instance be deployed?
  - Identify the VPC and optionally the subnet
- Should a public IP address be automatically assigned?
  - To make it internet-accessible
  - Public IP address is never assigned to instance directly
    - Via Floating IP



4. Attach IAM role & 5. User data script (optional)

- An AWS Identity and Access Management (IAM) role that is attached to an EC2 instance is kept in an instance profile.
- You are not restricted to attaching a role only at instance launch.
  - You can also attach a role to an instance that already exists.

- Optionally specify a user data script at instance launch
- Use user data scripts to customize the runtime environment of your instance
  - Script runs **only the first time** the instance starts



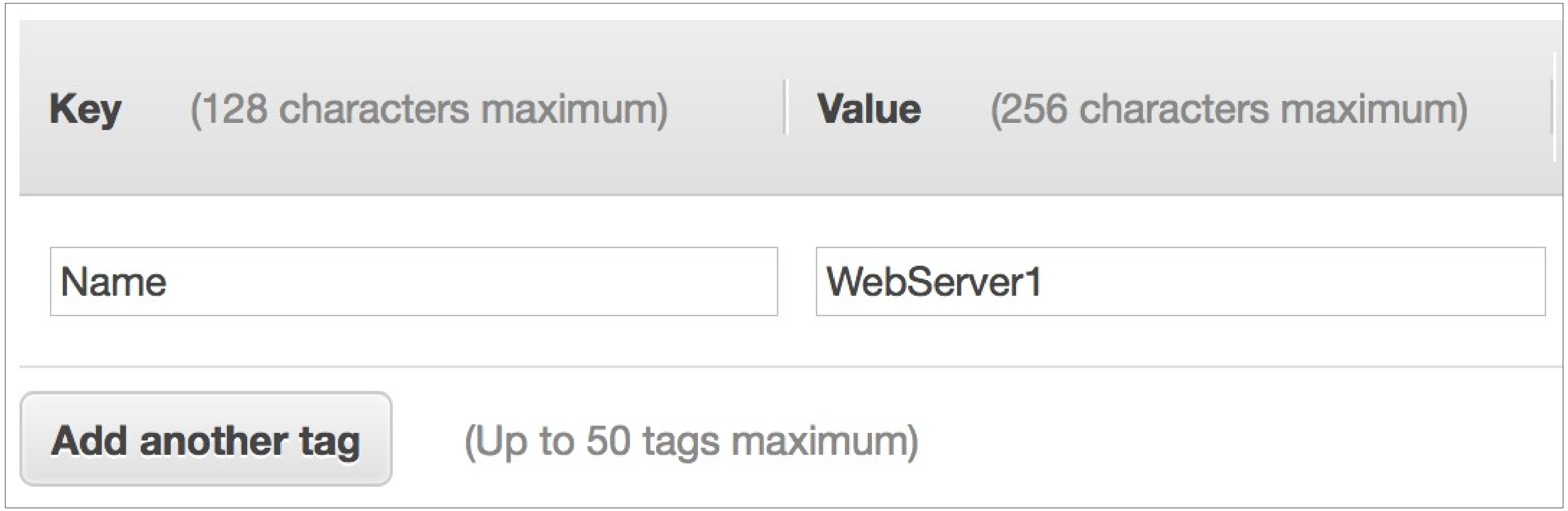
```
#!/bin/bash
apt update -y
```

6. Specify storage

- Configure the root volume
  - Where the guest operating system is installed
- Attach additional storage volumes (optional)
  - AMI might already include more than one volume
- For each volume, specify:
  - The size of the disk (in GB)
  - The volume type
    - Different types of solid state drives (SSDs) and hard disk drives (HDDs) are available
  - If the volume will be deleted when the instance is terminated
  - If encryption should be used

7. Add tags

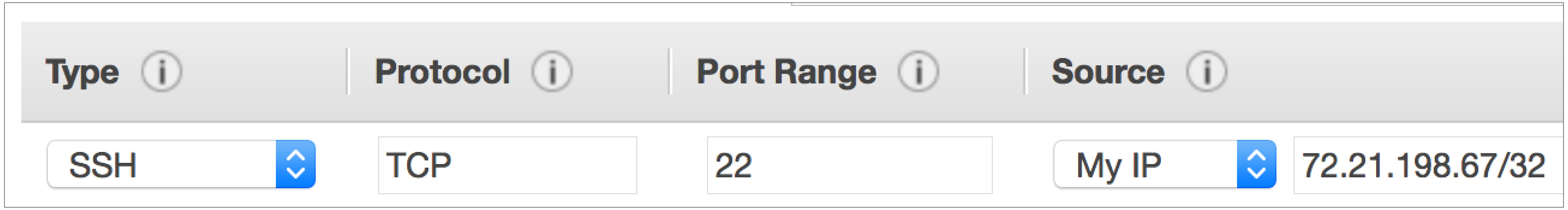
- A tag is a label that you can assign to an AWS resource
  - Consists of a key and an optional value
- Tagging is how you can attach metadata to an EC2 instance
- Potential benefits of tagging - Filtering, automation, cost allocation, and access control



Key (128 characters maximum)	Value (256 characters maximum)
Name	WebServer1

8. Security group settings

- A security group is a set of firewall rules that control traffic to the instance.
  - It exists outside of the instance's guest OS.
- Create rules that specify the source and which ports that network communications can use.
  - Specify the port number and the protocol, such as Transmission Control Protocol (TCP), User Datagram Protocol (UDP), or Internet Control Message Protocol (ICMP).
  - Specify the source (for example, an IP address or another security group) that is **allowed** to use the rule.

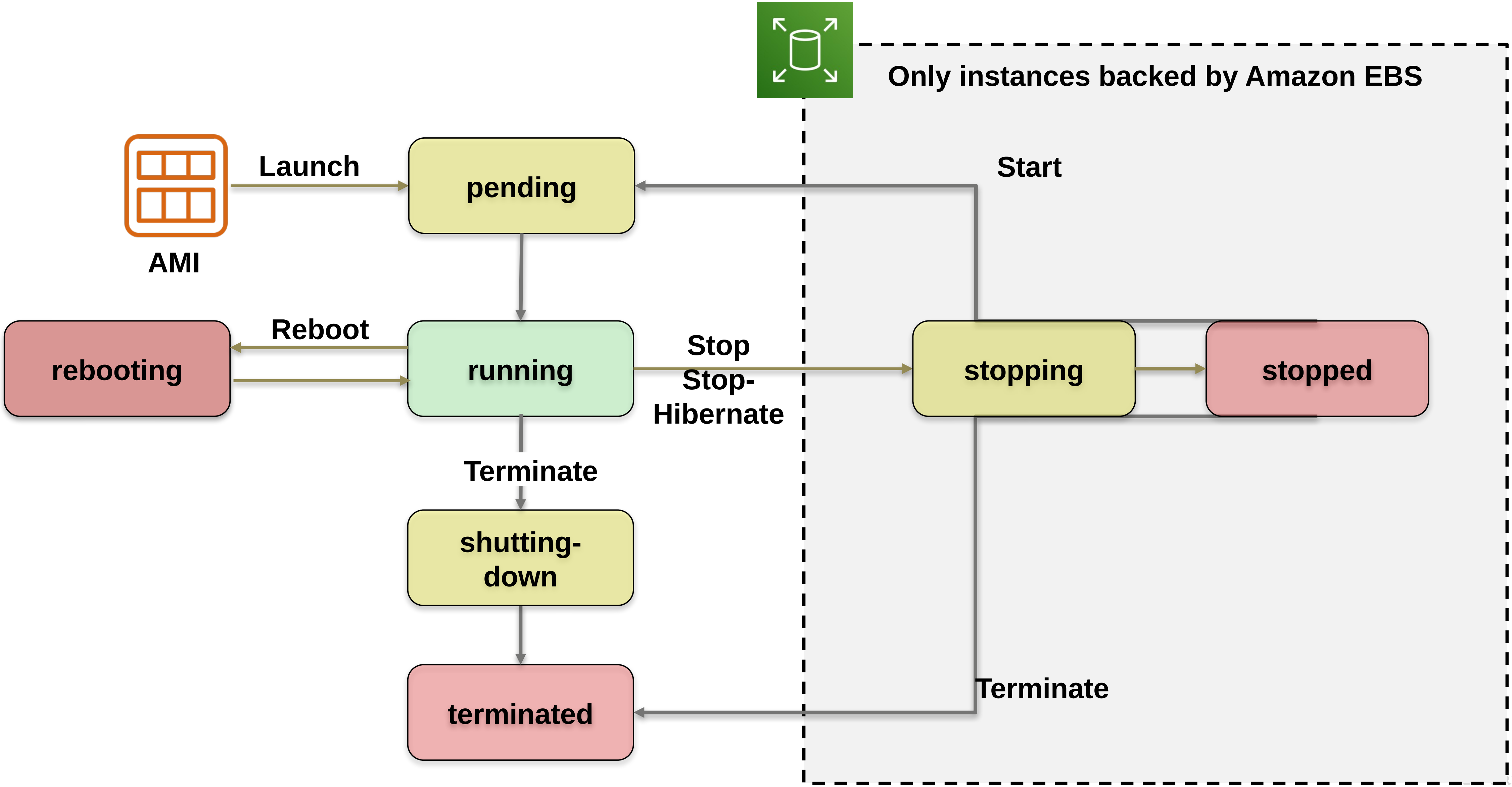


Type	Protocol	Port Range	Source	
SSH	TCP	22	My IP	72.21.198.67/32

9. Identify or create key pair

- At instance launch, you specify an existing key pair or create a new key pair.
- A key pair consists of
  - A public key that AWS stores.
  - A private key file that you store.
- It enables secure connections to the instance.
- For Windows AMIs
  - Use the private key to obtain the administrator password that you need to log in to your instance (SSH or RDP).
- For Linux AMIs
  - Use the private key to use SSH to securely connect to your instance.

Amazon EC2 instance lifecycle



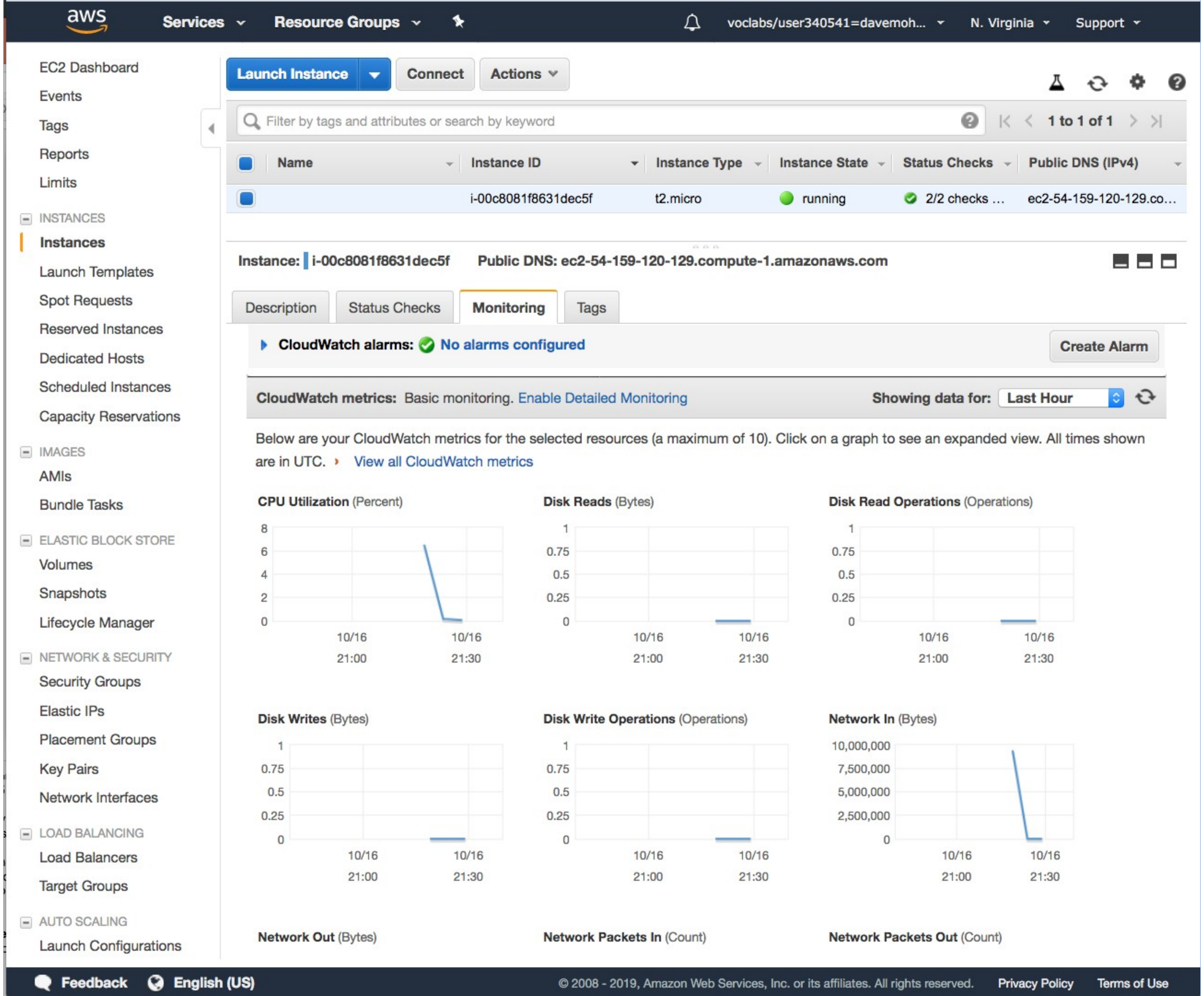
Amazon CloudWatch

- Use Amazon CloudWatch to monitor EC2 instances
  - Provides near-real-time metrics
  - Provides charts in the Amazon EC2 console Monitoring tab that you can view
  - Maintains 15 months of historical data

- Basic monitoring
  - Default, no additional cost
  - Metric data sent to CloudWatch every 5 minutes

- Detailed monitoring
  - Fixed monthly rate for seven pre-selected metrics
  - Metric data delivered every 1 minute

Amazon CloudWatch example



☐	Name	Instance ID	Instance Type	Instance State	Status Checks	Public DNS (IPv4)
■		i-00c8081f8631dec5f	t2.micro	● running	✓ 2/2 checks ...	ec2-54-159-120-129.co...

Instance: i-00c8081f8631dec5f Public DNS: ec2-54-159-120-129.compute-1.amazonaws.com

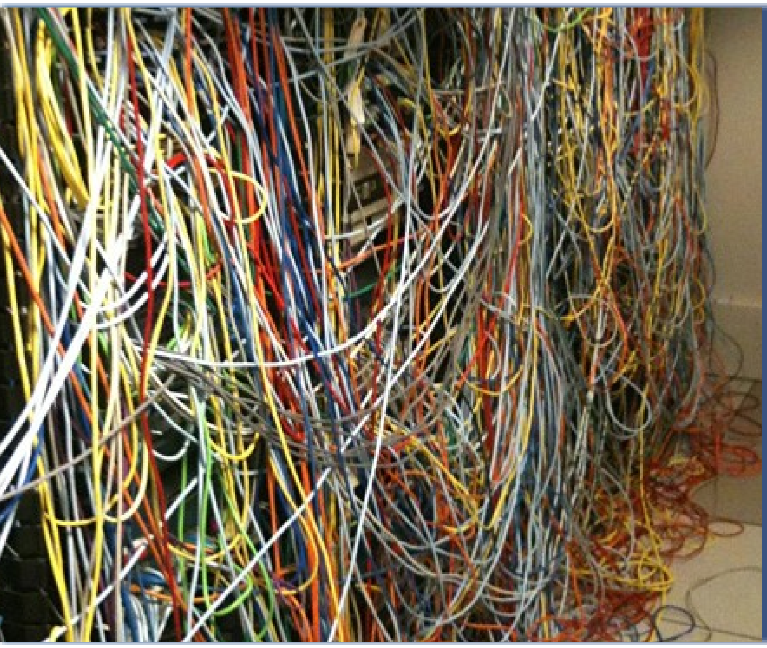
CloudWatch alarms: ✓ No alarms configured

CloudWatch metrics: Basic monitoring. Enable Detailed Monitoring

Showing data for: Last Hour

Below are your CloudWatch metrics for the selected resources (a maximum of 10). Click on a graph to see an expanded view. All times shown are in UTC. › View all CloudWatch metrics

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Amazon EC2 cost optimization

EC2 pricing models

- On-Demand Instances
  - Pay by the hour
  - No long-term commitments.
  - Eligible for the AWS Free Tier.
- Reserved Instances
  - Full, partial, or no upfront payment for instance you reserve.
  - Discount on hourly charge for that instance.
  - 1-year or 3-year term.
- Scheduled Reserved Instances
  - Purchase a capacity reservation that is always available on a recurring schedule you specify.
  - 1-year term.

EC2 pricing models (2)

- Spot Instances
  - Instances run as long as they are available and your bid is above the Spot Instance price.
  - They can be interrupted by AWS with a 2-minute notification.
  - Interruption options include terminated, stopped or hibernated.
  - Prices can be significantly less expensive compared to On-Demand Instances
  - Good choice when you have flexibility in when your applications can run.
- Dedicated Hosts
  - A physical server with EC2 instance capacity fully dedicated to your use.
- Dedicated Instances
  - Instances that run in a VPC on hardware that is dedicated to a single customer.

The four pillars of cost optimization

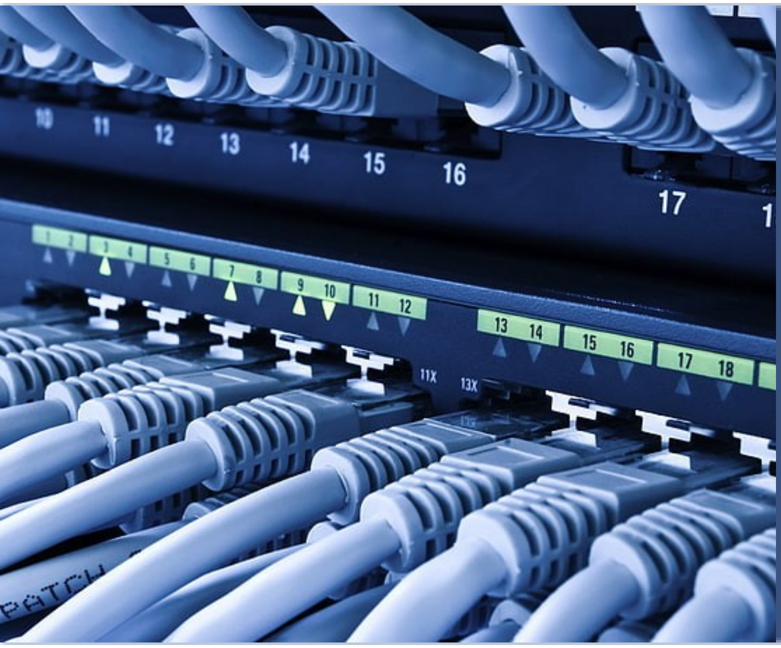
1. Right-size
   - Provision instances to match the need
   - Choose the right balance of instance types. Notice when servers can be either sized down or turned off, and still meet your performance requirements.

2. Increase elasticity
   - Use automatic scaling to match needs based on usage
   - Design your deployments to reduce the amount of server capacity that is idle by implementing deployments that are elastic, such as deployments that use automatic scaling to handle peak loads.

The four pillars of cost optimization (2)

3. Optimal pricing model
   - Optimize and **combine** purchase types
   - Recognize the available pricing options. Analyze your usage patterns so that you can run EC2 instances with the right mix of pricing options.

4. Optimize storage choices
   - Analyze the storage requirements of your deployments. Reduce unused storage overhead when possible, and choose less expensive storage options if they can still meet your requirements for storage performance.



Container services

What are containers?

- Containers are a method of operating system virtualization

- Benefits
  - Repeatable
  - Self-contained environments
  - Software runs the same in different environments
  - Developer's laptop, test, production
  - Faster to launch and stop or terminate than virtual machines

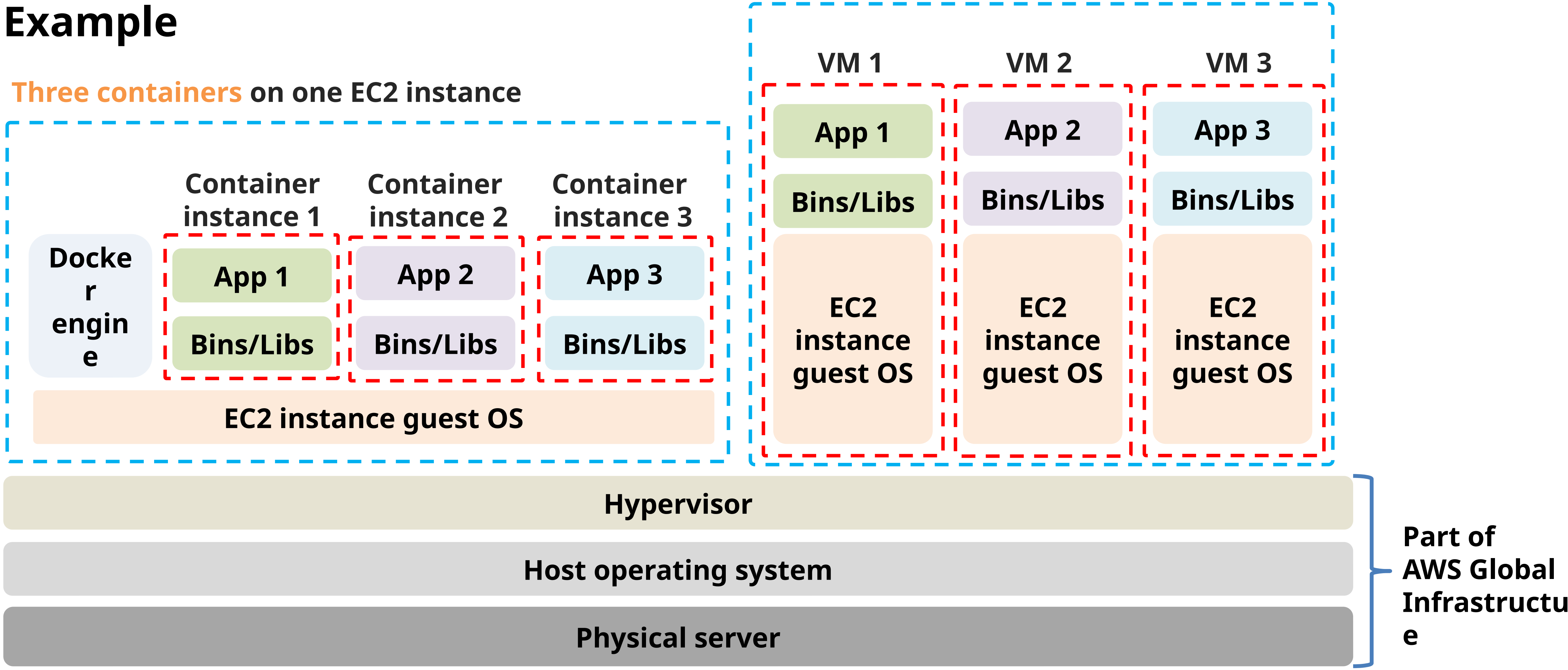
Docker

- Docker is a software platform that enables you to build, test, and deploy applications
- Containers are created from a template called an image
- A container has everything a software application needs to run
- There are Linux containers and Windows containers

Containers vs EC2 instances (virtual machines)

Example

Three containers on one EC2 instance



Container instance 1

Container instance 2

Container instance 3

Docke r engin e

App 1

Bins/Libs

App 2

Bins/Libs

App 3

Bins/Libs

EC2 instance guest OS

VM 1

VM 2

VM 3

App 1

Bins/Libs

EC2 instance guest OS

App 2

Bins/Libs

EC2 instance guest OS

App 3

Bins/Libs

EC2 instance guest OS

Hypervisor

Host operating system

Physical server

Part of AWS Global Infrastructu e

Amazon Elastic Container Service (Amazon ECS)

- Amazon Elastic Container Service (Amazon ECS) –
- A highly scalable, fast, **container management service**

- Key benefits
  - Orchestrates the running of Docker containers
  - Maintains and scales the fleet of nodes that run your containers
  - Removes the complexity of standing up the infrastructure

- Integrated with features that are familiar to Amazon EC2 service users –
  - Elastic Load Balancing
  - Amazon EC2 security groups
  - Amazon EBS volumes
  - IAM roles

Amazon Elastic Container Service

Kubernetes (K8s)

- Kubernetes is open source software for container orchestration.
  - Deploy and manage containerized applications at scale.
  - The same toolset can be used on premises and in the cloud.
- Complements Docker.
  - Docker enables you to run multiple containers on a single OS host.
  - Kubernetes orchestrates multiple Docker hosts (nodes).
- Automates
  - Container provisioning.
  - Networking.
  - Load distribution.
  - Scaling.

Amazon Elastic Kubernetes Service (Amazon EKS)

- Amazon Elastic Kubernetes Service (Amazon EKS)
  - Enables you to run Kubernetes on AWS
  - Certified Kubernetes conformant (supports easy migration)
  - Supports Linux and Windows containers
  - Compatible with Kubernetes community tools and supports popular Kubernetes add-ons

Amazon Elastic Kubernetes Service

- Use Amazon EKS to
  - Manage clusters of Amazon EC2 compute instances
  - Run containers that are orchestrated by Kubernetes on those instances

Ak chcete pridať obrázok, kliknite na ikonu

Introduction to AWS Lambda

AWS Lambda: Run code without servers

AWS Lambda

- AWS Lambda is event-driven a serverless compute service
  - Enables you to run code without provisioning or managing servers
- It supports multiple programming languages
  - Java, Go, PowerShell, Node.js, C#, Python, Ruby
- You pay only for the requests that are served and the compute time that is required to run your code
  - Billing is metered in increments of 100 milliseconds

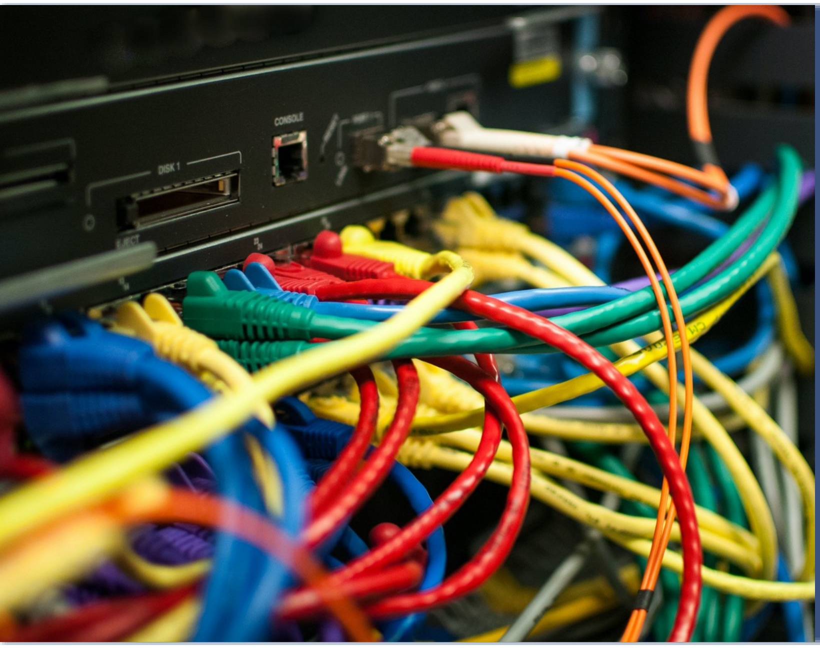
AWS Lambda quotas

Soft limits per Region:

- Concurrent executions = 1,000
- Function and layer storage = 75 GB

Hard limits for individual functions:

- Maximum function memory allocation = 3,008 MB
- Function timeout = 15 minutes
- Deployment package size = 250 MB unzipped, including layers



Introduction to AWS Elastic Beanstalk

AWS Elastic Beanstalk

- An easy way to get web applications up and running

- A managed service that automatically handles –
  - Infrastructure provisioning and configuration
  - Deployment
  - Load balancing
  - Automatic scaling
  - Health monitoring
  - Analysis and debugging
  - Logging

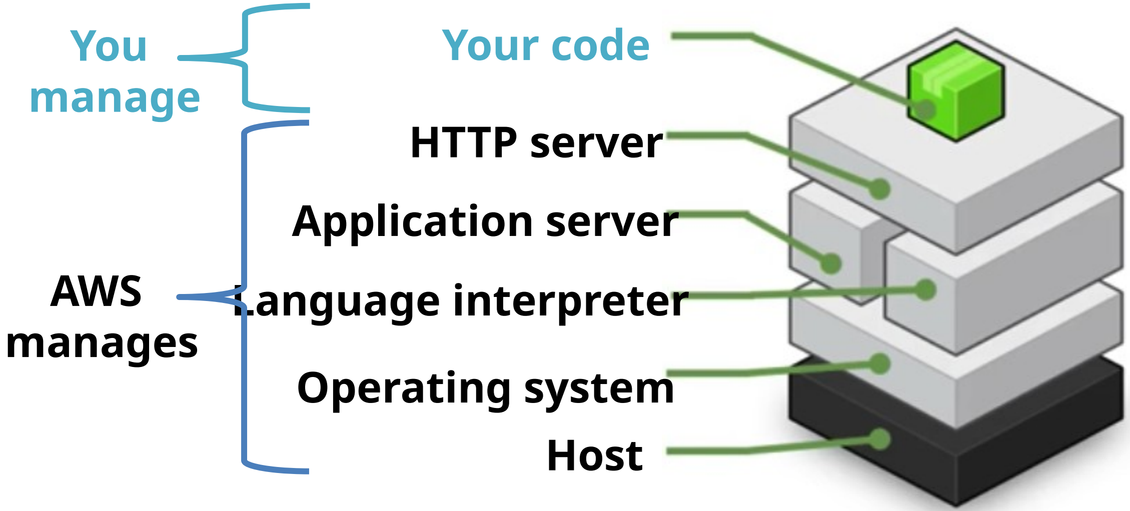
- No additional charge for Elastic Beanstalk
  - Pay only for the underlying resources that are used

AWS Elastic Beanstalk

AWS Elastic Beanstalk

- It supports web applications written for common platforms
  - Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker

- You upload your code
- Elastic Beanstalk automatically handles the deployment
  - Deploys on servers such as Apache, NGINX, Passenger, Puma, and Microsoft Internet Information Services (IIS)



UNIVERSITY OF ŽILINA
Faculty of Management Science
and Informatics

MINISTERSTVO
ŠKOLSTVA, VEDY,
VÝSKUMU A ŠPORTU
SLOVENSKEJ REPUBLIKY

Thank you for your attention.

The content was chapter from AWS Foundations Module 6 - Compute

Vytvorené v rámci projektu **KEGA 026TUKE-4/2021**