# Lesson 12 Auto Scaling Group

Michael Yang

# Why High Availability

- The following scenarios could cause an outage of your virtual machine:

- A software problem causes the virtual machine's OS to fail.
- A software problem occurs on the host machine, causing the virtual machine to fail (either the OS of the host machine fails or the virtualization layer does).
- The compute, storage, or networking hardware of the physical host fails.
- Parts of the data center that the virtual machine depends on fail: network connectivity, the power, or the cooling system.

# What is High Availability

- *High availability* describes a system that is operating with almost no downtime. Even if a failure occurs, the system can provide its services most of the time. The Harvard Research Group (HRG) defines high availability with the classification AEC-2, which requires an uptime of 99.99% over a year, or not more than 52 minutes and 35.7 seconds of downtime per year. You can achieve 99.99% uptime with EC2 instances if you follow the instructions in the rest of this chapter. Although a short interruption might be necessary to recover from a failure, no human intervention is needed to instigate the recovery

# AWS HA

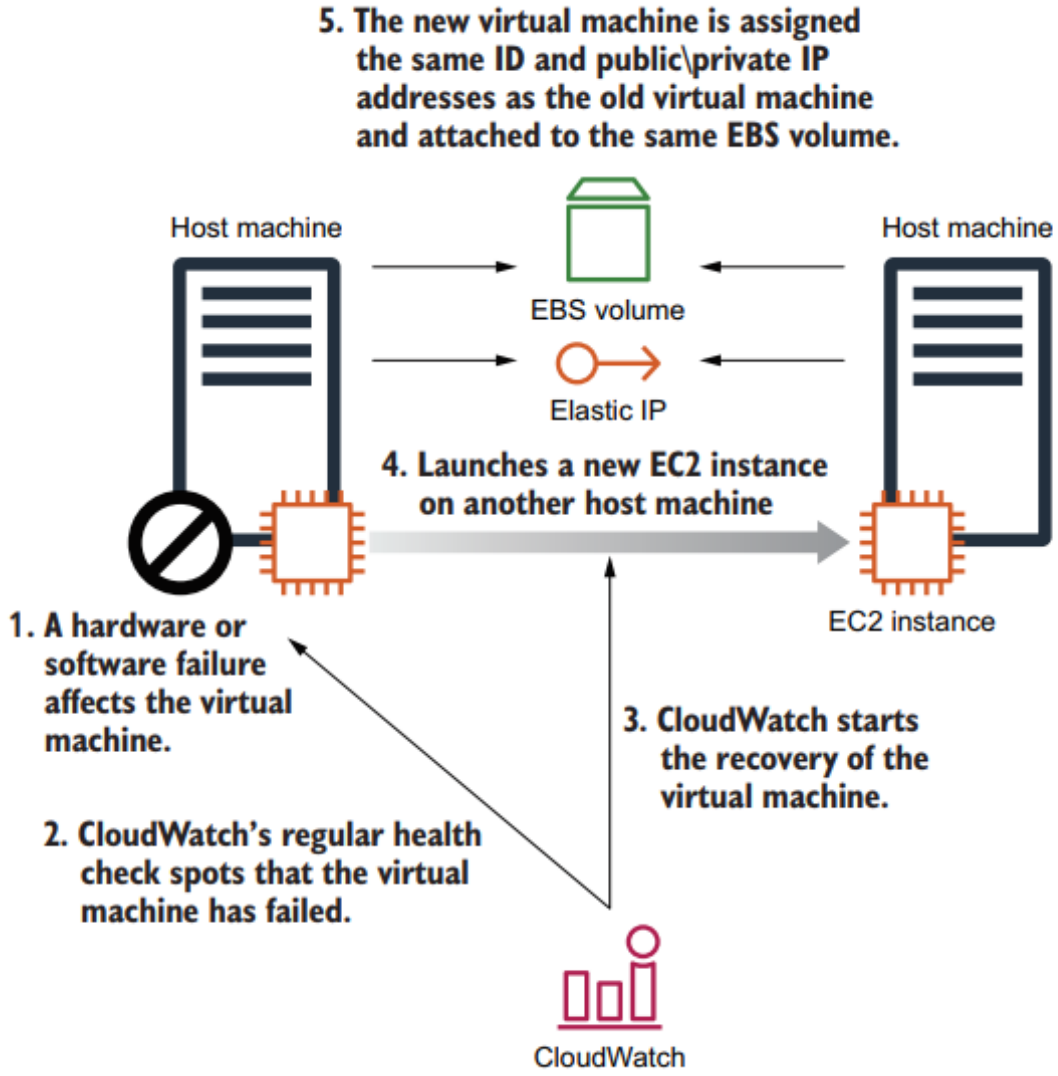
- Building a highly available infrastructure by using groups of isolated data centers, called availability zones, within a region.
- Monitoring the health of virtual machines with CloudWatch and triggering recovery automatically, if needed. This option fits best for workloads that need to run on a single virtual machine.
- Using autoscaling to guarantee a certain number of virtual machines are up and running and replace failed instances automatically. Use this approach when distributing your workload among multiple virtual machines is an option.

# Recovering with CloudWatch

In case the EC2 instance fails, AWS will not replace the instance automatically under all circumstances. Therefore, you need to create a CloudWatch alarm to trigger the recovery of the virtual machine automatically. A CloudWatch alarm consists of the following:

- A metric that monitors data (health check, CPU usage, and so on)
- A rule defining a threshold based on a statistical function over a period of time
- Actions to trigger if the state of the alarm changes (such as triggering a recovery of an EC2 instance if the state changes to `ALARM`)

# Recovering with CloudWatch



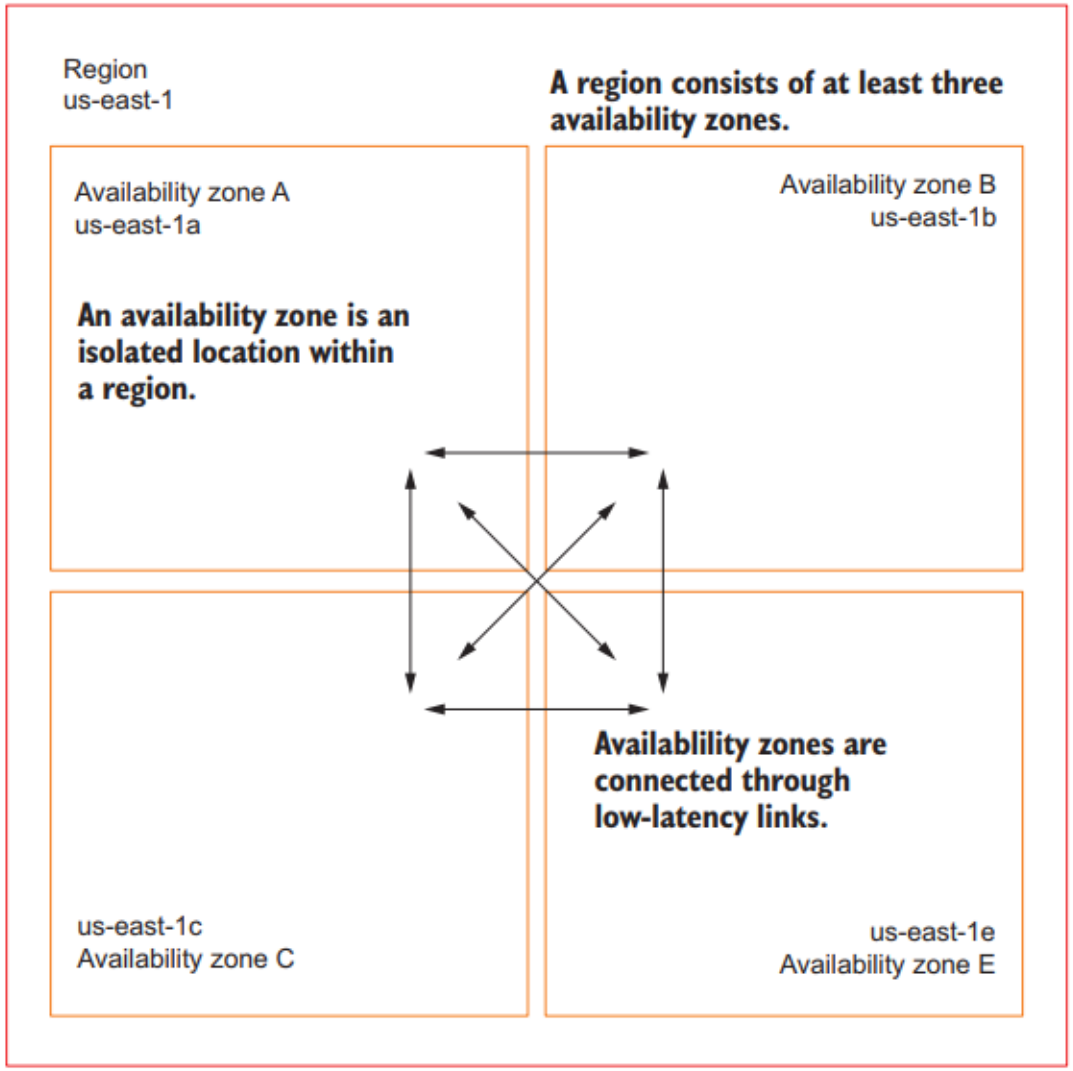
# Requirements for EC2 Recovering

- It must be running on a VPC network.
- The instance family must be A1, C3, C4, C5, C5a, C5n, C6g, C6gn, Inf1, C6i, M3, M4, M5, M5a, M5n, M5zn, M6g, M6i, P3, R3, R4, R5, R5a, R5b, R5n, R6g, R6i, T2, T3, T3a, T4g, X1, or X1e. Other instance families aren't supported.
- The EC2 instance must use EBS volumes exclusively, because data on instance storage would be lost after the instance was recovered.

# Recovering with ASG

- AWS is built for failure, even in the rare case that an entire data center fails. The AWS regions consist of multiple data centers grouped into availability zones. By distributing your workload among multiple availability zones, you are able to recover from an data center outage.

# Availability Zones(AZ)



# Recovering with ASG

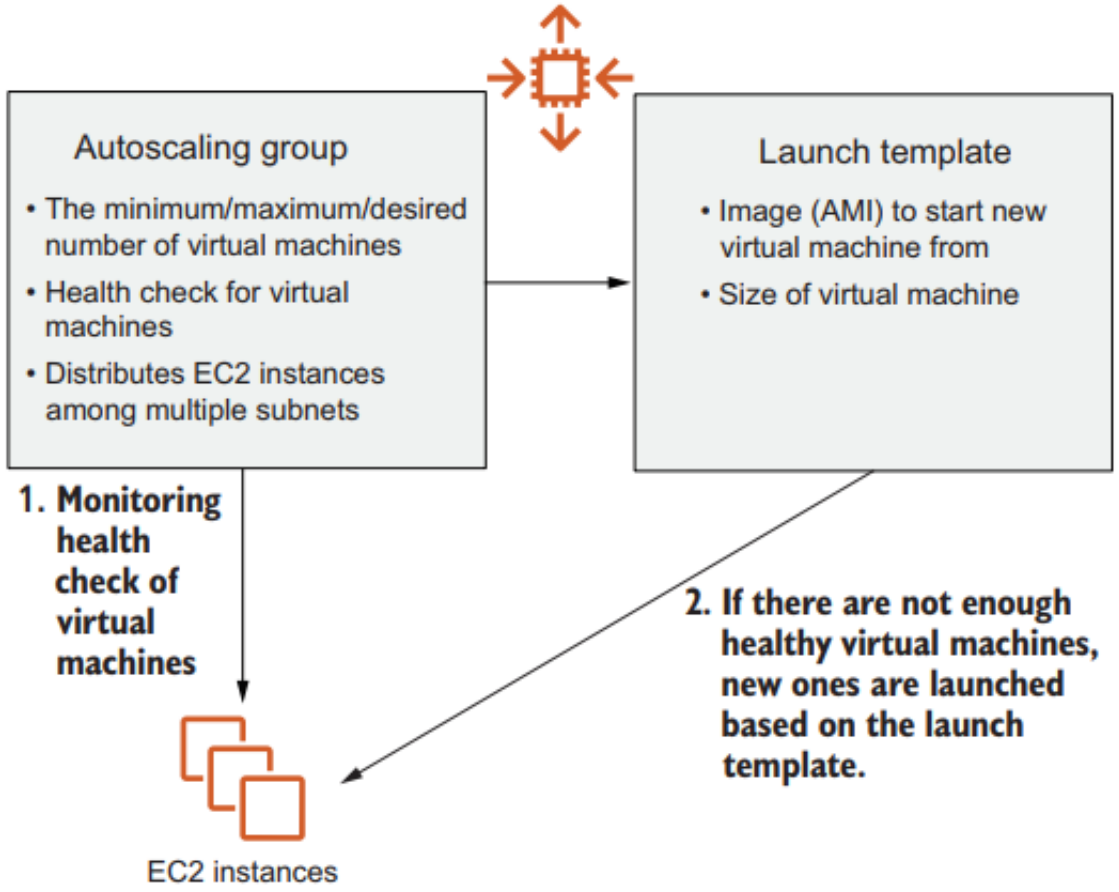
- Failing over into another availability zone is possible with the help of *autoscaling*.
- Autoscaling is part of the EC2 service and helps you to ensure that a specified number of EC2 instances is running

# How to Auto Scale

To configure autoscaling, you need to create the following two parts of the configuration:

- A *launch template* contains all information needed to launch an EC2 instance: instance type (size of virtual machine) and image (AMI) to start from.
- An *Auto Scaling group* tells the EC2 service how many virtual machines should be started with a specific launch template, how to monitor the instances, and in which subnets EC2 instances should be started.

# How to Auto Scale




Autoscaling ensures that a specified number of EC2 instances are running.

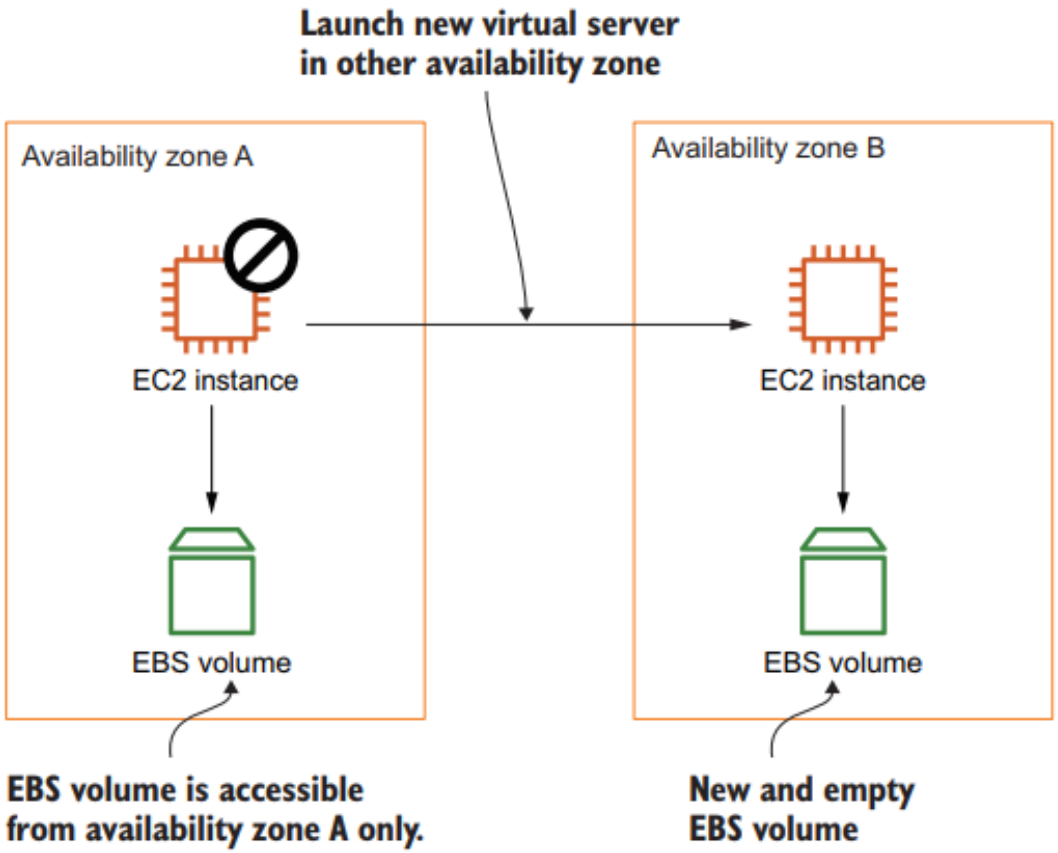
# Auto Scaling Parameters

| Context | Property | Description | Values |
|---|---|---|---|
| `LaunchTemplate` | `ImageId` | The ID of the AMI from which the virtual machine should be started | Any AMI ID accessible from your account |
| `LaunchTemplate` | `InstanceType` | The size of the virtual machine | All available instance sizes, such as `t2.micro`, `m3.medium`, and `c3.large` |
| `LaunchTemplate` | `SecurityGroupIds` | References the security groups for the EC2 instance | Any security group belonging to the same VPC |
| `LaunchTemplate` | `UserData` | Script executed during bootstrap to install the Jenkins CI server | Any bash script |

# Auto Scaling Parameters

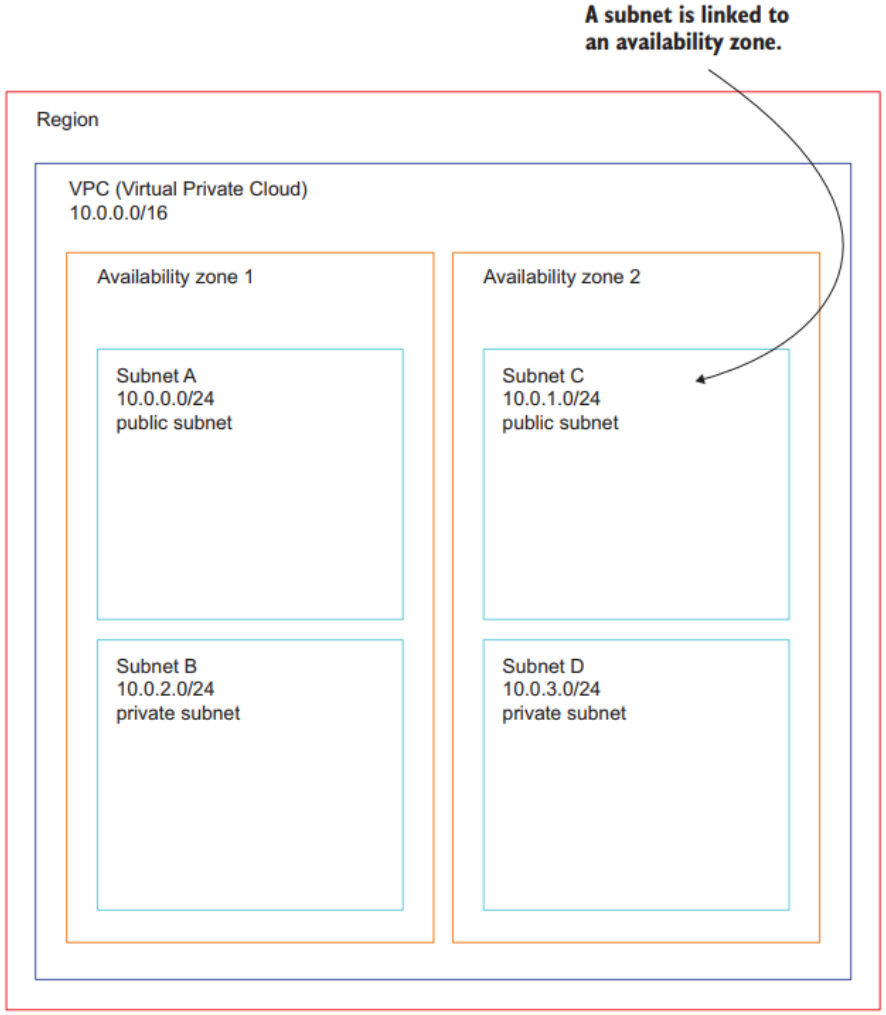
| Context | Property | Description | Values |
|---|---|---|---|
| `AutoScalingGroup` | `MinSize` | The minimum value for the `DesiredCapacity` | Any positive integer—use `1` if you want a single virtual machine to be started based on the launch template. |
| `AutoScalingGroup` | `MaxSize` | The maximum value for the `DesiredCapacity` | Any positive integer (greater than or equal to the `MinSize` value); use 1 if you want a single virtual machine to be started based on the launch template. |
| `AutoScalingGroup` | `VPCZoneIdentifier` | The subnet IDs in which you want to start virtual machines | Any subnet ID from a VPC from your account. Subnets must belong to the same VPC. |
| `AutoScalingGroup` | `HealthCheckType` | The health check used to identify failed virtual machines. If the health check fails, the Auto Scaling group replaces the virtual machine with a new one. | `EC2` to use the status checks of the virtual machine, or `ELB` to use the health check of the load balancer (see chapter 16). |

# Pitfall: Recovering network-attached storage



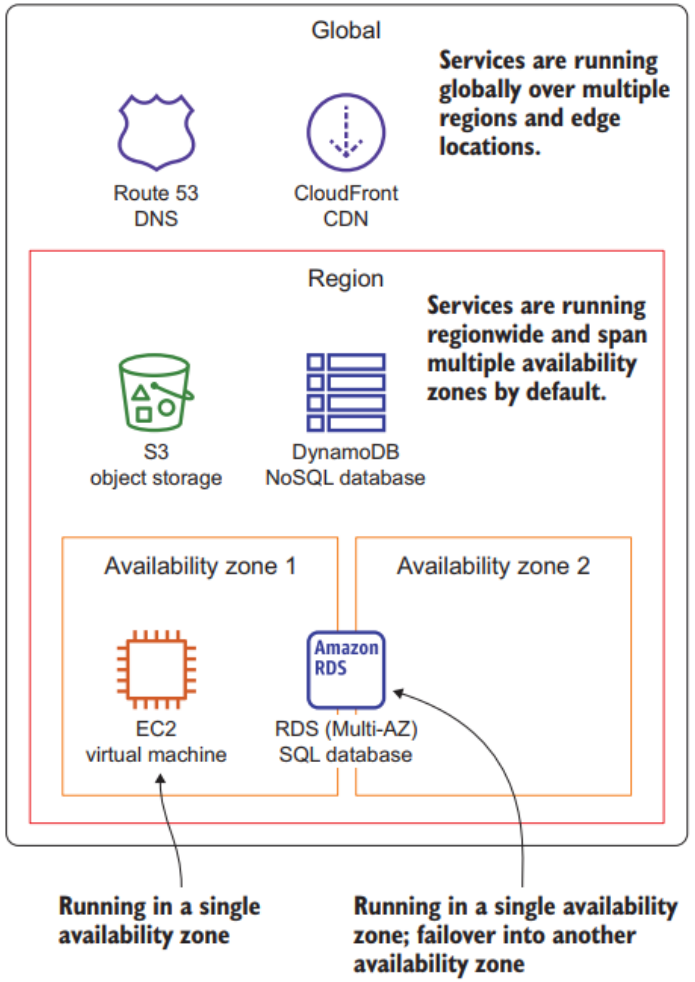

An EBS volume is available only in a single availability zone.

# Pitfall: Network interface recovery




A VPC is bound to a region, and a subnet is linked to an availability zone.

# AWS services HA guarantee




AWS services can operate in a single availability zone, over multiple availability zones within a region, or even globally.