AWS SA ( Associate ) AutoScaling QA ( sample )

Q1. Your architecture for an application currently consists of EC2 Instances sitting behind a classic ELB. The EC2 Instances are used to serve an application and are accessible through the internet. What can be done to improve this architecture in the event that the number of users accessing the application increases?

**A.**Add another ELB to the architecture.

**B.**Use Auto Scaling Groups.

**C.**Use an Application Load Balancer instead.

**D.**Use the Elastic Container Service.

**B.**Use Auto Scaling Groups.  
AWS Documentation mentions the following:  
AWS Auto Scaling monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost. Using AWS Auto Scaling, it is easy to setup application scaling for multiple resources across multiple services in minutes.  
For more information on AWS Auto Scaling, please visit the following URL:  
<https://aws.amazon.com/autoscaling/>

Q2. You create an Auto Scaling Group which is used to spin up instances On Demand. As an architect, you need to ensure that the instances are pre-installed with a software when they are launched. What are the ways in which you can achieve this? **Choose 2 answers from the options given below.**

**A.**Add the software installation to the configuration for the Auto Scaling Group.

**B.**Add the scripts for the installation in the User data section.

**C.**Create a golden image and then create a launch configuration.

**D.**Ask the IT operations team to install the software as soon as the instance is launched.

**B. & C.**  
The User data section of an instance launch can be used to pre-configure software after the instance is initially booted.  
For more information on User data, please visit the below URL:  
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/user-data.html  
Also, you can create an AMI or a golden image with the already installed software, then create a launch configuration which can be used by that Auto Scaling Group.  
For more information on AMIs please visit the below URL:  
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AMIs.html>

Q3.You have a business-critical two-tier web application currently deployed in 2 Availability Zones in a single region, using Elastic Load Balancing and Auto Scaling. The app depends on synchronous replication at the database layer. The application needs to remain fully available even if one application AZ goes offline and if Auto Scaling cannot launch new instances in the remaining AZ. How can the current architecture be enhanced to ensure this?

**A.**Deploy in 2 regions using Weighted Round Robin with Auto Scaling minimums set at 50%peak load per region.

**B.**Deploy in 3 AZ with Auto Scaling minimum set to handle 33 per cent peak load per zone.

**C.**Deploy in 3 AZ with Auto Scaling minimum set to handle 50 per cent peak load per zone.

**D.**Deploy in 2 regions using Weighted Round Robin with Auto Scaling minimums set at 100%peak load per region.

**C.**Deploy in 3 AZ with Auto Scaling minimum set to handle 50 per cent peak load per zone.  
Since the requirement states that the application should never go down even if an AZ is not available, we need to maintain 100% availability.  
Options A and D are incorrect because region deployment is not possible for ELB. ELBs can manage traffic within a region and not between regions. Option B is incorrect because even if one AZ goes down, we would be operating at only 66% and not the required 100%.  
For more information on Auto Scaling, please visit the below URL:  
<https://aws.amazon.com/autoscaling/>

Q4. An infrastructure is being hosted in AWS using the following resources:  
a)A couple of EC2 Instances serving a Web-Based application  
b)An Elastic Balancer in front of the EC2 Instances  
c)An AWS RDS which has Multi-AZ enabled  
Which of the following can be added to the setup to ensure scalability?

**A.**Add another ELB to the setup.

**B.**Add more EC2 Instances to the setup.

**C.**Enable Read Replicas for the AWS RDS.

**D.**Add an Auto Scaling Group to the setup.

**D.**Add an Auto Scaling Group to the setup.  
AWS Documentation mentions the following:  
AWS Auto Scaling enables you to configure automatic scaling for the scalable AWS resources for your application in a matter of minutes. AWS Auto Scaling uses the Auto Scaling and Application Auto Scaling services to configure scaling policies for your scalable AWS resources.  
For more information on AWS Auto Scaling, please visit the URL below.  
<https://docs.aws.amazon.com/autoscaling/plans/userguide/what-is-aws-auto-scaling.html>

Q5. Your Operations department is using an incident based application hosted on a set of EC2 Instances. These instances are placed behind an Auto Scaling Group to ensure the right number of instances are in place to support the application. The Operations department has expressed dissatisfaction with regard to poor application performance at 9:00 AM each day. However, it is also noted that the system performance returns to optimal at 9:45 AM.What can be done to ensure that this issue gets fixed?

**A.**Create another Dynamic Scaling Policy to ensure that the scaling happens at 9:00 AM.

**B.**Add another Auto Scaling group to support the current one.

**C.**Change the Cool Down Timers for the existing Auto Scaling Group.

**D.**Add a Scheduled Scaling Policy at 8:30 AM.

**D.**Add a Scheduled Scaling Policy at 8:30 AM.  
Scheduled Scaling can be used to ensure that the capacity is peaked before 9:00 AM each day.  
AWS Documentation further mentions the following on Scheduled Scaling:  
Scaling based on a schedule allows you to scale your application in response to predictable load changes. For example, every week the traffic to your web application starts to increase on Wednesday, remains high on Thursday, and starts to decrease on Friday. You can plan your scaling activities based on the predictable traffic patterns of your web application.  
For more information on Scheduled Scaling, please refer to the below URL:  
<https://docs.aws.amazon.com/autoscaling/ec2/userguide/schedule_time.html>

Q6. An application consists of the following architecture:  
a. EC2 Instances in a single AZ behind an ELB  
b. A NAT Instance which is used to ensure that instances can download updates from the Internet  
Which of the following can be used to ensure better fault tolerance in this setup? Choose 2 answers from the options given below.

**A.**Add more instances in the existing Availability Zone.

**B.**Add an Auto Scaling Group to the setup.

**C.**Add more instances in another Availability Zone.

**D.**Add another ELB for more fault tolerance.

**B. & C.**  
AWS Documentation mentions the following:  
Adding Auto Scaling to your application architecture is one way to maximize the benefits of the AWS Cloud. When you use Auto Scaling, your applications gain the following benefits:  
Better fault tolerance. Auto Scaling can detect when an instance is unhealthy, terminate it, and launch an instance to replace it. You can also configure Auto Scaling to use multiple Availability Zones. If one Availability Zone becomes unavailable, Auto Scaling can launch instances in another one to compensate. Better availability. Auto Scaling can help you ensure that your application always has the right amount of capacity to handle the current traffic demands.  
For more information on the benefits of Auto Scaling, please visit the following URL:  
<https://docs.aws.amazon.com/autoscaling/ec2/userguide/auto-scaling-benefits.html>

Q7. While reviewing the Auto Scaling events for your application, you notice that your application is scaling up and down multiple times in the same hour.  
What design choice could you make to optimize costs while preserving elasticity?  
**Choose 2 answers from the options given below**.

**A.**Modify the Auto Scaling group termination policy to terminate the older instance first.

**B.**Modify the Auto Scaling group termination policy to terminate the newest instance first.

**C.**Modify the Auto Scaling group cool down timers.

**D.**Modify the CloudWatch alarm period that triggers your Auto Scaling scale down policy

**C. & D**  
Here, not enough time is being given for the scaling activity to take effect and for the entire infrastructure to stabilize after the scaling activity. This can be taken care of by increasing the Auto Scaling group CoolDown timers.

For more information on Auto Scaling CoolDown, please visit the following URL:

https://docs.aws.amazon.com/autoscaling/ec2/userguide/Cooldown.html  
You will also have to define the right threshold for the CloudWatch alarm for triggering the scale down policy.

For more information on Auto Scaling Dynamic Scaling, please visit the following URL:

https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-scale-based-on-demand.html

Q8.An application hosted on EC2 Instances has its promotional campaign due to start in 2 weeks. There is a mandate from the management to ensure that no performance problems are encountered due to traffic growth during this time. Which of the following must be done to the Auto Scaling Group to ensure this requirement can be fulfilled?

**A.**Configure Step scaling for the Auto Scaling Group.

**B.**Configure Dynamic Scaling and use Target tracking scaling Policy

**C.**Configure Scheduled scaling for the Auto Scaling Group

**D.**Configure Static scaling for the Auto Scaling Group

**B.**Configure Dynamic Scaling and use Target tracking scaling Policy  
If you are scaling is based on a metric, which is an utilization metric that increases or decreases proportionally to the number of instances in the Auto Scaling group, we recommend that you use a target tracking scaling policy instead.

In Target tracking scaling policies you select a predefined metric or configure a customized metric, and set a target value. EC2 Auto Scaling creates and manages the CloudWatch alarms that trigger the scaling policy and calculates the scaling adjustment based on the metric and the target value. The scaling policy adds or removes capacity as required to keep the metric at, or close to, the specified target value.

Scheduled scaling works better when you can predict the load changes and also when you know how long you need to run. Here in our scenario we just know that there will be a heavy traffic during the campaign period (period is not specified) but not sure about the actual traffic. Don’t have any history to predict it either.

For more information on Auto Scaling Scheduled Scaling, please visit the following URL:

https://docs.aws.amazon.com/autoscaling/ec2/userguide/schedule\_time.html  
https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-scaling-simple-step.html  
https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-scaling-target-tracking.html

Q9.You have an application hosted on AWS consisting of EC2 Instances launched via an Auto Scaling Group. You notice that the EC2 Instances are not scaling up on demand. What checks can be done to ensure that the scaling occurs as expected?

**A.**Ensure that the right metrics are being used to trigger the scale out.

**B.**Ensure that ELB health checks are being used.

**C.**Ensure that the instances are placed across multiple Availability Zones.

**D.**Ensure that the instances are placed across multiple regions.

**A.**Ensure that the right metrics are being used to trigger the scale out.  
If your scaling events are not based on the right metrics and do not have the right threshold defined, then the scaling will not occur as you want it to happen.  
For more information on Auto Scaling Dynamic Scaling, please visit the following URL:  
https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-scale-based-on-demand.html