## **EC2 :Elastic Cloud Compute**

Placement groups:

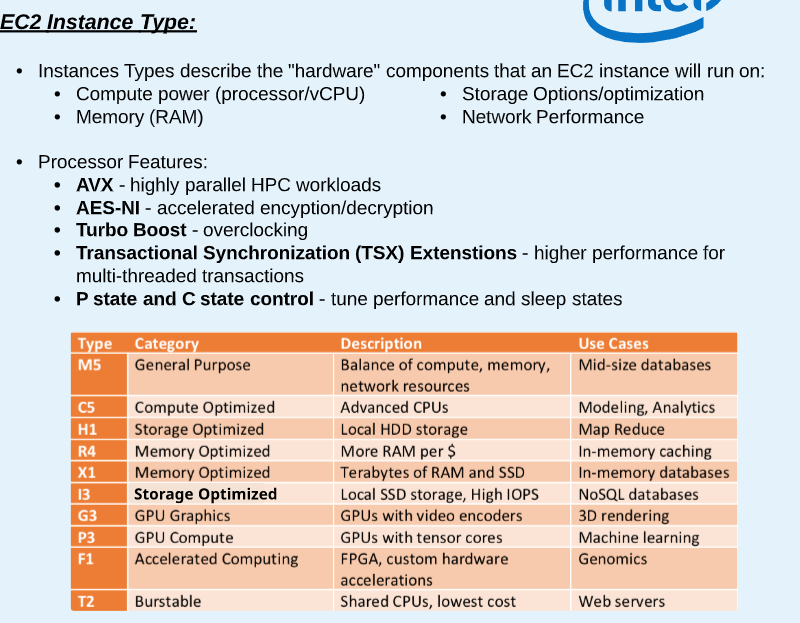
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-groups.html>

EC2-quiz:

<https://quizlet.com/335697525/ec2-quiz-flash-cards/>



**Instance types with use and Examples**



AWS EBS vs Object store(S3) 

EBS can only be attached to 1 EC2 at a time and

Both EC2 and EBS must be in the same AZ.

EFS is a block level storage that can be mounted to n number of ec2-instances within the same region. EFS replicates itself across multiple AZ within a region.

EFS can be mounted to on-prem with Direct Connect.

No need of preplanning for size like EBS

**Note:** After creating Security group,

By default,

* All outbound traffic is allowed.
* No inbound traffic is allowed.

Can be modified anytime, applies immediately to instances after security group’s rules are changed.

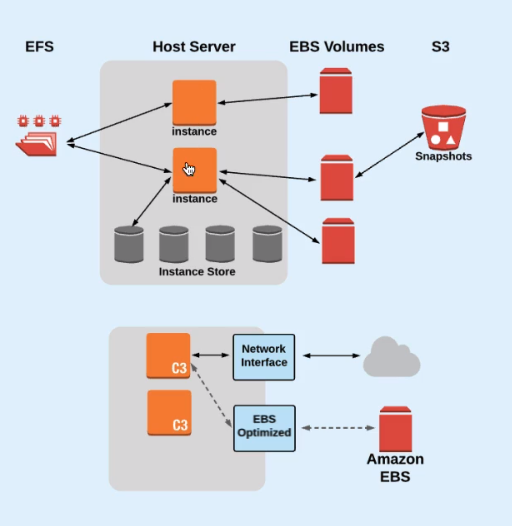
## **Security Group Rules**

The rules of a security group control the inbound traffic that's allowed to reach the instances that are associated with the security group and the outbound traffic that's allowed to leave them.

The following are the characteristics of security group rules:

* By default, security groups allow all outbound traffic.
* Security group rules are always permissive; you can't create rules that deny access.
* Security groups are stateful — if you send a request from your instance, the response traffic for that request is allowed to flow in regardless of inbound security group rules. For VPC security groups, this also means that responses to allowed inbound traffic are allowed to flow out, regardless of outbound rules.
* You can add and remove rules at any time. Your changes are automatically applied to the instances associated with the security group.

EC2 storage options:



EFS can be shared with multiple instances

EBS to only one instance at a time

Note: EBS volumes can be backed up as volumes in S3 (Snapshots)

Want a new instance with different configuration. How to do it with EBS and EC2?

1. Create a new EC2 with new required configuration
2. Attach the EBS volume of old instance to the newly created instance
3. Delete the old EC2 instance

Note: EBS has 99.999% availability

If EBS fails, the snapshot can be applied to a new EBS volume and attach it to the same instance

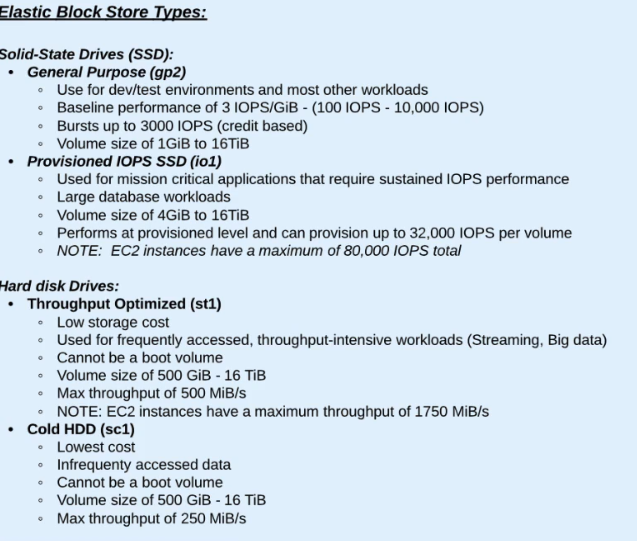
Note: When you are using an EBS volume that has been created from a Snapshot, the files stored in the snapshot, the files stored in the snapshot are nit copied to the blocks of the volume until you read that block (Lazy Loading)

Reason to do it: Volume available right away, no waiting

If we don’t want lazy loading, write a script to read data from every block on the volume

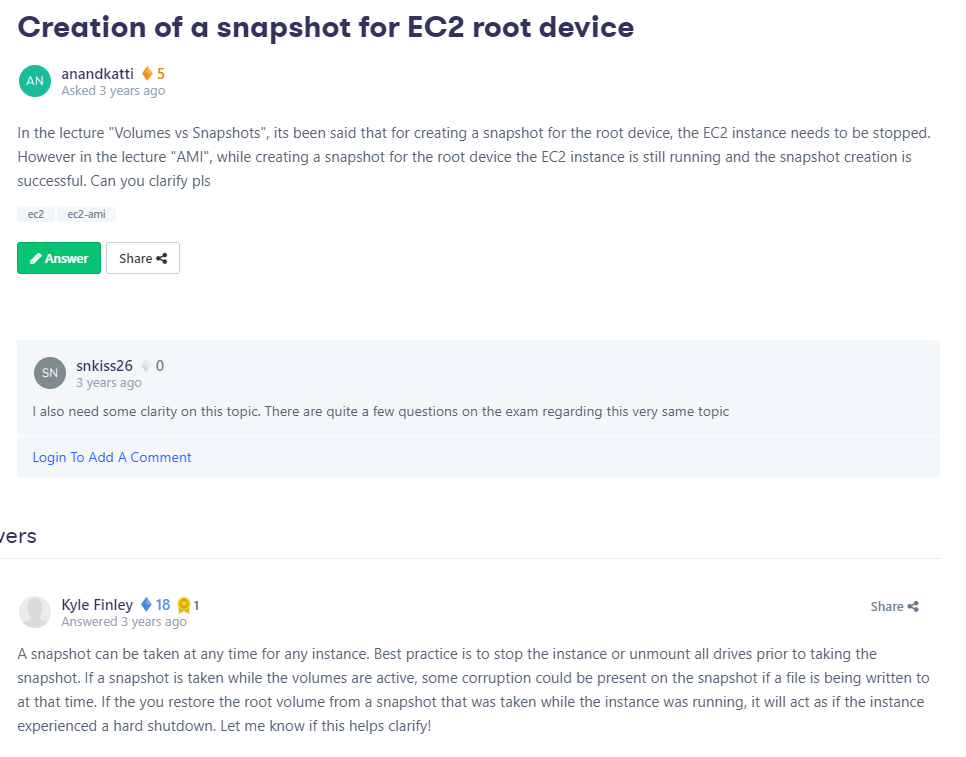
**NOTE**:  Instance Store volumes do not offer a snapshot capability.

**EBS Types :**

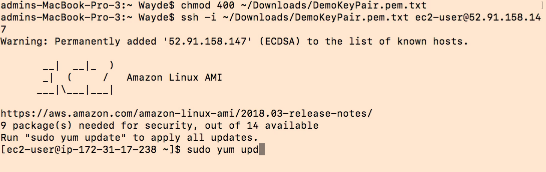


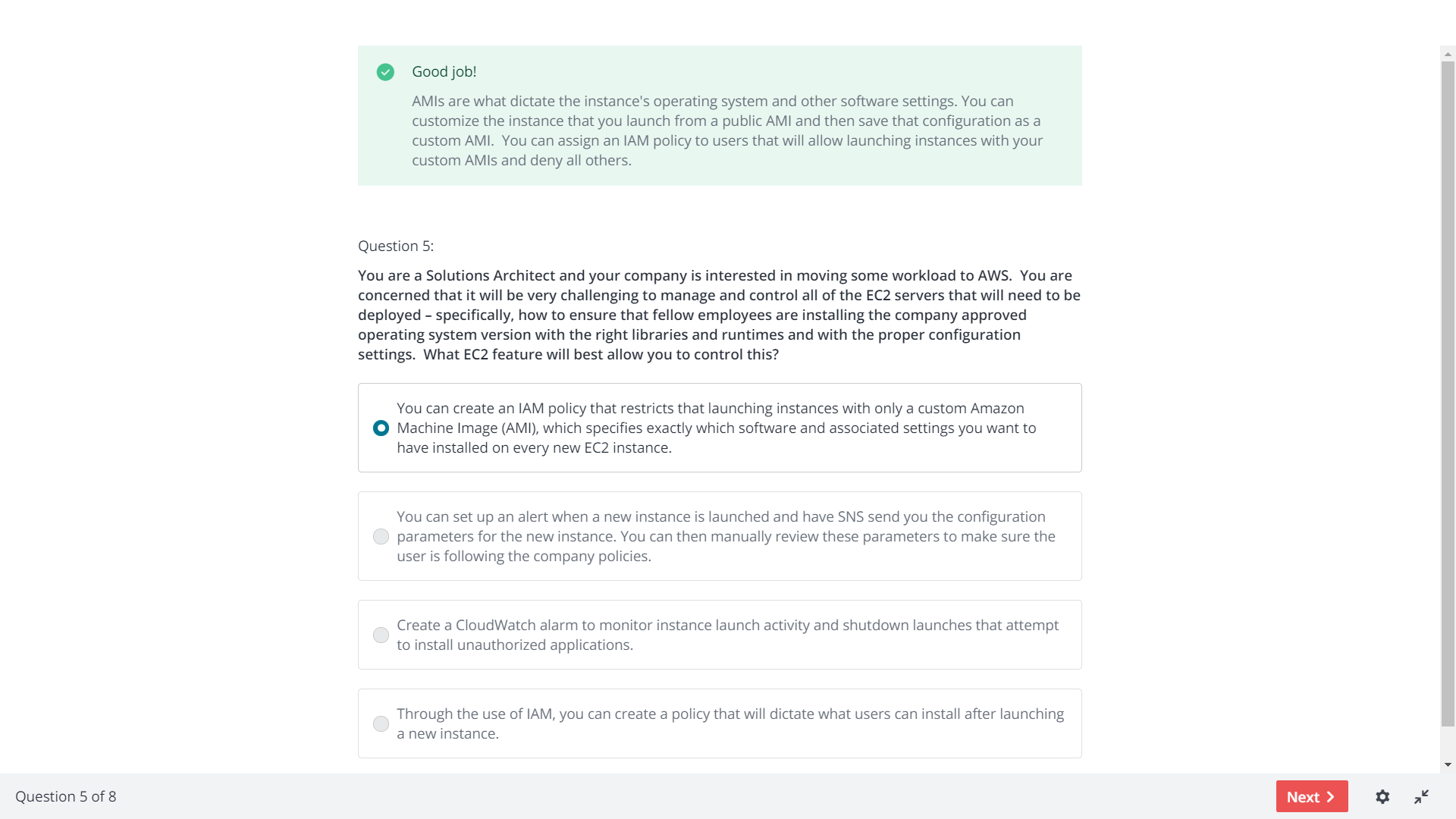
IOPS : run out of credits -> forced to baseline performance -> no termination

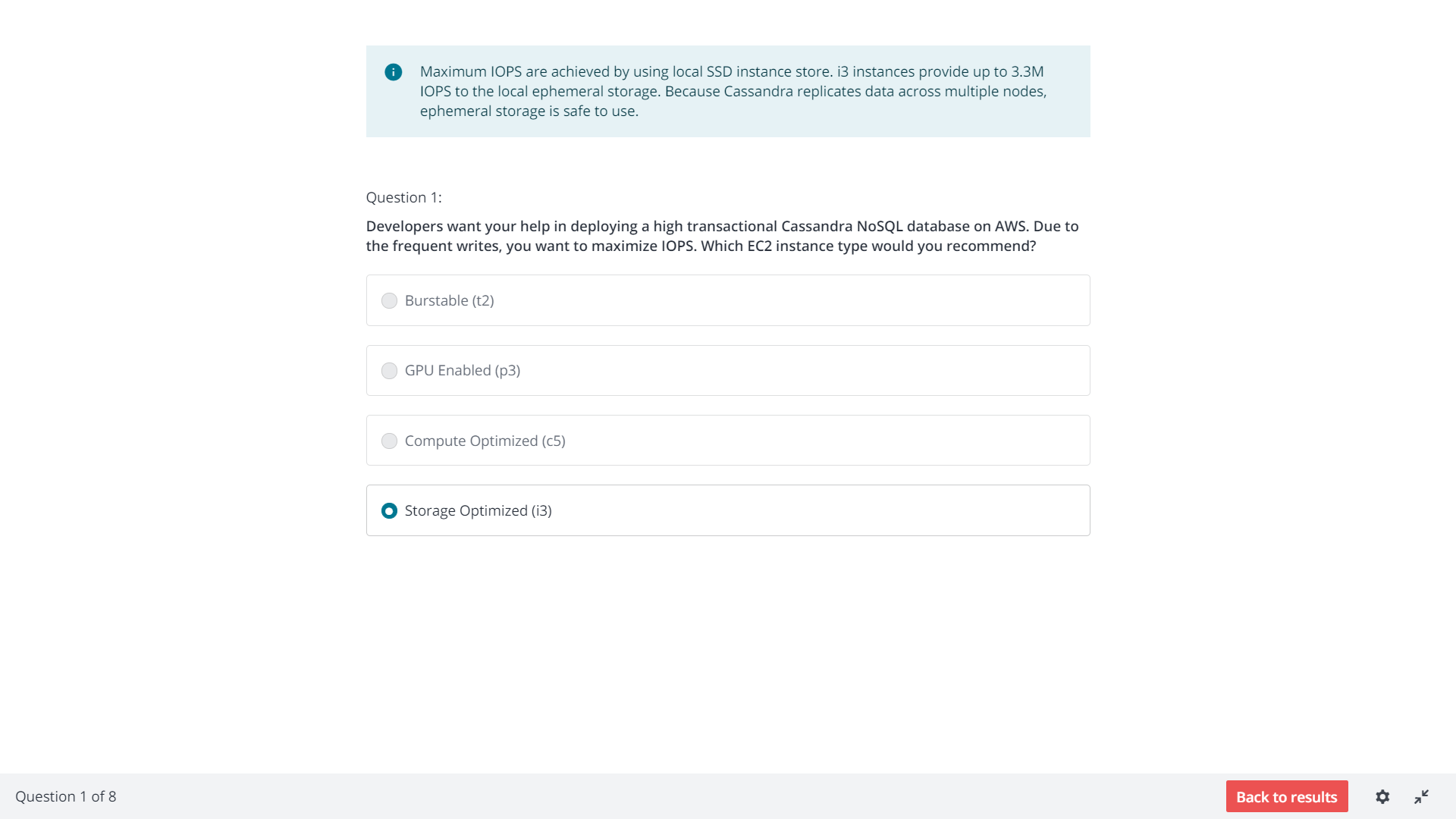
EBS Snapshot clarification:

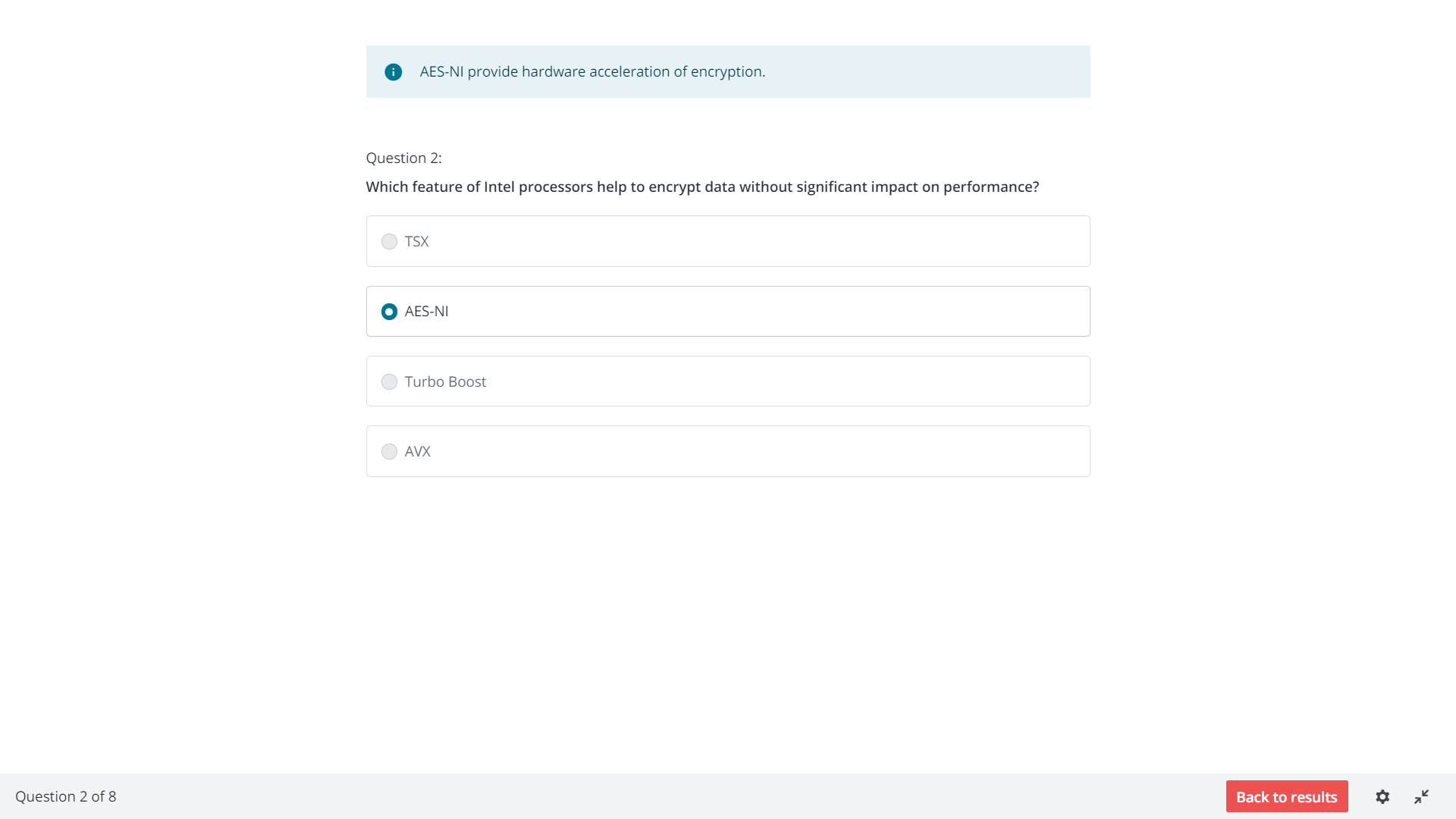


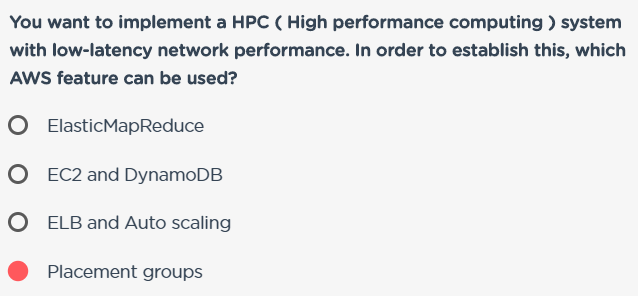
Connecting to EC2-instance via standard SSH without 3rd party apps like Putty ,Termius etc.

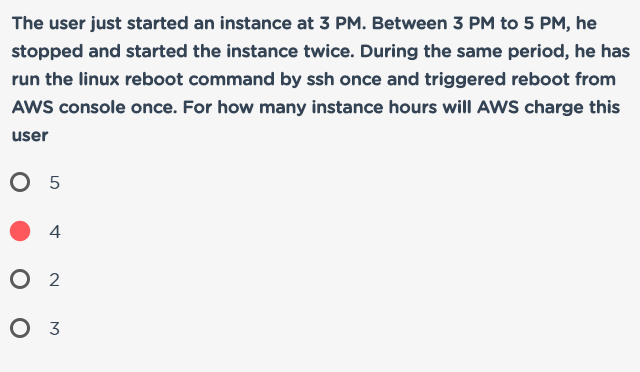




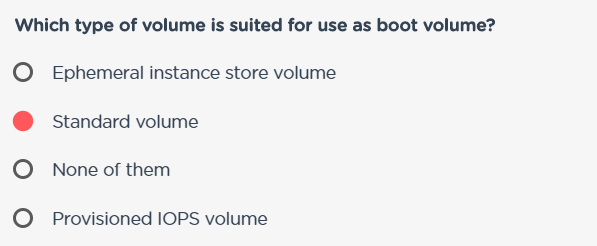


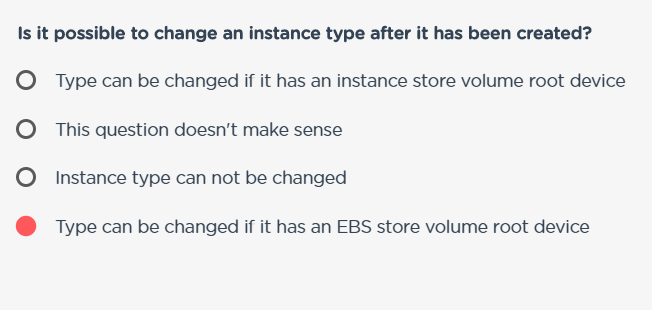


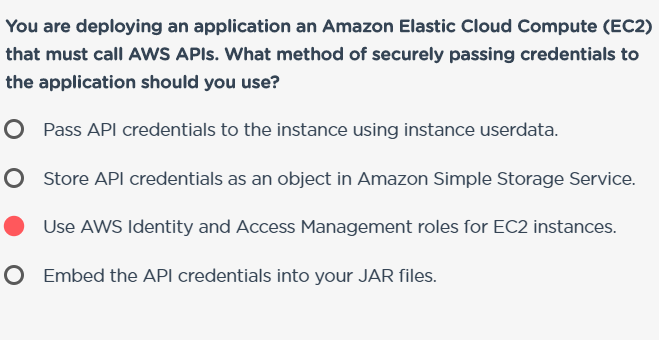


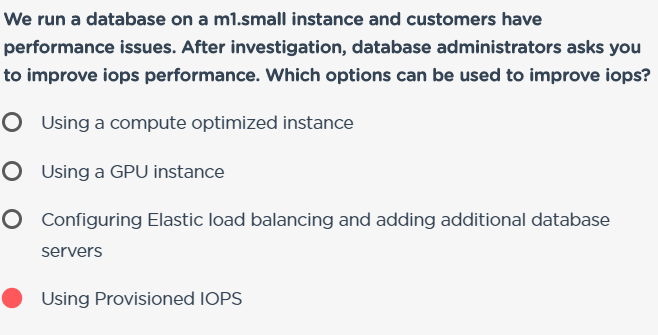


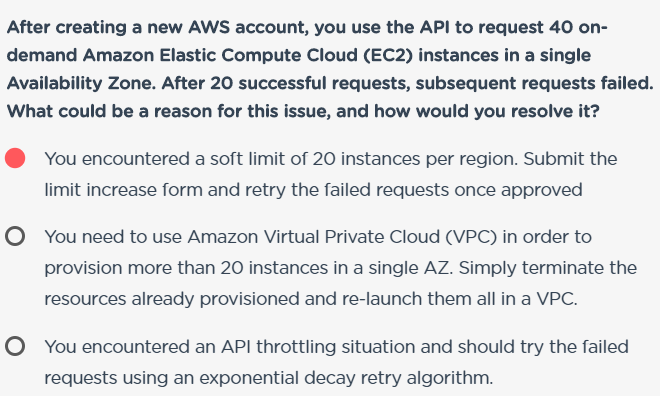


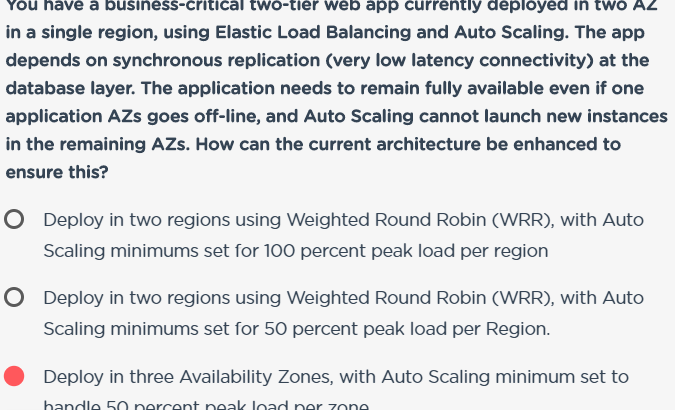


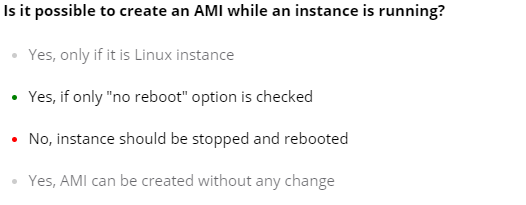






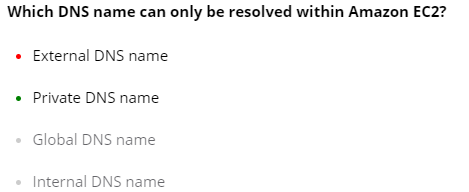




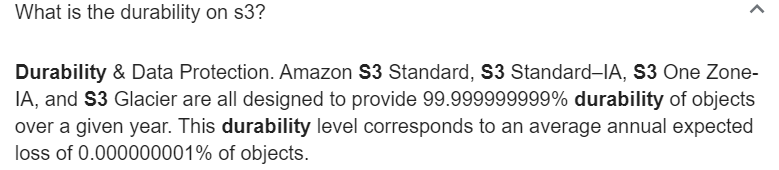


**Useful info for the above question :**

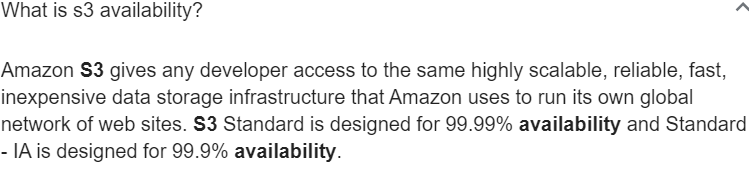
Amazon EC2 powers down the instance before creating the AMI to ensure that everything on the instance is stopped and in a consistent state during the creation process. If you're confident that your instance is in a consistent state appropriate for AMI creation, you can add the --no-reboot flag to ec2-create-image or CreateImage that tells Amazon EC2 not to power down and reboot the instance. With this flag, the instance remains running throughout the AMI creation process. Some file systems, such as xfs, can freeze and unfreeze activity, making it safe to create the image without rebooting the instance.  
  
Source: <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html>



S3 durability :



S3 availability:



**Old feature:**

