A stateful server is a server which maintains a state of data between simultaneous access. FTP / SMTP / Telnet servers are stateful servers, because it knows who you are once you have logged in and can track you.  HTTP server on the other hand is stateless unless used with an application layer which can utilize sessions to load a state for the user. For example a PHP backend can use sessions to give user a state or provide login facilities over a HTTP server.

Adding Lifecycle Hooks to the Auto Scaling group puts the instance into wait state before termination. During this wait state, you can perform custom activities to retrieve critical operational data from a stateful instance. Default Wait period is 1 hour. ● Option A is incorrect as the cooldown period will not help to copy data from the instance before termination. ● Option C is incorrect as Termination policy is used to specify which instances to terminate first during scale in, configuring termination policy for the Auto Scaling group will not copy data before instance termination. ● Option D is incorrect as Suspending Terminate policy will not prevent data loss but will disrupt other processes & prevent scale in. For more information on lifecycle-hooks, refer to the following URLs: ● https://docs.aws.amazon.com/autoscaling/ec2/userguide/lifecycle-hooks.html ● https://aws.amazon.com/ec2/autoscaling/faqs/

https://linuxacademy.com/blog/amazon-web-services-2/understand-lifecycle-hooks/ https://docs.aws.amazon.com/autoscaling/ec2/userguide/lifecycle-hooks.html

● https://aws.amazon.com/blogs/aws/new-ec2-auto-scaling-groups-with-multiple-instance-types-purch ● ase-options/ ● https://docs.aws.amazon.com/autoscaling/ec2/userguide/AutoScalingGroup.html#asg-purchase-optio ns

Auto Scaling group supports a mix of On-Demand & Spot instance which help to design a cost-optimized solution without any impact on the performance. You can choose the percentage of On-Demand & Spot instance based on the application requirements. With Option D, Auto Scaling group will have 2 instances initially as the On-Demand instances while remaining instances will be launched in a ratio of 20 % On-Demand instance & 80% Spot Instance. ● Option A is incorrect. With t2. micro, there would be a reduction in cost, but it will impact the performance of the application. ● Option B is incorrect as there would not be any cost reduction with all On-Demand instances. ● Option C is incorrect. Although this will reduce cost, all spot instance in an auto-scaling group may cause inconsistencies in the application & lead to poor performance. For more information on Auto Scaling with multiple Instance types & purchase options, refer to the following URLs: ● https://aws.amazon.com/blogs/aws/new-ec2-auto-scaling-groups-with-multiple-instance-types-purch ● ase-options/ ● https://docs.aws.amazon.com/autoscaling/ec2/userguide/AutoScalingGroup.html#asg-purchase-optio ns