

# IT-DUMPS Q&A

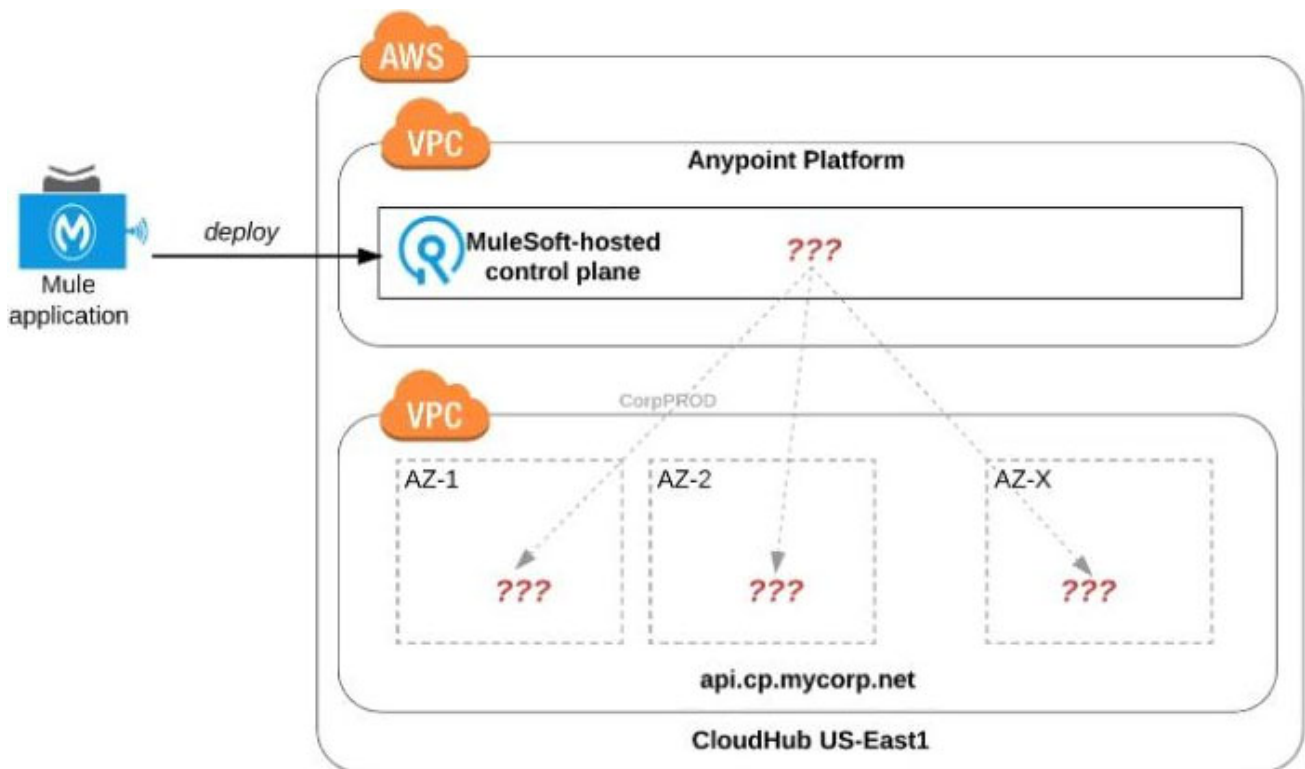
Accurate study guides, High passing rate!  
IT-dumps provides update free of charge in one year!

**Exam : MCPA-Level 1 Maintenance**

**Title : MuleSoft Certified Platform  
Architect - Level 1  
MAINTENANCE**

**Version : DEMO**

1.Refer to the exhibit.



An organization uses one specific CloudHub (AWS) region for all CloudHub deployments. How are CloudHub workers assigned to availability zones (AZs) when the organization's Mule applications are deployed to CloudHub in that region?

- A. Workers belonging to a given environment are assigned to the same AZ within that region
- B. AZs are selected as part of the Mule application's deployment configuration
- C. Workers are randomly distributed across available AZs within that region
- D. An AZ is randomly selected for a Mule application, and all the Mule application's CloudHub workers are assigned to that one AZ

**Answer: D**

#### Explanation

Correct Answer. Workers are randomly distributed across available AZs within that region.

\*\*\*\*\*

>> Currently, we only have control to choose which AWS Region to choose but there is no control at all using any configurations or deployment options to decide what Availability Zone (AZ) to assign to what worker.

>> There are NO fixed or implicit rules on platform too w.r.t assignment of AZ to workers based on environment or application.

>> They are completely assigned in random. However, cloudhub definitely ensures that HA is achieved by assigning the workers to more than on AZ so that all workers are not assigned to same AZ for same application.

: <https://help.mulesoft.com/s/question/0D52T000051rqDj/one-cloudhub-aws-region-how-cloudhub-workers-are-assigned-to-availability-zones-azs->

Graphical user interface, application

Description automatically generated

Bottom of Form

Top of Form

2.A set of tests must be performed prior to deploying API implementations to a staging environment. Due to data security and access restrictions, untested APIs cannot be granted access to the backend systems, so instead mocked data must be used for these tests. The amount of available mocked data and its contents is sufficient to entirely test the API implementations with no active connections to the backend systems.

What type of tests should be used to incorporate this mocked data?

- A. Integration tests
- B. Performance tests
- C. Functional tests (Blackbox)
- D. Unit tests (Whitebox)

**Answer: D**

**Explanation**

Correct Answer. Unit tests (Whitebox)

\*\*\*\*\*

Reference: <https://docs.mulesoft.com/mule-runtime/3.9/testing-strategies>

As per general IT testing practice and MuleSoft recommended practice, Integration and Performance tests should be done on full end to end setup for right evaluation.

Which means all end systems should be connected while doing the tests. So, these options are OUT and we are left with Unit Tests and Functional Tests.

As per attached reference documentation from MuleSoft:

Unit Tests - are limited to the code that can be realistically exercised without the need to run it inside Mule itself. So good candidates are Small pieces of modular code, Sub Flows, Custom transformers, Custom components, Custom expression evaluators etc.

Functional Tests - are those that most extensively exercise your application configuration. In these tests, you have the freedom and tools for simulating happy and unhappy paths. You also have the possibility to create stubs for target services and make them success or fail to easily simulate happy and unhappy paths respectively.

As the scenario in the question demands for API implementation to be tested before deployment to Staging and also clearly indicates that there is enough/ sufficient amount of mock data to test the various components of API implementations with no active connections to the backend systems, Unit Tests are the one to be used to incorporate this mocked data.

3.An organization makes a strategic decision to move towards an IT operating model that emphasizes consumption of reusable IT assets using modern APIs (as defined by MuleSoft).

What best describes each modern API in relation to this new IT operating model?

- A. Each modern API has its own software development lifecycle, which reduces the need for documentation and automation
- B. Each modern API must be treated like a product and designed for a particular target audience (for instance, mobile app developers)
- C. Each modern API must be easy to consume, so should avoid complex authentication mechanisms such as SAML or JWT D

D. Each modern API must be REST and HTTP based

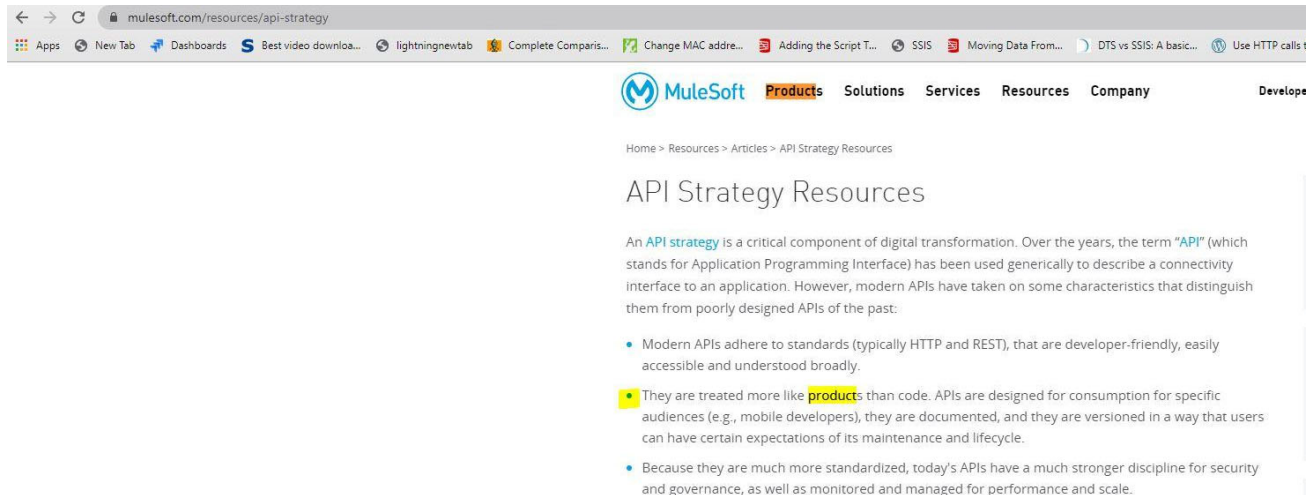
**Answer: B**

**Explanation**

Correct Answers:

1. Each modern API must be treated like a product and designed for a particular target audience (for instance mobile app developers)

\*\*\*\*\*



Bottom of Form

Top of Form

4.What CANNOT be effectively enforced using an API policy in Anypoint Platform?

- A. Guarding against Denial of Service attacks
- B. Maintaining tamper-proof credentials between APIs
- C. Logging HTTP requests and responses
- D. Backend system overloading

**Answer: A**

**Explanation**

Correct Answer. Guarding against Denial of Service attacks

\*\*\*\*\*

>> Backend system overloading can be handled by enforcing "Spike Control Policy"  
>> Logging HTTP requests and responses can be done by enforcing "Message Logging Policy"  
>> Credentials can be tamper-proofed using "Security" and "Compliance" Policies However, unfortunately, there is no proper way currently on Anypoint Platform to guard against DOS attacks.

Reference: <https://help.mulesoft.com/s/article/DDos-Dos-at>

5.What is the main change to the IT operating model that MuleSoft recommends to organizations to improve innovation and clock speed?

- A. Drive consumption as much as production of assets; this enables developers to discover and reuse assets from other projects and encourages standardization
- B. Expose assets using a Master Data Management (MDM) system; this standardizes projects and enables developers to quickly discover and reuse assets from other projects

C. Implement SOA for reusable APIs to focus on production over consumption; this standardizes on XML and WSDL formats to speed up decision making

D. Create a lean and agile organization that makes many small decisions everyday; this speeds up decision making and enables each line of business to take ownership of its projects

**Answer: A**

**Explanation**

Correct Answer. Drive consumption as much as production of assets; this enables developers to discover and reuse assets from other projects and encourages standardization

\*\*\*\*\*

>> The main motto of the new IT Operating Model that MuleSoft recommends and made popular is to change the way that they are delivered from a production model to a production + consumption model, which is done through an API strategy called API-led connectivity.

>> The assets built should also be discoverable and self-serveable for reusability across LOBs and organization.

>> MuleSoft's IT operating model does not talk about SDLC model (Agile/ Lean etc) or MDM at all. So, options suggesting these are not valid.

References:

<https://blogs.mulesoft.com/biz/connectivity/what-is-a-center-for-enablement-c4e/>

<https://www.mulesoft.com/resources/api/secret-to-managing-it-projects>