Number: MCPA-Level 1
Passing Score: 800
Time Limit: 120
File Version: 3.0

Exam Code: MCPA-Level-1
Exam Name: MuleSoft Certified Platform Architect - Level 1

Exam A

QUESTION 1
An organization is deploying their new implementation of the OrderStatus System API to multiple workers in CloudHub. This API fronts the organization's on-premises Order Management System, which is accessed by the API implementation over an IPsec tunnel.
What type of error typically does NOT result in a service outage of the OrderStatus System API?

A. A CloudHub worker fails with an out-of-memory exception
B. API Manager has an extended outage during the initial deployment of the API implementation
C. The AWS region goes offline with a major network failure to the relevant AWS data centers
D. The Order Management System is Inaccessible due to a network outage in the organization's onpremises data center

Correct Answer: A, C, L, O, U, D, H, U, B, W, O, R, K, E, R, F, A, I, L, S, W, I, T, H, A, N, O, U, T, O, F, M, E, M, O, R, Y, E, X, C, E, P, T, I, O, N
Section:
Explanation:
Answer: A CloudHub worker fails with an out-of-memory exception.
****
>> An AWS Region itself going down will definitely result in an outage as it does not matter how many workers are assigned to the Mule App as all of those in that region will go down. This is a complete downtime and outage.
>> Extended outage of API manager during initial deployment of API implementation will of course cause issues in proper application startup itself as the API Autodiscovery might fail or API policy templates and polices may not be downloaded to embed at the time of applicaiton startup etc... there are many reasons that could cause issues.
>> A network outage onpremises would of course cause the Order Management System not accessible and it does not matter how many workers are assigned to the app they all will fail and cause outage for sure.
The only option that does NOT result in a service outage is if a cloudhub worker fails with an out-ofmemory exception. Even if a worker fails and goes down, there are still other workers to handle the requests and keep the API UP and
Running. So, this is the right answer.

QUESTION 2
An Order API must be designed that contains significant amounts of integration logic and involves the invocation of the Product API.
The power relationship between Order API and Product API is one of "Customer/Supplier", because the Product API is used heavily throughout the organization and is developed by a dedicated development team located in the office of the CTO.
What strategy should be used to deal with the API data model of the Product API within the Order API?

A. Convince the development team of the Product API to adopt the API data model of the Order API such that the integration logic of the Order API can work with one consistent internal data model
B. Work with the API data types of the Product API directly when implementing the integration logic of the Order API such that the Order API uses the same (unchanged) data types as the Product API
C. Implement an anti-corruption layer in the Order API that transforms the Product API data model into internal data types of the Order API
D. Start an organization-wide data modeling initiative that will result in an Enterprise Data Model that will then be used in both the Product API and the Order API

Correct Answer: C, O, N, V, I, N, C, E, T, H, E, D, E, V, E, L, O, P, M, E, N, T, T, E, A, M, O, F, T, H, E, P, R, O, D, U, C, T, A, P, I, T, O, A, D, O, P, T, T, H, E, A, P, I, D, A, T, A, M, O, D, E, L, O, F, T, H, E, O, R, D, E, R, A, P, I, S, U, C, H, T, H, A, T, I, N, T, E, G, R, A, T, I, O, N, L, O, G, I, C, O, F, T, H, E, O, R, D, E, R, A, P, I, C, A, N, W, O, R, K, W, I, T, H, O, N, E, C, O, N, S, I, S, T, E, N, T, I, N, T, E, R, N, A, L, D, A, T, A, M, O, D, E, L
Section:
Explanation:
Answer: Convince the development team of the product API to adopt the API data model of the Order API such that integration logic of the Order API can work with one consistent internal data model
****
Key details to note from the given scenario:
>> Power relationship between Order API and Product API is customer/supplier So, as per below rules of "Power Relationships", the caller (in this case Order API) would request for features to the called (Product API team) and the Product
API team would need to accomodate those requests.

QUESTION 3
An API implementation is being designed that must invoke an Order API, which is known to repeatedly experience downtime.
For this reason, a fallback API is to be called when the Order API is unavailable.
What approach to designing the invocation of the fallback API provides the best resilience?

A. Search Anypoint Exchange for a suitable existing fallback API, and then implement invocations to this fallback API in addition to the Order API
B. Create a separate entry for the Order API in API Manager, and then invoke this API as a fallback API if the primary Order API is unavailable
C. Redirect client requests through an HTTP 307 Temporary Redirect status code to the fallback APIwhenever the Order API is unavailable
D. Set an option in the HTTP Requester component that invokes the Order API to instead invoke afallback API whenever an HTTP 4xx or 5xx response status code is returned from the Order API

Correct Answer: A, A, A, A, A, A, A, A, A, A, A, A, A, A, A, A, B, B, B, C, C, C, C, C, D, D, D, D, E, E, E, E, E, E, E, E, E, E, F, F, F, G, G, H, H, H, H, H, I, I, I, I, I, I, I, I, I, I, I, I, I, K, K, L, L, L, L, L, L, M, M, N, N, N, N, N, N, N, N, N, N, O, O, O, O, O, O, O, O, P, P, P, P, P, R, R, R, R, S, S, S, S, S, T, T, T, T, T, T, T, T, T, T, T, U, V, X, X, Y
Section:
Explanation:
Answer: Search Anypoint exchange for a suitable existing fallback API, and then implement invocations to this fallback API in addition to the order API **********
>> It is not ideal and good approach, until unless there is a pre-approved agreement with the API clients that they will receive a HTTP 3xx temporary redirect status code and they have to implement fallback logic their side to call another
API.
>> Creating separate entry of same Order API in API manager would just create an another instance of it on top of same API implementation. So, it does NO GOOD by using clone od same API as a fallback API. Fallback API should be ideally a different API implementation that is not same as primary one.
>> There is NO option currently provided by Anypoint HTTP Connector that allows us to invoke afallback API when we receive certain HTTP status codes in response.
The only statement TRUE in the given options is to Search Anypoint exchange for a suitable existing fallback API, and then implement invocations to this fallback API in addition to the order API.

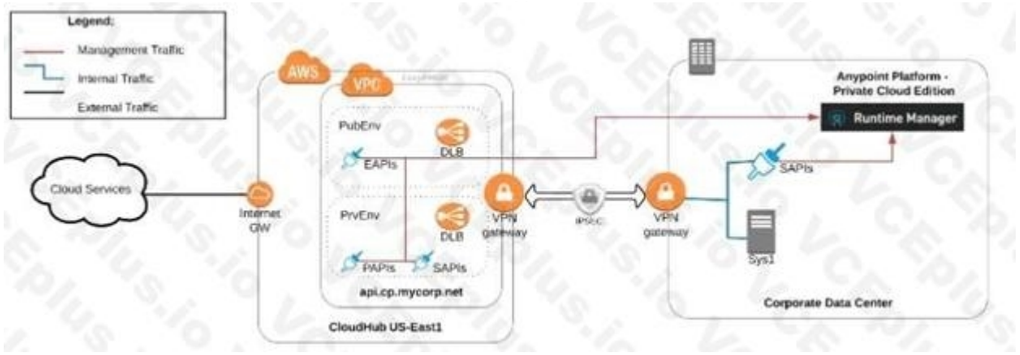
QUESTION 4
How are an API implementation, API client, and API consumer combined to invoke and process an API?

A. The API consumer creates an API implementation, which receives API invocations from an API such that they are processed for an API client
B. The API client creates an API consumer, which receives API invocations from an API such that they are processed for an API implementation
C. The ApI consumer creates an API client, which sends API invocations to an API such that they are processed by an API implementation
D. The ApI client creates an API consumer, which sends API invocations to an API such that they are processed by an API implementation

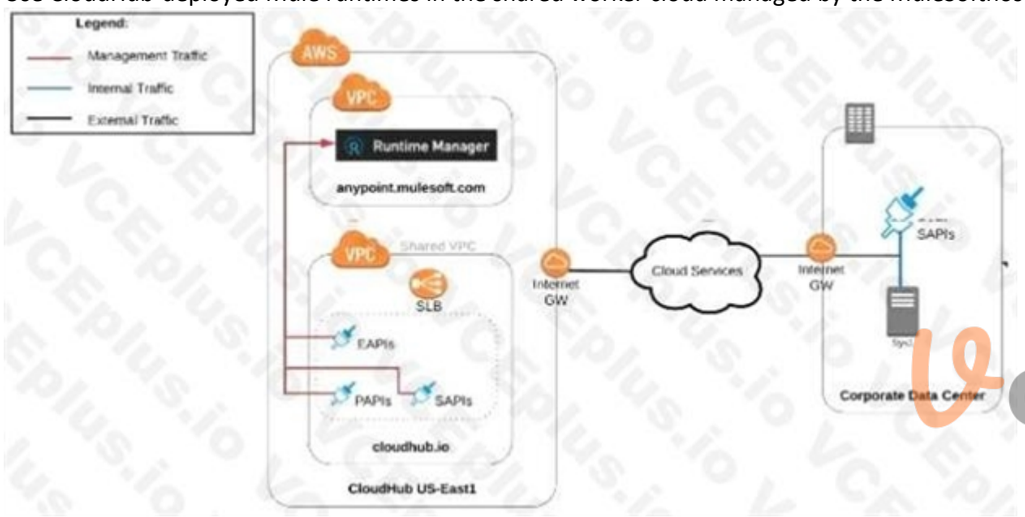
Correct Answer: T, H, E, A, P, I, C, O, N, S, U, M, E, R, C, R, E, A, T, E, S, A, N, A, P, I, C, L, I, E, N, T, W, H, I, C, H, S, E, N, D, S, A, P, I, I, N, V, O, C, A, T, I, O, N, S, T, O, A, N, A, P, I, S, U, C, H, T, H, A, T, T, H, E, Y, A, R, E, P, R, O, C, E, S, S, E, D, B, Y, A, N, A, P, I, I, M, P, L, E, M, E, N, T, A, T, I, O, N
Section:
Explanation:
Answer: The API consumer creates an API client, which sends API invocations to an API such that they are processed by an API implementation
****
Terminology:
>> API Client - It is a piece of code or program the is written to invoke an API
>> API Consumer - An owner/entity who owns the API Client. API Consumers write API clients.
>> API - The provider of the API functionality. Typically an API Instance on API Manager where they are managed and operated.
>> API Implementation - The actual piece of code written by API provider where the functionality of the API is implemented. Typically, these are Mule Applications running on Runtime Manager.

QUESTION 5
An organization uses various cloud-based SaaS systems and multiple on-premises systems. The onpremises systems are an important part of the organization's application network and can only be accessed from within the organization's intranet.
What is the best way to configure and use Anypoint Platform to support integrations with both the cloud-based SaaS systems and on-premises systems?

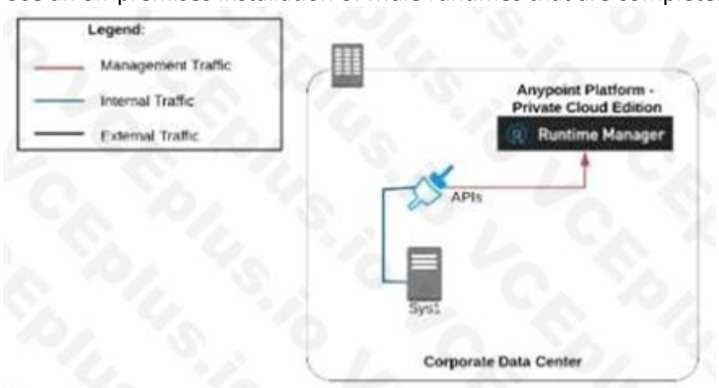
A. Use CloudHub-deployed Mule runtimes in an Anypoint VPC managed by Anypoint Platform Private Cloud Edition control plane



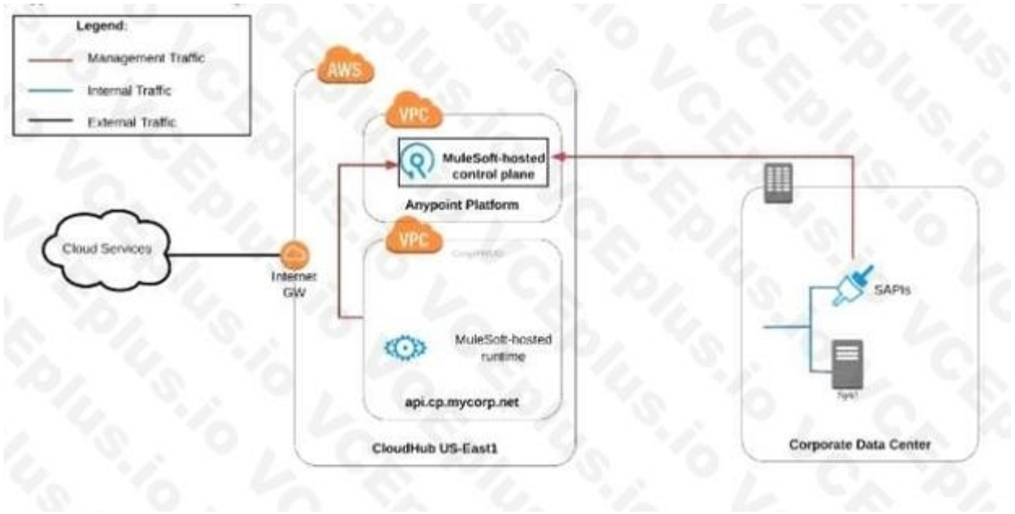
B. Use CloudHub-deployed Mule runtimes in the shared worker cloud managed by the MuleSofthosted Anypoint Platform control plane



C. Use an on-premises installation of Mule runtimes that are completely isolated with NO external network access, managed by the Anypoint Platform Private Cloud Edition control plane



D. Use a combination of Cloud Hub-deployed and manually provisioned on-premises Mule runtimes managed by the MuleSoft-hosted Anypoint Platform control plane



Correct Answer: U, S, E, A, C, O, M, B, I, N, A, T, I, O, N, O, F, C, L, O, U, D, H, U, B, D, E, P, L, O, Y, E, D, A, N, D, M, A, N, U, A, L, L, Y, P, R, O, V, I, S, I, O, N, E, D, O, N, P, R, E, M, I, S, E, S, M, U, L, E, R, U, N, T, I, M, E, S, M, A, N, A, G, E, D, B, Y, T, H, E, M, U, L, E, S, O, F, T, H, O, S, T, E, D, P, L, A, T, F, O, R, M, C, O, N, T, R, O, L, P, L, A, N, E
Section:
Explanation:
Answer: Use a combination of CloudHub-deployed and manually provisioned on-premises Mule runtimes managed by the MuleSoft-hosted Platform control plane.
****
Key details to be taken from the given scenario:
>> Organization uses BOTH cloud-based and on-premises systems
>> On-premises systems can only be accessed from within the organization's intranet Let us evaluate the given choices based on above key details:
>> CloudHub-deployed Mule runtimes can ONLY be controlled using MuleSoft-hosted control plane.
We CANNOT use Private Cloud Edition's control plane to control CloudHub Mule Runtimes. So, option suggesting this is INVALID
>> Using CloudHub-deployed Mule runtimes in the shared worker cloud managed by the MuleSofthosted Anypoint Platform is completely IRRELEVANT to given scenario and silly choice. So, option suggesting this is INVALID
>> Using an on-premises installation of Mule runtimes that are completely isolated with NO external network access, managed by the Anypoint Platform Private Cloud Edition control plane would work for On-premises integrations. However, with NO external access, integrations cannot be done to SaaS-based apps. Moreover CloudHub-hosted apps are best-fit for integrating with SaaS-based applications. So, option suggesting this is BEST WAY.
The best way to configure and use Anypoint Platform to support these mixed/hybrid integrations is to use a combination of CloudHub-deployed and manually provisioned on-premises Mule runtimes managed by the MuleSoft-hosted
Platform control plane.

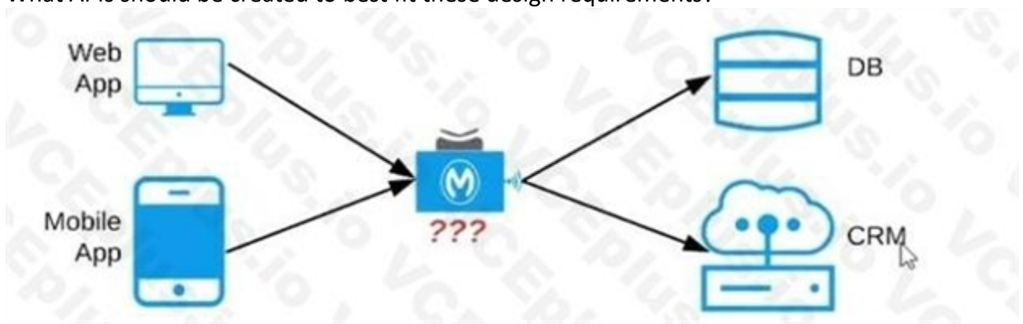
QUESTION 6
When must an API implementation be deployed to an Anypoint VPC?

A. When the API Implementation must invoke publicly exposed services that are deployed outside of CloudHub in a customer- managed AWS instance
B. When the API implementation must be accessible within a subnet of a restricted customer-hosted network that does not allow public access
C. When the API implementation must be deployed to a production AWS VPC using the Mule Maven plugin
D. When the API Implementation must write to a persistent Object Store

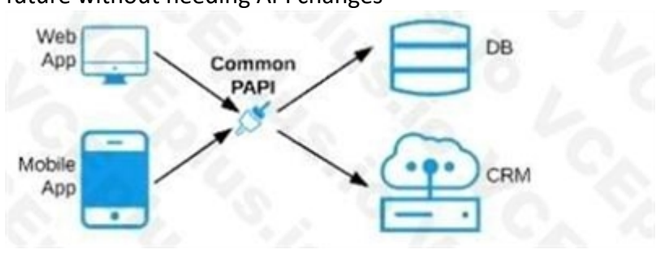
Correct Answer: A
Section:

QUESTION 7
Refer to the exhibit. An organization needs to enable access to their customer data from both a mobile app and a web application, which each need access to common fields as well as certain unique fields.

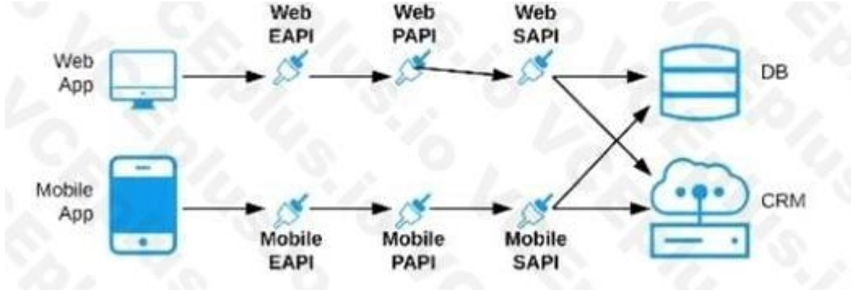
The data is available partially in a database and partially in a 3rd-party CRM system.
What APIs should be created to best fit these design requirements?



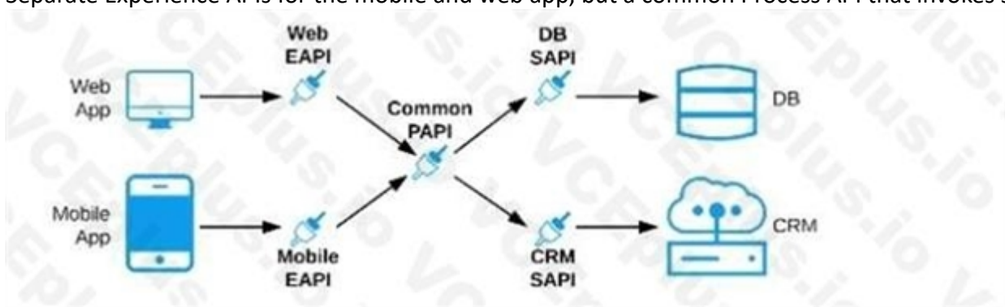
A. A Process API that contains the data required by both the web and mobile apps, allowing these applications to invoke it directly and access the data they need thereby providing the flexibility to add more fields in the future without needing API changes



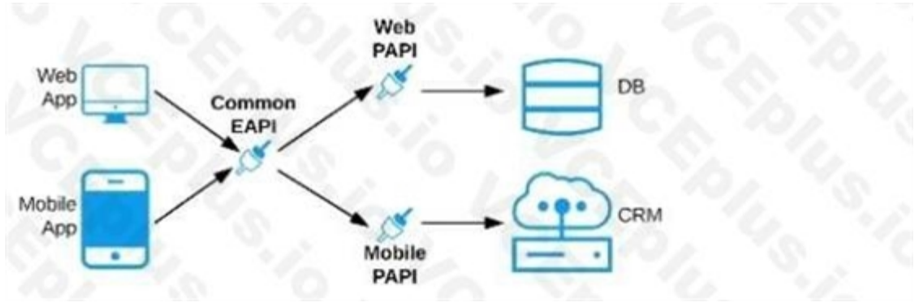
B. One set of APIs (Experience API, Process API, and System API) for the web app, and another set for the mobile app



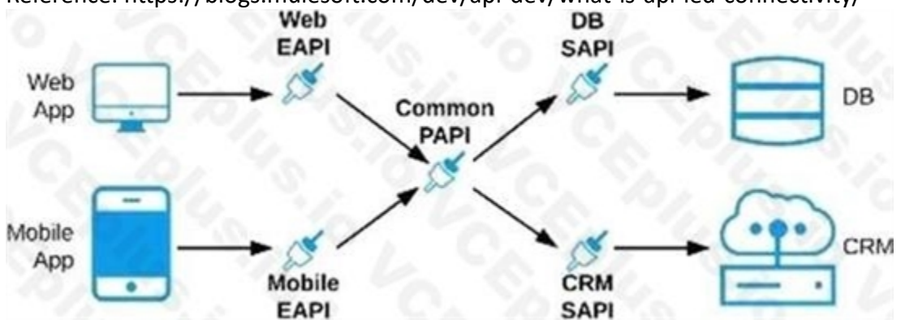
C. Separate Experience APIs for the mobile and web app, but a common Process API that invokes separate System APIs created for the database and CRM system



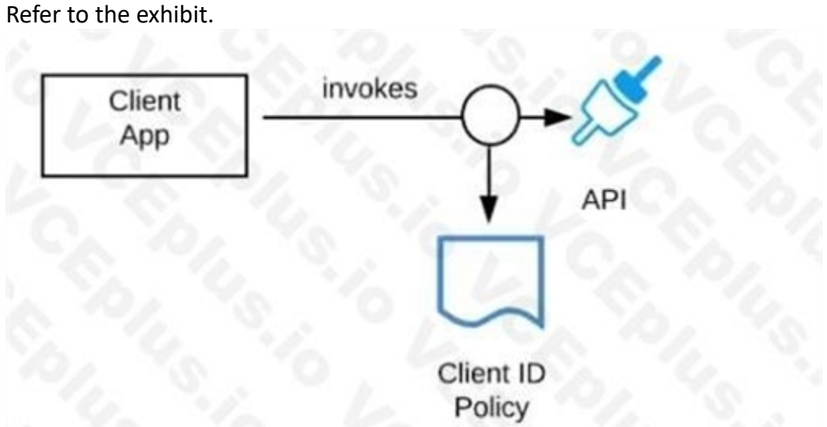
D. A common Experience API used by both the web and mobile apps, but separate Process APIs for the web and mobile apps that interact with the database and the CRM System



Correct Answer: A, A, A, A, A, A, A, A, A, A, A, A, A, A, A, A, A, A, B, B, B, B, C, C, C, C, C, C, C, D, D, D, D, D, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, F, F, F, H, H, H, I, I, I, I, I, I, I, I, I, K, L, L, L, M, M, M, M, M, M, M, N, N, N, N, N, N, N, O, O, O, O, O, O, O, O, O, O, P, P, P, P, P, P, P, P, P, P, P, R, R, R, R, R, R, R, R, R, S, S, S, S, S, S, S, S, S, S, S, S, S, S, S, T, T, T, T, T, T, T, T, T, T, T, T, T, T, U, U, V, V, W, X, Y, Y, Y
Section:
Explanation:
Answer: Separate Experience APIs for the mobile and web app, but a common Process API that invokes separate System APIs created for the database and CRM system ********** As per MuleSoft's API-led connectivity:
>> Experience APIs should be built as per each consumer needs and their experience.
>> Process APIs should contain all the orchestration logic to achieve the business functionality.
>> System APIs should be built for each backend system to unlock their data.
Reference: https://blogs.mulesoft.com/dev/api-dev/what-is-api-led-connectivity/



QUESTION 8
Refer to the exhibit.



A developer is building a client application to invoke an API deployed to the STAGING environment that is governed by a client ID enforcement policy.
What is required to successfully invoke the API?

A. The client ID and secret for the Anypoint Platform account owning the API in the STAGING environment

B. The client ID and secret for the Anypoint Platform account's STAGING environment
C. The client ID and secret obtained from Anypoint Exchange for the API instance in the STAGING environment
D. A valid OAuth token obtained from Anypoint Platform and its associated client ID and secret

Correct Answer: T, H, E, C, L, I, E, N, T, I, D, A, N, D, S, E, C, R, E, T, O, B, T, A, I, N, E, D, F, R, O, M, A, N, Y, P, O, I, N, T, E, X, C, H, A, N, G, E, F, O, R, T, H, E, A, P, I, I, N, S, T, A, N, C, E, I, N, T, H, E, S, T, A, G, I, N, G, E, N, V, I, R, O, N, M, E, N, T
Section:
Explanation:
Answer: The client ID and secret obtained from Anypoint Exchange for the API instance in the STAGING environment **********
>> We CANNOT use the client ID and secret of Anypoint Platform account or any individual environments for accessing the APIs
>> As the type of policy that is enforced on the API in question is "Client ID Enforcment Policy", OAuth token based access won't work.
Right way to access the API is to use the client ID and secret obtained from Anypoint Exchange for the API instance in a particular environment we want to work on.
References:
Managing API instance Contracts on API Manager
https://docs.mulesoft.com/api-manager/1.x/request-access-to-api-task
https://docs.mulesoft.com/exchange/to-request-access
https://docs.mulesoft.com/api-manager/2.x/policy-mule3-client-id-based-policies

QUESTION 9
In an organization, the InfoSec team is investigating Anypoint Platform related data traffic.
From where does most of the data available to Anypoint Platform for monitoring and alerting originate?

A. From the Mule runtime or the API implementation, depending on the deployment model
B. From various components of Anypoint Platform, such as the Shared Load Balancer, VPC, and Mule runtimes
C. From the Mule runtime or the API Manager, depending on the type of data
D. From the Mule runtime irrespective of the deployment model

Correct Answer: F, R, O, M, T, H, E, M, U, L, E, R, U, N, T, I, M, E, I, R, R, E, S, P, E, C, T, I, V, E, O, F, T, H, E, D, E, P, L, O, Y, M, E, N, T, M, O, D, E, L
Section:
Explanation:
Answer: From the Mule runtime irrespective of the deployment model **********
>> Monitoring and Alerting metrics are always originated from Mule Runtimes irrespective of the deployment model.
>> It may seems that some metrics (Runtime Manager) are originated from Mule Runtime and some are (API Invocations/ API Analytics) from API Manager. However, this is realistically NOT TRUE. The reason is, API manager is just a management tool for API instances but all policies upon applying on APIs eventually gets executed on Mule Runtimes only (Either Embedded or API Proxy).
>> Similarly all API Implementations also run on Mule Runtimes.
So, most of the day required for monitoring and alerts are originated fron Mule Runtimes only irrespective of whether the deployment model is MuleSoft-hosted or Customer-hosted or Hybrid.

QUESTION 10
When designing an upstream API and its implementation, the development team has been advised to NOT set timeouts when invoking a downstream API, because that downstream API has no SLA that can be relied upon. This is the only downstream API dependency of that upstream API.
Assume the downstream API runs uninterrupted without crashing. What is the impact of this advice?

A. An SLA for the upstream API CANNOT be provided
B. The invocation of the downstream API will run to completion without timing out
C. A default timeout of 500 ms will automatically be applied by the Mule runtime in which the upstream API implementation executes
D. A toad-dependent timeout of less than 1000 ms will be applied by the Mule runtime in which the downstream API implementation executes

Correct Answer: A, N, S, L, A, F, O, R, T, H, E, U, P, S, T, R, E, A, M, A, P, I, C, A, N, N, O, T, B, E, P, R, O, V, I, D, E, D

Section:
Explanation:
Answer: An SLA for the upstream API CANNOT be provided.
****
>> First thing first, the default HTTP response timeout for HTTP connector is 10000 ms (10 seconds).
NOT 500 ms.
>> Mule runtime does NOT apply any such "load-dependent" timeouts. There is no such behavior currently in Mule.
>> As there is default 10000 ms time out for HTTP connector, we CANNOT always guarantee that theinvocation of the downstream API will run to completion without timing out due to its unreliable SLAtimes. If the response time crosses 10 seconds then the request may time out.
The main impact due to this is that a proper SLA for the upstream API CANNOT be provided.
Reference: https://docs.mulesoft.com/http-connector/1.5/http-documentation#parameters-3

QUESTION 11
What best explains the use of auto-discovery in API implementations?

A. It makes API Manager aware of API implementations and hence enables it to enforce policies
B. It enables Anypoint Studio to discover API definitions configured in Anypoint Platform
C. It enables Anypoint Exchange to discover assets and makes them available for reuse
D. It enables Anypoint Analytics to gain insight into the usage of APIs

Correct Answer: I, T, M, A, K, E, S, A, P, I, M, A, N, A, G, E, R, A, W, A, R, E, O, F, A, P, I, I, M, P, L, E, M, E, N, T, A, T, I, O, N, S, A, N, D, H, E, N, C, E, E, N, A, B, L, E, S, I, T, T, O, E, N, F, O, R, C, E, P, O, L, I, C, I, E, S
Section:
Explanation:
Answer: It makes API Manager aware of API implementations and hence enables it to enforce policies.
****
>> API Autodiscovery is a mechanism that manages an API from API Manager by pairing the deployed application to an API created on the platform.
>> API Management includes tracking, enforcing policies if you apply any, and reporting API analytics.
>> Critical to the Autodiscovery process is identifying the API by providing the API name and version.
References:
https://docs.mulesoft.com/api-manager/2.x/api-auto-discovery-new-concept
https://docs.mulesoft.com/api-manager/1.x/api-auto-discovery
https://docs.mulesoft.com/api-manager/2.x/api-auto-discovery-new-concept

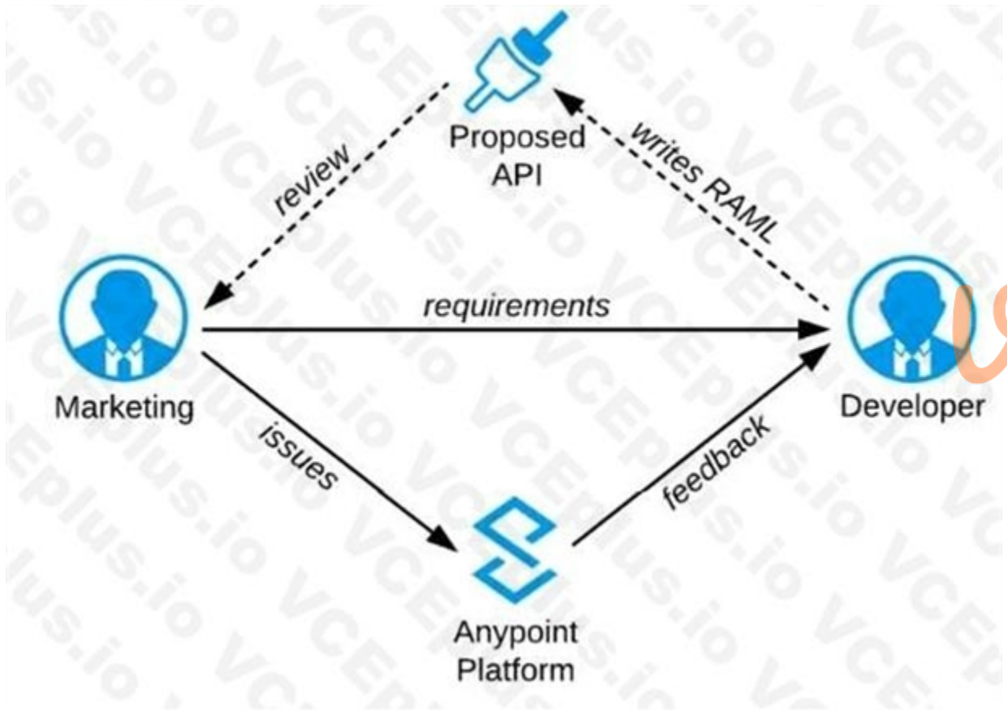
QUESTION 12
What should be ensured before sharing an API through a public Anypoint Exchange portal?

A. The visibility level of the API instances of that API that need to be publicly accessible should be set to public visibility
B. The users needing access to the API should be added to the appropriate role in Anypoint Platform
C. The API should be functional with at least an initial implementation deployed and accessible for users to interact with
D. The API should be secured using one of the supported authentication/authorization mechanisms to ensure that data is not compromised

Correct Answer: T, H, E, V, I, S, I, B, I, L, I, T, Y, L, E, V, E, L, O, F, T, H, E, A, P, I, I, N, S, T, A, N, C, E, S, O, F, T, H, A, T, A, P, I, T, H, A, T, N, E, E, D, T, O, B, E, P, U, B, L, I, C, L, Y, A, C, C, E, S, S, I, B, L, E, S, H, O, U, L, D, B, E, S, E, T, T, O, P, U, B, L, I, C, V, I, S, I, B, I, L, I, T, Y
Section:
Explanation:

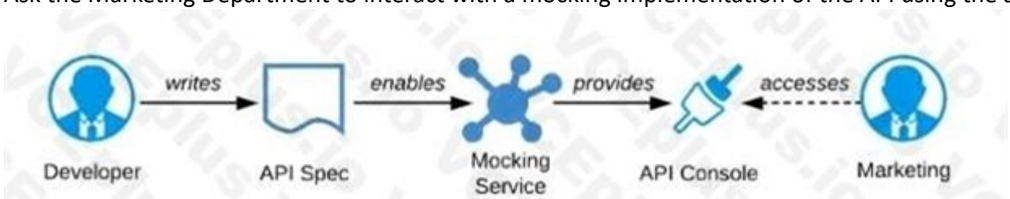
Answer: The visibility level of the API instances of that API that need to be publicly accessible should be set to public visibility.
****
Reference: https://docs.mulesoft.com/exchange/to-share-api-asset-to-portal
https://docs.mulesoft.com/exchange/to-share-api-asset-to-portal

QUESTION 13
Refer to the exhibit.



A RAML definition has been proposed for a new Promotions Process API, and has been published to Anypoint Exchange.
The Marketing Department, who will be an important consumer of the Promotions API, has important requirements and expectations that must be met.
What is the most effective way to use Anypoint Platform features to involve the Marketing Department in this early API design phase?

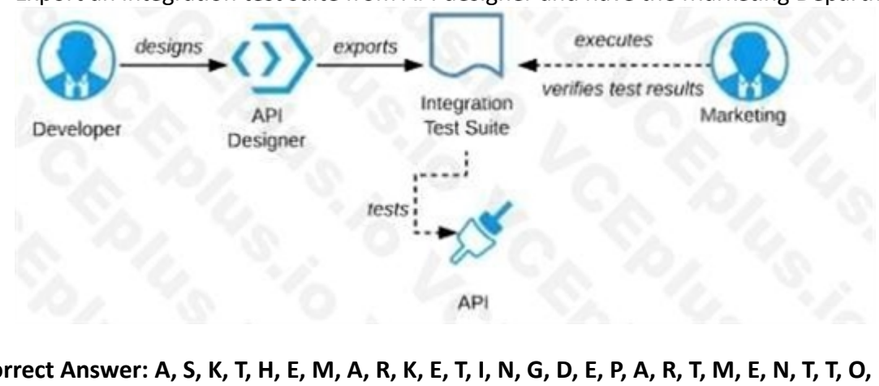
A. Ask the Marketing Department to interact with a mocking implementation of the API using the automatically generated API Console



B. Organize a design workshop with the DBAs of the Marketing Department in which the database schema of the Marketing IT systems is translated into RAML

C. Use Anypoint Studio to Implement the API as a Mule application, then deploy that API implementation to CloudHub and ask the Marketing Department to interact with it

D. Export an integration test suite from API designer and have the Marketing Department execute the tests In that suite to ensure they pass



Correct Answer: A, S, K, T, H, E, M, A, R, K, E, T, I, N, G, D, E, P, A, R, T, M, E, N, T, T, O, I, N, T, E, R, A, C, T, W, I, T, H, A, M, O, C, K, I, N, G, I, M, P, L, E, M, E, N, T, A, T, I, O, N, O, F, T, H, E, A, P, I, U, S, I, N, G, T, H, E, A, U, T, O, M, A, T, I, C, A, L, L, Y, G, E, N, E, R, A, T, E, D, A, P, I, C, O, N, S, O, L, E

Section:

Explanation:

Answer: Ask the Marketing Department to interact with a mocking implementation of the API using the automatically generated API Console.

****

As per MuleSoft's IT Operating Model:

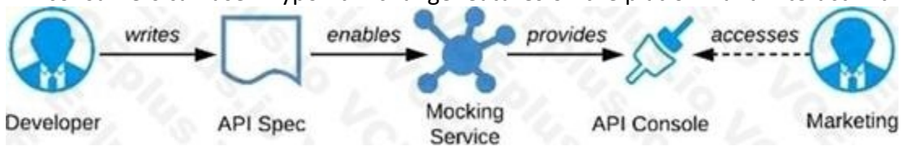
>> API consumers need NOT wait until the full API implementation is ready.

>> NO technical test-suites needs to be shared with end users to interact with APIs.

>> Anypoint Platform offers a mocking capability on all the published API specifications to Anypoint Exchange which also will be rich in documentation covering all details of API functionalities and working nature.

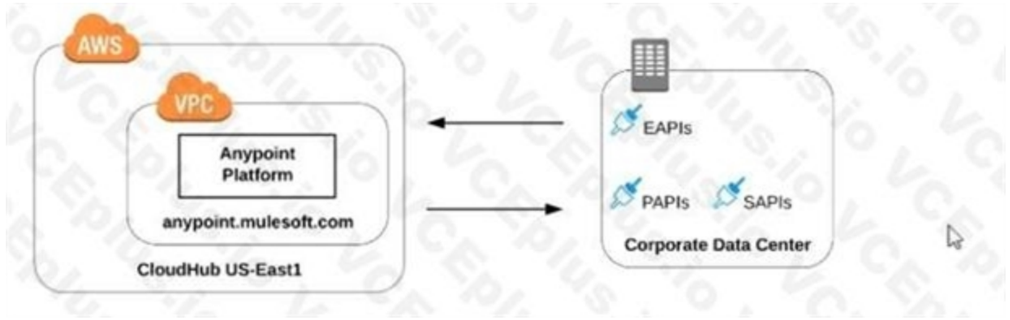
>> No needs of arranging days of workshops with end users for feedback.

API consumers can use Anypoint Exchange features on the platform and interact with the API using its mocking feature. The feedback can be shared quickly on the same to incorporate any changes.



QUESTION 14

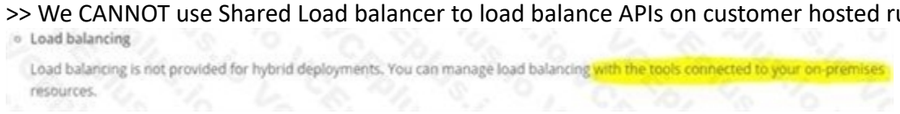
Refer to the exhibit.



what is true when using customer-hosted Mule runtimes with the MuleSoft-hosted Anypoint Platform control plane (hybrid deployment)?

A. Anypoint Runtime Manager initiates a network connection to a Mule runtime in order to deploy Mule applications
B. The MuleSoft-hosted Shared Load Balancer can be used to load balance API invocations to the Mule runtimes
C. API implementations can run successfully in customer-hosted Mule runtimes, even when they are unable to communicate with the control plane
D. Anypoint Runtime Manager automatically ensures HA in the control plane by creating a new Mule runtime instance in case of a node failure

Correct Answer: A, P, I, I, M, P, L, E, M, E, N, T, A, T, I, O, N, S, C, A, N, R, U, N, S, U, C, C, E, S, S, F, U, L, L, Y, I, N, C, U, S, T, O, M, E, R, H, O, S, T, E, D, M, U, L, E, R, U, N, T, I, M, E, S, E, V, E, N, W, H, E, N, T, H, E, Y, A, R, E, U, N, A, B, L, E, T, O, C, O, M, M, U, N, I, C, A, T, E, W, I, T, H, T, H, E, C, O, N, T, R, O, L, P, L, A, N, E
Section:
Explanation:
Answer: API implementations can run successfully in customer-hosted Mule runtimes, even when they are unable to communicate with the control plane.
****
>> We CANNOT use Shared Load balancer to load balance APIs on customer hosted runtimes



>> For Hybrid deployment models, the on-premises are first connected to Runtime Manager using Runtime Manager agent. So, the connection is initiated first from On-premises to Runtime Manager. Then all control can be done from Runtime Manager.
>> Anypoint Runtime Manager CANNOT ensure automatic HA. Clusters/Server Groups etc should be configured before hand.
Only TRUE statement in the given choices is, API implementations can run successfully in customerhosted Mule runtimes, even when they are unable to communicate with the control plane. There are several references below to justify this statement.
References:
https://docs.mulesoft.com/runtime-manager/deployment-strategies#hybrid-deployments
https://help.mulesoft.com/s/article/On-Premise-Runtimes-Disconnected-From-US-Control-Plane-June-18th-2018
https://help.mulesoft.com/s/article/Runtime-Manager-cannot-manage-On-Prem-Applications-and-Servers-from-US-Control-Plane-June-25th-2019
https://help.mulesoft.com/s/article/On-premise-Runtimes-Appear-Disconnected-in-Runtime-Manager-May-29th-2018

On-Premise Runtimes Disconnected From US Control Plane - June 18th 2018

Jun 19, 2018 · RCA

Content

Impacted Platforms	Impacted Duration
Anypoint Runtime Manager / On-Prem Runtimes	During this time frame, on-prem runtimes appeared disconnected from the US Anypoint Control Plane: June 18, 2018 10:35 AM PST to June 18, 2018 11:12 AM PST

Incident Description

On-premises applications weren't able to connect to Anypoint Runtime Manager during the length of the incident, which made on-premises runtimes to threw errors in their logs because they received network disconnect messages from the control plane. Other than generating the log as mentioned above entries, on-premises runtimes and applications were not impacted.

==========

Runtime Manager cannot manage On-Prem Applications and Servers from US Control Plane - June 25th 2019

Jul 3, 2019 · RCA

Content

Incident Summary

Between 2:51 p.m. PT June 25th and 12:41 a.m. PT June 26th, customers were not able to manage their On-Prem applications and servers. The availability of running applications and runtimes were not impacted.

Impacted Platforms	Impact Duration
US-Prod	9 hours and 50 minutes

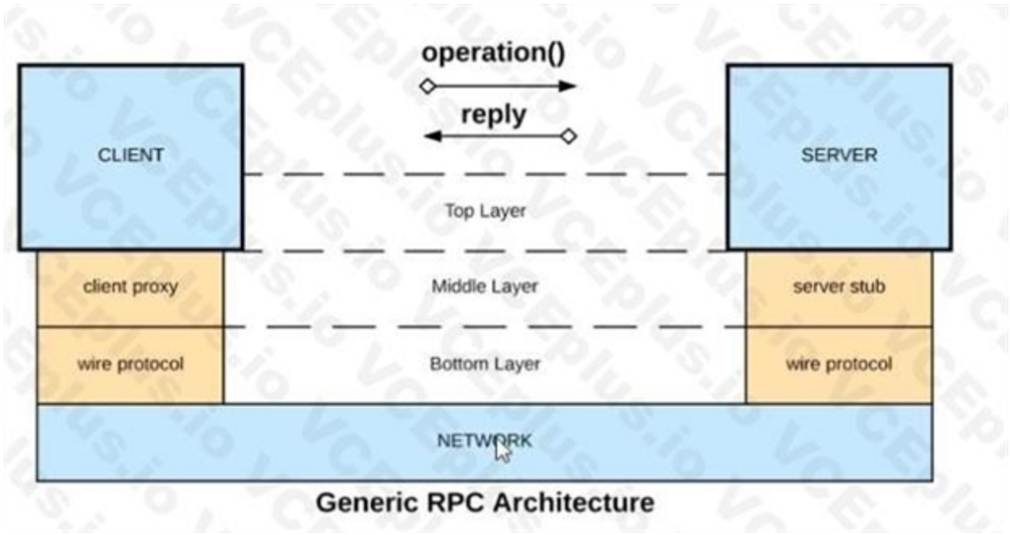
==========

QUESTION 15
A System API is designed to retrieve data from a backend system that has scalability challenges.
What API policy can best safeguard the backend system?

A. IPwhitelist
B. SLA-based rate limiting
C. Auth 2 token enforcement
D. Client ID enforcement

Correct Answer: S, L, A, B, A, S, E, D, R, A, T, E, L, I, M, I, T, I, N, G
Section:
Explanation:
Answer: SLA-based rate limiting
****
>> Client Id enforement policy is a "Compliance" related NFR and does not help in maintaining the "Quality of Service (QoS)". It CANNOT and NOT meant for protecting the backend systems from scalability challenges.
>> IP Whitelisting and OAuth 2.0 token enforcement are "Security" related NFRs and again does not help in maintaining the "Quality of Service (QoS)". They CANNOT and are NOT meant for protecting the backend systems from scalability challenges.
Rate Limiting, Rate Limiting-SLA, Throttling, Spike Control are the policies that are "Quality of Service (QOS)" related NFRs and are meant to help in protecting the backend systems from getting overloaded.
https://dzone.com/articles/how-to-secure-apis

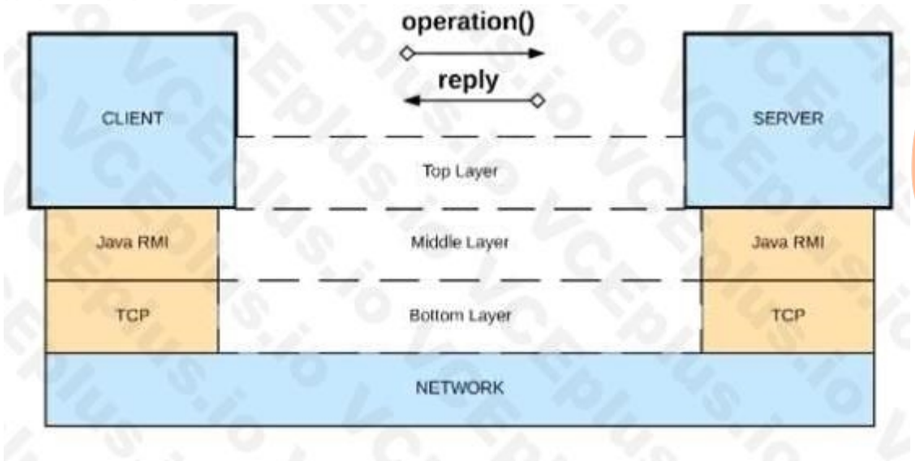
QUESTION 16
Refer to the exhibit.



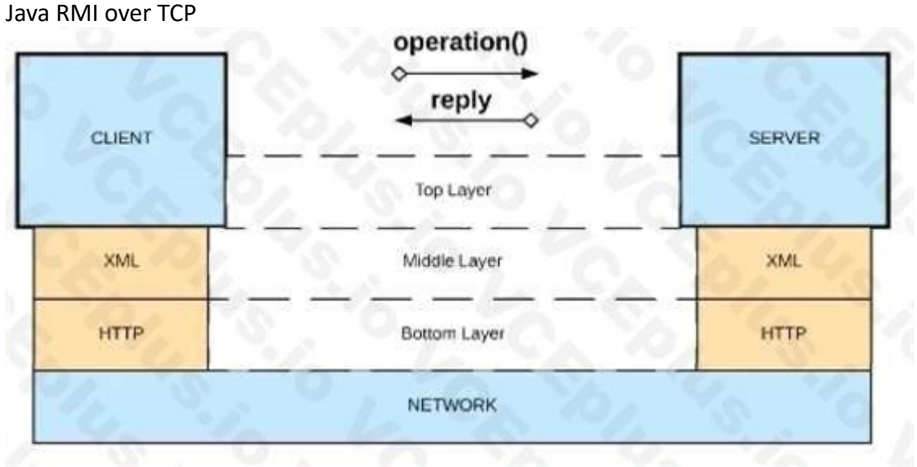
Generic RPC Architecture

What is a valid API in the sense of API-led connectivity and application networks?

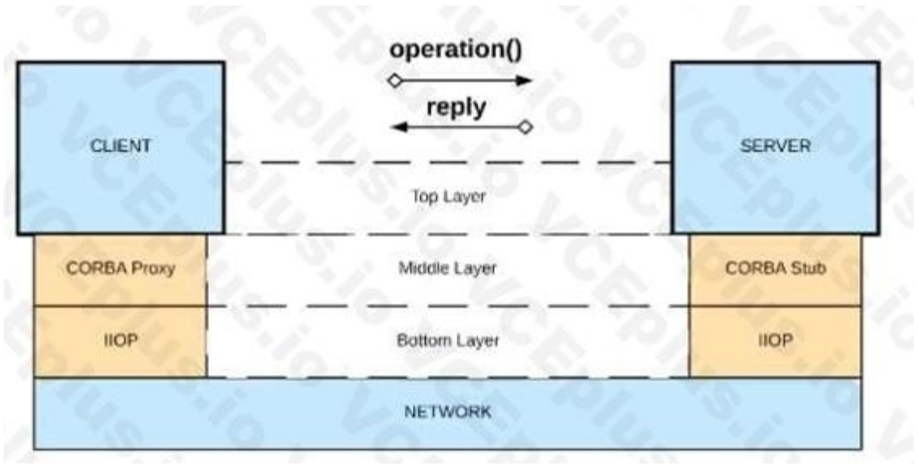
A. Java RMI over TCP



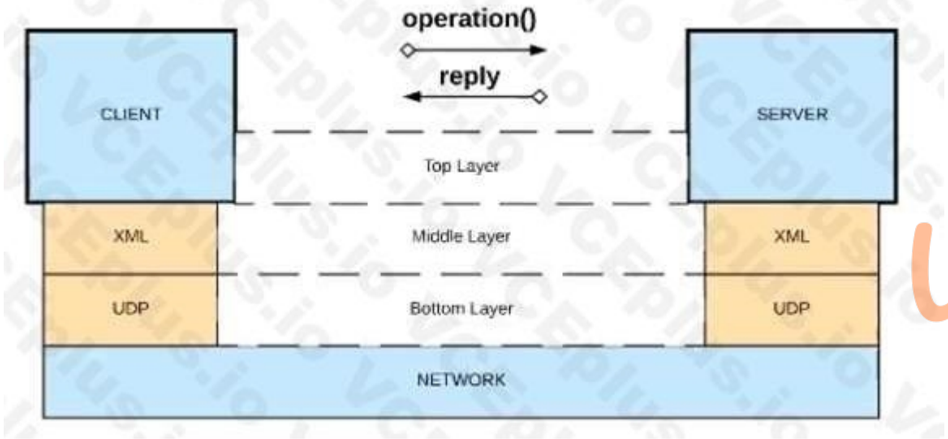
B. Java RMI over TCP



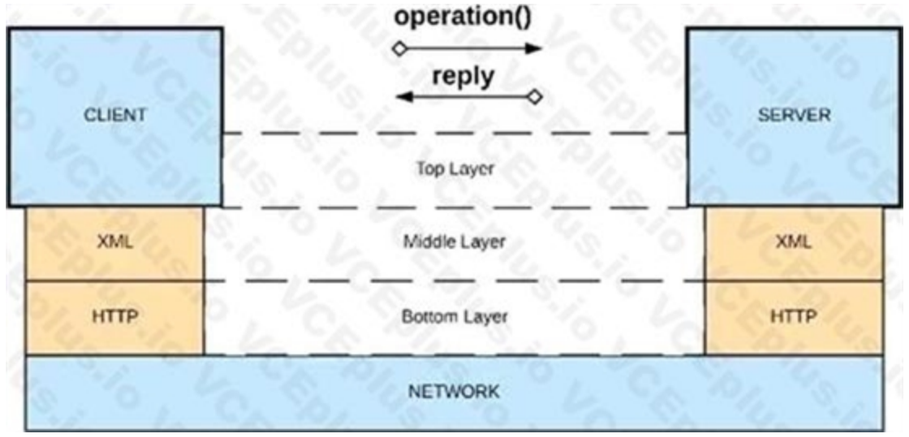
C. CORBA over HOP



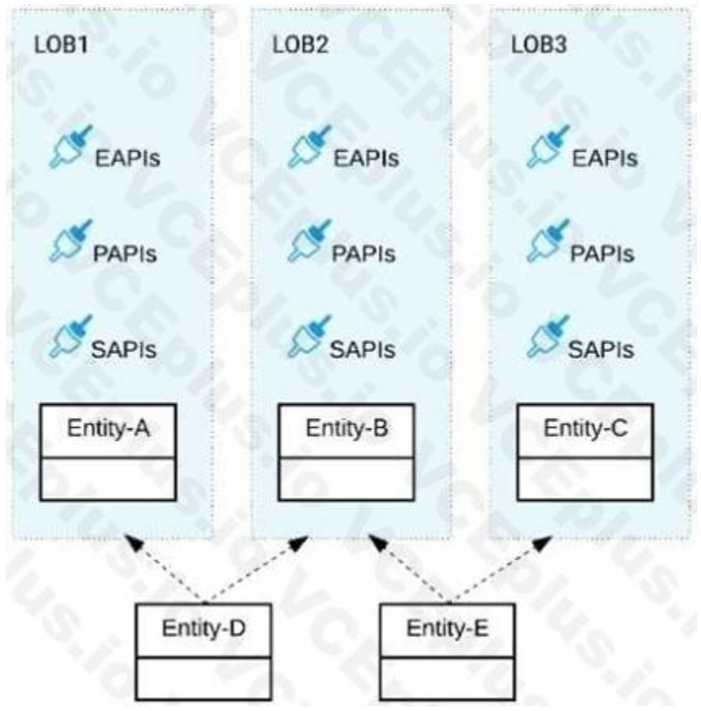
D. XML over UDP



Correct Answer: X, M, L, O, V, E, R, H, T, T, P
Section:
Explanation:
Answer: XML over HTTP
****
>> API-led connectivity and Application Networks urge to have the APIs on HTTP based protocols forbuilding most effective APIs and networks on top of them.
>> The HTTP based APIs allow the platform to apply various varities of policies to address many NFRs
>> The HTTP based APIs also allow to implement many standard and effective implementationpatterns that adhere to HTTP based w3c rules.

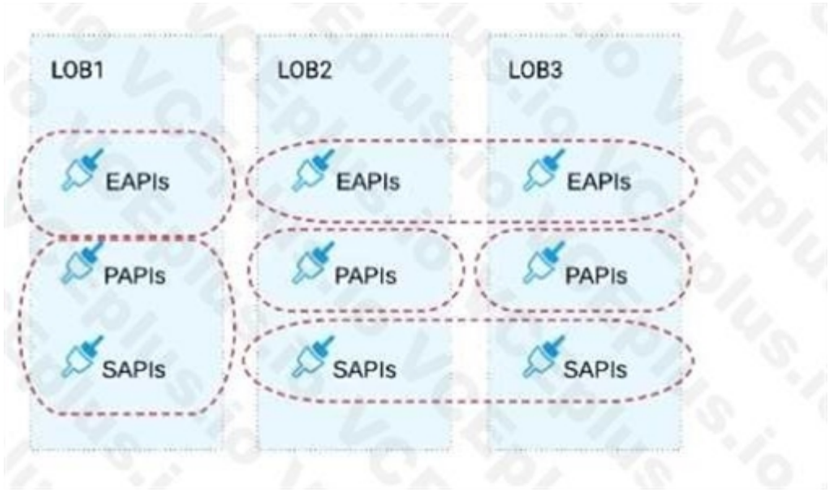


QUESTION 17
Refer to the exhibit.



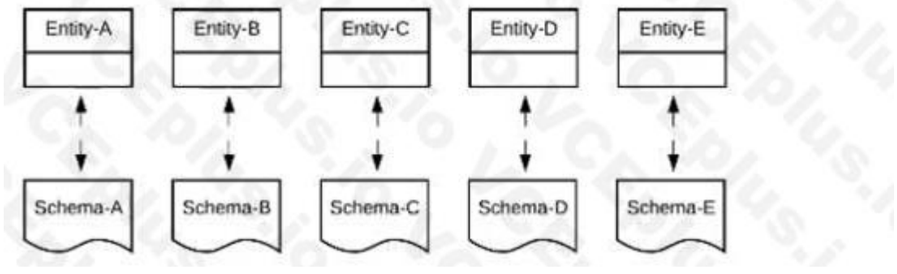
Three business processes need to be implemented, and the implementations need to communicate with several different SaaS applications.
These processes are owned by separate (siloed) LOBs and are mainly independent of each other, but do share a few business entities. Each LOB has one development team and their own budget In this organizational context, what is the most effective approach to choose the API data models for the APIs that will implement these business processes with minimal redundancy of the data models?

A. Build several Bounded Context Data Models that align with coherent parts of the business processes and the definitions of associated business entities

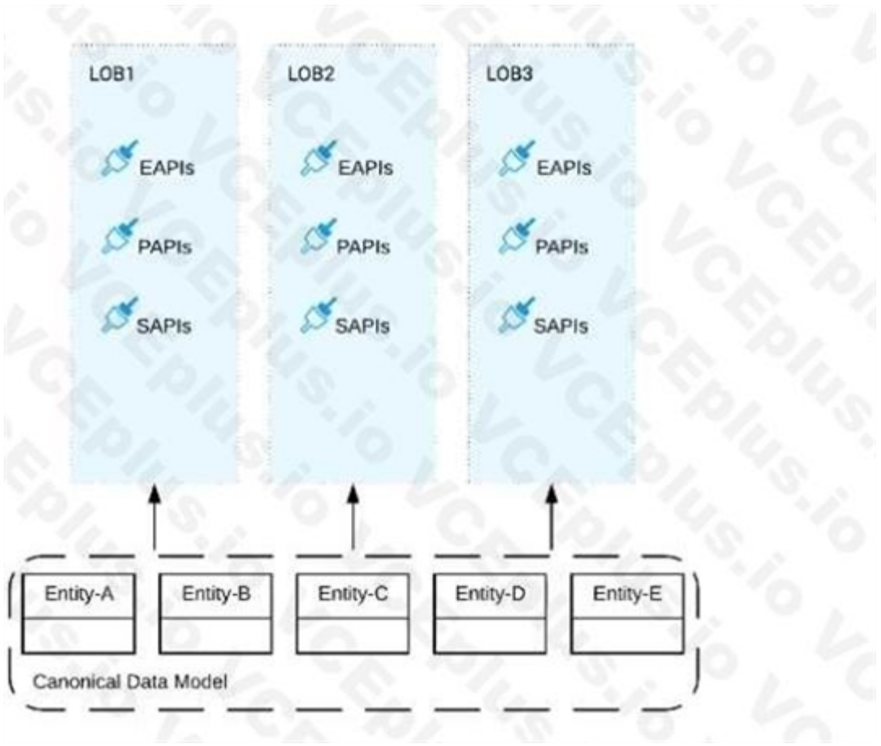


B. Build distinct data models for each API to follow established micro-services and Agile API-centric practices

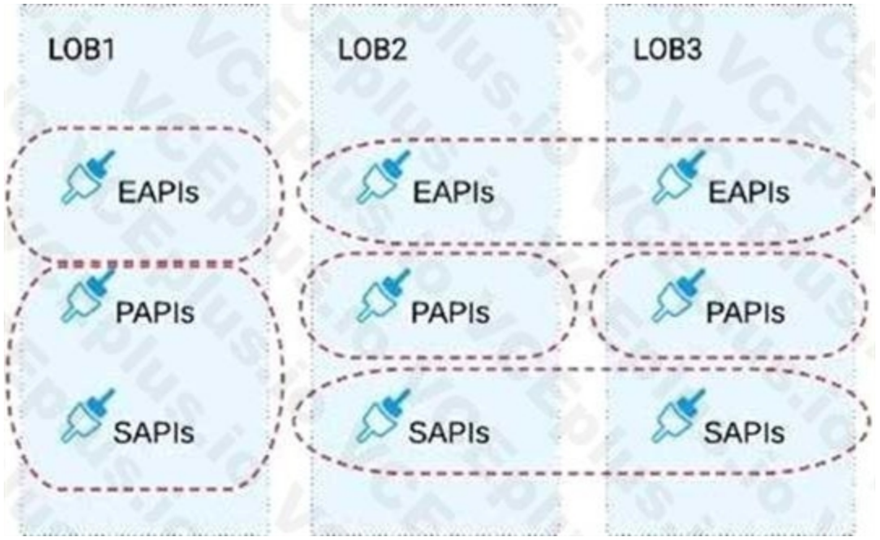
C. Build all API data models using XML schema to drive consistency and reuse across the organization



D. Build one centralized Canonical Data Model (Enterprise Data Model) that unifies all the data types from all three business processes, ensuring the data model is consistent and non-redundant



Correct Answer: B, U, I, L, D, S, E, V, E, R, A, L, B, O, U, N, D, E, D, C, O, N, T, E, X, T, D, A, T, A, M, O, D, E, L, S, T, H, A, T, A, L, I, G, N, W, I, T, H, C, O, H, E, R, E, N, T, P, A, R, T, S, O, F, T, H, E, B, U, S, I, N, E, S, S, P, R, O, C, E, S, S, E, S, A, N, D, T, H, E, D, E, F, I, N, I, T, I, O, N, S, O, F, A, S, S, O, C, I, A, T, E, D, B, U, S, I, N, E, S, S, E, N, T, I, T, I, E, S
Section:
Explanation:
Answer: Build several Bounded Context Data Models that align with coherent parts of the business processes and the definitions of associated business entities.
****
>> The options w.r.t building API data models using XML schema/ Agile API-centric practices are irrelevant to the scenario given in the question. So these two are INVALID.
>> Building EDM (Enterprise Data Model) is not feasible or right fit for this scenario as the teams and LOBs work in silo and they all have different initiatives, budget etc.. Building EDM needs intensive coordination among all the team which evidently seems not possible in this scenario.
So, the right fit for this scenario is to build several Bounded Context Data Models that align with coherent parts of the business processes and the definitions of associated business entities.



QUESTION 18
What best describes the Fully Qualified Domain Names (FQDNs), also known as DNS entries, created when a Mule application is deployed to the CloudHub Shared Worker Cloud?

A. A fixed number of FQDNs are created, IRRESPECTIVE of the environment and VPC design
B. The FQDNs are determined by the application name chosen, IRRESPECTIVE of the region
C. The FQDNs are determined by the application name, but can be modified by an administrator after deployment
D. The FQDNs are determined by both the application name and the Anypoint Platform organization

Correct Answer: T, H, E, F, Q, D, N, S, A, R, E, D, E, T, E, R, M, I, N, E, D, B, Y, T, H, E, A, P, P, L, I, C, A, T, I, O, N, N, A, M, E, C, H, O, S, E, N, I, R, R, E, S, P, E, C, T, I, V, E, O, F, T, H, E, R, E, G, I, O, N
Section:
Explanation:
Answer: The FQDNs are determined by the application name chosen, IRRESPECTIVE of the region **********
>> When deploying applications to Shared Worker Cloud, the FQDN are always determined by application name chosen.
>> It does NOT matter what region the app is being deployed to.
>> Although it is fact and true that the generated FQDN will have the region included in it (Ex: expsalesorder- api.au-s1.cloudhub.io), it does NOT mean that the same name can be used when deploying to another CloudHub region.
>> Application name should be universally unique irrespective of Region and Organization and solely determines the FQDN for Shared Load Balancers.

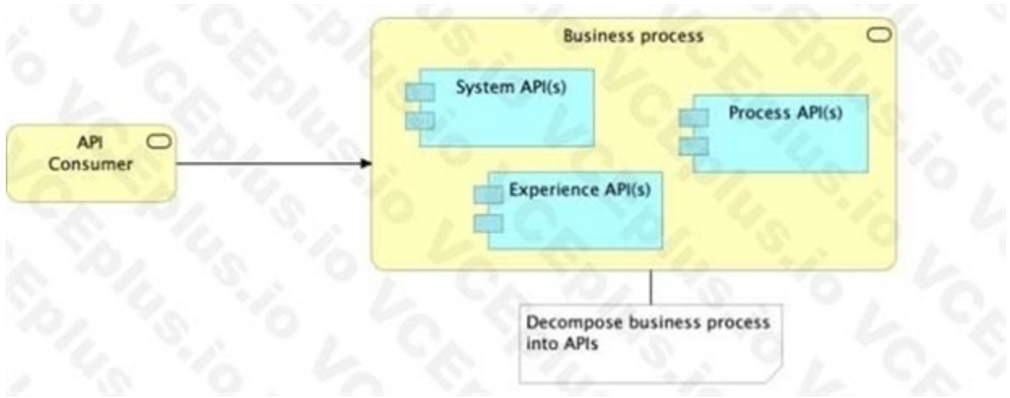
QUESTION 19
When using CloudHub with the Shared Load Balancer, what is managed EXCLUSIVELY by the API implementation (the Mule application) and NOT by Anypoint Platform?

A. The assignment of each HTTP request to a particular CloudHub worker
B. The logging configuration that enables log entries to be visible in Runtime Manager
C. The SSL certificates used by the API implementation to expose HTTPS endpoints
D. The number of DNS entries allocated to the API implementation

Correct Answer: T, H, E, S, S, L, C, E, R, T, I, F, I, C, A, T, E, S, U, S, E, D, B, Y, T, H, E, A, P, I, I, M, P, L, E, M, E, N, T, A, T, I, O, N, T, O, E, X, P, O, S, E, H, T, T, P, S, E, N, D, P, O, I, N, T, S
Section:
Explanation:
Answer: The SSL certificates used by the API implementation to expose HTTPS endpoints**********
>> The assignment of each HTTP request to a particular CloudHub worker is taken care by AnypointPlatform itself. We need not manage it explicitly in the API implementation and in fact we CANNOTmanage it in the API

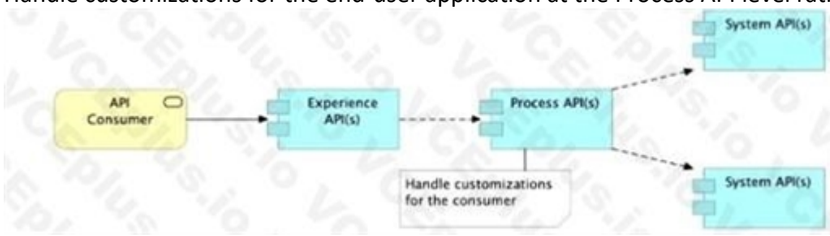
implementation.
>> The logging configuration that enables log entries to be visible in Runtime Manager is ALWAYS managed in the API implementation and NOT just for SLB. So this is not something we do EXCLUSIVELY when using SLB.
>> We DO NOT manage the number of DNS entries allocated to the API implementation inside the code. Anypoint Platform takes care of this.
It is the SSL certificates used by the API implementation to expose HTTPS endpoints that is to bemanaged EXCLUSIVELY by the API implementation. Anypoint Platform does NOT do this when usingSLBs.

QUESTION 20
Refer to the exhibit.

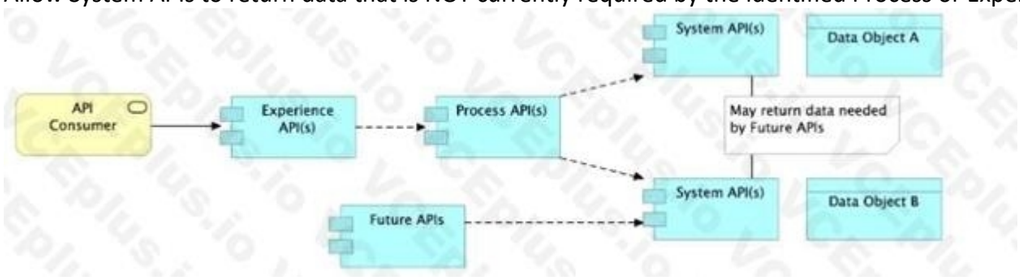


What is the best way to decompose one end-to-end business process into a collaboration of Experience, Process, and System APIs?

A. Handle customizations for the end-user application at the Process API level rather than the Experience API level

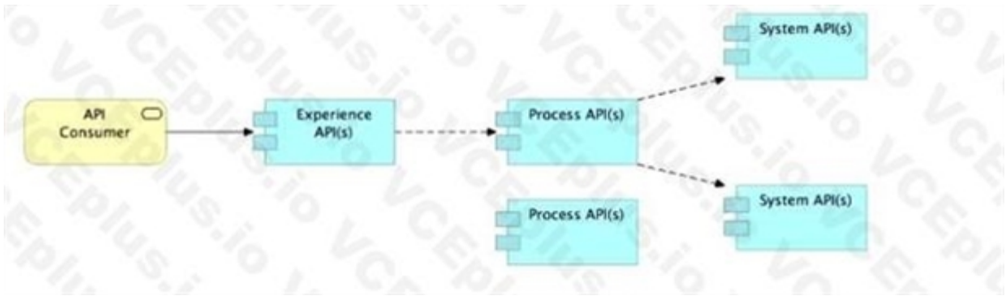


B. Allow System APIs to return data that is NOT currently required by the identified Process or Experience APIs

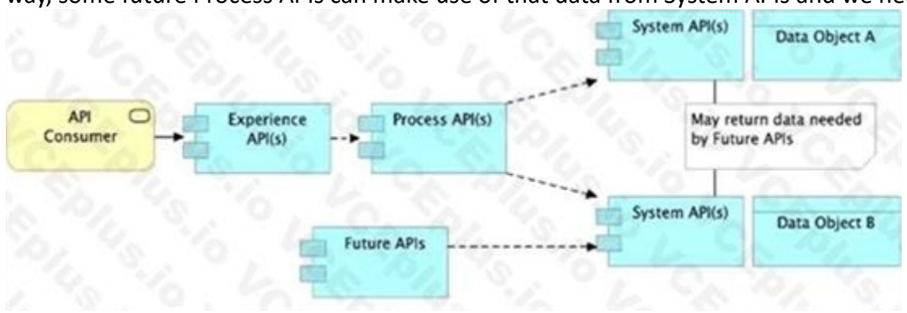


C. Always use a tiered approach by creating exactly one API for each of the 3 layers (Experience, Process and System APIs)

D. Use a Process API to orchestrate calls to multiple System APIs, but NOT to other Process APIs



Correct Answer: A, L, L, O, W, S, Y, S, T, E, M, A, P, I, S, T, O, R, E, T, U, R, N, D, A, T, A, T, H, A, T, I, S, N, O, T, C, U, R, R, E, N, T, L, Y, R, E, Q, U, I, R, E, D, B, Y, T, H, E, I, D, E, N, T, I, F, I, E, D, P, R, O, C, E, S, S, O, R, E, X, P, E, R, I, E, N, C, E, A, P, I, S
Section:
Explanation:
Answer: Allow System APIs to return data that is NOT currently required by the identified Process or Experience APIs.
****
>> All customizations for the end-user application should be handled in "Experience API" only. Not in Process API
>> We should use tiered approach but NOT always by creating exactly one API for each of the 3 layers. Experience APIs might be one but Process APIs and System APIs are often more than one.
System APIs for sure will be more than one all the time as they are the smallest modular APIs built in front of end systems.
>> Process APIs can call System APIs as well as other Process APIs. There is no such anti-design pattern in API-Led connectivity saying Process APIs should not call other Process APIs.
So, the right answer in the given set of options that makes sense as per API-Led connectivity principles is to allow System APIs to return data that is NOT currently required by the identified Process or Experience APIs. This way, some future Process APIs can make use of that data from System APIs and we need NOT touch the System layer APIs again and again.



QUESTION 21
What is true about where an API policy is defined in Anypoint Platform and how it is then applied to API instances?

A. The API policy Is defined In Runtime Manager as part of the API deployment to a Mule runtime, and then ONLY applied to the specific API Instance
B. The API policy Is defined In API Manager for a specific API Instance, and then ONLY applied to the specific API instance
C. The API policy Is defined in API Manager and then automatically applied to ALL API instances
D. The API policy is defined in API Manager, and then applied to ALL API instances in the specified environment

Correct Answer: T, H, E, A, P, I, P, O, L, I, C, Y, I, S, D, E, F, I, N, E, D, I, N, A, P, I, M, A, N, A, G, E, R, F, O, R, A, S, P, E, C, I, F, I, C, A, P, I, I, N, S, T, A, N, C, E, A, N, D, T, H, E, N, O, N, L, Y, A, P, P, L, I, E, D, T, O, T, H, E, S, P, E, C, I, F, I, C, A, P, I, I, N, S, T, A, N, C, E
Section:
Explanation:
Answer: The API policy is defined in API Manager for a specific API instance, and then ONLY applied to the specific API instance.
****
>> Once our API specifications are ready and published to Exchange, we need to visit API Manager and register an API instance for each API.
>> API Manager is the place where management of API aspects takes place like addressing NFRs by enforcing policies on them.
>> We can create multiple instances for a same API and manage them differently for different purposes.

>> One instance can have a set of API policies applied and another instance of same API can have different set of policies applied for some other purpose.
>> These APIs and their instances are defined PER environment basis. So, one need to manage them seperately in each environment.
>> We can ensure that same configuration of API instances (SLAs, Policies etc..) gets promoted when promoting to higher environments using platform feature. But this is optional only. Still one can change them per environment basis if they have to.
>> Runtime Manager is the place to manage API Implementations and their Mule Runtimes but NOT APIs itself. Though API policies gets executed in Mule Runtimes, We CANNOT enforce API policies in Runtime Manager. We would need to do that via API Manager only for a cherry picked instance in an environment.
So, based on these facts, right statement in the given choices is - "The API policy is defined in API Manager for a specific API instance, and then ONLY applied to the specific API instance".
Reference: https://docs.mulesoft.com/api-manager/2.x/latest-overview-concept

QUESTION 22
An API implementation is deployed to CloudHub.
What conditions can be alerted on using the default Anypoint Platform functionality, where the alert conditions depend on the end-to-end request processing of the API implementation?

A. When the API is invoked by an unrecognized API client
B. When a particular API client invokes the API too often within a given time period
C. When the response time of API invocations exceeds a threshold
D. When the API receives a very high number of API invocations

Correct Answer: W, H, E, N, T, H, E, R, E, S, P, O, N, S, E, T, I, M, E, O, F, A, P, I, I, N, V, O, C, A, T, I, O, N, S, E, X, C, E, E, D, S, A, T, H, R, E, S, H, O, L, D
Section:
Explanation:
Answer: When the response time of API invocations exceeds a threshold **********
>> Alerts can be setup for all the given options using the default Anypoint Platform functionality
>> However, the question insists on an alert whose conditions depend on the end-to-end request processing of the API implementation.
>> Alert w.r.t "Response Times" is the only one which requires end-to-end request processing of API implementation in order to determine if the threshold is exceeded or not.
Reference: https://docs.mulesoft.com/api-manager/2.x/using-api-alerts

QUESTION 23
A Mule application exposes an HTTPS endpoint and is deployed to the CloudHub Shared WorkerCloud. All traffic to that Mule application must stay inside the AWS VPC.
To what TCP port do API invocations to that Mule application need to be sent?

A. 443
B. 8081
C. 8091
D. 8082

Correct Answer:
Section:
Explanation:
Answer: 8082
****
>> 8091 and 8092 ports are to be used when keeping your HTTP and HTTPS app private to the LOCALVPC respectively.
>> Above TWO ports are not for Shared AWS VPC/ Shared Worker Cloud.
>> 8081 is to be used when exposing your HTTP endpoint app to the internet through Shared LB
>> 8082 is to be used when exposing your HTTPS endpoint app to the internet through Shared LBSo, API invocations should be sent to port 8082 when calling this HTTPS based app.
References:
https://docs.mulesoft.com/runtime-manager/cloudhub-networking-guide
https://help.mulesoft.com/s/article/Configure-Cloudhub-Application-to-Send-a-HTTPS-Request-Directly-to-Another-Cloudhub-Application
https://help.mulesoft.com/s/question/0D52T00004mXXULSA4/multiple-http-listerners-oncloudhub-one-with-port-9090

QUESTION 24
What is a key requirement when using an external Identity Provider for Client Management in Anypoint Platform?

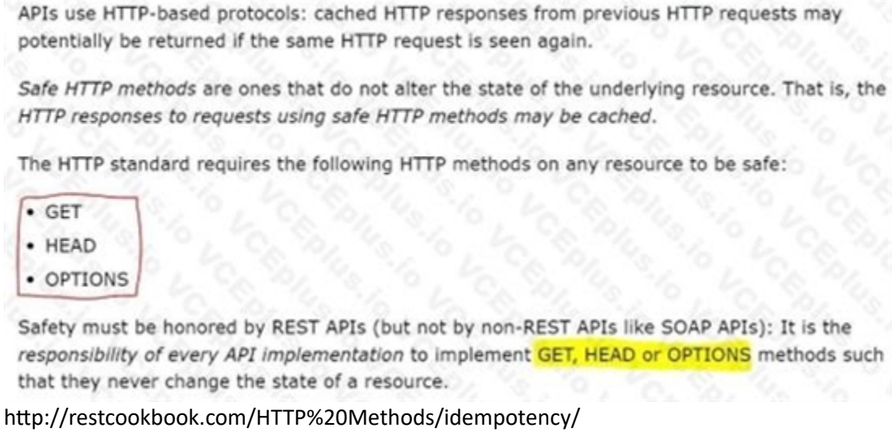
A. Single sign-on is required to sign in to Anypoint Platform
B. The application network must include System APIs that interact with the Identity Provider
C. To invoke OAuth 2.0-protected APIs managed by Anypoint Platform, API clients must submit access tokens issued by that same Identity Provider
D. APIs managed by Anypoint Platform must be protected by SAML 2.0 policies

Correct Answer: T, O, I, N, V, O, K, E, O, A, U, T, H, P, R, O, T, E, C, T, E, D, A, P, I, S, M, A, N, A, G, E, D, B, Y, A, N, Y, P, O, I, N, T, P, L, A, T, F, O, R, M, A, P, I, C, L, I, E, N, T, S, M, U, S, T, S, U, B, M, I, T, A, C, C, E, S, S, T, O, K, E, N, S, I, S, S, U, E, D, B, Y, T, H, A, T, S, A, M, E, I, D, E, N, T, I, T, Y, P, R, O, V, I, D, E, R
Section:
Explanation:
https://www.folkstalk.com/2019/11/mulesoft-integration-and-platform.html
Explanation:
Answer: To invoke OAuth 2.0-protected APIs managed by Anypoint Platform, API clients must submit access tokens issued by that same Identity Provider **********
>> It is NOT necessary that single sign-on is required to sign in to Anypoint Platform because we are using an external Identity Provider for Client Management
>> It is NOT necessary that all APIs managed by Anypoint Platform must be protected by SAML 2.0 policies because we are using an external Identity Provider for Client Management
>> Not TRUE that the application network must include System APIs that interact with the Identity Provider because we are using an external Identity Provider for Client Management Only TRUE statement in the given options is - "To invoke OAuth 2.0-protected APIs managed by Anypoint Platform, API clients must submit access tokens issued by that same Identity Provider" References:
https://docs.mulesoft.com/api-manager/2.x/external-oauth-2.0-token-validation-policy
https://blogs.mulesoft.com/dev/api-dev/api-security-ways-to-authenticate-and-authorize/

QUESTION 25
The responses to some HTTP requests can be cached depending on the HTTP verb used in therequest. According to the HTTP specification, for what HTTP verbs is this safe to do?

A. PUT, POST, DELETE
B. GET, HEAD, POST
C. GET, PUT, OPTIONS
D. GET, OPTIONS, HEAD

Correct Answer: G, E, T, O, P, T, I, O, N, S, H, E, A, D
Section:
Explanation:
Answer: GET, OPTIONS, HEAD


APIs use HTTP-based protocols: cached HTTP responses from previous HTTP requests may potentially be returned if the same HTTP request is seen again.

Safe HTTP methods are ones that do not alter the state of the underlying resource. That is, the *HTTP responses to requests using safe HTTP methods may be cached.*

The HTTP standard requires the following HTTP methods on any resource to be safe:

- GET
- HEAD
- OPTIONS

Safety must be honored by REST APIs (but not by non-REST APIs like SOAP APIs): It is the *responsibility of every API implementation* to implement GET, HEAD or OPTIONS methods such that they never change the state of a resource.

http://restcookbook.com/HTTP%20Methods/idempotency/

QUESTION 26

What is the most performant out-of-the-box solution in Anypoint Platform to track transaction state in an asynchronously executing long-running process implemented as a Mule application deployed to multiple CloudHub workers?

A. Redis distributed cache
B. java.util.WeakHashMap
C. Persistent Object Store
D. File-based storage

Correct Answer: P, E, R, S, I, S, T, E, N, T, O, B, J, E, C, T, S, T, O, R, E
Section:
Explanation:
Answer: Persistent Object Store
****
>> Redis distributed cache is performant but NOT out-of-the-box solution in Anypoint Platform
>> File-storage is neither performant nor out-of-the-box solution in Anypoint Platform
>> java.util.WeakHashMap needs a completely custom implementation of cache from scratch using Java code and is limited to the JVM where it is running. Which means the state in the cache is not worker aware when running on multiple workers. This type of cache is local to the worker. So, this is neither out-of-the-box nor worker-aware among multiple workers on cloudhub.
https://www.baeldung.com/java-weakhashmap
>> Persistent Object Store is an out-of-the-box solution provided by Anypoint Platform which is performant as well as worker aware among multiple workers running on CloudHub.
https://docs.mulesoft.com/object-store/
So, Persistent Object Store is the right answer.

QUESTION 27
How can the application of a rate limiting API policy be accurately reflected in the RAML definition of an API?

A. By refining the resource definitions by adding a description of the rate limiting policy behavior
B. By refining the request definitions by adding a remaining Requests query parameter with description, type, and example
C. By refining the response definitions by adding the out-of-the-box Anypoint Platform rate-limitenforcement securityScheme with description, type, and example
D. By refining the response definitions by adding the x-ratelimit-* response headers with description, type, and example

Correct Answer: B, Y, R, E, F, I, N, I, N, G, T, H, E, R, E, S, P, O, N, S, E, D, E, F, I, N, I, T, I, O, N, S, B, Y, A, D, D, I, N, G, T, H, E, X, R, A, T, E, L, I, M, I, T, R, E, S, P, O, N, S, E, H, E, A, D, E, R, S, W, I, T, H, D, E, S, C, R, I, P, T, I, O, N, T, Y, P, E, A, N, D, E, X, A, M, P, L, E
Section:
Explanation:
Answer: By refining the response definitions by adding the x-ratelimit-* response headers with description, type, and example
****

Response Headers

The following access-limiting policies return headers having information about the current state of the request:

- X-Ratelimit-Remaining: The amount of available quota.
- X-Ratelimit-Limit: The maximum available requests per window.
- X-Ratelimit-Reset: The remaining time, in milliseconds, until a new window starts.

Response Headers

Three headers are included in request responses that inform users about the SLA restrictions and inform them when nearing the threshold. When the SLA enforces multiple policies that limit request throughput, a single set of headers pertaining to the most restrictive of the policies provides this information.

For example, a user of your API may receive a response that includes these headers:

```
X-RateLimit-Limit: 20
X-RateLimit-Remaining: 14
X-RateLimit-Reset: 19100
```

Within the next 19100 milliseconds, only 14 more requests are allowed by the SLA, which is set to allow 20 within this time-window.

References:
https://docs.mulesoft.com/api-manager/2.x/rate-limiting-and-throttling#response-headers
https://docs.mulesoft.com/api-manager/2.x/rate-limiting-and-throttling-sla-basedpolicies#response-headers

QUESTION 28
An organization has several APIs that accept JSON data over HTTP POST. The APIs are all publiclyavailable and are associated with several mobile applications and web applications.
The organization does NOT want to use any authentication or compliance policies for these APIs, but at the same time, is worried that some bad actor could send payloads that could somehow compromise the applications or servers running the API implementations.
What out-of-the-box Anypoint Platform policy can address exposure to this threat?

A. Shut out bad actors by using HTTPS mutual authentication for all API invocations
B. Apply an IP blacklist policy to all APIs; the blacklist will Include all bad actors
C. Apply a Header injection and removal policy that detects the malicious data before it is used
D. Apply a JSON threat protection policy to all APIs to detect potential threat vectors

Correct Answer: A, P, P, L, Y, A, J, S, O, N, T, H, R, E, A, T, P, R, O, T, E, C, T, I, O, N, P, O, L, I, C, Y, T, O, A, L, L, A, P, I, S, T, O, D, E, T, E, C, T, P, O, T, E, N, T, I, A, L, T, H, R, E, A, T, V, E, C, T, O, R, S
Section:
Explanation:
Answer: Apply a JSON threat protection policy to all APIs to detect potential threat vectors **********
>> Usually, if the APIs are designed and developed for specific consumers (known consumers/customers) then we would IP Whitelist the same to ensure that traffic only comes from them.

>> However, as this scenario states that the APIs are publicly available and being used by so many mobile and web applications, it is NOT possible to identify and blacklist all possible bad actors.
>> So, JSON threat protection policy is the best chance to prevent any bad JSON payloads from such bad actors.

QUESTION 29
An API experiences a high rate of client requests (TPS) vwth small message paytoads. How can usage limits be imposed on the API based on the type of client application?

A. Use an SLA-based rate limiting policy and assign a client application to a matching SLA tier based on its type
B. Use a spike control policy that limits the number of requests for each client application type
C. Use a cross-origin resource sharing (CORS) policy to limit resource sharing between client applications, configured by the client application type
D. Use a rate limiting policy and a client ID enforcement policy, each configured by the client application type

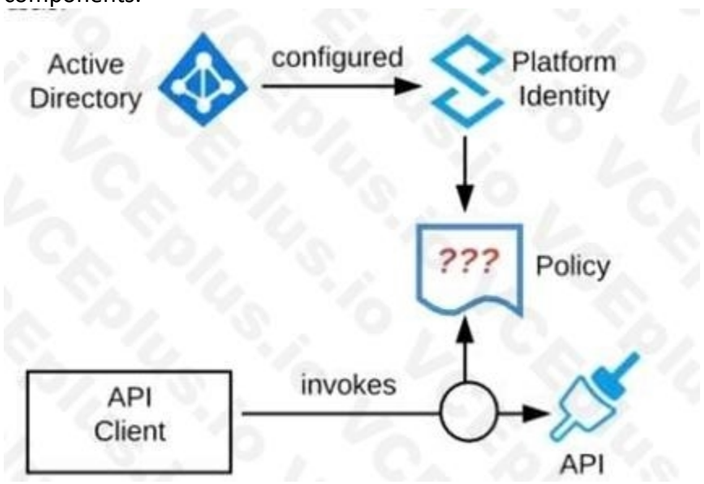
Correct Answer: U, S, E, A, N, S, L, A, B, A, S, E, D, R, A, T, E, L, I, M, I, T, I, N, G, P, O, L, I, C, Y, A, N, D, A, S, S, I, G, N, A, C, L, I, E, N, T, A, P, P, L, I, C, A, T, I, O, N, T, O, A, M, A, T, C, H, I, N, G, S, L, A, T, I, E, R, B, A, S, E, D, O, N, I, T, S, T, Y, P, E
Section:
Explanation:
Answer: Use an SLA-based rate limiting policy and assign a client application to a matching SLA tier based on its type.
****
>> SLA tiers will come into play whenever any limits to be imposed on APIs based on client type
Reference: https://docs.mulesoft.com/api-manager/2.x/rate-limiting-and-throttling-sla-basedpolicies

QUESTION 30
A code-centric API documentation environment should allow API consumers to investigate and execute API client source code that demonstrates invoking one or more APIs as part of representative scenarios.
What is the most effective way to provide this type of code-centric API documentation environment using Anypoint Platform?

A. Enable mocking services for each of the relevant APIs and expose them via their Anypoint Exchange entry
B. Ensure the APIs are well documented through their Anypoint Exchange entries and API Consoles and share these pages with all API consumers
C. Create API Notebooks and include them in the relevant Anypoint Exchange entries
D. Make relevant APIs discoverable via an Anypoint Exchange entry

Correct Answer: C, R, E, A, T, E, A, P, I, N, O, T, E, B, O, O, K, S, A, N, D, I, N, C, L, U, D, E, T, H, E, M, I, N, T, H, E, R, E, L, E, V, A, N, T, A, N, Y, P, O, I, N, T, E, X, C, H, A, N, G, E, E, N, T, R, I, E, S
Section:
Explanation:
Answer: Create API Notebooks and Include them in the relevant Anypoint exchange entries
****
>> API Notebooks are the one on Anypoint Platform that enable us to provide code-centric API documentation
Reference: https://docs.mulesoft.com/exchange/to-use-api-notebook

QUESTION 31
Refer to the exhibit. An organization is running a Mule standalone runtime and has configured Active Directory as the Anypoint Platform external Identity Provider. The organization does not have budget for other system components.



What policy should be applied to all instances of APIs in the organization to most effecuvelyKestrict access to a specific group of internal users?

A. Apply a basic authentication - LDAP policy; the internal Active Directory will be configured as the LDAP source for authenticating users
B. Apply a client ID enforcement policy; the specific group of users will configure their client applications to use their specific client credentials
C. Apply an IP whitelist policy; only the specific users' workstations will be in the whitelist
D. Apply an OAuth 2.0 access token enforcement policy; the internal Active Directory will be configured as the OAuth server

Correct Answer: A, P, P, L, Y, A, B, A, S, I, C, A, U, T, H, E, N, T, I, C, A, T, I, O, N, L, D, A, P, P, O, L, I, C, Y, T, H, E, I, N, T, E, R, N, A, L, A, C, T, I, V, E, D, I, R, E, C, T, O, R, Y, W, I, L, L, B, E, C, O, N, F, I, G, U, R, E, D, A, S, T, H, E, L, D, A, P, S, O, U, R, C, E, F, O, R, A, U, T, H, E, N, T, I, C, A, T, I, N, G, U, S, E, R, S
Section:
Explanation:
Answer: Apply a basic authentication - LDAP policy; the internal Active Directory will be configured as the LDAP source for authenticating users.
****
>> IP Whitelisting does NOT fit for this purpose. Moreover, the users workstations may not necessarily have static IPs in the network.
>> OAuth 2.0 enforcement requires a client provider which isn't in the organizations system components.
>> It is not an effective approach to let every user create separate client credentials and configure those for their usage.

The effective way it to apply a basic authentication - LDAP policy and the internal Active Directory will be configured as the LDAP source for authenticating users.
Reference: https://docs.mulesoft.com/api-manager/2.x/basic-authentication-ldap-concept

QUESTION 32
What is a best practice when building System APIs?

A. Document the API using an easily consumable asset like a RAML definition
B. Model all API resources and methods to closely mimic the operations of the backend system
C. Build an Enterprise Data Model (Canonical Data Model) for each backend system and apply it to System APIs
D. Expose to API clients all technical details of the API implementation's interaction wifch the backend system

Correct Answer: M, O, D, E, L, A, L, L, A, P, I, R, E, S, O, U, R, C, E, S, A, N, D, M, E, T, H, O, D, S, T, O, C, L, O, S, E, L, Y, M, I, M, I, C, T, H, E, O, P, E, R, A, T, I, O, N, S, O, F, T, H, E, B, A, C, K, E, N, D, S, Y, S, T, E, M
Section:
Explanation:
Answer: Model all API resources and methods to closely mimic the operations of the backend system.
****
>> There are NO fixed and straight best practices while opting data models for APIs. They are completly contextual and depends on number of factors. Based upon those factors, an enterprise can choose if they have to go with Enterprise
Canonical Data Model or Bounded Context Model etc.
>> One should NEVER expose the technical details of API implementation to their API clients. Only the API interface/ RAML is exposed to API clients.
>> It is true that the RAML definitions of APIs should be as detailed as possible and should reflect most of the documentation. However, just that is NOT enough to call your API as best documented API. There should be even more documentation on Anypoint Exchange with API Notebooks etc. to make and create a developer friendly API and repository..
>> The best practice always when creating System APIs is to create their API interfaces by modeling their resources and methods to closely reflect the operations and functionalities of that backend system.

QUESTION 33
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

Correct Answer: G, U, A, R, D, I, N, G, A, G, A, I, N, S, T, D, E, N, I, A, L, O, F, S, E, R, V, I, C, E, A, T, T, A, C, K, S
Section:
Explanation:
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

QUESTION 34
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 modem 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

Correct Answer: B
Section:
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)
****

QUESTION 35
What API policy would be LEAST LIKELY used when designing an Experience API that is intended to work with a consumer mobile phone or tablet application?

A. OAuth 2.0 access token enforcement
B. Client ID enforcement
C. JSON threat protection
D. IPwhitellst

Correct Answer: I, P, W, H, I, T, E, L, I, S, T
Section:
Explanation:
Answer: IP whitelist
****
>> OAuth 2.0 access token and Client ID enforcement policies are VERY common to apply on Experience APIs as API consumers need to register and access the APIs using one of these mechanisms
>> JSON threat protection is also VERY common policy to apply on Experience APIs to prevent bad or suspicious payloads hitting the API implementations.
>> IP whitelisting policy is usually very common in Process and System APIs to only whitelist the IP range inside the local VPC. But also applied occassionally on some experience APIs where the End User/ API Consumers are FIXED.
>> When we know the API consumers upfront who are going to access certain Experience APIs, then we can request for static IPs from such consumers and whitelist them to prevent anyone else hitting the API.
However, the experience API given in the question/ scenario is intended to work with a consumer mobile phone or tablet application. Which means, there is no way we can know all possible IPs that are to be whitelisted as mobile phones and tablets can so many in number and any device in the city/state/country/globe.
So, It is very LEAST LIKELY to apply IP Whitelisting on such Experience APIs whose consumers are typically Mobile Phones or Tablets.

QUESTION 36
A new upstream API Is being designed to offer an SLA of 500 ms median and 800 ms maximum (99th percentile) response time. The corresponding API implementation needs to sequentially invoke 3 downstream APIs of very similar complexity.
The first of these downstream APIs offers the following SLA for its response time: median: 100 ms, 80th percentile: 500 ms, 95th percentile: 1000 ms.
If possible, how can a timeout be set in the upstream API for the invocation of the first downstream API to meet the new upstream API's desired SLA?

A. Set a timeout of 50 ms; this times out more invocations of that API but gives additional room for retries
B. Set a timeout of 100 ms; that leaves 400 ms for the other two downstream APIs to complete
C. No timeout is possible to meet the upstream API's desired SLA; a different SLA must be negotiated with the first downstream API or invoke an alternative API
D. Do not set a timeout; the Invocation of this API Is mandatory and so we must wait until it responds

Correct Answer: S, E, T, A, T, I, M, E, O, U, T, O, F, M, S, T, H, A, T, L, E, A, V, E, S, M, S, F, O, R, O, T, H, E, R, T, W, O, D, O, W, N, S, T, R, E, A, M, A, P, I, S, T, O, C, O, M, P, L, E, T, E
Section:
Explanation:
Answer: Set a timeout of 100ms; that leaves 400ms for other two downstream APIs to complete
****
Key details to take from the given scenario:
>> Upstream API's designed SLA is 500ms (median). Lets ignore maximum SLA response times.
>> This API calls 3 downstream APIs sequentially and all these are of similar complexity.
>> The first downstream API is offering median SLA of 100ms, 80th percentile: 500ms; 95th percentile: 1000ms.
Based on the above details:
>> We can rule out the option which is suggesting to set 50ms timeout. Because, if the median SLA itself being offered is 100ms then most of the calls are going to timeout and time gets wasted in retried them and eventually gets exhausted with all retries. Even if some retries gets successful, the remaining time wont leave enough room for 2nd and 3rd downstream APIs to respond within time.
>> The option suggesting to NOT set a timeout as the invocation of this API is mandatory and so we must wait until it responds is silly. As not setting time out would go against the good implementation pattern and moreover if the first API is not responding within its offered median SLA 100ms then most probably it would either respond in 500ms (80th percentile) or 1000ms (95th percentile). In BOTH cases, getting a successful response from 1st downstream API does NO
GOOD because already by this time the Upstream API SLA of 500 ms is breached. There is no time left to call 2nd and 3rd downstream APIs.
>> It is NOT true that no timeout is possible to meet the upstream APIs desired SLA.
As 1st downstream API is offering its median SLA of 100ms, it means MOST of the time we would get the responses within that time. So, setting a timeout of 100ms would be ideal for MOST calls as it leaves enough room of 400ms for remaining 2 downstream API calls.

QUESTION 37
What is true about automating interactions with Anypoint Platform using tools such as Anypoint Platform REST APIs, Anypoint CU, or the Mule Maven plugin?

A. Access to Anypoint Platform APIs and Anypoint CU can be controlled separately through the roles and permissions in Anypoint Platform, so that specific users can get access to Anypoint CLI white others get access to the platform APIs
B. Anypoint Platform APIs can ONLY automate interactions with CloudHub, while the Mule Maven plugin is required for deployment to customer-hosted Mule runtimes
C. By default, the Anypoint CLI and Mule Maven plugin are NOT included in the Mule runtime, so are NOT available to be used by deployed Mule applications
D. API policies can be applied to the Anypoint Platform APIs so that ONLY certain LOBs have access to specific functions

Correct Answer: B, Y, D, E, F, A, U, L, T, T, H, E, A, N, Y, P, O, I, N, T, C, L, I, A, N, D, M, U, L, E, M, A, V, E, N, P, L, U, G, I, N, A, R, E, N, O, T, I, N, C, L, U, D, E, D, I, N, T, H, E, M, U, L, E, R, U, N, T, I, M, E, S, O, A, R, E, N, O, T, A, V, A, I, L, A, B, L, E, T, O, B, E, U, S, E, D, B, Y, D, E, P, L, O, Y, E, D, M, U, L, E, A, P, P, L, I, C, A, T, I, O, N, S
Section:
Explanation:
Answer: By default, the Anypoint CLI and Mule Maven plugin are NOT included in the Mule runtime, so are NOT available to be used by deployed Mule applications **********
>> We CANNOT apply API policies to the Anypoint Platform APIs like we can do on our custom written API instances. So, option suggesting this is FALSE.
>> Anypoint Platform APIs can be used for automating interactions with both CloudHub and customer-hosted Mule runtimes. Not JUST the CloudHub. So, option opposing this is FALSE.
>> Mule Maven plugin is NOT mandatory for deployment to customer-hosted Mule runtimes. It just helps your CI/CD to have smoother automation. But not a compulsory requirement to deploy. So, option opposing this is FALSE.

>> We DO NOT have any such special roles and permissions on the platform to separately control access for some users to have Anypoint CLI and others to have Anypoint Platform APIs. With proper general roles/permissions (API Owner,
Cloudhub Admin etc..), one can use any of the options
(Anypoint CLI or Platform APIs). So, option suggesting this is FALSE.
Only TRUE statement given in the choices is that - Anypoint CLI and Mule Maven plugin are NOT included in the Mule runtime, so are NOT available to be used by deployed Mule applications.
Maven is part of Studio or you can use other Maven installation for development.
CLI is convenience only. It is one of many ways how to install app to the runtime.
These are definitely NOT part of anything except your process of deployment or automation.

QUESTION 38
What Mule application deployment scenario requires using Anypoint Platform Private Cloud Edition or Anypoint Platform for Pivotal Cloud Foundry?

A. When it Is required to make ALL applications highly available across multiple data centers
B. When it is required that ALL APIs are private and NOT exposed to the public cloud
C. When regulatory requirements mandate on-premises processing of EVERY data item, including meta-data
D. When ALL backend systems in the application network are deployed in the organization's intranet

Correct Answer: W, H, E, N, R, E, G, U, L, A, T, O, R, Y, R, E, Q, U, I, R, E, M, E, N, T, S, M, A, N, D, A, T, E, O, N, P, R, E, M, I, S, E, S, P, R, O, C, E, S, S, I, N, G, O, F, E, V, E, R, Y, D, A, T, A, I, T, E, M, I, N, C, L, U, D, I, N, G, M, E, T, A, D, A, T, A
Section:
Explanation:
Answer: When regulatory requirements mandate on-premises processing of EVERY data item, including meta-data.
****
We need NOT require to use Anypoint Platform PCE or PCF for the below. So these options are OUT.
>> We can make ALL applications highly available across multiple data centers using CloudHub too.
>> We can use Anypoint VPN and tunneling from CloudHub to connect to ALL backend systems in the application network that are deployed in the organization's intranet.
>> We can use Anypoint VPC and Firewall Rules to make ALL APIs private and NOT exposed to the public cloud.
Only valid reason in the given options that requires to use Anypoint Platform PCE/ PCF is - When regulatory requirements mandate on-premises processing of EVERY data item, including meta-data.

QUESTION 39
What is typically NOT a function of the APIs created within the framework called API-led connectivity?

A. They provide an additional layer of resilience on top of the underlying backend system, thereby insulating clients from extended failure of these systems.
B. They allow for innovation at the user Interface level by consuming the underlying assets without being aware of how data Is being extracted from backend systems.
C. They reduce the dependency on the underlying backend systems by helping unlock data from backend systems In a reusable and consumable way.
D. They can compose data from various sources and combine them with orchestration logic to create higher level value.

Correct Answer: T, H, E, Y, P, R, O, V, I, D, E, A, N, A, D, D, I, T, I, O, N, A, L, L, A, Y, E, R, O, F, R, E, S, I, L, I, E, N, C, E, O, N, T, O, P, O, F, T, H, E, U, N, D, E, R, L, Y, I, N, G, B, A, C, K, E, N, D, S, Y, S, T, E, M, T, H, E, R, E, B, Y, I, N, S, U, L, A, T, I, N, G, C, L, I, E, N, T, S, F, R, O, M, E, X, T, E, N, D, E, D, F, A, I, L, U, R, E, O, F, T, H, E, S, E, S, Y, S, T, E, M, S
Section:
Explanation:
Answer: They provide an additional layer of resilience on top of the underlying backend system, thereby insulating clients from extended failure of these systems.
****
In API-led connectivity,
>> Experience APIs - allow for innovation at the user interface level by consuming the underlying assets without being aware of how data is being extracted from backend systems.
>> Process APIs - compose data from various sources and combine them with orchestration logic to create higher level value
>> System APIs - reduce the dependency on the underlying backend systems by helping unlock data from backend systems in a reusable and consumable way.
However, they NEVER promise that they provide an additional layer of resilience on top of the underlying backend system, thereby insulating clients from extended failure of these systems.
https://dzone.com/articles/api-led-connectivity-with-mule

QUESTION 40
An organization has implemented a Customer Address API to retrieve customer address information.
This API has been deployed to multiple environments and has been configured to enforce client IDs everywhere.
A developer is writing a client application to allow a user to update their address. The developer has found the Customer Address API in Anypoint Exchange and wants to use it in their client application.
What step of gaining access to the API can be performed automatically by Anypoint Platform?

A. Approve the client application request for the chosen SLA tier
B. Request access to the appropriate API Instances deployed to multiple environments using the client application's credentials
C. Modify the client application to call the API using the client application's credentials
D. Create a new application in Anypoint Exchange for requesting access to the API

Correct Answer: A, P, P, R, O, V, E, T, H, E, C, L, I, E, N, T, A, P, P, L, I, C, A, T, I, O, N, R, E, Q, U, E, S, T, F, O, R, T, H, E, C, H, O, S, E, N, S, L, A, T, I, E, R
Section:
Explanation:
Answer: Approve the client application request for the chosen SLA tier **********
>> Only approving the client application request for the chosen SLA tier can be automated
>> Rest of the provided options are not valid
Reference: https://docs.mulesoft.com/api-manager/2.x/defining-sla-tiers#defining-a-tier

QUESTION 41
What is a typical result of using a fine-grained rather than a coarse-grained API deployment model to implement a given business process?

A. A decrease in the number of connections within the application network supporting the business process
B. A higher number of discoverable API-related assets in the application network
C. A better response time for the end user as a result of the APIs being smaller in scope and complexity
D. An overall tower usage of resources because each fine-grained API consumes less resources

Correct Answer: A, H, I, G, H, E, R, N, U, M, B, E, R, O, F, D, I, S, C, O, V, E, R, A, B, L, E, A, P, I, R, E, L, A, T, E, D, A, S, S, E, T, S, I, N, T, H, E, A, P, P, L, I, C, A, T, I, O, N, N, E, T, W, O, R, K
Section:
Explanation:
Answer: A higher number of discoverable API-related assets in the application network.
****
>> We do NOT get faster response times in fine-grained approach when compared to coarse-grained approach.
>> In fact, we get faster response times from a network having coarse-grained APIs compared to a network having fine-grained APIs model. The reasons are below.
Fine-grained approach:
1. will have more APIs compared to coarse-grained
2. So, more orchestration needs to be done to achieve a functionality in business process.
3. Which means, lots of API calls to be made. So, more connections will needs to be established. So, obviously more hops, more network i/o, more number of integration points compared to coarsegrained approach where fewer APIs with bulk functionality embedded in them.
4. That is why, because of all these extra hops and added latencies, fine-grained approach will have bit more response times compared to coarse-grained.
5. Not only added latencies and connections, there will be more resources used up in fine-grained approach due to more number of APIs.
That's why, fine-grained APIs are good in a way to expose more number of resuable assets in your network and make them discoverable. However, needs more maintenance, taking care of integration points, connections, resources with a little compromise w.r.t network hops and response times.

QUESTION 42
What correctly characterizes unit tests of Mule applications?

A. They test the validity of input and output of source and target systems

B. They must be run in a unit testing environment with dedicated Mule runtimes for the environment
C. They must be triggered by an external client tool or event source
D. They are typically written using MUnit to run in an embedded Mule runtime that does not require external connectivity

Correct Answer: A, A, A, A, A, B, C, C, C, D, D, D, D, E, E, E, E, E, E, E, E, E, E, E, E, E, G, H, H, I, I, I, I, I, I, I, I, I, L, L, L, L, M, M, M, M, N, N, N, N, N, N, N, N, N, N, N, O, O, O, O, P, Q, R, R, R, R, R, R, R, S, S, T, T, T, T, T, T, T, T, T, T, T, T, T, T, U, U, U, U, U, U, V, W, X, Y, Y, Y, Y
Section:
Explanation:
Answer: They are typically written using MUnit to run in an embedded Mule runtime that does not require external connectivity.
****
Below TWO are characteristics of Integration Tests but NOT unit tests:
>> They test the validity of input and output of source and target systems.
>> They must be triggered by an external client tool or event source.
It is NOT TRUE that Unit Tests must be run in a unit testing environment with dedicated Mule runtimes for the environment.
MuleSoft offers MUnit for writing Unit Tests and they run in an embedded Mule Runtime without needing any separate/ dedicated Runtimes to execute them. They also do NOT need any external connectivity as MUnit supports mocking via stubs.
https://dzone.com/articles/munit-framework

QUESTION 43
What is true about API implementations when dealing with legal regulations that require all data processing to be performed within a certain jurisdiction (such as in the USA or the EU)?

A. They must avoid using the Object Store as it depends on services deployed ONLY to the US East region
B. They must use a Jurisdiction-local external messaging system such as Active MQ rather than Anypoint MQ
C. They must te deployed to Anypoint Platform runtime planes that are managed by Anypoint Platform control planes, with both planes in the same Jurisdiction
D. They must ensure ALL data is encrypted both in transit and at rest

Correct Answer: T, H, E, Y, M, U, S, T, B, E, D, E, P, L, O, Y, E, D, T, O, A, N, Y, P, O, I, N, T, P, L, A, T, F, O, R, M, R, U, N, T, I, M, E, P, L, A, N, E, S, T, H, A, T, A, R, E, M, A, N, A, G, E, D, B, Y, A, N, Y, P, O, I, N, T, P, L, A, T, F, O, R, M, C, O, N, T, R, O, L, P, L, A, N, E, S, W, I, T, H, B, O, T, H, P, L, A, N, E, S, I, N, T, H, E, S, A, M, E, J, U, R, I, S, D, I, C, T, I, O, N
Section:
Explanation:
Answer: They must be deployed to Anypoint Platform runtime planes that are managed by Anypoint Platform control planes, with both planes in the same Jurisdiction.
****
>> As per legal regulations, all data processing to be performed within a certain jurisdiction.
Meaning, the data in USA should reside within USA and should not go out. Same way, the data in EU should reside within EU and should not go out.
>> So, just encrypting the data in transit and at rest does not help to be compliant with the rules. We need to make sure that data does not go out too.
>> The data that we are talking here is not just about the messages that are published to Anypoint MQ. It includes the apps running, transaction states, application logs, events, metric info and any other metadata. So, just replacing Anypoint
MQ with a locally hosted ActiveMQ does NOT help.
>> The data that we are talking here is not just about the key/value pairs that are stored in Object Store. It includes the messages published, apps running, transaction states, application logs, events, metric info and any other metadata. So, just avoiding using Object Store does NOT help.
>> The only option left and also the right option in the given choices is to deploy application on runtime and control planes that are both within the jurisdiction.

QUESTION 44
An API has been updated in Anypoint Exchange by its API producer from version 3.1.1 to 3.2.0 following accepted semantic versioning practices and the changes have been communicated via the API's public portal.
The API endpoint does NOT change in the new version.
How should the developer of an API client respond to this change?

A. The update should be identified as a project risk and full regression testing of the functionality that uses this API should be run

B. The API producer should be contacted to understand the change to existing functionality
C. The API producer should be requested to run the old version in parallel with the new one
D. The API client code ONLY needs to be changed if it needs to take advantage of new features

Correct Answer: D
Section:
Explanation:
Reference: https://docs.mulesoft.com/exchange/to-change-raml-version

QUESTION 45
Mule applications that implement a number of REST APIs are deployed to their own subnet that is inaccessible from outside the organization.
External business-partners need to access these APIs, which are only allowed to be invoked from a separate subnet dedicated to partners - called Partner-subnet. This subnet is accessible from the public internet, which allows these external partners to reach it.
Anypoint Platform and Mule runtimes are already deployed in Partner-subnet. These Mule runtimes can already access the APIs.
What is the most resource-efficient solution to comply with these requirements, while having the least impact on other applications that are currently using the APIs?

A. Implement (or generate) an API proxy Mule application for each of the APIs, then deploy the API proxies to the Mule runtimes
B. Redeploy the API implementations to the same servers running the Mule runtimes
C. Add an additional endpoint to each API for partner-enablement consumption
D. Duplicate the APIs as Mule applications, then deploy them to the Mule runtimes

Correct Answer: A
Section:

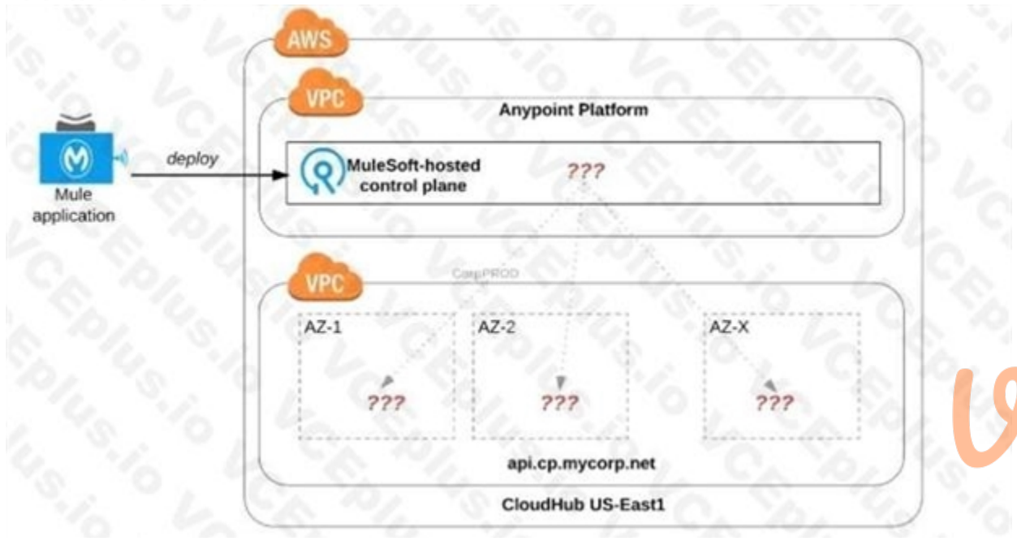
QUESTION 46
When could the API data model of a System API reasonably mimic the data model exposed by the corresponding backend system, with minimal improvements over the backend system's data model?

A. When there is an existing Enterprise Data Model widely used across the organization
B. When the System API can be assigned to a bounded context with a corresponding data model
C. When a pragmatic approach with only limited isolation from the backend system is deemed appropriate
D. When the corresponding backend system is expected to be replaced in the near future

Correct Answer: W, H, E, N, A, P, R, A, G, M, A, T, I, C, A, P, P, R, O, A, C, H, W, I, T, H, O, N, L, Y, L, I, M, I, T, E, D, I, S, O, L, A, T, I, O, N, F, R, O, M, T, H, E, B, A, C, K, E, N, D, S, Y, S, T, E, M, I, S, D, E, E, M, E, D, A, P, P, R, O, P, R, I, A, T, E
Section:
Explanation:
Answer: When a pragmatic approach with only limited isolation from the backend system is deemed appropriate.
****
General guidance w.r.t choosing Data Models:
>> If an Enterprise Data Model is in use then the API data model of System APIs should make use of data types from that Enterprise Data Model and the corresponding API implementation should translate between these data types from the Enterprise Data Model and the native data model of the backend system.
>> If no Enterprise Data Model is in use then each System API should be assigned to a Bounded Context, the API data model of System APIs should make use of data types from the corresponding Bounded Context Data Model and the corresponding API implementation should translate between these data types from the Bounded Context Data Model and the native data model of the backend system. In this scenario, the data types in the Bounded Context Data Model are defined purely in terms of their business characteristics and are typically not related to the native data model of the backend system. In other words, the translation effort may be significant.
>> If no Enterprise Data Model is in use, and the definition of a clean Bounded Context Data Model is considered too much effort, then the API data model of System APIs should make use of data types that approximately mirror those from the backend system, same semantics and naming as backend system, lightly sanitized, expose all fields needed for the given System API's functionality, but not significantly more and making good use of REST conventions.
The latter approach, i.e., exposing in System APIs an API data model that basically mirrors that of the backend system, does not provide satisfactory isolation from backend systems through the System API tier on its own. In

particular, it will typically not be possible to "swap out" a backend system without significantly changing all System APIs in front of that backend system and therefore the API implementations of all Process APIs that depend on those System APIs! This is so because it is not desirable to prolong the life of a previous backend system's data model in the form of the API data model of System APIs that now front a new backend system. The API data models of System APIs following this approach must therefore change when the backend system is replaced.
On the other hand:
>> It is a very pragmatic approach that adds comparatively little overhead over accessing the backend system directly
>> Isolates API clients from intricacies of the backend system outside the data model (protocol, authentication, connection pooling, network address, …)
>> Allows the usual API policies to be applied to System APIs
>> Makes the API data model for interacting with the backend system explicit and visible, by exposing it in the RAML definitions of the System APIs
>> Further isolation from the backend system data model does occur in the API implementations of the Process API tier

QUESTION 47
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

Correct Answer: W, O, R, K, E, R, S, A, R, E, R, A, N, D, O, M, L, Y, D, I, S, T, R, I, B, U, T, E, D, A, C, R, O, S, S, A, V, A, I, L, A, B, L, E, A, Z, S, W, I, T, H, I, N, T, H, A, T, R, E, G, I, O, N
Section:
Explanation:
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.
Reference: https://help.mulesoft.com/s/question/0D52T000051rqDj/one-cloudhub-aws-region-howcloudhub-workers-are-assigned-to-availability-zones-azs-

QUESTION 48
What is most likely NOT a characteristic of an integration test for a REST API implementation?

A. The test needs all source and/or target systems configured and accessible
B. The test runs immediately after the Mule application has been compiled and packaged
C. The test is triggered by an external HTTP request
D. The test prepares a known request payload and validates the response payload

Correct Answer: T, H, E, T, E, S, T, R, U, N, S, I, M, M, E, D, I, A, T, E, L, Y, A, F, T, E, R, T, H, E, M, U, L, E, A, P, P, L, I, C, A, T, I, O, N, H, A, S, B, E, E, N, C, O, M, P, I, L, E, D, A, N, D, P, A, C, K, A, G, E, D
Section:
Explanation:
Answer: The test runs immediately after the Mule application has been compiled and packaged **********
>> Integration tests are the last layer of tests we need to add to be fully covered.
>> These tests actually run against Mule running with your full configuration in place and are tested from external source as they work in PROD.
>> These tests exercise the application as a whole with actual transports enabled. So, external systems are affected when these tests run.
So, these tests do NOT run immediately after the Mule application has been compiled and packaged.
FYI... Unit Tests are the one that run immediately after the Mule application has been compiled and packaged.
Reference: https://docs.mulesoft.com/mule-runtime/3.9/testing-strategies#integration-testing

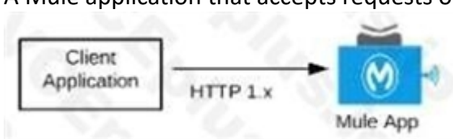
QUESTION 49
An API implementation is updated. When must the RAML definition of the API also be updated?

A. When the API implementation changes the structure of the request or response messages
B. When the API implementation changes from interacting with a legacy backend system deployed on-premises to a modern, cloud-based (SaaS) system
C. When the API implementation is migrated from an older to a newer version of the Mule runtime
D. When the API implementation is optimized to improve its average response time

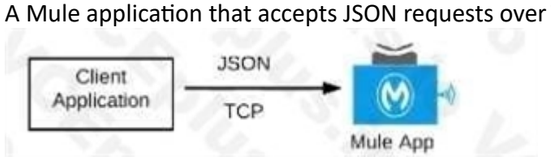
Correct Answer: W, H, E, N, T, H, E, A, P, I, I, M, P, L, E, M, E, N, T, A, T, I, O, N, C, H, A, N, G, E, S, T, H, E, S, T, R, U, C, T, U, R, E, O, F, T, H, E, R, E, Q, U, E, S, T, O, R, R, E, S, P, O, N, S, E, M, E, S, S, A, G, E, S
Section:
Explanation:
Answer: When the API implementation changes the structure of the request or response messages
****
>> RAML definition usually needs to be touched only when there are changes in the request/response schemas or in any traits on API.
>> It need not be modified for any internal changes in API implementation like performance tuning, backend system migrations etc..

QUESTION 50
What Mule application can have API policies applied by
Anypoint Platform to the endpoint exposed by that Mule application?

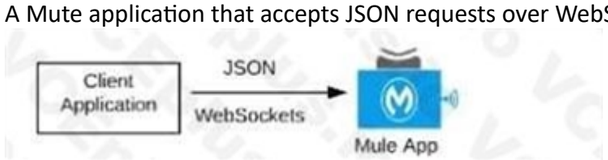
A. A Mule application that accepts requests over HTTP/1.x



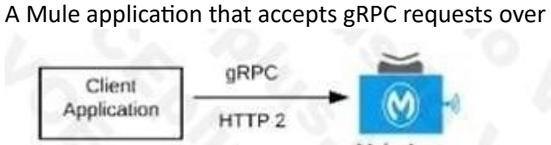
B. A Mule application that accepts JSON requests over TCP but is NOT required to provide a response



C. A Mute application that accepts JSON requests over WebSocket



D. A Mule application that accepts gRPC requests over HTTP/2



Correct Answer: O, P, T, I, O, N, A
Section:
Explanation:
Answer: Option A
****
>> Anypoint API Manager and API policies are applicable to all types of HTTP/1.x APIs.
>> They are not applicable to WebSocket APIs, HTTP/2 APIs and gRPC APIs
Reference: https://docs.mulesoft.com/api-manager/2.x/using-policies

QUESTION 51
What Anypoint Connectors support transactions?

A. Database, JMS, VM
B. Database, 3MS, HTTP
C. Database, JMS, VM, SFTP
D. Database, VM, File

Correct Answer: A
Section:

QUESTION 52
A REST API is being designed to implement a Mule application.
What standard interface definition language can be used to define REST APIs?

A. Web Service Definition Language(WSDL)
B. OpenAPI Specification (OAS)
C. YAML
D. AsyncAPI Specification

Correct Answer: B
Section:

QUESTION 53
A retail company is using an Order API to accept new orders. The Order API uses a JMS queue to submit orders to a backend order management service. The normal load for orders is being handled using two (2) CloudHub workers, each configured with 0.2 vCore. The CPU load of each CloudHub worker normally runs well below 70%. However, several times during the year the Order API gets four times (4x) the average number of orders. This causes the CloudHub worker CPU load to exceed 90% and the order submission time to exceed 30 seconds. The cause, however, is NOT the backend order management service, which still responds fast enough to meet the response SLA for the Order API. What is the MOST resource-efficient way to configure the Mule application's CloudHub deployment to help the company cope with this performance challenge?

A. Permanently increase the size of each of the two (2) CloudHub workers by at least four times (4x) to one (1) vCore
B. Use a vertical CloudHub autoscaling policy that triggers on CPU utilization greater than 70%
C. Permanently increase the number of CloudHub workers by four times (4x) to eight (8) CloudHub workers
D. Use a horizontal CloudHub autoscaling policy that triggers on CPU utilization greater than 70%

Correct Answer: U, S, E, A, H, O, R, I, Z, O, N, T, A, L, C, L, O, U, D, H, U, B, A, U, T, O, S, C, A, L, I, N, G, P, O, L, I, C, Y, T, H, A, T, T, R, I, G, G, E, R, S, O, N, C, P, U, U, T, I, L, I, Z, A, T, I, O, N, G, R, E, A, T, E, R, T, H, A, N
Section:
Explanation:
Answer: Use a horizontal CloudHub autoscaling policy that triggers on CPU utilization greater than 70%
****
The scenario in the question is very clearly stating that the usual traffic in the year is pretty well handled by the existing worker configuration with CPU running well below 70%. The problem occurs only "sometimes" occasionally when there is spike in the number of orders coming in.
So, based on above, We neither need to permanently increase the size of each worker nor need to permanently increase the number of workers. This is unnecessary as other than those "occasional" times the resources are idle and wasted.
We have two options left now. Either to use horizontal Cloudhub autoscaling policy to automatically increase the number of workers or to use vertical Cloudhub autoscaling policy to automatically increase the vCore size of each worker.
Here, we need to take two things into consideration:
1. CPU
2. Order Submission Rate to JMS Queue
>> From CPU perspective, both the options (horizontal and vertical scaling) solves the issue. Both helps to bring down the usage below 90%.
>> However, If we go with Vertical Scaling, then from Order Submission Rate perspective, as the application is still being load balanced with two workers only, there may not be much improvement in the incoming request processing rate and order submission rate to JMS queue. The throughput would be same as before. Only CPU utilization comes down.
>> But, if we go with Horizontal Scaling, it will spawn new workers and adds extra hand to increase the throughput as more workers are being load balanced now. This way we can address both CPU and Order Submission rate.
Hence, Horizontal CloudHub Autoscaling policy is the right and best answer.

QUESTION 54
A company has started to create an application network and is now planning to implement a Center for Enablement (C4E) organizational model. What key factor would lead the company to decide upon a federated rather than a centralized C4E?

A. When there are a large number of existing common assets shared by development teams
B. When various teams responsible for creating APIs are new to integration and hence need extensive training
C. When development is already organized into several independent initiatives or groups
D. When the majority of the applications in the application network are cloud based

Correct Answer: W, H, E, N, D, E, V, E, L, O, P, M, E, N, T, I, S, A, L, R, E, A, D, Y, O, R, G, A, N, I, Z, E, D, I, N, T, O, S, E, V, E, R, A, L, I, N, D, E, P, E, N, D, E, N, T, I, N, I, T, I, A, T, I, V, E, S, O, R, G, R, O, U, P, S
Section:
Explanation:
Answer: When development is already organized into several independent initiatives or groups **********
>> It would require lot of process effort in an organization to have a single C4E team coordinating with multiple already organized development teams which are into several independent initiatives. A single C4E works well with different teams having at least a common initiative. So, in this scenario, federated C4E works well instead of centralized C4E.

QUESTION 55
An organization wants MuleSoft-hosted runtime plane features (such as HTTP load balancing, zerodowntime, and horizontal and vertical scaling) in its Azure environment. What runtime planeminimizes the organization's effort to achieve these features?

A. Anypoint Runtime Fabric
B. Anypoint Platform for Pivotal Cloud Foundry
C. CloudHub
D. A hybrid combination of customer-hosted and MuleSoft-hosted Mule runtimes

Correct Answer: A, N, Y, P, O, I, N, T, R, U, N, T, I, M, E, F, A, B, R, I, C
Section:
Explanation:
Answer: Anypoint Runtime Fabric
****
>> When a customer is already having an Azure environment, It is not at all an ideal approach to go with hybrid model having some Mule Runtimes hosted on Azure and some on MuleSoft. This is unnecessary and useless.
>> CloudHub is a Mulesoft-hosted Runtime plane and is on AWS. We cannot customize to point CloudHub to customer's Azure environment.
>> Anypoint Platform for Pivotal Cloud Foundry is specifically for infrastructure provided by Pivotal Cloud Foundry
>> Anypoint Runtime Fabric is right answer as it is a container service that automates the deployment and orchestration of Mule applications and API gateways. Runtime Fabric runs within a customer-managed infrastructure on AWS,
Azure, virtual machines (VMs), and bare-metal servers.
-Some of the capabilities of Anypoint Runtime Fabric include:
-Isolation between applications by running a separate Mule runtime per application.
-Ability to run multiple versions of Mule runtime on the same set of resources.
-Scaling applications across multiple replicas.
-Automated application fail-over.
-Application management with Anypoint Runtime Manager.
Reference: https://docs.mulesoft.com/runtime-fabric/1.7/

QUESTION 56
Say, there is a legacy CRM system called CRM-Z which is offering below functions:

A. Customer creation
B. Amend details of an existing customer
C. Retrieve details of a customer
D. Suspend a customer
E. Implement a system API named customerManagement which has all the functionalities wrapped in it as various operations/resources
F. Implement different system APIs named createCustomer, amendCustomer, retrieveCustomer and suspendCustomer as they are modular and has seperation of concerns
G. Implement different system APIs named createCustomerInCRMZ, amendCustomerInCRMZ, retrieveCustomerFromCRMZ and suspendCustomerInCRMZ as they are modular and has seperation of concerns

Correct Answer: I, M, P, L, E, M, E, N, T, D, I, F, F, E, R, E, N, T, S, Y, S, T, E, M, A, P, I, S, N, A, M, E, D, C, R, E, A, T, E, C, U, S, T, O, M, E, R, A, M, E, N, D, C, U, S, T, O, M, E, R, R, E, T, R, I, E, V, E, C, U, S, T, O, M, E, R, A, N, D, S, U, S, P, E, N, D, C, U, S, T, O, M, E, R, A, S, T, H, E, Y, A, R, E, M, O, D, U, L, A, R, A, N, D, H, A, S, S, E, P, E, R, A, T, I, O, N, O, F, C, O, N, C, E, R, N, S

Section:
Explanation:
Answer: Implement different system APIs named createCustomer, amendCustomer, retrieveCustomer and suspendCustomer as they are modular and has seperation of concerns
****
>> It is quite normal to have a single API and different Verb + Resource combinations. However, this fits well for an Experience API or a Process API but not a best architecture style for System APIs. So, option with just one customerManagement API is not the best choice here.
>> The option with APIs in createCustomerInCRMZ format is next close choice w.r.t modularization and less maintenance but the naming of APIs is directly coupled with the legacy system. A better foreseen approach would be to name your APIs by abstracting the backend system names as it allows seamless replacement/migration of any backend system anytime. So, this is not the correct choice too.
>> createCustomer, amendCustomer, retrieveCustomer and suspendCustomer is the right approach and is the best fit compared to other options as they are both modular and same time got the names decoupled from backend system and it has covered all requirements a System API needs.

QUESTION 57
An Anypoint Platform organization has been configured with an external identity provider (IdP) for identity management and client management. What credentials or token must be provided to Anypoint CLI to execute commands against the
Anypoint Platform APIs?

A. The credentials provided by the IdP for identity management
B. The credentials provided by the IdP for client management
C. An OAuth 2.0 token generated using the credentials provided by the IdP for client management
D. An OAuth 2.0 token generated using the credentials provided by the IdP for identity management

Correct Answer: T, H, E, C, R, E, D, E, N, T, I, A, L, S, P, R, O, V, I, D, E, D, B, Y, T, H, E, I, D, P, F, O, R, I, D, E, N, T, I, T, Y, M, A, N, A, G, E, M, E, N, T
Section:
Explanation:
Answer: The credentials provided by the IdP for identity management **********
Reference: https://docs.mulesoft.com/runtime-manager/anypoint-platform-cli#authentication
>> There is no support for OAuth 2.0 tokens from client/identity providers to authenticate via Anypoint CLI. Only possible tokens are "bearer tokens" that too only generated using Anypoint Organization/Environment Client Id and Secret from https://anypoint.mulesoft.com/accounts/login.
Not the client credentials of client provider. So, OAuth 2.0 is not possible. More over, the token is mainly for API Manager purposes and not associated with a user. You can NOT use it to call most APIs (for example Cloudhub and etc) as per this Mulesoft Knowledge article.
>> The other option allowed by Anypoint CLI is to use client credentials. It is possible to use client credentials of a client provider but requires setting up Connected Apps in client management but such details are not given in the scenario explained in the question.
>> So only option left is to use user credentials from identify provider

QUESTION 58
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

Correct Answer: D, R, I, V, E, C, O, N, S, U, M, P, T, I, O, N, A, S, M, U, C, H, A, S, P, R, O, D, U, C, T, I, O, N, O, F, A, S, S, E, T, S, T, H, I, S, E, N, A, B, L, E, S, D, E, V, E, L, O, P, E, R, S, T, O, D, I, S, C, O, V, E, R, A, N, D, R, E, U, S, E, A, S, S, E, T, S, F, R, O, M, O, T, H, E, R, P, R, O, J, E, C, T, S, A, N, D, E, N, C, O, U, R, A, G, E, S, S, T, A, N, D, A, R, D, I, Z, A, T, I, O, N
Section:
Explanation:
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 reusablity 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

QUESTION 59
Version 3.0.1 of a REST API implementation represents time values in PST time using ISO 8601 hh:mm:ss format. The API implementation needs to be changed to instead represent time values in CEST time using ISO 8601 hh:mm:ss format. When following the semver.org semantic versioning specification, what version should be assigned to the updated API implementation?

A. 3.0.2
B. 4.0.0
C. 3.1.0
D. 3.0.1

Correct Answer:
Section:
Explanation:
Answer: 4.0.0
****
As per semver.org semantic versioning specification:
Given a version number MAJOR.MINOR.PATCH, increment the:
- MAJOR version when you make incompatible API changes.
- MINOR version when you add functionality in a backwards compatible manner.
- PATCH version when you make backwards compatible bug fixes.
As per the scenario given in the question, the API implementation is completely changing its behavior. Although the format of the time is still being maintained as hh:mm:ss and there is no change in schema w.r.t format, the API will start functioning different after this change as the times are going to come completely different.
Example: Before the change, say, time is going as 09:00:00 representing the PST. Now on, after the change, the same time will go as 18:00:00 as Central European Summer Time is 9 hours ahead of Pacific Time.
>> This may lead to some uncertain behavior on API clients depending on how they are handling the times in the API response. All the API clients need to be informed that the API functionality is going to change and will return in CEST format. So, this considered as a MAJOR change and the version of API for this new change would be 4.0.0

QUESTION 60
A company wants to move its Mule API implementations into production as quickly as possible. To protect access to all Mule application data and metadata, the company requires that all Mule applications be deployed to the company's customer-hosted infrastructure within the corporate firewall. What combination of runtime plane and control plane options meets these project lifecycle goals?

A. Manually provisioned customer-hosted runtime plane and customer-hosted control plane
B. MuleSoft-hosted runtime plane and customer-hosted control plane
C. Manually provisioned customer-hosted runtime plane and MuleSoft-hosted control plane
D. iPaaS provisioned customer-hosted runtime plane and MuleSoft-hosted control plane

Correct Answer: M, A, N, U, A, L, L, Y, P, R, O, V, I, S, I, O, N, E, D, C, U, S, T, O, M, E, R, H, O, S, T, E, D, R, U, N, T, I, M, E, P, L, A, N, E, A, N, D, C, U, S, T, O, M, E, R, H, O, S, T, E, D, C, O, N, T, R, O, L, P, L, A, N, E
Section:
Explanation:
Answer: Manually provisioned customer-hosted runtime plane and customer-hosted control plane
****
There are two key factors that are to be taken into consideration from the scenario given in the question.
>> Company requires both data and metadata to be resided within the corporate firewall
>> Company would like to go with customer-hosted infrastructure.
Any deployment model that is to deal with the cloud directly or indirectly (Mulesoft-hosted or Customer's own cloud like Azure, AWS) will have to share atleast the metadata.

Application data can be controlled inside firewall by having Mule Runtimes on customer hosted runtime plane. But if we go with Mulsoft-hosted/ Cloud-based control plane, the control plane required atleast some minimum level of metadata to be sent outside the corporate firewall.
As the customer requirement is pretty clear about the data and metadata both to be within the corporate firewall, even though customer wants to move to production as quickly as possible, unfortunately due to the nature of their security requirements, they have no other option but to go with manually provisioned customer-hosted runtime plane and customer-hosted control plane.

QUESTION 61
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?

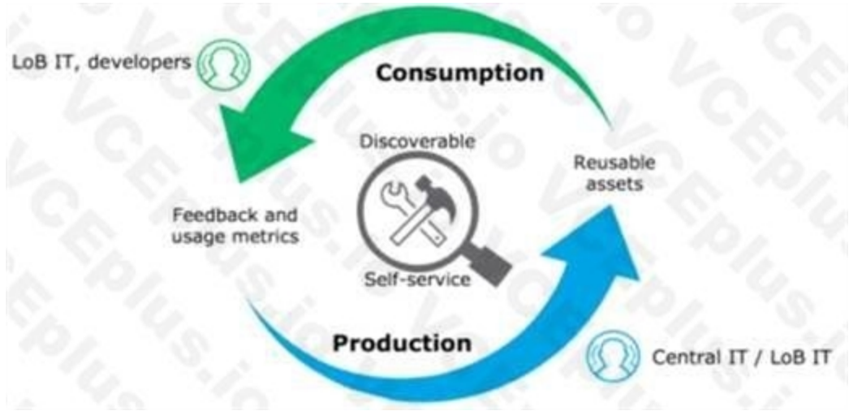
A. Integration tests
B. Performance tests
C. Functional tests (Blackbox)
D. Unit tests (Whitebox)

Correct Answer: U, N, I, T, T, E, S, T, S, W, H, I, T, E, B, O, X
Section:
Explanation:
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.

QUESTION 62
Which of the below, when used together, makes the IT Operational Model effective?

A. Create reusable assets, Do marketing on the created assets across organization, Arrange time to time LOB reviews to ensure assets are being consumed or not
B. Create reusable assets, Make them discoverable so that LOB teams can self-serve and browse the APIs, Get active feedback and usage metrics
C. Create resuable assets, make them discoverable so that LOB teams can self-serve and browse the APIs

Correct Answer: C, R, E, A, T, E, R, E, U, S, A, B, L, E, A, S, S, E, T, S, M, A, K, E, T, H, E, M, D, I, S, C, O, V, E, R, A, B, L, E, S, O, T, H, A, T, L, O, B, T, E, A, M, S, C, A, N, S, E, L, F, S, E, R, V, E, A, N, D, B, R, O, W, S, E, T, H, E, A, P, I, S, G, E, T, A, C, T, I, V, E, F, E, E, D, B, A, C, K, A, N, D, U, S, A, G, E, M, E, T, R, I, C, S
Section:
Explanation:
Answer: Create reusable assets, Make them discoverable so that LOB teams can self-serve and browse the APIs, Get active feedback and usage metrics.
****



QUESTION 63
Which of the following sequence is correct?

A. API Client implementes logic to call an API >> API Consumer requests access to API >> API Implementation routes the request to >> API
B. API Consumer requests access to API >> API Client implementes logic to call an API >> API routes the request to >> API Implementation
C. API Consumer implementes logic to call an API >> API Client requests access to API >> API Implementation routes the request to >> API
D. API Client implementes logic to call an API >> API Consumer requests access to API >> API routes the request to >> API Implementation

Correct Answer: A, P, I, C, O, N, S, U, M, E, R, R, E, Q, U, E, S, T, S, A, C, C, E, S, S, T, O, A, P, I, A, P, I, C, L, I, E, N, T, I, M, P, L, E, M, E, N, T, E, S, L, O, G, I, C, T, O, C, A, L, L, A, N, A, P, I
Section:
Explanation:
Answer: API Consumer requests access to API >> API Client implementes logic to call an API
>> API routes the request to >> API Implementation **********
>> API consumer does not implement any logic to invoke APIs. It is just a role. So, the option stating "API Consumer implementes logic to call an API" is INVALID.
>> API Implementation does not route any requests. It is a final piece of logic where functionality of target systems is exposed. So, the requests should be routed to the API implementation by some other entity. So, the options stating "API
Implementation routes the request to >> API" is INVALID
>> The statements in one of the options are correct but sequence is wrong. The sequence is given as "API Client implementes logic to call an API >> API Consumer requests access to API >> API routes the request to >> API Implementation". Here, the statements in the options are VALID but sequence is WRONG.
>> Right option and sequence is the one where API consumer first requests access to API on Anypoint Exchange and obtains client credentials. API client then writes logic to call an API by using the access client credentials requested by
API consumer and the requests will be routed to API implementation via the API which is managed by API Manager.

QUESTION 64
An organization has created an API-led architecture that uses various API layers to integrate mobile clients with a backend system. The backend system consists of a number of specialized components and can be accessed via a REST API. The process and experience APIs share the same boundedcontext model that is different from the backend data model. What additional canonical models, bounded-context models, or anti-corruption layers are best added to this architecture to help process data consumed from the backend system?

A. Create a bounded-context model for every layer and overlap them when the boundary contexts overlap, letting API developers know about the differences between upstream and downstream data models
B. Create a canonical model that combines the backend and API-led models to simplify and unify data models, and minimize data transformations.
C. Create a bounded-context model for the system layer to closely match the backend data model, and add an anti-corruption layer to let the different bounded contexts cooperate across the system and process layers
D. Create an anti-corruption layer for every API to perform transformation for every data model to match each other, and let data simply travel between APIs to avoid the complexity and overhead of building canonical models

Correct Answer: C, R, E, A, T, E, A, B, O, U, N, D, E, D, C, O, N, T, E, X, T, M, O, D, E, L, F, O, R, T, H, E, S, Y, S, T, E, M, L, A, Y, E, R, T, O, C, L, O, S, E, L, Y, M, A, T, C, H, T, H, E, B, A, C, K, E, N, D, D, A, T, A, M, O, D, E, L, A, N, D, A, D,

D, A, N, A, N, T, I, C, O, R, R, U, P, T, I, O, N, L, A, Y, E, R, T, O, L, E, T, T, H, E, D, I, F, F, E, R, E, N, T, B, O, U, N, D, E, D, C, O, N, T, E, X, T, S, C, O, O, P, E, R, A, T, E, A, C, R, O, S, S, T, H, E, S, Y, S, T, E, M, A, N, D, P, R, O, C, E, S, S, L, A, Y, E, R, S
Section:
Explanation:
Answer: Create a bounded-context model for the system layer to closely match the backend data model, and add an anti-corruption layer to let the different bounded contexts cooperate across the system and process layers
****
>> Canonical models are not an option here as the organization has already put in efforts and created bounded-context models for Experience and Process APIs.
>> Anti-corruption layers for ALL APIs is unnecessary and invalid because it is mentioned that experience and process APIs share same bounded-context model. It is just the System layer APIs that need to choose their approach now.
>> So, having an anti-corruption layer just between the process and system layers will work well. Also to speed up the approach, system APIs can mimic the backend system data model.

QUESTION 65
An API client calls one method from an existing API implementation. The API implementation is later updated. What change to the API implementation would require the API client's invocation logic to also be updated?

A. When the data type of the response is changed for the method called by the API client
B. When a new method is added to the resource used by the API client
C. When a new required field is added to the method called by the API client
D. When a child method is added to the method called by the API client

Correct Answer: W, H, E, N, A, N, E, W, R, E, Q, U, I, R, E, D, F, I, E, L, D, I, S, A, D, D, E, D, T, O, T, H, E, M, E, T, H, O, D, C, A, L, L, E, D, B, Y, T, H, E, A, P, I, C, L, I, E, N, T
Section:
Explanation:
Answer: When a new required field is added to the method called by the API client **********
>> Generally, the logic on API clients need to be updated when the API contract breaks.
>> When a new method or a child method is added to an API , the API client does not break as it can still continue to use its existing method. So these two options are out.
>> We are left for two more where "datatype of the response if changed" and "a new required field is added".
>> Changing the datatype of the response does break the API contract. However, the question is insisting on the "invocation" logic and not about the response handling logic. The API client can still invoke the API successfully and receive the response but the response will have a different datatype for some field.
>> Adding a new required field will break the API's invocation contract. When adding a new required field, the API contract breaks the RAML or API spec agreement that the API client/API consumer and API provider has between them. So this requires the API client invocation logic to also be updated.

QUESTION 66
Traffic is routed through an API proxy to an API implementation. The API proxy is managed by API Manager and the API implementation is deployed to a CloudHub VPC using Runtime Manager. API policies have been applied to this API.
In this deployment scenario, at what point are the API policies enforced on incoming API client requests?

A. At the API proxy
B. At the API implementation
C. At both the API proxy and the API implementation
D. At a MuleSoft-hosted load balancer

Correct Answer: A, T, T, H, E, A, P, I, P, R, O, X, Y
Section:
Explanation:
Answer: At the API proxy
****
>> API Policies can be enforced at two places in Mule platform.
>> One - As an Embedded Policy enforcement in the same Mule Runtime where API implementation is running.
>> Two - On an API Proxy sitting in front of the Mule Runtime where API implementation is running.

>> As the deployment scenario in the question has API Proxy involved, the policies will be enforced at the API Proxy.

QUESTION 67
Once an API Implementation is ready and the API is registered on API Manager, who should request the access to the API on Anypoint Exchange?

A. None
B. Both
C. API Client
D. API Consumer

Correct Answer: A, P, I, C, O, N, S, U, M, E, R
Section:
Explanation:
Answer: API Consumer
****
>> API clients are piece of code or programs that use the client credentials of API consumer but does not directly interact with Anypoint Exchange to get the access
>> API consumer is the one who should get registered and request access to API and then API client needs to use those client credentials to hit the APIs So, API consumer is the one who needs to request access on the API from Anypoint
Exchange

QUESTION 68
In which layer of API-led connectivity, does the business logic orchestration reside?

A. System Layer
B. Experience Layer
C. Process Layer

Correct Answer: P, R, O, C, E, S, S, L, A, Y, E, R
Section:
Explanation:
Answer: Process Layer
****
>> Experience layer is dedicated for enrichment of end user experience. This layer is to meet the needs of different API clients/ consumers.
>> System layer is dedicated to APIs which are modular in nature and implement/ expose various individual functionalities of backend systems
>> Process layer is the place where simple or complex business orchestration logic is written by invoking one or many System layer modular APIs So, Process Layer is the right answer.

QUESTION 69
A system API is deployed to a primary environment as well as to a disaster recovery (DR) environment, with different DNS names in each environment. A process API is a client to the system API and is being rate limited by the system API, with different limits in each of the environments. The system API's DR environment provides only 20% of the rate limiting offered by the primary environment. What is the best API fault-tolerant invocation strategy to reduce overall errors in the process API, given these conditions and constraints?

A. Invoke the system API deployed to the primary environment; add timeout and retry logic to the process API to avoid intermittent failures; if it still fails, invoke the system API deployed to the DR environment
B. Invoke the system API deployed to the primary environment; add retry logic to the process API to handle intermittent failures by invoking the system API deployed to the DR environment
C. In parallel, invoke the system API deployed to the primary environment and the system API deployed to the DR environment; add timeout and retry logic to the process API to avoid intermittent failures; add logic to the process API to combine the results
D. Invoke the system API deployed to the primary environment; add timeout and retry logic to the process API to avoid intermittent failures; if it still fails, invoke a copy of the process API deployed to the DR environment

Correct Answer: I, N, V, O, K, E, T, H, E, S, Y, S, T, E, M, A, P, I, D, E, P, L, O, Y, E, D, T, O, T, H, E, P, R, I, M, A, R, Y, E, N, V, I, R, O, N, M, E, N, T, A, D, D, T, I, M, E, O, U, T, A, N, D, R, E, T, R, Y, L, O, G, I, C, T, O, T, H, E, P, R, O, C, E, S,

S, A, P, I, T, O, A, V, O, I, D, I, N, T, E, R, M, I, T, T, E, N, T, F, A, I, L, U, R, E, S, I, F, I, T, S, T, I, L, L, F, A, I, L, S, I, N, V, O, K, E, T, H, E, S, Y, S, T, E, M, A, P, I, D, E, P, L, O, Y, E, D, T, O, T, H, E, D, R, E, N, V, I, R, O, N, M, E, N, T
Section:
Explanation:
Answer: Invoke the system API deployed to the primary environment; add timeout and retry logic to the process API to avoid intermittent failures; if it still fails, invoke the system API deployed to the DR environment
********** There is one important consideration to be noted in the question which is - System API in DR environment provides only 20% of the rate limiting offered by the primary environment. So, comparitively, very less calls will be allowed into the DR environment API opposed to its primary environment. With this in mind, lets analyse what is the right and best fault-tolerant invocation strategy.
1. Invoking both the system APIs in parallel is definitely NOT a feasible approach because of the 20% limitation we have on DR environment. Calling in parallel every time would easily and quickly exhaust the rate limits on DR environment and may not give chance to genuine intermittent error scenarios to let in during the time of need.
2. Another option given is suggesting to add timeout and retry logic to process API while invoking primary environment's system API. This is good so far. However, when all retries failed, the option is suggesting to invoke the copy of process
API on DR environment which is not right or recommended.
Only system API is the one to be considered for fallback and not the whole process API. Process APIs usually have lot of heavy orchestration calling many other APIs which we do not want to repeat again by calling DR's process API. So this option is NOT right.
3. One more option given is suggesting to add the retry (no timeout) logic to process API to directly retry on DR environment's system API instead of retrying the primary environment system API first.
This is not at all a proper fallback. A proper fallback should occur only after all retries are performed and exhausted on Primary environment first. But here, the option is suggesting to directly retry fallback API on first failure itself without trying main API. So, this option is NOT right too.
This leaves us one option which is right and best fit.
- Invoke the system API deployed to the primary environment
- Add Timeout and Retry logic on it in process API
- If it fails even after all retries, then invoke the system API deployed to the DR environment.

QUESTION 70
A company uses a hybrid Anypoint Platform deployment model that combines the EU control plane with customer-hosted Mule runtimes. After successfully testing a Mule API implementation in the Staging environment, the Mule API implementation is set with environment-specific properties and must be promoted to the Production environment. What is a way that MuleSoft recommends to configure the Mule API implementation and automate its promotion to the Production environment?

A. Bundle properties files for each environment into the Mule API implementation's deployable archive, then promote the Mule API implementation to the Production environment using Anypoint CLI or the Anypoint Platform REST APIsB.
B. Modify the Mule API implementation's properties in the API Manager Properties tab, then promote the Mule API implementation to the Production environment using API Manager
C. Modify the Mule API implementation's properties in Anypoint Exchange, then promote the Mule API implementation to the Production environment using Runtime Manager
D. Use an API policy to change properties in the Mule API implementation deployed to the Staging environment and another API policy to deploy the Mule API implementation to the Production environment

Correct Answer: B, U, N, D, L, E, P, R, O, P, E, R, T, I, E, S, F, I, L, E, S, F, O, R, E, A, C, H, E, N, V, I, R, O, N, M, E, N, T, I, N, T, O, T, H, E, M, U, L, E, A, P, I, I, M, P, L, E, M, E, N, T, A, T, I, O, N, S, D, E, P, L, O, Y, A, B, L, E, A, R, C, H, I, V, E, T, H, E, N, P, R, O, M, O, T, E, T, H, E, M, U, L, E, A, P, I, I, M, P, L, E, M, E, N, T, A, T, I, O, N, T, O, T, H, E, P, R, O, D, U, C, T, I, O, N, E, N, V, I, R, O, N, M, E, N, T, U, S, I, N, G, A, N, Y, P, O, I, N, T, C, L, I, O, R, T, H, E, A, N, Y, P, O, I, N, T, P, L, A, T, F, O, R, M
Section:
Explanation:
Answer: Bundle properties files for each environment into the Mule API implementation's deployable archive, then promote the Mule API implementation to the Production environment using Anypoint CLI or the Anypoint Platform
REST APIs **********
>> Anypoint Exchange is for asset discovery and documentation. It has got no provision to modify the properties of Mule API implementations at all.
>> API Manager is for managing API instances, their contracts, policies and SLAs. It has also got no provision to modify the properties of API implementations.
>> API policies are to address Non-functional requirements of APIs and has again got no provision to modify the properties of API implementations.
So, the right way and recommended way to do this as part of development practice is to bundle properties files for each environment into the Mule API implementation and just point and refer to respective file per environment.

QUESTION 71
An organization wants to make sure only known partners can invoke the organization's APIs. To achieve this security goal, the organization wants to enforce a Client ID Enforcement policy in API Manager so that only

registered partner applications can invoke the organization's APIs. In what type of API implementation does MuleSoft recommend adding an API proxy to enforce the Client ID Enforcement policy, rather than embedding the policy directly in the application's JVM?

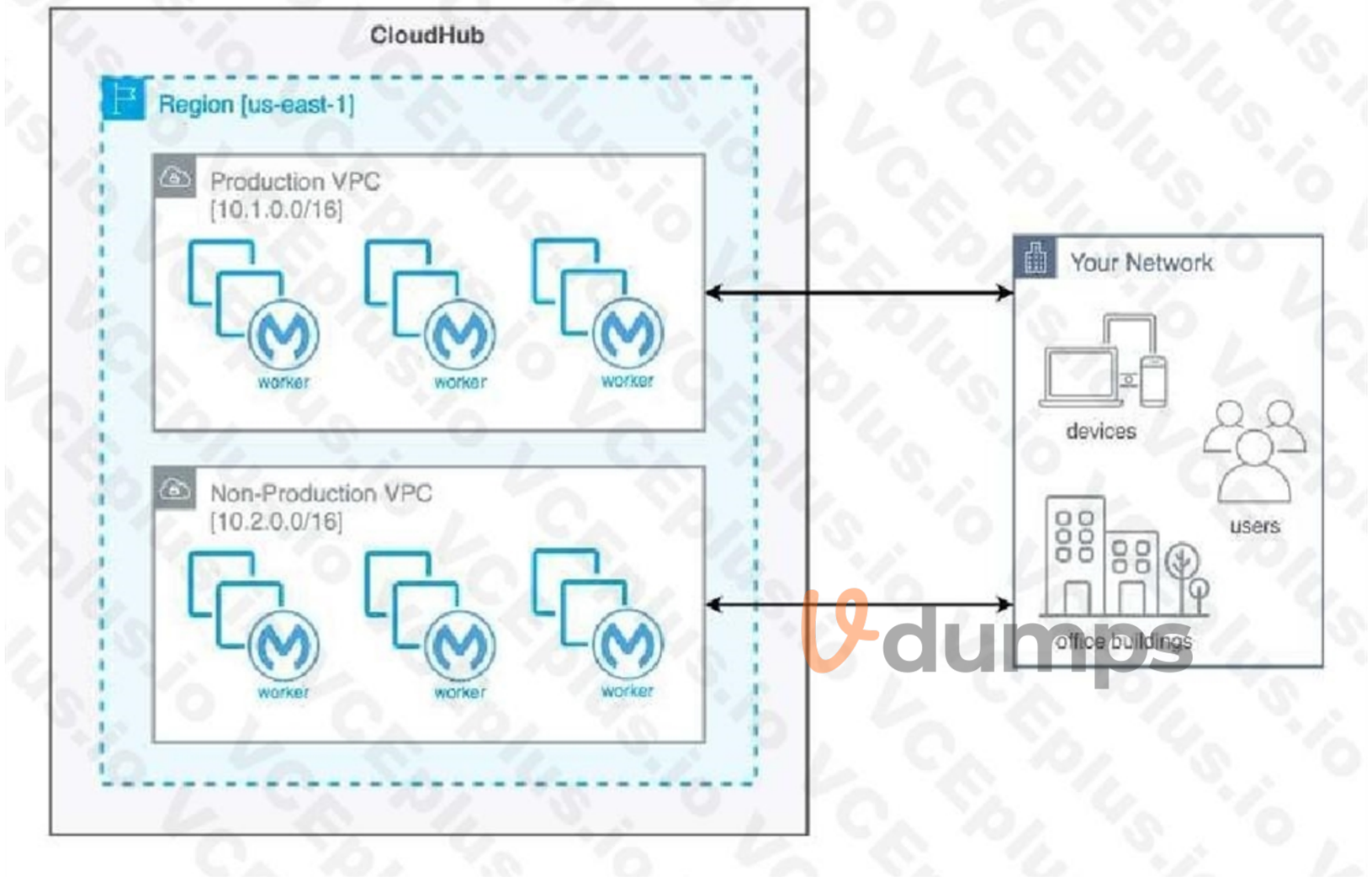
A. A Mule 3 application using APIkit
B. A Mule 3 or Mule 4 application modified with custom Java code
C. A Mule 4 application with an API specification
D. A Non-Mule application

Correct Answer: A, N, O, N, M, U, L, E, A, P, P, L, I, C, A, T, I, O, N
Section:
Explanation:
Answer: A Non-Mule application
****
>> All type of Mule applications (Mule 3/ Mule 4/ with APIkit/ with Custom Java Code etc) running on Mule Runtimes support the Embedded Policy Enforcement on them.
>> The only option that cannot have or does not support embedded policy enforcement and must have API Proxy is for Non-Mule Applications.
So, Non-Mule application is the right answer.

QUESTION 72
A company requires Mule applications deployed to CloudHub to be isolated between non-production and production environments. This is so Mule applications deployed to non-production environments can only access backend systems running in their customer-hosted non-production environment, and so Mule applications deployed to production environments can only access backend systems running in their customer-hosted production environment. How does MuleSoft recommend modifying Mule applications, configuring environments, or changing infrastructure to support this type of per-environment isolation between Mule applications and backend systems?

A. Modify properties of Mule applications deployed to the production Anypoint Platform environments to prevent access from non-production Mule applications
B. Configure firewall rules in the infrastructure inside each customer-hosted environment so that only IP addresses from the corresponding Anypoint Platform environments are allowed to communicate with corresponding backend systems
C. Create non-production and production environments in different Anypoint Platform business groups
D. Create separate Anypoint VPCs for non-production and production environments, then configure connections to the backend systems in the corresponding customer-hosted environments

Correct Answer: A, A, A, A, A, A, B, C, C, C, C, C, C, C, C, C, C, D, D, D, D, D, D, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, E, F, F, G, G, H, H, H, H, I, I, I, I, I, I, I, I, I, K, M, M, M, M, N, N, N, N, N, N, N, N, N, N, N, N, N, N, N, N, N, N, N, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, P, P, P, P, P, P, R, R, R, R, R, R, R, R, R, R, R, S, S, S, S, S, S, S, S, S, S, S, T, T, T, T, T, T, T, T, T, T, T, T, T, T, T, U, U, U, U, V, V, V, Y, Y
Section:
Explanation:
Answer: Create separate Anypoint VPCs for non-production and production environments, then configure connections to the backend systems in the corresponding customer-hosted environments.
****
>> Creating different Business Groups does NOT make any difference w.r.t accessing the non-prod and prod customer-hosted environments. Still they will be accessing from both Business Groups unless process network restrictions are put in place.
>> We need to modify or couple the Mule Application Implementations with the environment. In fact, we should never implements application coupled with environments by binding them in the properties. Only basic things like endpoint URL etc should be bundled in properties but not environment level access restrictions.
>> IP addresses on CloudHub are dynamic until unless a special static addresses are assigned. So it is not possible to setup firewall rules in customer-hosted infrastrcture. More over, even if static IP addresses are assigned, there could be 100s of applications running on cloudhub and setting up rules for all of them would be a hectic task, non-maintainable and definitely got a good practice.
>> The best practice recommended by Mulesoft (In fact any cloud provider), is to have your Anypoint VPCs seperated for Prod and Non-Prod and perform the VPC peering or VPN tunneling for these Anypoint VPCs to respective Prod and
Non-Prod customer-hosted environment networks.
Reference: https://docs.mulesoft.com/runtime-manager/virtual-private-cloud



QUESTION 73
A company has created a successful enterprise data model (EDM). The company is committed to building an application network by adopting modern APIs as a core enabler of the company's IT operating model. At what API tiers (experience, process, system) should the company require reusing the EDM when designing modern API data models?

A. At the experience and process tiers
B. At the experience and system tiers
C. At the process and system tiers
D. At the experience, process, and system tiers

Correct Answer: A, A, C, D, E, E, E, E, H, I, M, N, O, P, R, R, S, S, S, S, S, T, T, T, T, Y
Section:
Explanation:
Answer: At the process and system tiers
****

>> Experience Layer APIs are modeled and designed exclusively for the end user's experience. So, the data models of experience layer vary based on the nature and type of such API consumer. For example, Mobile consumers will need light-weight data models to transfer with ease on the wire, where as web-based consumers will need detailed data models to render most of the info on web pages, so on. So, enterprise data models fit for the purpose of canonical models but not of good use for experience APIs.
>> That is why, EDMs should be used extensively in process and system tiers but NOT in experience tier.

QUESTION 74
The application network is recomposable: it is built for change because it "bends but does not break"

A. TRUE
B. FALSE

Correct Answer: A
Section:
Explanation:
****
>> Application Network is a disposable architecture.
>> Which means, it can be altered without disturbing entire architecture and its components.
>> It bends as per requirements or design changes but does not break
Reference: https://www.mulesoft.com/resources/api/what-is-an-application-network

QUESTION 75
A system API has a guaranteed SLA of 100 ms per request. The system API is deployed to a primary environment as well as to a disaster recovery (DR) environment, with different DNS names in each environment. An upstream process API invokes the system API and the main goal of this process API is to respond to client requests in the least possible time. In what order should the system APIs be invoked, and what changes should be made in order to speed up the response time for requests from the process API?

A. In parallel, invoke the system API deployed to the primary environment and the system API deployed to the DR environment, and ONLY use the first response
B. In parallel, invoke the system API deployed to the primary environment and the system API deployed to the DR environment using a scatter-gather configured with a timeout, and then merge the responses
C. Invoke the system API deployed to the primary environment, and if it fails, invoke the system API deployed to the DR environment
D. Invoke ONLY the system API deployed to the primary environment, and add timeout and retry logic to avoid intermittent failures

Correct Answer: I, N, P, A, R, A, L, L, E, L, I, N, V, O, K, E, T, H, E, S, Y, S, T, E, M, A, P, I, D, E, P, L, O, Y, E, D, T, O, T, H, E, P, R, I, M, A, R, Y, E, N, V, I, R, O, N, M, E, N, T, A, N, D, T, H, E, S, Y, S, T, E, M, A, P, I, D, E, P, L, O, Y, E, D, T, O, T, H, E, D, R, E, N, V, I, R, O, N, M, E, N, T, A, N, D, O, N, L, Y, U, S, E, T, H, E, F, I, R, S, T, R, E, S, P, O, N, S, E
Section:
Explanation:
Answer: In parallel, invoke the system API deployed to the primary environment and the system API deployed to the DR environment, and ONLY use the first response.
****
>> The API requirement in the given scenario is to respond in least possible time.
>> The option that is suggesting to first try the API in primary environment and then fallback to API in DR environment would result in successful response but NOT in least possible time. So, this is NOT a right choice of implementation for given requirement.
>> Another option that is suggesting to ONLY invoke API in primary environment and to add timeout and retries may also result in successful response upon retries but NOT in least possible time. So, this is also NOT a right choice of implementation for given requirement.
>> One more option that is suggesting to invoke API in primary environment and API in DR environment in parallel using Scatter-Gather would result in wrong API response as it would return merged results and moreover, Scatter-Gather does things in parallel which is true but still completes its scope only on finishing all routes inside it. So again, NOT a right choice of implementation for given requirement The Correct choice is to invoke the API in primary environment and the API in DR environment parallelly, and using ONLY the first response received from one of them.

QUESTION 76
Which of the following best fits the definition of API-led connectivity?

A. API-led connectivity is not just an architecture or technology but also a way to organize people and processes for efficient IT delivery in the organization

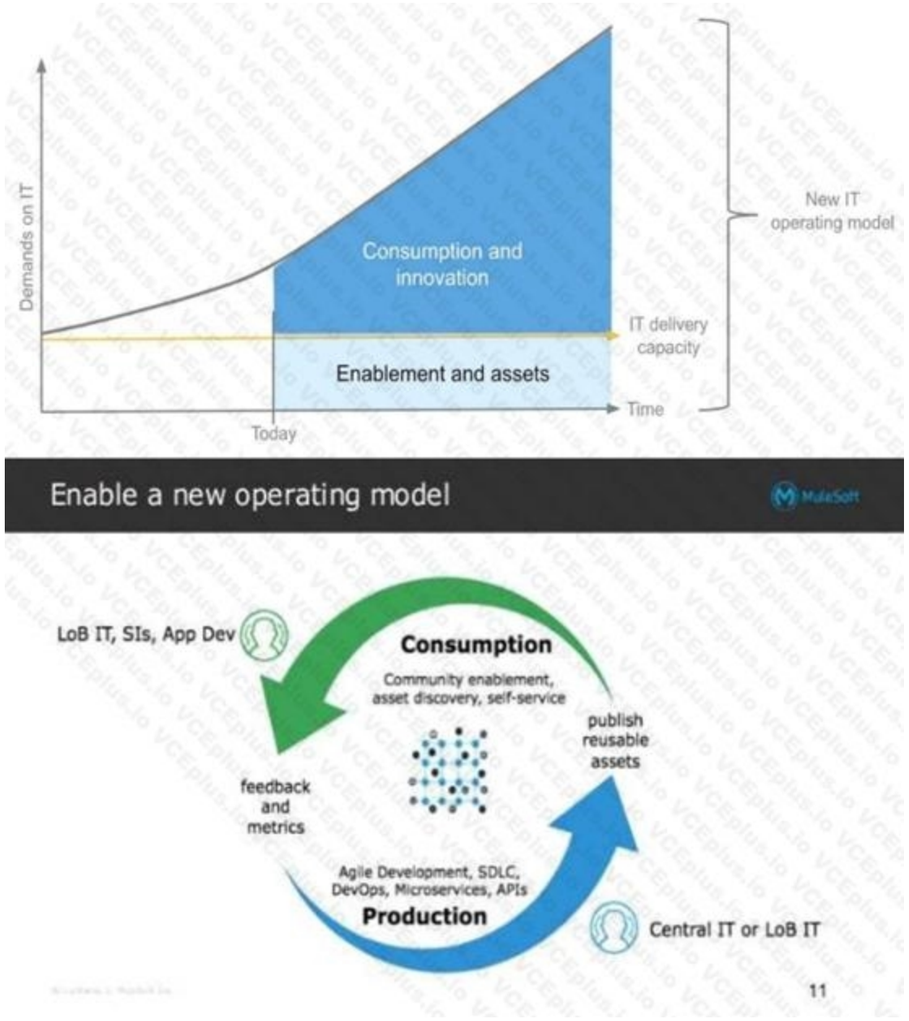
B. API-led connectivity is a 3-layered architecture covering Experience, Process and System layers
C. API-led connectivity is a technology which enabled us to implement Experience, Process and System layer based APIs

Correct Answer: A, P, I, L, E, D, C, O, N, N, E, C, T, I, V, I, T, Y, I, S, N, O, T, J, U, S, T, A, N, A, R, C, H, I, T, E, C, T, U, R, E, O, R, T, E, C, H, N, O, L, O, G, Y, B, U, T, A, L, S, O, A, W, A, Y, T, O, O, R, G, A, N, I, Z, E, P, E, O, P, L, E, A, N, D, P, R, O, C, E, S, S, E, S, F, O, R, E, F, F, I, C, I, E, N, T, I, T, D, E, L, I, V, E, R, Y, I, N, T, H, E, O, R, G, A, N, I, Z, A, T, I, O, N
Section:
Explanation:
Answer: API-led connectivity is not just an architecture or technology but also a way to organize people and processes for efficient IT delivery in the organization.
****
Reference: https://blogs.mulesoft.com/dev/api-dev/what-is-api-led-connectivity/

QUESTION 77
What are the major benefits of MuleSoft proposed IT Operating Model?

A. 1. Decrease the IT delivery gap
B. Meet various business demands without increasing the IT capacity
C. Focus on creation of reusable assets first. Upon finishing creation of all the possible assets then inform the LOBs in the organization to start using them
D. 1. Decrease the IT delivery gap
E. Meet various business demands by increasing the IT capacity and forming various IT departments 3. Make consumption of assets at the rate of production
F. 1. Decrease the IT delivery gap
G. Meet various business demands without increasing the IT capacity
H. Make consumption of assets at the rate of production

Correct Answer:
Section:
Explanation:
Answer:
1. Decrease the IT delivery gap
2. Meet various business demands without increasing the IT capacity
3. Make consumption of assets at the rate of production.
****
Reference: https://www.youtube.com/watch?v=U0FpYMnMjmM


Demands on IT
Consumption and innovation
Enablement and assets
IT delivery capacity
New IT operating model
Time
Today
Enable a new operating model
MuleSoft
LoB IT, SIs, App Dev
Consumption
Community enablement, asset discovery, self-service
publish reusable assets
feedback and metrics
Agile Development, SDLC, DevOps, Microservices, APIs
Production
Central IT or LoB IT
11