

Exam Questions MCPA-Level-1

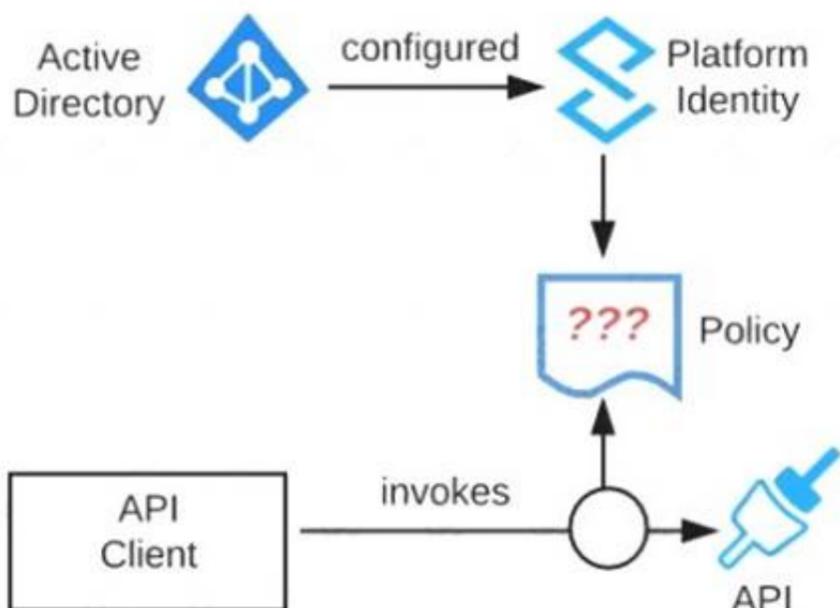
MuleSoft Certified Platform Architect - Level 1

<https://www.2passeasy.com/dumps/MCPA-Level-1/>



NEW QUESTION 1

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 effectively restrict 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

Answer: A

Explanation:

Correct 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.

NEW QUESTION 2

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

Answer: C

Explanation:

Correct 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.

NEW QUESTION 3

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

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

Answer: C

Explanation:

Correct 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.

NEW QUESTION 4

True or False. We should always make sure that the APIs being designed and developed are self-servable even if it needs more man-day effort and resources.

- A. FALSE
- B. TRUE

Answer: B

Explanation:

Correct Answer
TRUE

>> As per MuleSoft proposed IT Operating Model, designing APIs and making sure that they are discoverable and self-servable is VERY VERY IMPORTANT and decides the success of an API and its application network.

NEW QUESTION 5

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

Answer: B

Explanation:

Correct 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:

exp-salesorder-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.

NEW QUESTION 6

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

Answer: A

Explanation:

Correct 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>

NEW QUESTION 7

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

Answer: C

Explanation:

Correct 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.

NEW QUESTION 8

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

Answer: C

Explanation:

Correct 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.

NEW QUESTION 9

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

Answer: A

Explanation:

Correct 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.

NEW QUESTION 10

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

- A. * 1. Decrease the IT delivery gap* 2. Meet various business demands without increasing the IT capacity* 3. Focus on creation of reusable assets first
- B. Upon finishing creation of all the possible assets then inform the LOBs in the organization to start using them
- C. * 1. Decrease the IT delivery gap* 2. Meet various business demands by increasing the IT capacity and forming various IT departments* 3. Make consumption of assets at the rate of production
- D. * 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

Answer: C

Explanation:

Correct 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.

NEW QUESTION 10

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

Answer: C

Explanation:

Correct 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

NEW QUESTION 11

A Mule application exposes an HTTPS endpoint and is deployed to three CloudHub workers that do not use static IP addresses. The Mule application expects a high volume of client requests in short time periods. What is the most cost-effective infrastructure component that should be used to serve the high volume of client requests?

- A. A customer-hosted load balancer
- B. The CloudHub shared load balancer
- C. An API proxy
- D. Runtime Manager autoscaling

Answer: B

Explanation:

Correct Answer

The CloudHub shared load balancer

***** The scenario in this question can be split as below:

>> There are 3 CloudHub workers (So, there are already good number of workers to handle high volume of requests)

>> The workers are not using static IP addresses (So, one CANNOT use customer load-balancing solutions without static IPs)

>> Looking for most cost-effective component to load balance the client requests among the workers. Based on the above details given in the scenario:

>> Runtime autoscaling is NOT at all cost-effective as it incurs extra cost. Most over, there are already 3 workers running which is a good number.

>> We cannot go for a customer-hosted load balancer as it is also NOT most cost-effective (needs custom load balancer to maintain and licensing) and same time the Mule App is not having Static IP Addresses which limits from going with custom load balancing.

>> An API Proxy is irrelevant there as it has no role to play w.r.t handling high volumes or load balancing. So, the only right option to go with and fits the purpose of scenario being most cost-effective is - using a CloudHub Shared Load Balancer.

NEW QUESTION 15

A Mule application exposes an HTTPS endpoint and is deployed to the CloudHub Shared Worker Cloud. 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

Answer: D

Explanation:

Correct Answer 8082

>> 8091 and 8092 ports are to be used when keeping your HTTP and HTTPS app private to the LOCAL VPC 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 LB So, 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-An>

<https://help.mulesoft.com/s/question/0D52T00004mXXULSA4/multiple-http-listeners-on-cloudhub-one-with-p>

NEW QUESTION 16

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

Answer: A

Explanation:

Correct 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..

NEW QUESTION 20

What are 4 important Platform Capabilities offered by Anypoint Platform?

- A. API Versioning, API Runtime Execution and Hosting, API Invocation, API Consumer Engagement
- B. API Design and Development, API Runtime Execution and Hosting, API Versioning, API Deprecation
- C. API Design and Development, API Runtime Execution and Hosting, API Operations and Management, API Consumer Engagement
- D. API Design and Development, API Deprecation, API Versioning, API Consumer Engagement

Answer: C

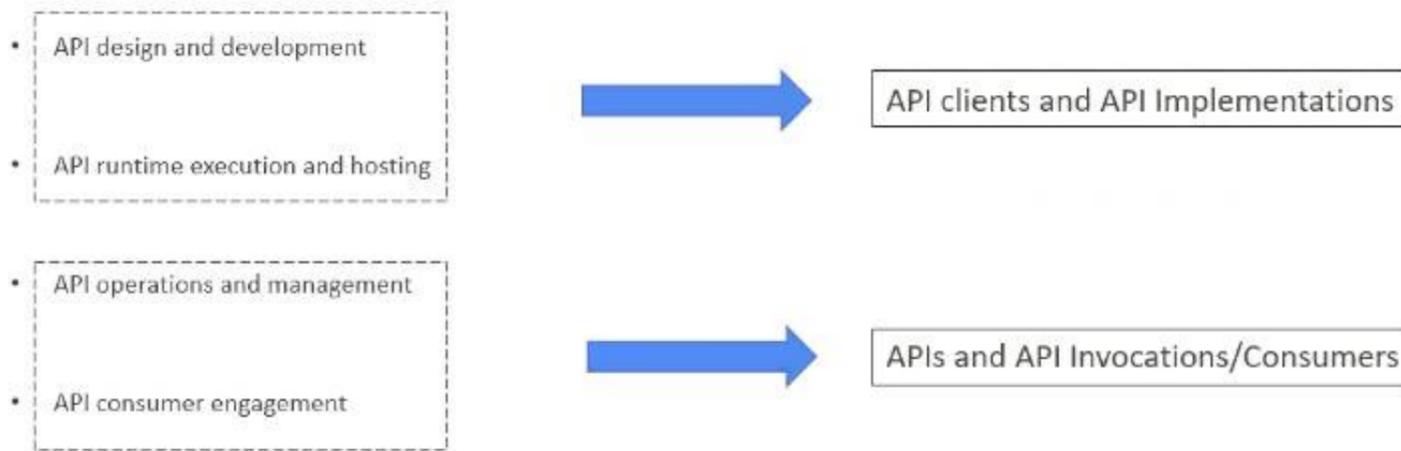
Explanation:

Correct Answer

API Design and Development, API Runtime Execution and Hosting, API Operations and Management, API Consumer Engagement

- >> API Design and Development - Anypoint Studio, Anypoint Design Center, Anypoint Connectors
- >> API Runtime Execution and Hosting - Mule Runtimes, CloudHub, Runtime Services
- >> API Operations and Management - Anypoint API Manager, Anypoint Exchange
- >> API Consumer Management - API Contracts, Public Portals, Anypoint Exchange, API Notebooks

Platform Capabilities



© Prasad Pokala

NEW QUESTION 25

What is true about the technology architecture of Anypoint VPCs?

- A. The private IP address range of an Anypoint VPC is automatically chosen by CloudHub
- B. Traffic between Mule applications deployed to an Anypoint VPC and on-premises systems can stay within a private network
- C. Each CloudHub environment requires a separate Anypoint VPC
- D. VPC peering can be used to link the underlying AWS VPC to an on-premises (non AWS) private network

Answer: B

Explanation:

Correct Answer

Traffic between Mule applications deployed to an Anypoint VPC and on-premises systems can stay within a private network

>> The private IP address range of an Anypoint VPC is NOT automatically chosen by CloudHub. It is chosen by us at the time of creating VPC using thr CIDR blocks.

CIDR Block: The size of the Anypoint VPC in Classless Inter-Domain Routing (CIDR) notation.

For example, if you set it to 10.111.0.0/24, the Anypoint VPC is granted 256 IP addresses from 10.111.0.0 to 10.111.0.255.

Ideally, the CIDR Blocks you choose for the Anypoint VPC come from a private IP space, and should not overlap with any other Anypoint VPC's CIDR Blocks, or any CIDR Blocks in use in your corporate network.

← Create VPC

[Learn more about VPCs](#)

General Information

Name: vpc1

Region: US East (N. Virginia)

CIDR Block: 10.0.0.0/16

Environments: Design

Set as default VPC

Business Groups: MyBusinessGroup (MyOrg)

that each CloudHub environment requires a separate Anypoint VPC. Once an Anypoint VPC is created, we can choose a same VPC by multiple environments. However, it is generally a best and recommended practice to always have separate Anypoint VPCs for Non-Prod and Prod environments. >> We use Anypoint VPN to link the underlying AWS VPC to an on-premises (non AWS) private network. NOT VPC Peering.

NEW QUESTION 29

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

Answer: A

NEW QUESTION 31

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

Answer: C

Explanation:

Correct 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 Anypoint Platform itself. We need not manage it explicitly in the API implementation and in fact we CANNOT manage 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 be managed EXCLUSIVELY by the API implementation. Anypoint Platform does NOT do this when using SLBs.

NEW QUESTION 34

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

Answer: C

Explanation:

Correct 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.

NEW QUESTION 37

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 API whenever the Order API is unavailable
- D. Set an option in the HTTP Requester component that invokes the Order API to instead invoke a fallback API whenever an HTTP 4xx or 5xx response status code is returned from the Order API

Answer: A

Explanation:

Correct 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 of 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 a fallback 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.

NEW QUESTION 39

Due to a limitation in the backend system, a system API can only handle up to 500 requests per second. What is the best type of API policy to apply to the system API to avoid overloading the backend system?

- A. Rate limiting
- B. HTTP caching
- C. Rate limiting - SLA based
- D. Spike control

Answer: D

Explanation:

Correct Answer

Spike control

>> First things first, HTTP Caching policy is for purposes different than avoiding the backend system from overloading. So this is OUT.

>> Rate Limiting and Throttling/ Spike Control policies are designed to limit API access, but have different intentions.

>> Rate limiting protects an API by applying a hard limit on its access.

>> Throttling/ Spike Control shapes API access by smoothing spikes in traffic. That is why, Spike Control is the right option.

NEW QUESTION 43

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

Answer: A

Explanation:

Correct 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.

NEW QUESTION 48

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

Answer: C

Explanation:

Correct 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

NEW QUESTION 50

What do the API invocation metrics provided by Anypoint Platform provide?

- A. ROI metrics from APIs that can be directly shared with business users
- B. Measurements of the effectiveness of the application network based on the level of reuse
- C. Data on past API invocations to help identify anomalies and usage patterns across various APIs
- D. Proactive identification of likely future policy violations that exceed a given threat threshold

Answer: C

Explanation:

Correct Answer

Data on past API invocations to help identify anomalies and usage patterns across various APIs

API Invocation metrics provided by Anypoint Platform:

- >> Does NOT provide any Return Of Investment (ROI) related information. So the option suggesting it is OUT.
 - >> Does NOT provide any information w.r.t how APIs are reused, whether there is effective usage of APIs or not etc...
 - >> Does NOT provide any prediction information as such to help us proactively identify any future policy violations.
- So, the kind of data/information we can get from such metrics is on past API invocations to help identify anomalies and usage patterns across various APIs.

NEW QUESTION 52

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

Answer: D

Explanation:

Correct 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:

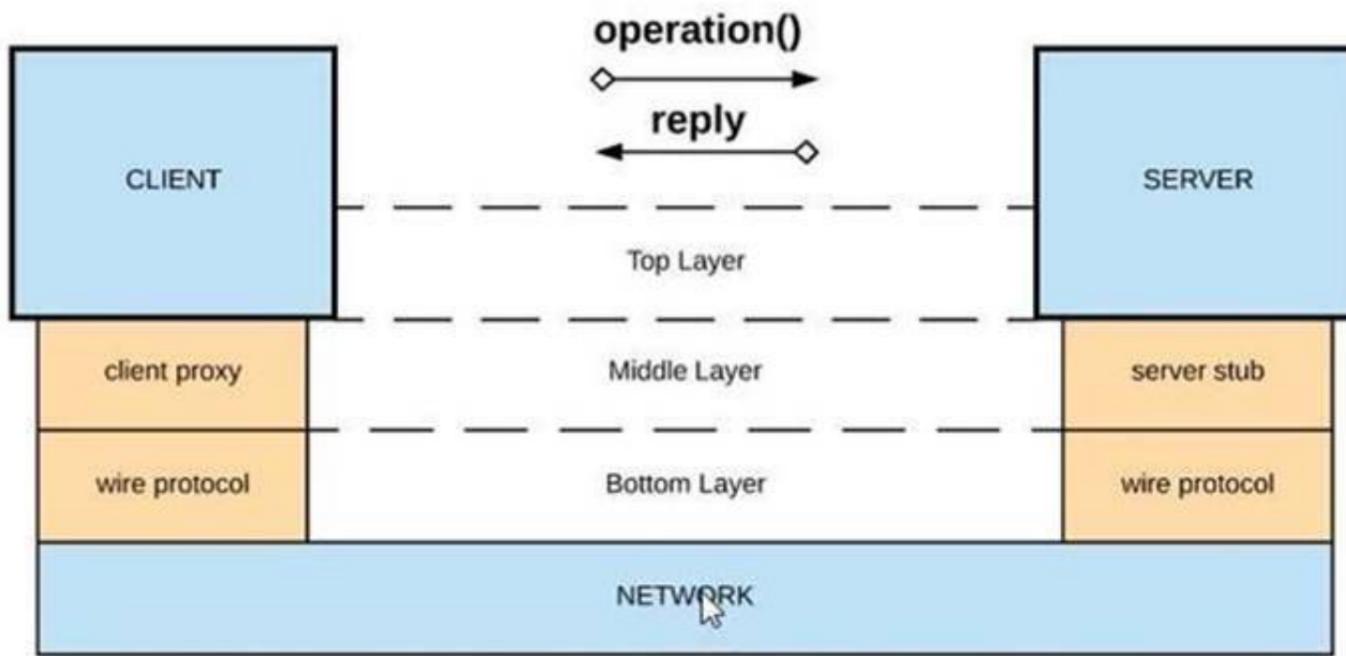
- >> They test the validity of input and output of source and target systems.
 - >> They must be triggered by an external client tool or event source.
- It is NOT TRUE that Unit Tests must be run in a unit testing environment with dedicated Mule runtimes for the environment.

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>

NEW QUESTION 54

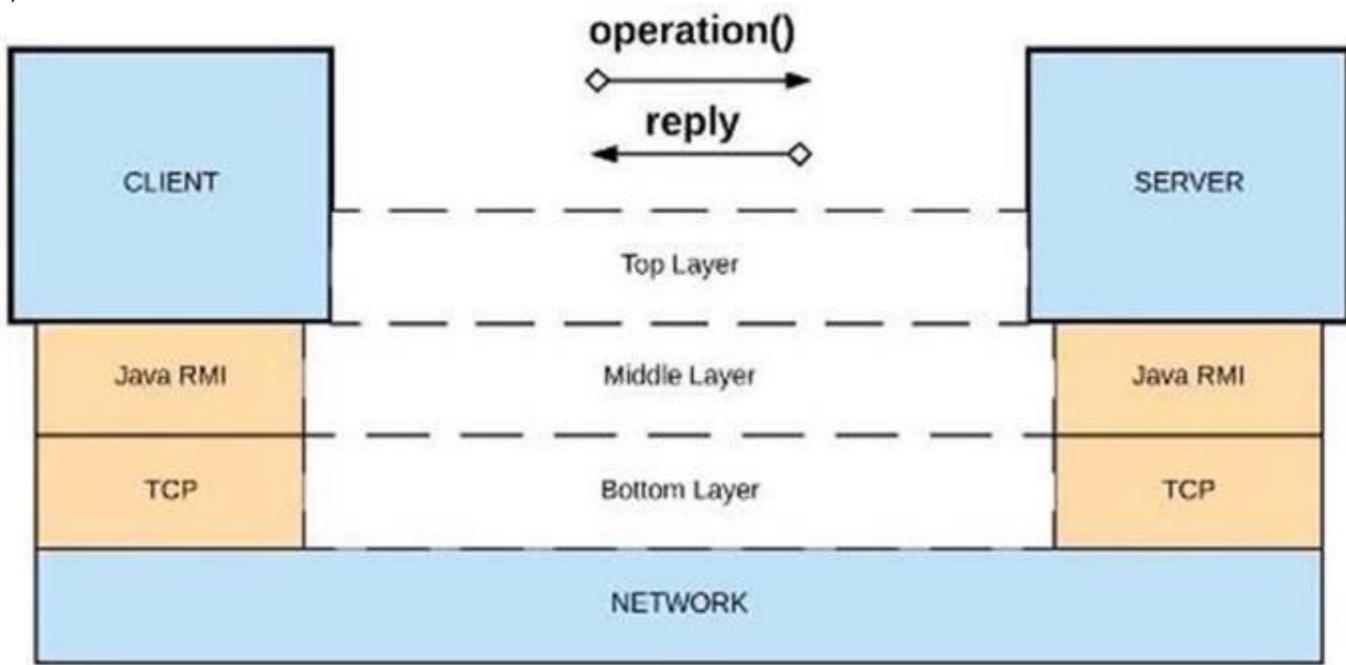
Refer to the exhibit.



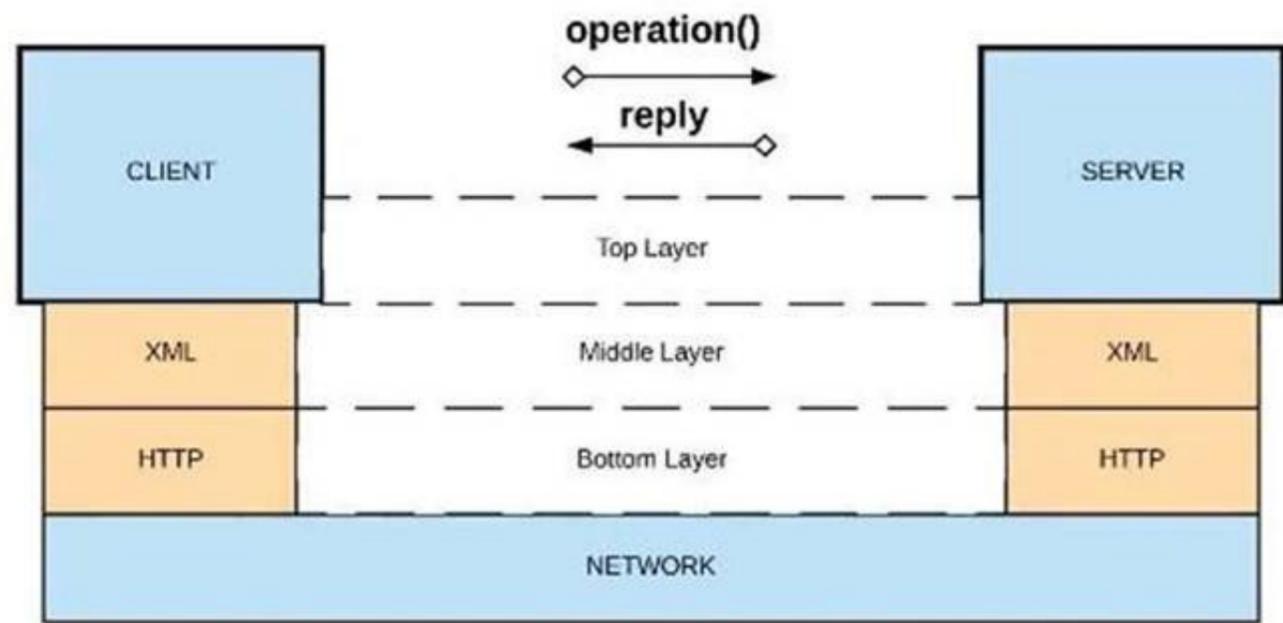
Generic RPC Architecture

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

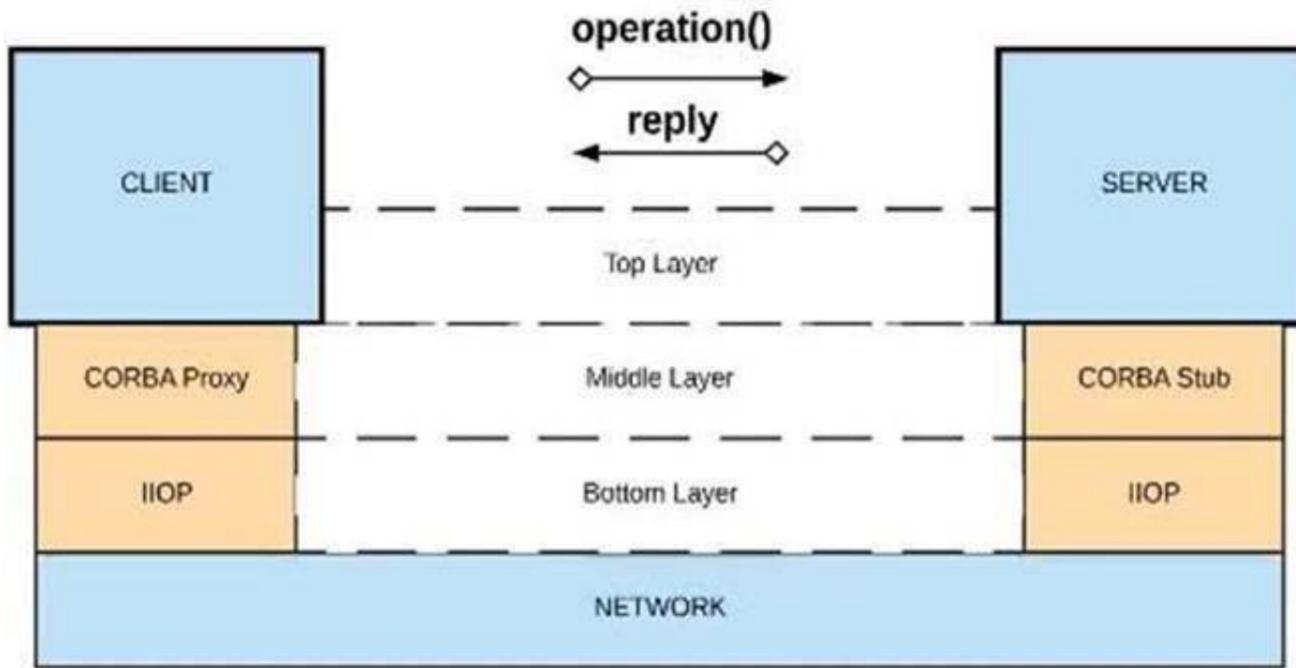
A) Java RMI over TCP



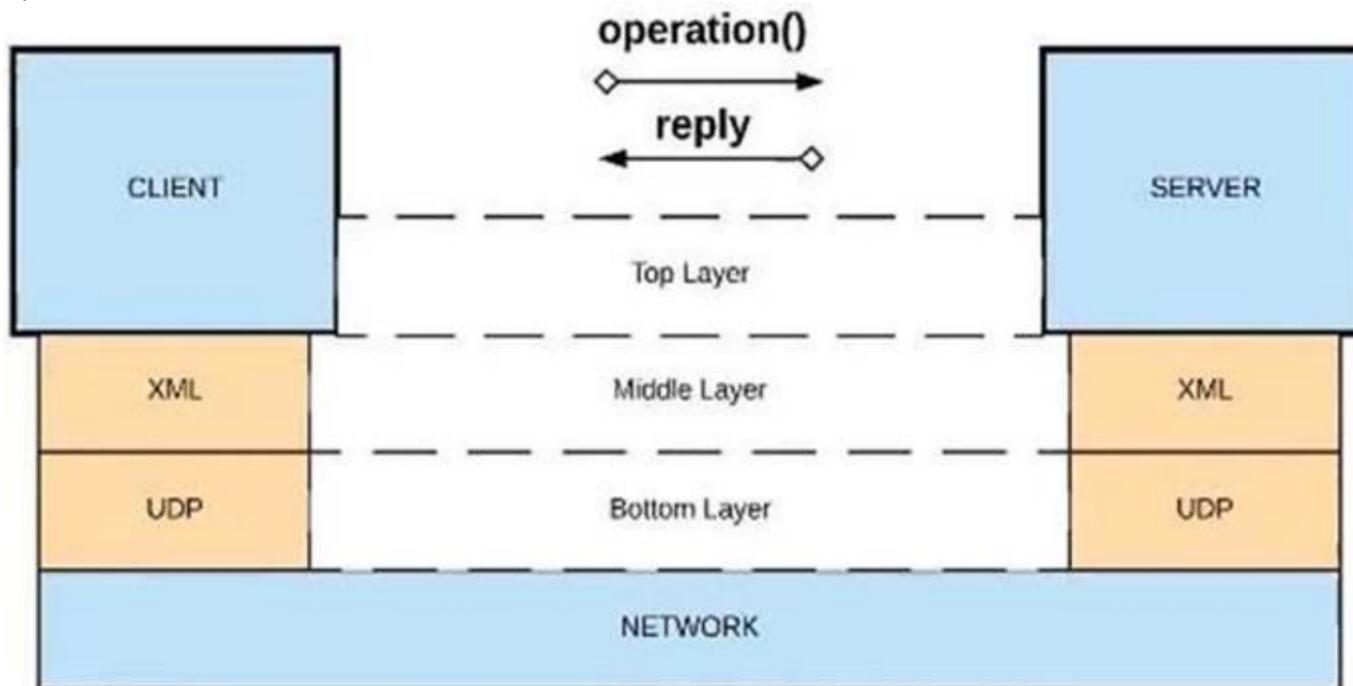
B) Java RMI over TCP



C) CORBA over IIOP



D) XML over UDP



- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: D

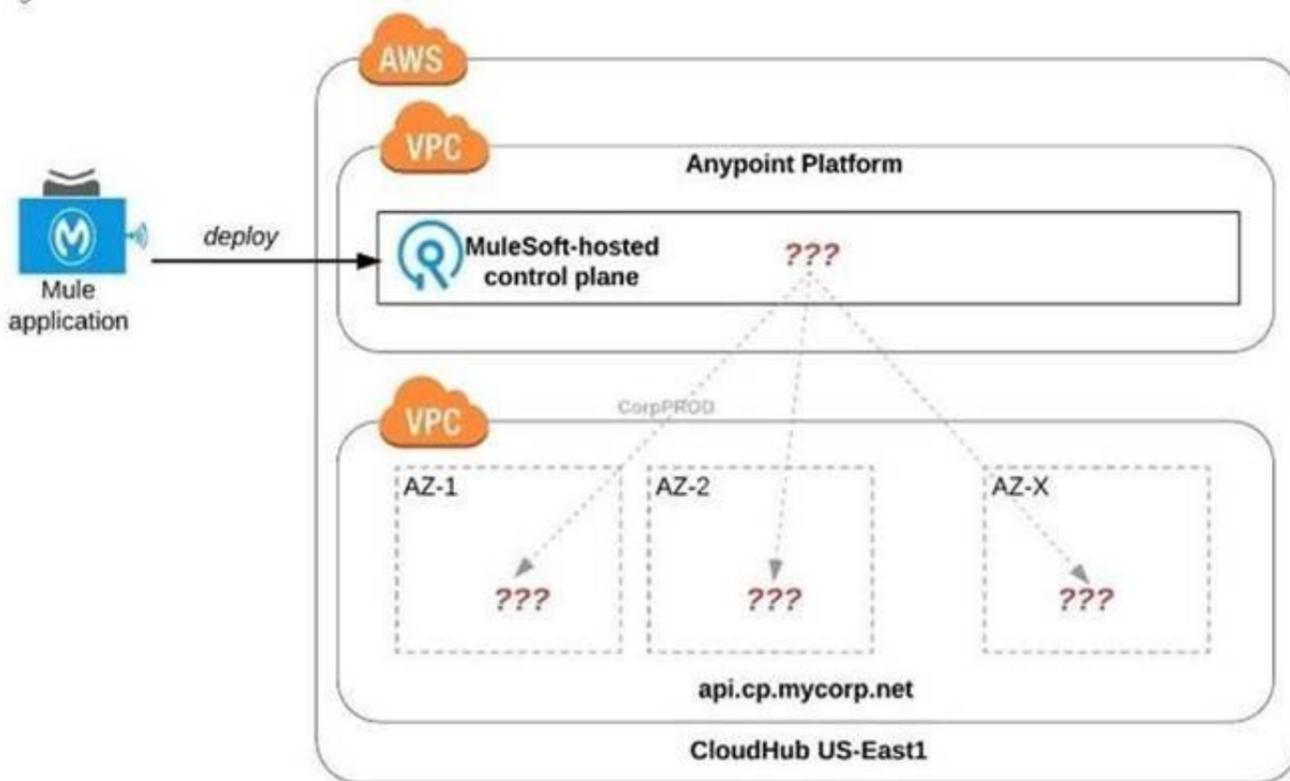
Explanation:

\Correct Answer
 XML over HTTP

>> API-led connectivity and Application Networks urge to have the APIs on HTTP based protocols for building most effective APIs and networks on top of them.
 >> The HTTP based APIs allow the platform to apply various varieties of policies to address many NFRs
 >> The HTTP based APIs also allow to implement many standard and effective implementation patterns that adhere to HTTP based w3c rules.
 Bottom of Form Top of Form

NEW QUESTION 59

Refer to the exhibit.



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

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

Answer: D

Explanation:

Correct Answer

Workers are randomly distributed across available AZs within that region.

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

>> There are

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 one AZ so that all workers are not assigned to same AZ for same application.

NEW QUESTION 63

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

Answer: A

NEW QUESTION 65

What is a key performance indicator (KPI) that measures the success of a typical C4E that is immediately apparent in responses from the Anypoint Platform APIs?

- A. The number of production outage incidents reported in the last 24 hours
- B. The number of API implementations that have a publicly accessible HTTP endpoint and are being managed by Anypoint Platform
- C. The fraction of API implementations deployed manually relative to those deployed using a CI/CD tool
- D. The number of API specifications in RAML or OAS format published to Anypoint Exchange

Answer: D

Explanation:

Correct Answer

The number of API specifications in RAML or OAS format published to Anypoint Exchange

>> The success of C4E always depends on their contribution to the number of reusable assets that they have helped to build and publish to Anypoint Exchange.

>> It is NOT due to any factors w.r.t # of outages, Manual vs CI/CD deployments or Publicly accessible HTTP endpoints

>> Anypoint Platform APIs helps us to quickly run and get the number of published RAML/OAS assets to Anypoint Exchange. This clearly depicts how successful a C4E team is based on number of returned assets in the response.

NEW QUESTION 69

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. IPwhitelist

Answer: D

Explanation:

Correct 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 occasionally 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.

NEW QUESTION 74

The responses to some HTTP requests can be cached depending on the HTTP verb used in the request. 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

Answer: D

Explanation:

Correct 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.

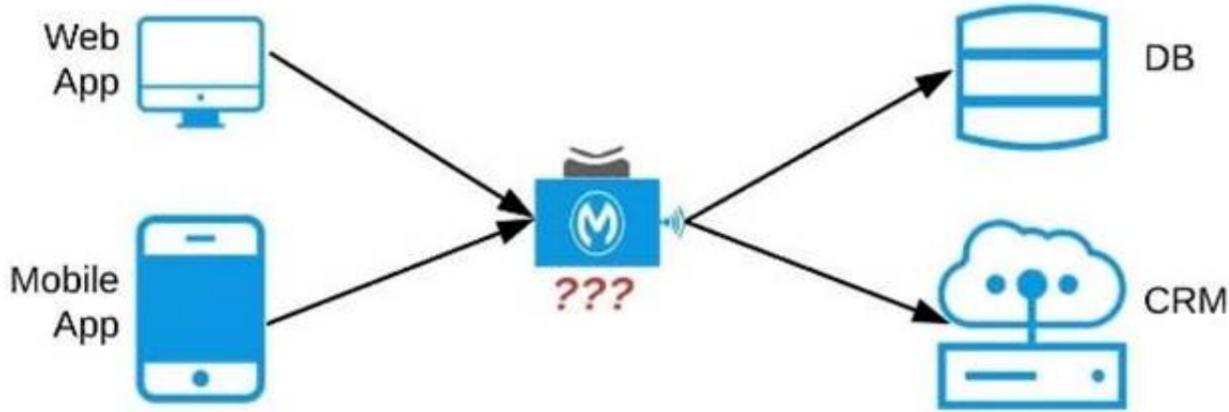
<http://restcookbook.com/HTTP%20Methods/idempotency/>

NEW QUESTION 77

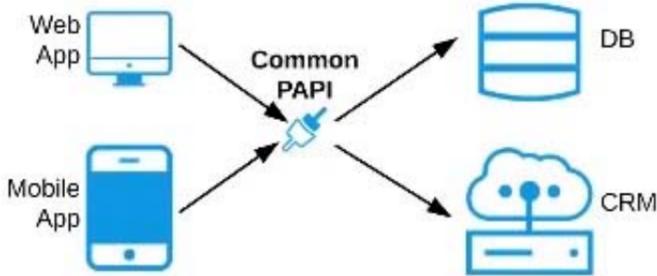
Refer to the exhibit. An organization needs to enable access to their customer data from both a mobile app and a web application, which each need access to common fields as well as certain unique fields.

The data is available partially in a database and partially in a 3rd-party CRM system.

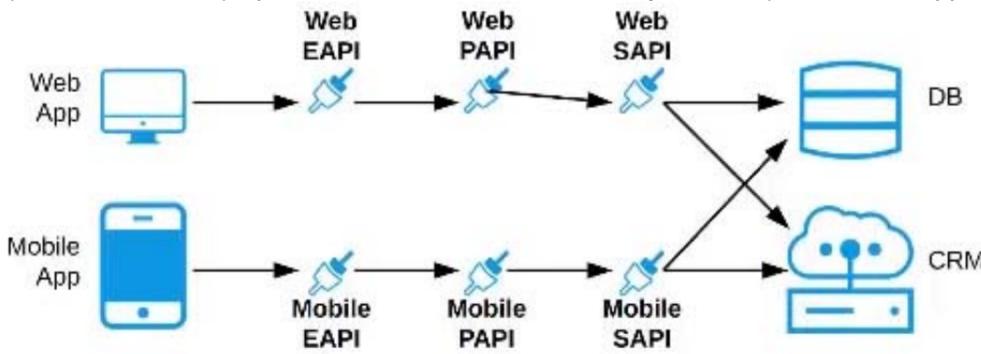
What APIs should be created to best fit these design requirements?



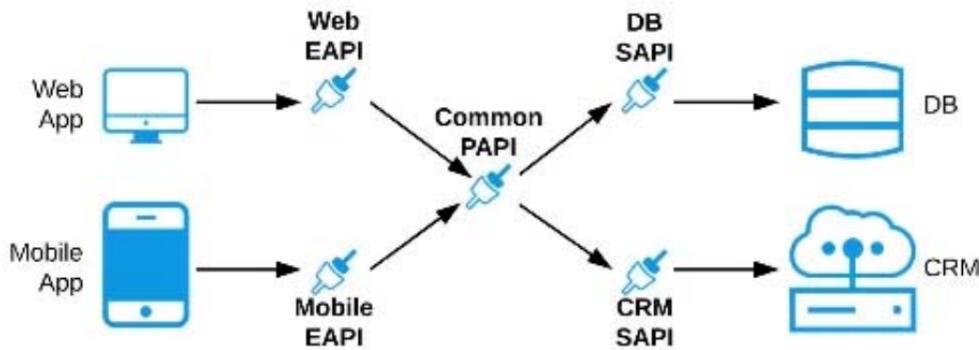
A) A Process API that contains the data required by both the web and mobile apps, allowing these applications to invoke it directly and access the data they need thereby providing the flexibility to add more fields in the future without needing API changes



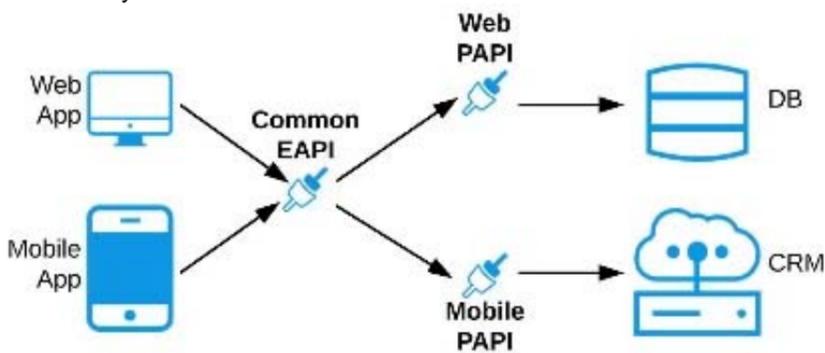
B) One set of APIs (Experience API, Process API, and System API) for the web app, and another set for the mobile app



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



- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: C

Explanation:

Correct 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.

NEW QUESTION 82

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

Answer: C

Explanation:

<https://www.folkstalk.com/2019/11/mulesoft-integration-and-platform.html>

Correct 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/>

NEW QUESTION 85

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.

Answer: A

Explanation:

Correct 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

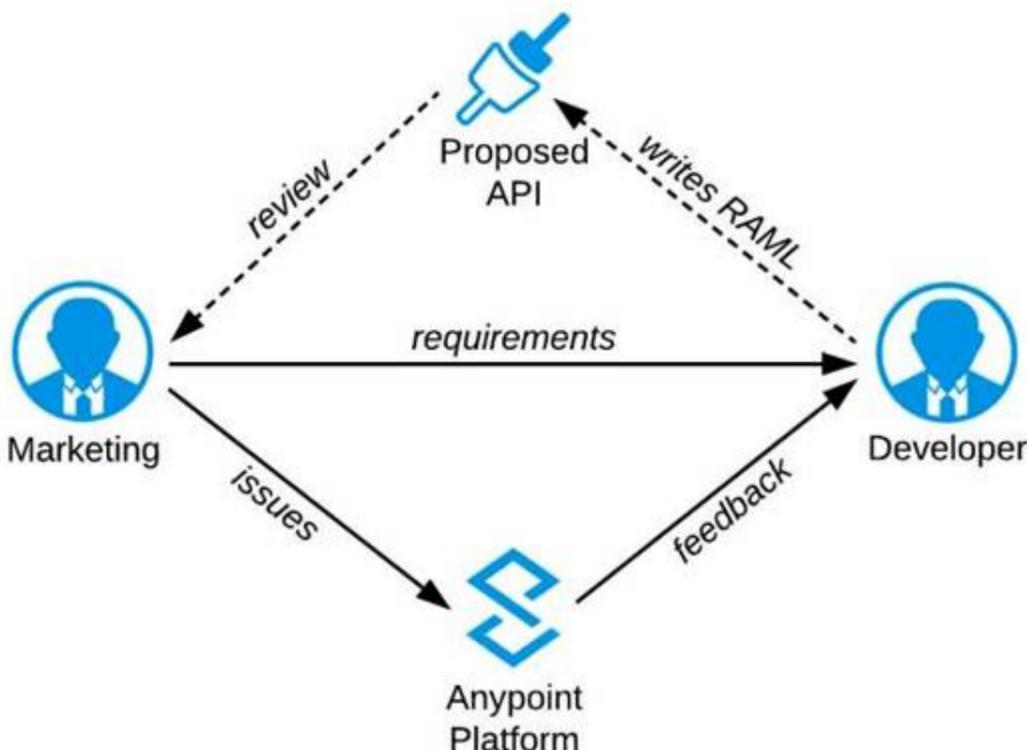
>> System APIs - reduce the dependency on the underlying backend systems by helping unlock data from backend systems in a reusable and consumable way.

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>

NEW QUESTION 88

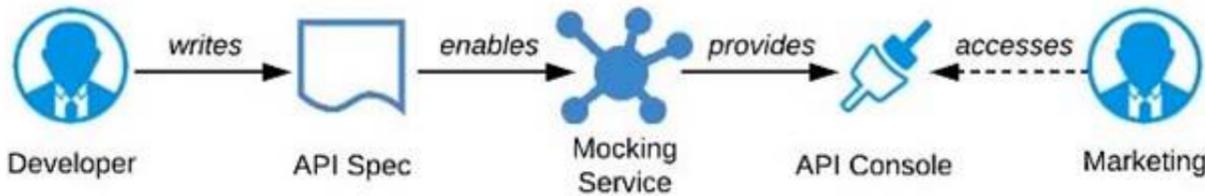
Refer to the exhibit.



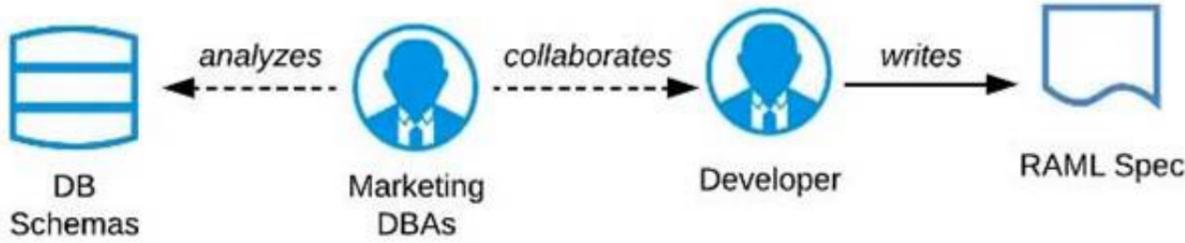
A RAML definition has been proposed for a new Promotions Process API, and has been published to Anypoint Exchange.

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?

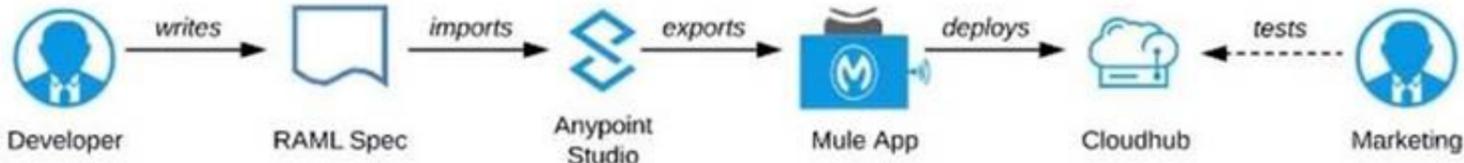
A) Ask the Marketing Department to interact with a mocking implementation of the API using the automatically generated API Console



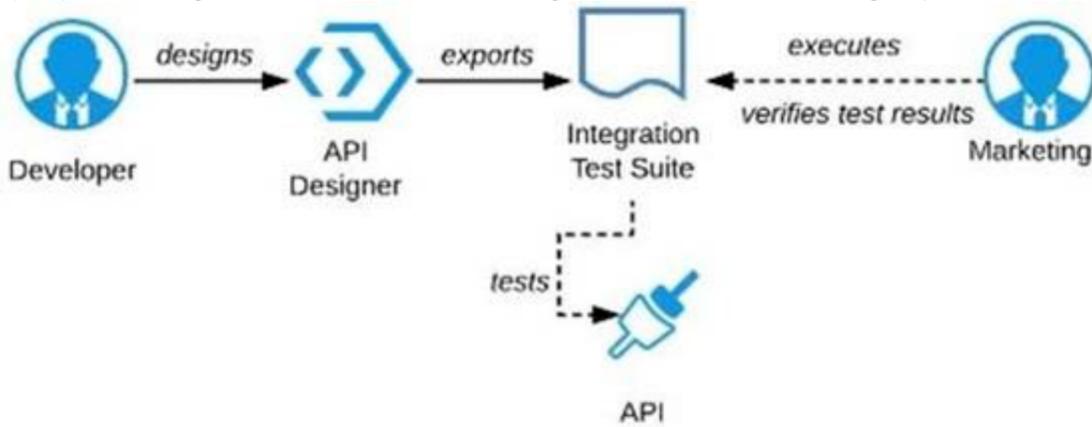
B) Organize a design workshop with the DBAs of the Marketing Department in which the database schema of the Marketing IT systems is translated into RAML



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



- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: A

Explanation:

Correct 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.

NEW QUESTION 90

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

Answer: B

Explanation:

Correct Answer

SLA-based rate limiting

>> Client Id enforcement 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>

NEW QUESTION 94

An organization has several APIs that accept JSON data over HTTP POST. The APIs are all publicly available 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

Answer: D

Explanation:

Correct 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.

NEW QUESTION 95

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

- A. Drive consumption as much as production of assets; this enables developers to discover and reuse assets from other projects and encourages standardization
- B. Expose assets using a Master Data Management (MDM) system; this standardizes projects and enables developers to quickly discover and reuse assets from other projects
- C. Implement SOA for reusable APIs to focus on production over consumption; this standardizes on XML and WSDL formats to speed up decision making
- D. Create a lean and agile organization that makes many small decisions everyday; this speeds up decision making and enables each line of business to take ownership of its projects

Answer: A

Explanation:

Correct Answer

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

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

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

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

References:

<https://blogs.mulesoft.com/biz/connectivity/what-is-a-center-for-enablement-c4e/> <https://www.mulesoft.com/resources/api/secret-to-managing-it-projects>

NEW QUESTION 99

What Anypoint Connectors support transactions?

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

Answer: A

NEW QUESTION 104

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

Answer: C

Explanation:

Correct 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.

NEW QUESTION 109

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 APIs 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 API producer should be requested to run the old version in parallel with the new one
- B. The API producer should be contacted to understand the change to existing functionality
- C. The API client code only needs to be changed if it needs to take advantage of the new features
- D. The API clients need to update the code on their side and need to do full regression

Answer: C

NEW QUESTION 112

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

Answer: A

Explanation:

Correct 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.

NEW QUESTION 116

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

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

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

Answer: B

Explanation:

Correct Answers

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

Bottom of Form Top of Form

NEW QUESTION 121

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

Answer: A

Explanation:

Correct Answer

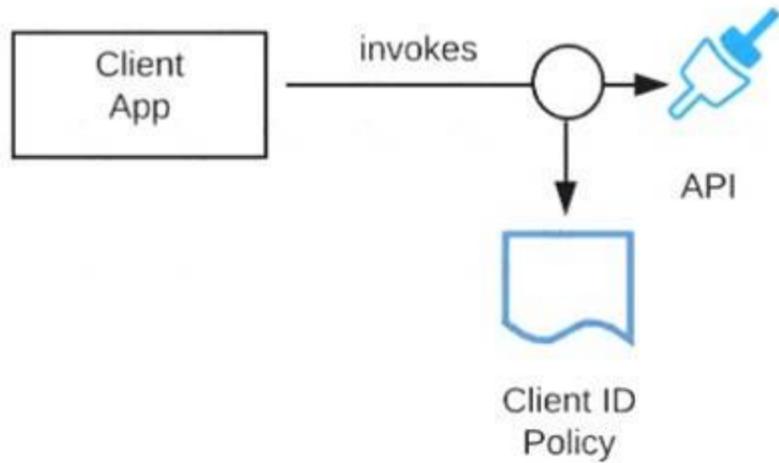
Approve the client application request for the chosen SLA tier

>> Only approving the client application request for the chosen SLA tier can be automated

>> Rest of the provided options are not valid

NEW QUESTION 122

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

Answer: C

Explanation:

Correct 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 Enforcement 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>

NEW QUESTION 126

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 bounded-context 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

Answer: C

Explanation:

Correct 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.

NEW QUESTION 130

.....

THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual MCPA-Level-1 Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the MCPA-Level-1 Product From:

<https://www.2passeasy.com/dumps/MCPA-Level-1/>

Money Back Guarantee

MCPA-Level-1 Practice Exam Features:

- * MCPA-Level-1 Questions and Answers Updated Frequently
- * MCPA-Level-1 Practice Questions Verified by Expert Senior Certified Staff
- * MCPA-Level-1 Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- * MCPA-Level-1 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year