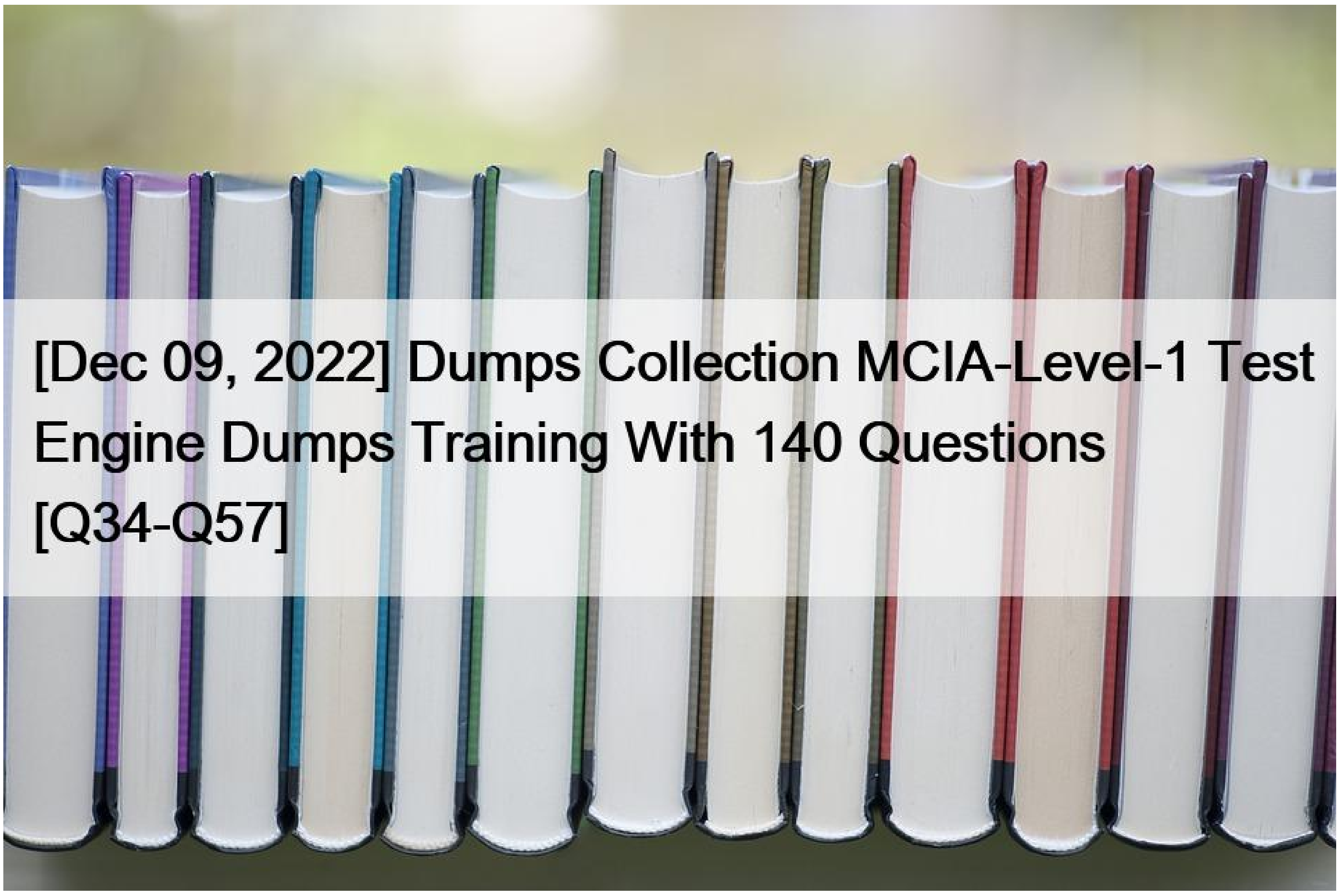
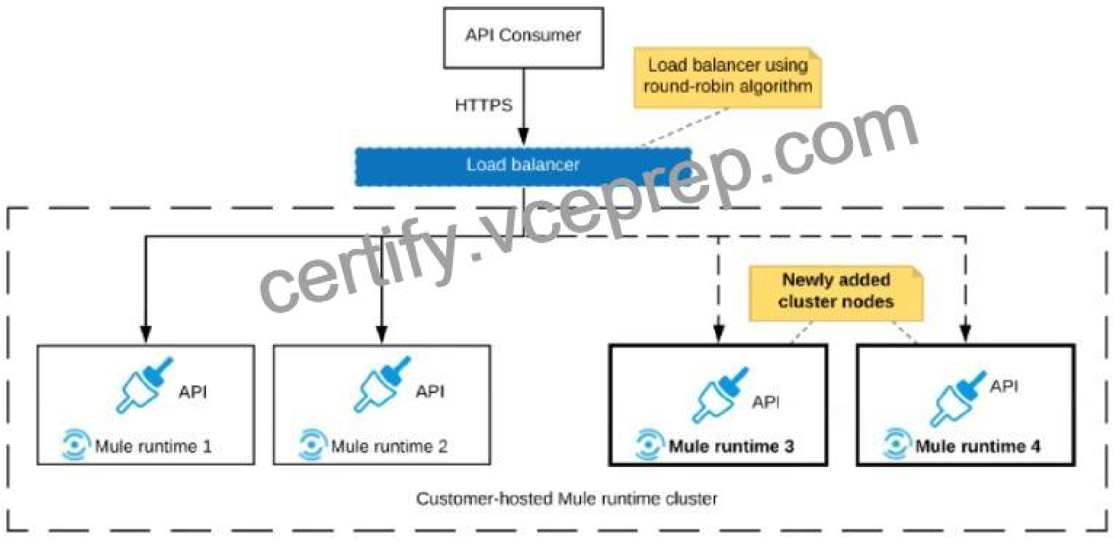
[Dec 09, 2022 Dumps Collection MCIA-Level-1 Test Engine Dumps Training With 140 Questions [Q34-Q57



[Dec 09, 2022] Dumps Collection MCIA-Level-1 Test Engine Dumps Training With 140 Questions
MuleSoft MCIA-Level-1 Dumps - 100% Cover Real Exam Questions

For more info read reference:
MuleSoft Training
MuleSoft Documents

NO.34 Refer to the exhibit.


API Consumer
HTTPS
Load balancer using round-robin algorithm
Load balancer
Newly added cluster nodes
API
Mule runtime 1
API
Mule runtime 2
API
Mule runtime 3
API
Mule runtime 4
Customer-hosted Mule runtime cluster

An organization uses a 2-node Mute runtime cluster to host one stateless API implementation. The API is accessed over HTTPS through a load balancer that uses round-robin for load distribution.

Two additional nodes have been added to the cluster and the load balancer has been configured to recognize the new nodes with no other change to the load balancer.

What average performance change is guaranteed to happen, assuming all cluster nodes are fully operational?
* 50% reduction in the response time of the API
* 100% increase in the throughput of the API
* 50% reduction In the JVM heap memory consumed by each node
* 50% reduction In the number of requests being received by each node

NO.35 A Mule application name Pub uses a persistence object store. The Pub Mule application is deployed to Cloudhub and it configured to use Object Store v2.

Another Mule application name sub is being developed to retrieve values from the Pub Mule application persistence object Store and will also be deployed to cloudhub.

What is the most direct way for the Sub Mule application to retrieve values from the Pub Mule application persistence object store with the least latency?
* Use an object store connector configured to access the Pub Mule application persistence object store
* Use a VM connector configured to directly access the persistence queue of the Pub Mule application persistence object store.
* Use an Anypoint MQ connector configured to directly access the Pub Mule application persistence object store
* Use the Object store v2 REST API configured to access the Pub Mule application persistence object store.

NO.36 In a Mule Application, a flow contains two (2) JMS consume operations that are used to connect to a JMS broker and consume messages from two(2) JMS destination. The Mule application then joins the two JMS messages together.

The JMS broker does not implement high availability (HA) and periodically experiences scheduled outages of upto 10 mins for routine maintenance.

What is the most idiomatic (used for its intented purpose) way to build the mule flow so it can best recover from the expected outages?
* Configure a reconnection strategy for the JMS connector
* Enclose the two(2) JMS operation in an Until Successful scope
* Consider a transaction for the JMS connector
* Enclose the two(2) JMS operations in a Try scope with an Error Continue error handler

NO.37 What requires configuration of both a key store and a trust store for an HTTP Listener?
* Support for TLS mutual (two-way) authentication with HTTP clients
* Encryption of requests to both subdomains and API resource endpoints fhttPs://aDi.customer.com/ and

https://customer.com/api)
* Encryption of both HTTP request and HTTP response bodies for all HTTP clients
* Encryption of both HTTP request header and HTTP request body for all HTTP clients

NO.38 In Anypoint Platform, a company wants to configure multiple identity providers(Idps) for various lines of business (LOBs) Multiple business groups and environments have been defined for the these LOBs. What Anypoint Platform feature can use multiple Idps access the company&#8217;s business groups and environment?
* User management
* Roles and permissions
* Dedicated load balancers
* Client Management
Correct answer is Client Management

* Anypoint Platform acts as a client provider by default, but you can also configure external client providers to authorize client applications.

* As an API owner, you can apply an OAuth 2.0 policy to authorize client applications that try to access your API. You need an OAuth 2.0 provider to use an OAuth 2.0 policy.

* You can configure more than one client provider and associate the client providers with different environments. If you configure multiple client providers after you have already created environments, you can associate the new client providers with the environment.

* You should review the existing client configuration before reassigning client providers to avoid any downtime with existing assets or APIs.

* When you delete a client provider from your master organization, the client provider is no longer available in environments that used it.

* Also, assets or APIs that used the client provider can no longer authorize users who want to access them.

&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;-MuleSoft Reference: https://docs.mulesoft.com/access-management/managing-api-clients

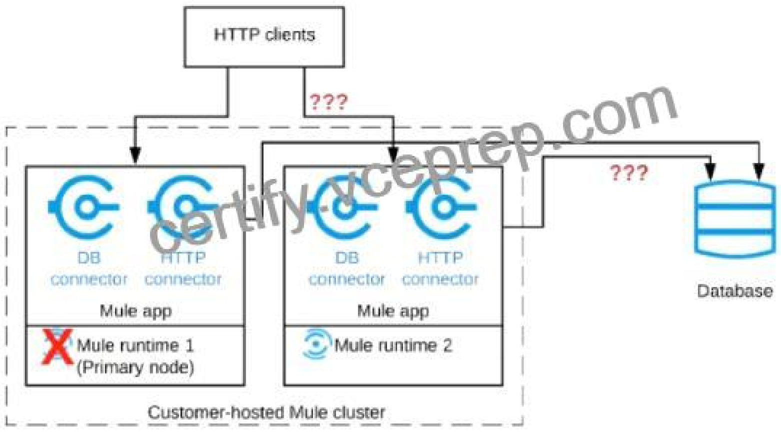
https://www.folkstalk.com/2019/11/mulesoft-integration-and-platform.html

NO.39 What requires configuration of both a key store and a trust store for an HTTP Listener?

* Support for TLS mutual (two-way) authentication with HTTP clients
* Encryption of requests to both subdomains and API resource endpoints fhttPs://aDi.customer.com/ and

https://customer.com/api)
* Encryption of both HTTP request and HTTP response bodies for all HTTP clients
* Encryptionof both HTTP request header and HTTP request body for all HTTP clients

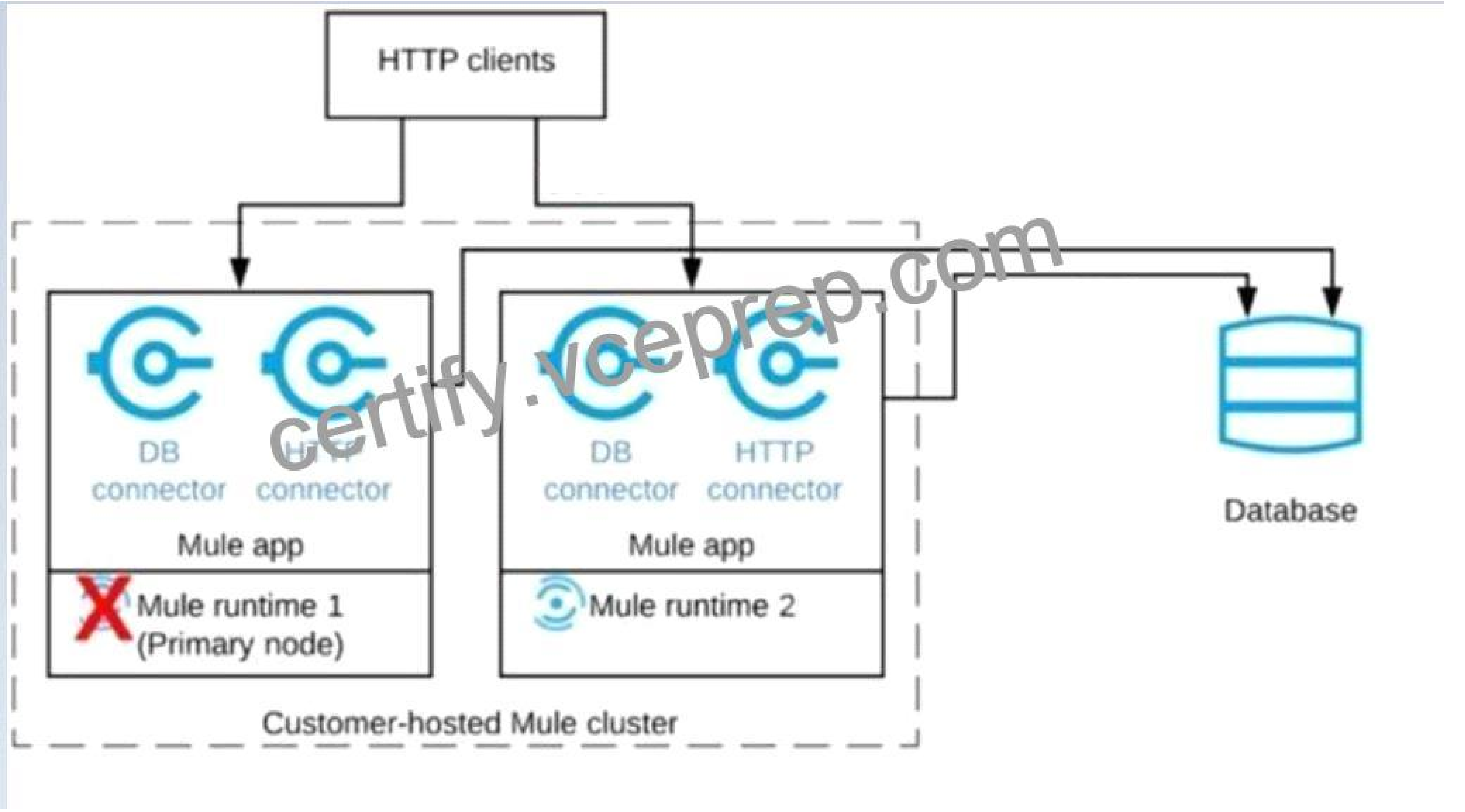
NO.40 Refer to the exhibit.



A Mule application is deployed to a cluster of two customer-hosted Mute runtimes. The Mute application has a flow that polls a database and another flow with an HTTP Listener.

HTTP clients send HTTP requests directly to individual cluster nodes.

What happens to database polling and HTTP request handling in the time after the primary (master) node of the cluster has railed, but before that node is restarted?
* Database polling continues Only HTTP requests sent to the remaining node continue to be accepted
* Database polling stops All HTTP requests continue to be accepted
* Database polling continues All HTTP requests continue to be accepted, but requests to the failed node Incur increased latency
* Database polling stops All HTTP requests are rejected
Correct answer is Database polling continues Only HTTP requests sent to the remaining node continue to be accepted. : Architecture descripted in the question could be described as follows.When node 1 is down , DB polling will still continue via node 2 . Also requests which are coming directly to node 2 will also be accepted and processed in BAU fashion. Only thing that wont work is when requests are sent to Node 1 HTTP connector. The flaw with this architecture is HTTP clients are sending HTTP requests directly to individual cluster nodes. By default, clustering Mule runtime engines ensures high system availability. If a Mule runtime engine node becomes unavailable due to failure or planned downtime, another node in the cluster can assume the workload and continue to process existing events and messages



NO.41 An XA transaction Is being configured that involves a JMS connector listening for Incoming JMS messages. What is the meaning of the timeout attribute of the XA transaction, and what happens after the timeout expires?
* The time that is allowed to pass between committing the transaction and the completion of the Mule flow After the timeout, flow processing triggers an error
* The time that Is allowed to pass between receiving JMS messages on the same JMS connection After the timeout, a new JMS connection Is established
* The time that Is allowed to pass without the transaction being ended explicitly After the timeout, the transaction Is forcefully rolled-back
* The time that Is allowed to pass for state JMS consumer threads to be destroyed After the timeout, a new JMS consumer thread is created

NO.42 A mule application uses an HTTP request operation to involve an external API.

The external API follows the HTTP specification for proper status code usage.

What is possible cause when a 3xx status code is returned to the HTTP Request operation from the external API?
* The request was not accepted by the external API
* The request was Redirected to a different URL by the external API
* The request was NOT RECEIVED by the external API
* The request was ACCEPTED by the external API
3xx HTTP status codes indicate a redirection that the user agent (a web browser or a crawler) needs to take further action when trying to access a particular resource.

NO.43 What aspects of a CI/CD pipeline for Mule applications can be automated using MuleSoft-provided Maven plugins?

* Compile, package, unit test, validate unit test coverage, deploy
* Compile, package, unit test, deploy, integration test (Incorrect)
* Compile, package, unit test, deploy, create associated API instances in API Manager
* Import from API designer, compile, package, unit test, deploy, publish to Anypoint Exchange

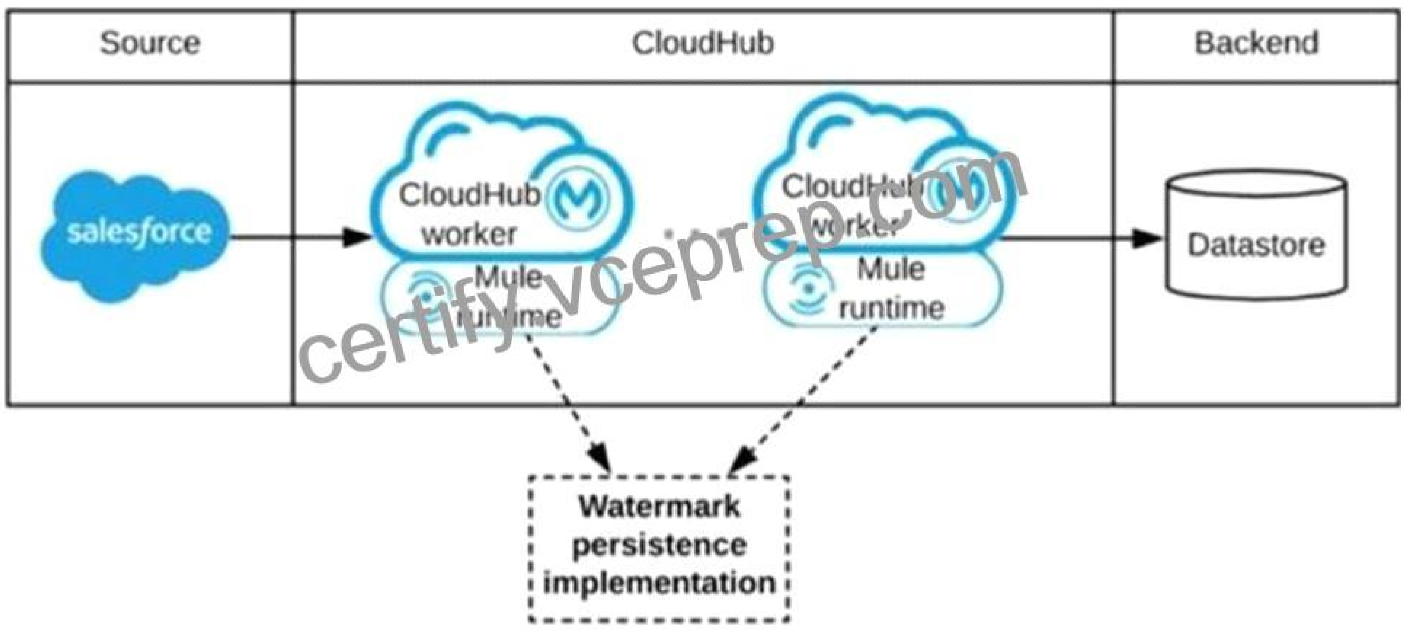
Correct answer is &#8220;Compile, package, unit test, validate unit test coverage, deploy&#8221; : Anypoint Platform supports continuous integration and continuous delivery using industry standard tools Mule Maven Plugin The Mule Maven plugin can automate building, packaging and deployment of Mule applications from source projects Using the Mule Maven plugin, you can automate your Mule application deployment to CloudHub, to Anypoint Runtime Fabric, or on-premises, using any of the following deployment strategies * CloudHub deployment * Runtime Fabric deployment * Runtime Manager REST API deployment * Runtime Manager agent deployment MUnit Maven Plugin The MUnit Maven plugin can automate test execution, and ties in with the Mule Maven plugin. It provides a full suite of integration and unit test capabilities, and is fully integrated with Maven and Surefire for integration with your continuous deployment environment. Since MUnit 2.x, the coverage report goal is integrated with the maven reporting section. Coverage Reports are generated during Maven&#8217;s site lifecycle, during the coverage-report goal. One of the features of MUnit Coverage is to fail the build if a certain coverage level is not reached. MUnit is not used for integration testing Also publishing to Anypoint Exchange or to create associated API instances in API Manager is not a part of CICD pipeline which can ne achieved using mulesoft provided maven plugin Architecture mentioned in the question can be diagrammatically put as below. Persistent Object Store is the correct answer .

* Mule Object Stores: An object store is a facility for storing objects in or across Mule applications. Mule uses object stores to persist data for eventual retrieval.

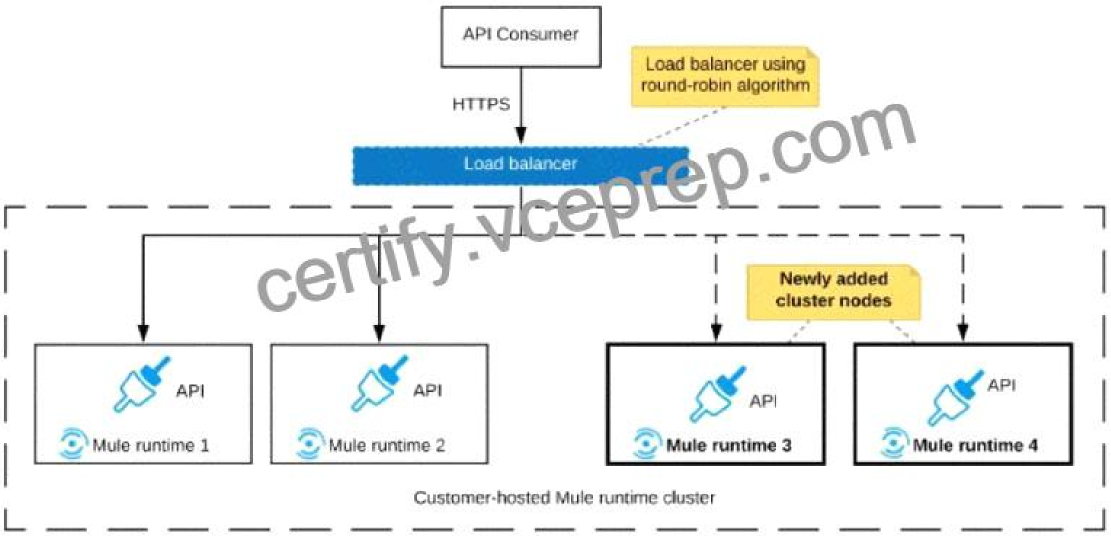
Mule provides two types of object stores:

1) In-memory store &#8211; stores objects in local Mule runtime memory. Objects are lost on shutdown of the Mule runtime. So we cant use in memory store in our scenario as we want to share watermark within all cloudhub workers

2) Persistent store &#8211; Mule persists data when an object store is explicitly configured to be persistent. Hence this watermark will be available even any of the worker goes down



NO.44 Refer to the exhibit.



An organization uses a 2-node Mute runtime cluster to host one stateless API implementation. The API is accessed over HTTPS through a load balancer that uses round-robin for load distribution.

Two additional nodes have been added to the cluster and the load balancer has been configured to recognize the new nodes with no other change to the load balancer.

What average performance change is guaranteed to happen, assuming all cluster nodes are fully operational?
* 50% reduction in the response time of the API
* 100% increase in the throughput of the API
* 50% reduction In the JVM heap memory consumed by each node
* 50% reduction In the number of requests being received by each node

NO.45 What aspects of a CI/CD pipeline for Mute applications can be automated using MuleSoft-provided Maven plugins?
* Compile, package, unit test, deploy, create associated API instances in API Manager B Import from API designer, compile, package, unit test, deploy, publish to Am/point Exchange
* Compile, package, unit test, validate unit test coverage, deploy
* Compile, package, unit test, deploy, integration test

NO.46 An organization has various integrations implemented as Mule applications. Some of these Mule applications are deployed to customhosted Mule runtimes (on-premises) while others execute in theMuleSoft-hosted runtime plane (CloudHub). To perform the Integra functionality, these Mule applications connect to various backend systems, with multiple applications typically needing to access the backend systems.

How can the organization most effectively avoid creating duplicates in each Mule application of the credentials required to access thebackend systems?
* Create a Mule domain project that maintains the credentials as Mule domain-shared resources Deploy the Mule applications to the Mule domain, so the credentials are available to the Mule applications
* Store the credentials in properties files in a shared folder within the organization&#8217;s data center Have the Mule applications load properties files from this shared location at startup

* Segregate the credentials for each backend system into environment-specific properties files Package these properties files in each Mule application, from where they are loaded at startup
* Configure or create a credentials service that returns the credentials for each backend system, and that is accessible from customer-hosted and MuleSoft-hosted Mule runtimes Have the Mule applications toad the properties at startup by invoking that credentials service

NO.47 What API policy would LEAST likely be applied to a Process API?
* Custom circuit breaker
* Client ID enforcement
* Rate limiting
* JSON threat protection
Key to this question lies in the fact that Process API are not meant to be accessed directly by clients. Lets analyze options one by one. Client ID enforcement : This is applied at process API level generally to ensure that identity of API clients is always known and available for API-based analytics Rate Limiting : This policy is applied on Process Level API to secure API&#8217;s against degradation of service that can happen in case load received is more than it can handle Custom circuit breaker : This is also quite useful feature on process level API&#8217;s as it saves the API client the wasted time and effort of invoking a failing API. JSON threat protection : This policy is not required at Process API and rather implemented as Experience API&#8217;s. This policy is used to safeguard application from malicious attacks by injecting malicious code in JSON object. As ideally Process API&#8217;s are never called from external world , this policy is never used on Process API&#8217;s Hence correct answer is JSON threat protection MuleSoft Documentation Reference : https://docs.mulesoft.com/api-manager/2.x/policy-mule3-json-threat

NO.48 Anypoint Exchange is required to maintain the source code of some of the assets committed to it, such as Connectors, Templates, and API specifications.

What is the best way to use an organization&#8217;s source-code management (SCM) system in this context?
* Organizations should continue to use an SCM system of their choice, in addition to keeping source code for these asset types in Anypoint Exchange, thereby enabling parallel development, branching, and merging
* Organizations need to use Anypoint Exchange as the main SCM system to centralize versioning and avoid code duplication
* Organizations can continue to use an SCM system of their choice for branching and merging, as long as they follow the branching and merging strategy enforced by Anypoint Exchange
* Organizations need to point Anypoint Exchange to their SCM system so Anypoint Exchange can pull source code when requested by developers and provide it to Anypoint Studio

NO.49 What is not true about Mule Domain Project?
* This allows Mule applications to share resources
* Expose multiple services within the Mule domain on the same port
* Only available Anypoint Runtime Fabric
* Send events (messages) to other Mule applications using VM queues
* Mule Domain Project is ONLY available for customer-hosted Mule runtimes, but not for Anypoint Runtime Fabric

* Mule domain project is available for Hybrid and Private Cloud (PCE). Rest all provide application isolation and can&#8217;t support domain project.

What is Mule Domain Project?

* A Mule Domain Project is implemented to configure the resources that are shared among different projects. These resources can be used by all the projects associated with this domain. Mule applications can be associated with only one domain, but a domain can be associated with multiple projects. Shared resources allow multiple development teams to work in parallel using the same set of reusable connectors. Defining these connectors as shared resources at the domain level allows the team to: &#8211; Expose multiple services within the domain through the same port. &#8211; Share the connection to persistent storage. &#8211; Share services

between apps through a well-defined interface. &#8211; Ensure consistency between apps upon any changes because the configuration is only set in one place.

* Use domains Project to share the same host and port among multiple projects. You can declare the http connector within a domain project and associate the domain project with other projects. Doing this also allows to control thread settings, keystore configurations, time outs for all the requests made within multiple applications. You may think that one can also achieve this by duplicating the http connector configuration across all the applications. But, doing this may pose a nightmare if you have to make a change and redeploy all the applications.

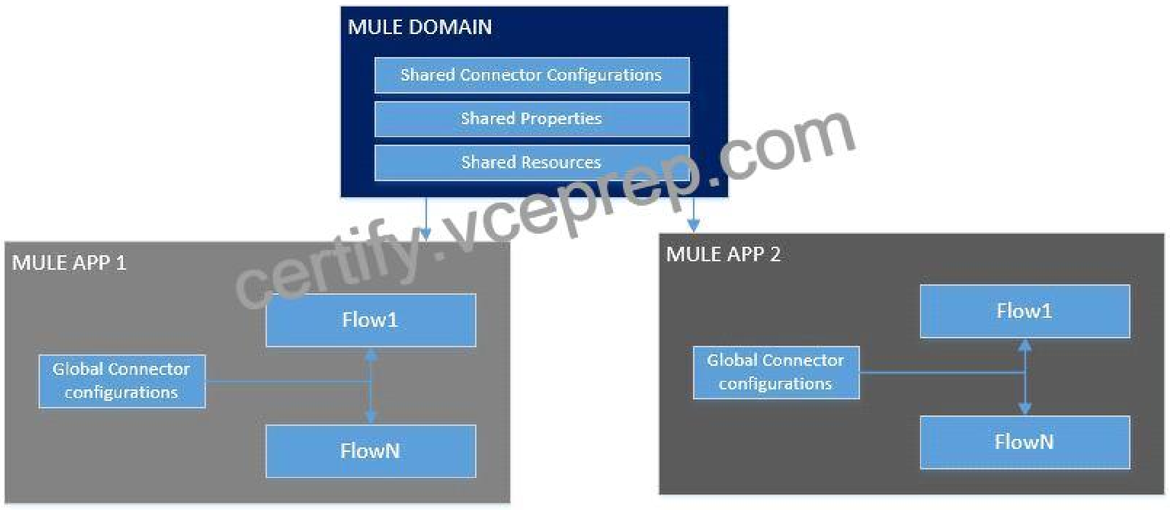
* If you use connector configuration in the domain and let all the applications use the new domain instead of a default domain, you will maintain only one copy of the http connector configuration. Any changes will require only the domain to the redeployed instead of all the applications.

You can start using domains in only three steps:

1) Create a Mule Domain project

2) Create the global connector configurations which needs to be shared across the applications inside the Mule Domain project

3) Modify the value of domain in mule-deploy.properties file of the applications


MULE DOMAIN
Shared Connector Configurations
Shared Properties
Shared Resources
MULE APP 1
Flow1
Global Connector configurations
FlowN
MULE APP 2
Flow1
Global Connector configurations
FlowN

NO.50 Refer to the exhibit.

A Mule application is deployed to a multi-node Mule runtime cluster. The Mule application uses the competing consumer pattern among its cluster replicas to receive JMS messages from a JMS queue. To process each received JMS message, the following steps are performed in a flow:

Step l: The JMS Correlation ID header is read from the received JMS message.

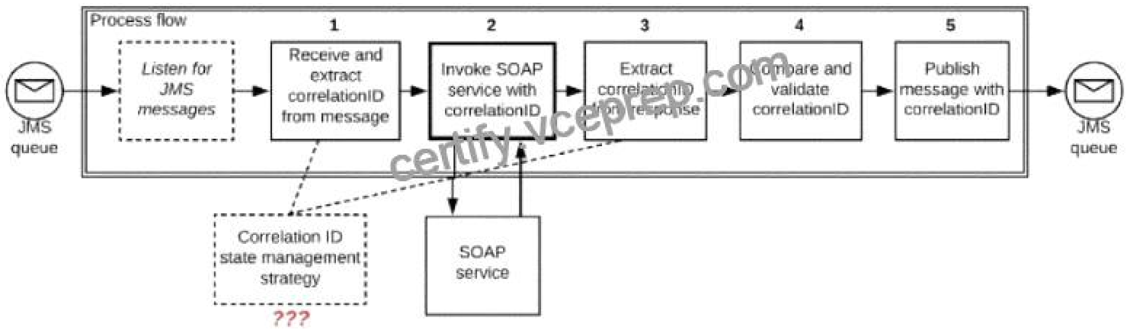
Step 2: The Mule application invokes an idempotent SOAP webservice over HTTPS, passing the JMS Correlation ID as one parameter in the SOAP request.

Step 3: The response from the SOAP webservice also returns the same JMS Correlation ID.

Step 4: The JMS Correlation ID received from the SOAP webservice is validated to be identical to the JMS Correlation ID received in Step 1.

Step 5: The Mule application creates a response JMS message, setting the JMS Correlation ID message header to the validated JMS Correlation ID and publishes that message to a response JMS queue.

Where should the Mule application store the JMS Correlation ID values received in Step 1 and Step 3 so that the validation in Step 4 can be performed, while also making the overall Mule application highly available, fault-tolerant, performant, and maintainable?


Process flow
JMS queue
Listen for JMS messages
1
Receive and extract correlationID from message
2
Invoke SOAP service with correlationID
3
Extract correlationID from response
4
Compare and validate correlationID
5
Publish message with correlationID
JMS queue
Correlation ID state management strategy
???
SOAP service

* Both Correlation ID values should be stored in a persistent object store
* Both Correlation ID values should be stored In a non-persistent object store
* The Correlation ID value in Step 1 should be stored in a persistent object store The Correlation ID value in step 3 should be stored as a Mule event vanable/attnbute
* Both Correlation ID values should be stored as Mule event vanabtes/attnbutes

NO.51 How are the API implementation , API client, and API consumer combined to invoke and process an API ?
* The API consumer creates an API implementation , which receives API invocations from an API such that they are processed for an API client
* The API consumer creates an API client which sends API invocations to an API such that they are processed by an API implementation
* An API client creates an API consumer, which receives API invocation from an API such that they are processed for an API implementation
* The API client creates an API consumer which sends API invocations to an API such that they are processed by API implementation
* Correct answer is The API consumer creates an API client which sends API invocations to an API such that they are processed by an API implementation This is based on below definitions API client * An application component * that accesses a service * by invoking an API of that service &#8211; by definition of the term API over HTTP API consumer * A business role, which is often assigned to an individual * that develops API clients, i.e., performs the activities necessary for enabling an API client

NO.52 A Mule application is built to support a local transaction for a series of operations on a single database. The Mule application has a Scatter-Gather that participates in the local transaction.

What is the behavior of the Scatter-Gather when running within this local transaction?
* Execution of each route within the Scatter-Gather occurs sequentially Any error that occurs inside the Scatter-Gather will result in a rollback of all the database operations
* Execution of all routes within the Scatter-Gather occurs in parallel Any error that occurs inside the Scatter-Gather will result in a

rollback of all the database operations
* Execution of each route within the Scatter-Gather occurs sequentially Any error that occurs inside the Scatter-Gather will NOT result in a rollback of any of the database operations
* Execution of each route within the Scatter-Gather occurs in parallel Any error that occurs inside the Scatter-Gather will NOT result in a rollback of any of the database operations

NO.53 An organization currently uses a multi-node Mule runtime deployment model within their datacenter, so each Mule runtime hosts several Mule applications. The organization is planning to transition to a deployment model based on Docker containers in a Kubernetes cluster. The organization has already created a standard Docker image containing a Mule runtime and all required dependencies (including a JVM), but excluding the Mule application itself.

What is an expected outcome of this transition to container-based Mule application deployments?
* Required redesign of Mule applications to follow microservice architecture principles
* Required migration to the Docker and Kubernetes-based Anypoint Platform &#8211; Private Cloud Edition
* Required change to the URL endpoints used by clients to send requests to the Mule applications
* Guaranteed consistency of execution environments across all deployments of a Mule application
* Organization can continue using existing load balancer even if backend application changes are there. So option A is ruled out.

* As Mule runtime is within their datacenter, this model is RTF and not PCE. So option C is ruled out.

Mule runtime deployment model within their datacenter, so each Mule runtime hosts several Mule applications &#8212; This mean PCE or Hybird not RTF &#8211; Also mentioned in Question is that &#8211; Mule runtime is hosting several Mule Application, so that also rules out RTF and as for hosting multiple Application it will have Domain project which need redesign to make it microservice architecture

&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212;&#8212; Correct answer: Required redesign of Mule applications to follow microservice architecture principles

NO.54 A mule application is deployed to a Single Cloudhub worker and the public URL appears in Runtime Manager as the APP URL.

Requests are sent by external web clients over the public internet to the mule application App url. Each of these requests routed to the HTTPS Listener event source of the running Mule application.

Later, the DevOps team edits some properties of this running Mule application in Runtime Manager.

Immediately after the new property values are applied in runtime manager, how is the current Mule application deployment affected and how will future web client requests to the Mule application be handled?
* Cloudhub will redeploy the Mule application to the OLD Cloudhub worker New web client requests will RETURN AN ERROR until the Mule application is redeployed to the OLD Cloudhub worker
* CloudHub will redeploy the Mule application to a NEW Cloudhub worker New web client requests will RETURN AN ERROR until the NEW Cloudhub worker is available
* Cloudhub will redeploy the Mule application to a NEW Cloudhub worker New web client requests are ROUTED to the OLD Cloudhub worker until the NEW Cloudhub worker is available.
* Cloudhub will redeploy the mule application to the OLD Cloudhub worker New web client requests are ROUTED to the OLD Cloudhub worker BOTH before and after the Mule application is redeployed.

NO.55 A Mule application uses the Database connector.

What condition can the Mule application automatically adjust to or recover from without needing to restart or redeploy the Mule application?
* One of the stored procedures being called by the Mule application has been renamed
* The database server has been updated and hence the database driver library/JAR needs a minor version upgrade
* The database server was unavailable for four hours due to a major outage but is now fully operational again
* The credentials for accessing the database have been updated and the previous credentials are no longer valid

NO.56 An API has been unit tested and is ready for integration testing. The API is governed by a Client ID Enforcement policy in all environments.

What must the testing team do before they can start integration testing the API in the Staging environment?
* They must access the API portal and create an API notebook using the Client ID and Client Secret supplied by the API portal in the Staging environment
* They must request access to the API instance in the Staging environment and obtain a Client ID and Client Secret to be used for testing the API
* They must be assigned as an API version owner of the API in the Staging environment
* They must request access to the Staging environment and obtain the Client ID and Client Secret for that environment to be used for testing the API

NO.57 What condition requires using a CloudHub Dedicated Load Balancer?
* When cross-region load balancing is required between separate deployments of the same Mule application
* When custom DNS names are required for API implementations deployed to customer-hosted Mule runtimes
* When API invocations across multiple CloudHub workers must be load balanced
* When server-side load-balanced TLS mutual authentication is required between API implementations and API clients

Correct answer is When server-side load-balanced TLS mutual authentication is required between API implementations and API clients CloudHub dedicated load balancers (DLBs) are an optional component of Anypoint Platform that enable you to route external HTTP and HTTPS traffic to multiple Mule applications deployed to CloudHub workers in a Virtual Private Cloud (VPC). Dedicated load balancers enable you to: * Handle load balancing among the different CloudHub workers that run your application. * Define SSL configurations to provide custom certificates and optionally enforce two-way SSL client authentication. * Configure proxy rules that map your applications to custom domains. This enables you to host your applications under a single domain