

Salesforce.MuleSoft-Integration-Architect-I.v2026-07-22.q100

Exam Code:	MuleSoft-Integration-Architect-I
Exam Name:	Salesforce Certified MuleSoft Integration Architect I
Certification Provider:	Salesforce
Free Question Number:	100
Version:	v2026-07-22
# of views:	122
# of Questions views:	1000
https://www.freeqas.com/qa/Salesforce/MuleSoft-Integration-Architect-I/Salesforce.MuleSoft-Integration-Architect-I.v2026-07-22.q100.html	

NEW QUESTION: 1

An IT integration delivery team begins a project by gathering all of the requirements, and proceeds to execute the remaining project activities as sequential, non-repeating phases. Which IT project delivery methodology is this team following?

- A. Kanban
- B. Scrum
- C. Waterfall
- D. Agile

Answer: C (LEAVE A REPLY)

The described approach of gathering all requirements upfront and proceeding through sequential, non-repeating phases is characteristic of the Waterfall methodology. Waterfall is a linear project management approach where each phase must be completed before the next one begins, and there is typically no going back to a previous phase once it is completed. This methodology is well-suited for projects with well-defined requirements and deliverables, but it can be inflexible if changes or new requirements arise during the project.

References:

Waterfall Model

Waterfall vs. Agile: Which is the Right Development Methodology for Your Project?

NEW QUESTION: 2

An airline is architecting an API connectivity project to integrate its flight data into an online aggregation website. The interface must allow for secure communication high-performance and asynchronous message exchange.

What are suitable interface technologies for this integration assuming that Mulesoft fully supports these technologies and that Anypoint connectors exist for these interfaces?

- A. AsyncAPI over HTTPS
AMQP with RabbitMQ JSON/REST over HTTPS
- B. XML over ActiveMQ XML over SFTP XML/REST over HTTPS
- C. CSV over FTP YAM L over TLS JSON over HTTPS
- D. SOAP over HTTPS HOP over TLS gRPC over HTTPS

Answer: A (LEAVE A REPLY)

For integrating flight data into an online aggregation website with secure, high-performance, and asynchronous message exchange, the most suitable interface technologies supported by MuleSoft are:

- * AsyncAPI over HTTPS: AsyncAPI is a specification for asynchronous APIs, which is ideal for high- performance, event-driven communication. Using HTTPS ensures secure communication.
- * AMQP with RabbitMQ: AMQP (Advanced Message Queuing Protocol) is a protocol for message- oriented middleware. RabbitMQ is a popular message broker that supports AMQP, allowing for reliable and asynchronous message exchange.
- * JSON/REST over HTTPS: JSON is a lightweight data-interchange format, and REST (Representational State Transfer) is an architectural style for designing networked applications. Using HTTPS ensures secure communication.

Implementation Steps:

- * Define the AsyncAPI specification for the flight data events and set up the infrastructure to handle these events over HTTPS.
- * Set up RabbitMQ and configure AMQP for message queuing. Ensure that your MuleSoft application is capable of consuming and producing messages using the RabbitMQ connector.
- * Implement RESTful APIs using JSON over HTTPS to expose the flight data securely.

These technologies collectively ensure secure, asynchronous, and high-performance integration.

References:

- * MuleSoft Documentation: AsyncAPI
- * MuleSoft Documentation: RabbitMQ Connector
- * MuleSoft Documentation: JSON and REST APIs

NEW QUESTION: 3

How does timeout attribute help inform design decisions while using JMS connector listening for incoming messages in an extended architecture (XA) transaction?

- A.** The timeout defines the time that is allowed to pass without the transaction ending explicitly and after the timeout expires, the transaction rolls back
- B.** The time allowed to pass between committing the transaction and the completion of the mule flow and then after the timeout flow processing triggers an error
- C.** The timeout specifies the time allowed to pass between receiving JMS messages on the same JMS connection and then after the timeout new JMS connection is established
- D.** After the timeout is exceeded, stale JMS consumer threads are destroyed and new threads are created

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 4

According to MuleSoft, which deployment characteristic applies to a microservices application architecture?

- A.** Services exist as independent deployment artifacts and can be scaled -independently of other services
- B.** A deployment to enhance one capability requires a redeployment of all capabilities
- C.** All services of an application can be deployed together as single Java WAR file
- D.** Core business capabilities are encapsulated in a single, deployable application

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 5

Which Anypoint Platform component helps integration developers discovers and share reusable APIs, connectors, and templates?

- A.** Anypoint Studio
- B.** API Manager
- C.** Design Center
- D.** Anypoint Exchange

Answer: **D** ([LEAVE A REPLY](#))

NEW QUESTION: 6

An organization's security policies mandate complete control of the login credentials used to log in to Anypoint Platform. What feature of Anypoint Platform should be used to meet this requirement?

- A. Enterprise Security Module
- B. Client ID Secret
- C. Federated Identity Management
- D. Federated Client Management

Answer: C ([LEAVE A REPLY](#))

Correct answer is Federated Identity Management As the Anypoint Platform organization administrator, you can configure identity management in Anypoint Platform to set up users for single sign-on (SSO). Configure identity management using one of the following single sign-on standards: OpenID Connect: End user identity verification by an authorization server including SSO SAML 2.0: Web-based authorization including cross- domain SSO Where as Client Management is where 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

NEW QUESTION: 7

Refer to the exhibit.

This Mule application is deployed to multiple Cloudhub workers with persistent queue enabled. The retrievefile flow event source reads a CSV file from a remote SFTP server and then publishes each record in the CSV file to a VM queue. The processCustomerRecords flow's VM Listener receives messages from the same VM queue and then processes each message separately.

How are messages routed to the cloudhub workers as messages are received by the VM Listener?

- A. Each messages routes to ONE of the available Clouhub workers in a NON- DETERMINSTIC non round-robin fashion thereby APPROXIMATELY BALANCING messages among the cloudhub workers
- B. Each message is routed to the SAME Cloudhub worker that retrieved the file, thereby BINDING ALL messages to ONLY that ONE Cloudhub worker
- C. Each message is duplicated to ALL of the Cloudhub workers, thereby SHARING EACH message with ALL the Cloudhub workers.
- D. Each message is routed to ONE of the Cloudhub workers in a DETERMINSTIC round robin fashion thereby EXACTLY BALANCING messages among the cloudhub workers

Answer: A ([LEAVE A REPLY](#))

NEW QUESTION: 8

In Anypoint Platform, a company wants to configure multiple identity providers (IdPs) for multiple lines of business (LOBs). Multiple business groups, teams, and environments have been defined for these LOBs.

What Anypoint Platform feature can use multiple IdPs across the company's business groups, teams, and environments?

- A. MuleSoft-hosted (CloudHub) dedicated load balancers
- B. Client (application) management
- C. Virtual private clouds
- D. Permissions

Answer: A ([LEAVE A REPLY](#))

To use a dedicated load balancer in your environment, you must first create an Anypoint VPC. Because you can associate multiple environments with the same Anypoint VPC, you can use the same dedicated load balancer for your different environments.

Reference: <https://docs.mulesoft.com/runtime-manager/cloudhub-dedicated-load-balancer>

NEW QUESTION: 9

A company is modernizing its legal systems to accelerate access to applications and data while supporting the adoption of new technologies. The key to achieving this business goal is unlocking the companies' key systems and data including microservices running under Docker and Kubernetes containers using APIs.

Considering the current aggressive backlog and project delivery requirements the company wants to take a strategic approach in the first phase of its transformation projects by quickly deploying APIs in a mule runtime that are able to scale, connect to on-premises systems and migrate as needed.

Which runtime deployment option supports company's goals?

- A. Customer hosted self provisioned runtimes
- B. Cloudhub runtimes
- C. Runtime fabric on self managed Kubernetes
- D. Runtime fabric on VMware metal

Answer: C (LEAVE A REPLY)

To support the company's goals of unlocking key systems and data, quickly deploying scalable APIs, and connecting to on-premises systems, while also preparing for future migrations, the best runtime deployment option is using Runtime Fabric on self-managed Kubernetes. Here's why:

- * Scalability: Kubernetes is designed to scale applications easily and efficiently. By deploying Mule runtimes on a self-managed Kubernetes cluster, the company can dynamically scale its APIs based on demand, ensuring performance and reliability.
- * Flexibility: Running on self-managed Kubernetes allows the company to have control over the infrastructure. They can customize the environment to meet their specific needs, integrate with existing on-premises systems, and support their microservices architecture running in Docker containers.
- * Future-Proofing: Kubernetes supports hybrid and multi-cloud deployments, making it easier to migrate workloads between different environments as needed. This aligns with the company's goal of being able to migrate their systems as part of their strategic approach.
- * Efficiency: By leveraging Kubernetes, the company can automate deployment, scaling, and management of containerized applications, reducing the burden on IT and DevOps teams.

References:

- * MuleSoft Documentation on Runtime Fabric: <https://docs.mulesoft.com/runtime-fabric/1.9/>
- * Kubernetes Documentation: <https://kubernetes.io/docs/home/>

NEW QUESTION: 10

An organization uses a set of customer-hosted Mule runtimes that are managed using the Mulesoft-hosted control plane. What is a condition that can be alerted on from Anypoint Runtime Manager without any custom components or custom coding?

- A. When a Mule runtime on a given customer-hosted server is experiencing high memory consumption during certain periods
- B. When an SSL certificate used by one of the deployed Mule applications is about to expire
- C. When the Mule runtime license installed on a Mule runtime is about to expire
- D. When a Mule runtime's customer-hosted server is about to run out of disk space

Answer: A (LEAVE A REPLY)

Correct answer is When a Mule runtime on a given customer-hosted server is experiencing high memory consumption during certain periods Using Anypoint Monitoring, you can configure two different types of alerts: Basic alerts for servers and Mule apps Limit per organization: Up to 50 basic alerts for users who do not have a Titanium subscription to Anypoint Platform You can set up basic alerts to trigger email notifications when a metric you are measuring passes a specified threshold. You can create basic alerts for the following metrics for servers or Mule apps: For on-premises servers and CloudHub apps: * CPU utilization * Memory utilization * Thread count Advanced alerts for graphs in custom dashboards in Anypoint Monitoring.

You must have a Titanium subscription to use this feature. Limit per organization: Up to 20 advanced alerts

NEW QUESTION: 11

An organization is designing a Mule application to periodically poll an SFTP location for new files containing sales order records and then process those sales orders. Each sales order must be processed exactly once.

To support this requirement, the Mule application must identify and filter duplicate sales orders on the basis of a unique ID contained in each sales order record and then only send the new sales orders to the downstream system.

What is the most idiomatic (used for its intended purpose) Anypoint connector, validator, or scope that can be configured in the Mule application to filter duplicate sales orders on the basis of the unique ID field contained in each sales order record?

- A.** Configure a Cache scope to filter and store each record from the received file by the order ID
- B.** Configure a Database connector to filter and store each record by the order ID
- C.** Configure an Idempotent Message Validator component to filter each record by the order ID
- D.** Configure a watermark In an On New or Updated File event source to filter unique records by the order ID

Answer: C (LEAVE A REPLY)

Reference: <https://docs.mulesoft.com/mule-runtime/3.9/idempotent-filter>

NEW QUESTION: 12

When the mule application using VM is deployed to a customer-hosted cluster or multiple cloudhub workers, how are messages consumed by the Mule engine?

- A.** in non-deterministic way
- B.** by starting an XA transaction for each new message
- C.** in a deterministic way
- D.** the primary only in order to avoid duplicate processing

Answer: A (LEAVE A REPLY)

When a Mule application using VM (Virtual Machine) queues is deployed to a customer-hosted cluster or multiple CloudHub workers, the messages are consumed by the Mule engine in a non-deterministic way. Here' s an in-depth explanation:

* VM Queues Overview:

* VM queues in Mule applications are used for intra-application communication, allowing different flows within the same application to exchange messages.

* Deployment in Clusters or Multiple Workers:

* Customer-Hosted Cluster: In a customer-hosted cluster, multiple Mule runtime instances work together to process messages. Each instance can pick up messages from the VM queue.

* CloudHub Workers: When deployed on CloudHub, multiple worker instances can run the same Mule application, and each worker can access the VM queue.

* Non-Deterministic Message Consumption:

* Load Distribution: Messages are distributed among the available nodes or workers based on their availability. This means any node or worker that is ready to process a message can pick it up from the queue.

* No Guaranteed Order: Because any available node or worker can consume messages, the order in which messages are processed is not guaranteed, making the consumption non-deterministic.

* Parallel Processing: This approach allows for parallel processing of messages, which improves the scalability and throughput of the application.

* Advantages:

* Scalability: By allowing multiple nodes or workers to process messages, the system can handle increased load more effectively.

* Fault Tolerance: If one node or worker fails, other nodes/workers can continue processing messages from the VM queue, providing higher availability.

* Considerations:

* Idempotency: Ensure that the processing logic is idempotent, meaning that processing the same message more than once does not produce different outcomes. This is crucial in a non-deterministic consumption environment to avoid issues with data consistency.

* Transaction Management: Proper transaction management should be in place to handle scenarios where a message might need to be reprocessed due to errors.

MuleSoft Documentation on VM Connector

MuleSoft Documentation on High Availability Clustering

MuleSoft Documentation on Deploying to CloudHub

NEW QUESTION: 13

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?

- A.** One of the stored procedures being called by the Mule application has been renamed
- B.** The database server was unavailable for four hours due to a major outage but is now fully operational again
- C.** The credentials for accessing the database have been updated and the previous credentials are no longer valid
- D.** The database server has been updated and hence the database driver library/JAR needs a minor version upgrade

Answer: B ([LEAVE A REPLY](#))

* Any change in the application will require a restart except when the issue outside the app. For below situations , you would need to redeploy the code after doing necessary changes

-- One of the stored procedures being called by the Mule application has been renamed. In this case, in the Mule application you will have to do changes to accommodate the new stored procedure name.

-- Required redesign of Mule applications to follow microservice architecture principles. As code is changed, deployment is must

-- If the credentials changed and you need to update the connector or the properties.

-- The credentials for accessing the database have been updated and the previous credentials are no longer valid. In this situation you need to restart or redeploy depending on how credentials are configured in Mule application.

* So Correct answer is The database server was unavailable for four hours due to a major outage but is now fully operational again as this is the only external issue to application.

NEW QUESTION: 14

Which Salesforce API is invoked to deploy, retrieve, create or delete customization information such as custom object definitions using a Mule Salesforce connector in a Mule application?

- A.** Metadata API
- B.** REST API
- C.** SOAP API
- D.** Bulk API

Answer: A ([LEAVE A REPLY](#))

* urpose of Metadata API:

* The Metadata API is specifically designed to manage customization and configuration data in Salesforce. It allows users to deploy, retrieve, create, update, or delete customization information such as custom object definitions, page layouts, and more.

* Use in MuleSoft:

* When using the Mule Salesforce connector in a Mule application, invoking the Metadata API enables you to programmatically manage and deploy Salesforce metadata. This is essential for tasks such as deploying new custom objects, updating existing configurations, or deleting obsolete customizations.

* Why Not Other APIs:

* REST API: Primarily used for accessing Salesforce data and interacting with standard and custom objects.

* SOAP API: Similar to REST API but uses SOAP protocol; also mainly for data access and manipulation.

* Bulk API: Designed for handling large volumes of data for insert, update, delete operations but not for metadata management.

References:

* Salesforce Metadata API Overview: Salesforce Metadata API

* MuleSoft Documentation on Salesforce Connector

NEW QUESTION: 15

An organization plans to use the Anypoint Platform audit logging service to log Anypoint MQ actions.

What consideration must be kept in mind when leveraging Anypoint MQ Audit Logs?

- A. Anypoint MQ Audit Logs include logs for sending, receiving, or browsing messages
- B. Anypoint MQ Audit Logs include logs for failed Anypoint MQ operations
- C. Anypoint MQ Audit Logs include logs for queue create, delete, modify, and purge operations

Answer: C (LEAVE A REPLY)

When leveraging Anypoint MQ Audit Logs, it's important to note that they include logs for operations such as creating, deleting, modifying, and purging queues. These logs are crucial for auditing and monitoring the state and changes made to the message queues within Anypoint MQ. However, they do not include logs for individual message actions like sending, receiving, or browsing messages.

References

* MuleSoft Documentation on Anypoint MQ Audit Logs

* Anypoint Platform Audit Logging Overview

NEW QUESTION: 16

A Mule application is deployed to a cluster of two(2) customer-hosted Mule runtimes. Currently the node name Alice is the primary node and node named Bob is the secondary node. The mule application has a flow that polls a directory on a file system for new files.

The primary node Alice fails for an hour and then restarted.

After the Alice node completely restarts, from what node are the files polled, and what node is now the primary node for the cluster?

- A. Files are polled from Alice node Alice is now the primary node
- B. Files are polled from Bob node Alice is now the primary node
- C. Files are polled from Alice node Bob is now the primary node
- D. Files are polled from Bob node Bob is now the primary node

Answer: D (LEAVE A REPLY)

* Mule High Availability Clustering provides basic failover capability for Mule. * When the primary Mule Runtime becomes unavailable, for example, because of a fatal JVM or hardware failure or it's taken offline for maintenance, a backup Mule Runtime immediately becomes the primary node and resumes processing where the failed instance left off. * After a system administrator recovers a failed Mule Runtime server and puts it back online, that server automatically becomes the backup node. In this case, Alice, once up, will become backup
----- Reference: <https://docs.mulesoft.com/mule-runtime/4.3/hadr-guide> So correct choice is : Files are polled from Bob node Bob is now the primary node

Valid MuleSoft-Integration-Architect-I Dumps shared by PrepPdf.com for Helping Passing MuleSoft-Integration-Architect-I Exam! PrepPdf.com now offer the **newest MuleSoft-Integration-Architect-I exam dumps**, the PrepPdf.com MuleSoft-Integration-Architect-I exam **questions have been updated** and **answers have been corrected** get the **newest** PrepPdf.com MuleSoft-Integration-Architect-I dumps with Test Engine here: <https://www.preppdf.com/Salesforce/MuleSoft-Integration-Architect-I-prepaway-exam-dumps.html> (275 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 17

When a Mule application using VM queues is deployed to a customer-hosted cluster or multiple CloudHub v1.

0 workers/replicas, how are messages consumed across the nodes?

- A. Sequentially, from a dedicated Anypoint MQ queue
- B. Sequentially, only from the primary node
- C. In a non-deterministic way

D. Round-robin, within an XA transaction

Answer: ([SHOW ANSWER](#))

When a Mule application using VM queues is deployed to a customer-hosted cluster or multiple CloudHub v1.

0 workers/replicas, messages are consumed in a non-deterministic way. This means that any of the nodes in the cluster or any of the workers can consume the messages from the VM queues, but there is no guaranteed order or specific pattern (such as round-robin or sequential processing).

This non-deterministic message consumption helps in distributing the load and handling messages more efficiently across multiple nodes or workers, improving the scalability and reliability of the application.

References

- * MuleSoft Documentation on VM Queues and Clustering
- * Best Practices for Deploying Mule Applications in Clusters

NEW QUESTION: 18

An organization's IT team must secure all of the internal APIs within an integration solution by using an API proxy to apply required authentication and authorization policies.

Which integration technology, when used for its intended purpose, should the team choose to meet these requirements if all other relevant factors are equal?

- A.** Robotic Process Automation (RPA)
- B.** API Management (APIM)
- C.** Integration Platform-as-a-service (PaaS)
- D.** Electronic Data Interchange (EDI)

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 19

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's source-code management (SCM) system in this context?

- A.** 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
- B.** Organizations need to use Anypoint Exchange as the main SCM system to centralize versioning and avoid code duplication
- C.** 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
- D.** 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

Answer: **B** ([LEAVE A REPLY](#))

* Organization should continue to use SCM system of their choice, in addition to keeping source code for these asset types in Anypoint Exchange, thereby enabling parallel development, branching.

* Reason is that Anypoint exchange is not full fledged version repositories like GitHub.

* But at same time it is tightly coupled with Mule assets

NEW QUESTION: 20

An integration Mule application consumes and processes a list of rows from a CSV file. Each row must be read from the CSV file, validated, and the row data sent to a JMS queue, in the exact order as in the CSV file.

If any processing step for a row falls, then a log entry must be written for that row, but processing of other rows must not be affected.

What combination of Mule components is most idiomatic (used according to their intended purpose) when Implementing the above requirements?

- A.** Scatter-Gather component On Error Continue scope
- B.** VM connector first Successful scope On Error Propagate scope

C. For Each scope On Error Continue scope

D. Async scope On Error Propagate scope

Answer: C ([LEAVE A REPLY](#))

* On Error Propagate halts execution and sends error to the client. In this scenario it's mentioned that

"processing of other rows must not be affected" so Option B and C are ruled out.

* Scatter gather is used to club multiple responses together before processing. In this scenario, we need sequential processing. So option A is out of choice.

* Correct answer is For Each scope & On Error Continue scope Below requirement can be fulfilled in the below way

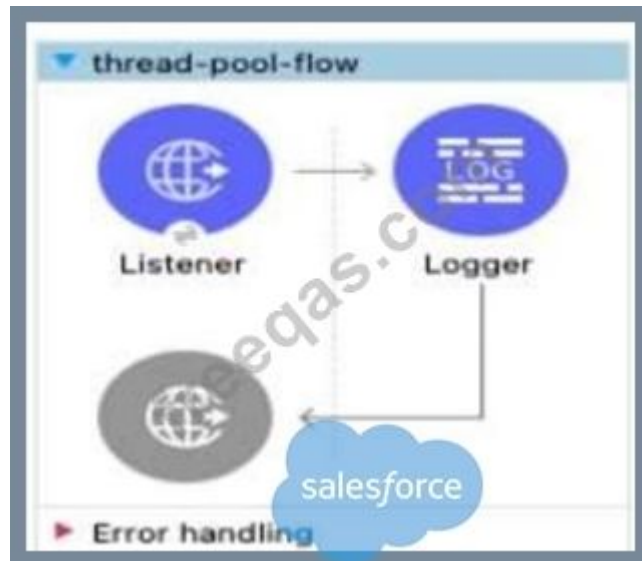
1) Using For Each scope , which will send each row from csv file sequentially. each row needs to be sent sequentially as requirement is to send the message in exactly the same way as it is mentioned in the csv file

2) Also other part of requirement is if any processing step for a row fails then it should log an error but should not affect other record processing . This can be achieved using On error Continue scope on these set of activities. so that error will not halt the processing. Also logger needs to be added in error handling section so that it can be logged.

* Attaching diagram for reference. Here it's try scope, but similar would be the case with For Each loop.

NEW QUESTION: 21

Refer to the exhibit.



A customer is running Mule applications on Runtime Fabric for Self-Managed Kubernetes (RTF-BYOKS) in a multi-cloud environment.

Based on this configuration, how do Agents and Runtime Manager communicate, and what is exchanged between them?

A. Shared NIO Selector Pool, CPU_LITE

B. CPU_LITE, CPU_INTENSIVE

C. UBER, Dedicated NIO Selector Pool

D. BLOCKING_IO, UBER

Answer: (SHOW ANSWER)

NEW QUESTION: 22

A mule application must periodically process a large dataset which varies from 6 GB to 8 GB from a back-end database and write transform data to an FTPS server using a properly configured batch job scope.

The performance requirements of an application are approved to run in the cloud hub 0.2 vCore with 8 GB storage capacity and currency requirements are met.

How can the high rate of records be effectively managed in this application?

- A. Use streaming with a file storage repeatable strategy for reading records from the database and batch aggregator with streaming to write to FTPS
- B. Use streaming with an in-memory reputable store strategy for reading records from the database and batch aggregator with streaming to write to FTPS
- C. Use streaming with a file store repeatable strategy for reading records from the database and batch aggregator with an optimal size
- D. Use streaming with a file store repeatable strategy reading records from the database and batch aggregator without any required configuration

Answer: A ([LEAVE A REPLY](#))

For handling large datasets in a Mule application, streaming is an effective strategy. Streaming allows the application to process large amounts of data in chunks, reducing memory usage and improving performance.

Using a file storage repeatable strategy for reading records from the database ensures that the data is read in manageable chunks and stored temporarily in files, which can be re-read if necessary, enhancing reliability.

The batch aggregator with streaming to write to an FTPS server ensures that data is written in chunks, aligning with the processing capabilities of the application running in CloudHub with 0.2 vCore and 8 GB storage. This configuration optimizes performance by balancing the load and managing the dataset size effectively, ensuring that the high rate of records can be processed and written to the FTPS server without overwhelming the system.

References:

- * MuleSoft Documentation on Streaming
- * MuleSoft Documentation on Batch Processing

NEW QUESTION: 23

An organization designing a hybrid, load balanced, single cluster production environment. Due to performance service level agreement goals, it is looking into running the Mule applications in an active-active multi node cluster configuration.

What should be considered when running its Mule applications in this type of environment?

- A. All event sources, regardless of time , can be configured as the target source by the primary node in the cluster
- B. An external load balancer is required to distribute incoming requests throughout the cluster nodes
- C. A Mule application deployed to multiple nodes runs in an isolation from the other nodes in the cluster
- D. Although the cluster environment is fully installed configured and running, it will not process any requests until an outage condition is detected by the primary node in the cluster.

Answer: B ([LEAVE A REPLY](#))

In a hybrid, load-balanced, single cluster production environment running Mule applications in an active- active multi-node configuration, several considerations are critical for ensuring performance and reliability.

The key consideration is the use of an external load balancer:

- * Active-Active Multi-Node Cluster Configuration:
 - * An active-active cluster means that all nodes are actively handling traffic, providing high availability and better resource utilization.
- * External Load Balancer Requirement:
 - * Distribution of Requests: An external load balancer is essential to evenly distribute incoming requests across all active nodes in the cluster. This prevents any single node from becoming a bottleneck and ensures balanced load distribution.
 - * Scalability and Failover: The load balancer provides scalability by allowing more nodes to be added seamlessly. It also handles failover, rerouting traffic to healthy nodes if one node goes down.
- * Load Balancer Configuration:
 - * Setup: Configure the load balancer to include all the nodes of the Mule cluster.
 - * Health Checks: Implement health checks to monitor the status of each node and ensure traffic is only directed to healthy nodes.
 - * Session Persistence: If required, enable session persistence (sticky sessions) to ensure that user sessions remain consistent across requests.
- * Mule Application Isolation:
 - * Each Mule application instance runs in isolation but shares the same configuration and state. This isolation ensures that the failure of one node does not impact others.

* Handling Requests:

* In an active-active configuration, all nodes handle incoming requests simultaneously. The load balancer's role is to manage the distribution of these requests efficiently.

* Benefits:

* High Availability: Ensures that the system remains available even if some nodes fail.

* Improved Performance: Balances the load, preventing any single node from being overwhelmed.

* Scalability: Makes it easy to scale horizontally by adding more nodes.

MuleSoft Documentation on Mule Clustering

Best Practices for Load Balancing

NEW QUESTION: 24

A payment processing company has implemented a Payment Processing API Mule application to process credit card and debit card transactions, Because the Payment Processing API handles highly sensitive information, the payment processing company requires that data must be encrypted both In-transit and at-rest.

To meet these security requirements, consumers of the Payment Processing API must create request message payloads in a JSON format specified by the API, and the message payload values must be encrypted.

How can the Payment Processing API validate requests received from API consumers?

A. A Transport Layer Security (TLS) - Inbound policy can be applied in API Manager to decrypt the message payload and the Mule application implementation can then use the JSON Validation module to validate the JSON data

B. The Mule application implementation can use the APIkit module to decrypt and then validate the JSON data

C. The Mule application implementation can use the Validation module to decrypt and then validate the JSON data

D. The Mule application implementation can use DataWeave to decrypt the message payload and then use the JSON Scheme Validation module to validate the JSON data

Answer: A ([LEAVE A REPLY](#))

To ensure that data is encrypted both in-transit and at-rest, and to validate incoming requests to the Payment Processing API, the following approach is recommended:

* TLS Inbound Policy: Apply a Transport Layer Security (TLS) - Inbound policy in API Manager. This policy ensures that the data is encrypted during transmission and can be decrypted by the API Manager before it reaches the Mule application.

* Decryption: With the TLS policy applied, the message payload is decrypted when it is received by the API Manager.

* JSON Validation: After decryption, the Mule application can use the JSON Validation module to validate the structure and content of the JSON data. This ensures that the payload conforms to the specified format and contains valid data.

This approach ensures that data is securely transmitted and properly validated upon receipt.

References:

* Transport Layer Security (TLS) Policies

* JSON Validation Module

NEW QUESTION: 25

Refer to the exhibit.



The HTTP Listener and the Logger are being handled from which thread pools respectively?

- A. CPU_INTENSIVE and Dedicated Selector pool
- B. UBER and NONBLOCKING
- C. Shared Selector Pool and CPU LITE
- D. BLOCKING_IO and UBER

Answer: (SHOW ANSWER)

In Mule applications, different components are handled by specific thread pools to optimize performance and resource utilization.

* HTTP Listener: This component handles incoming HTTP requests and is managed by the Shared Selector Pool. The Shared Selector Pool is responsible for handling non-blocking IO operations efficiently.

* Logger: The Logger component is lightweight and does not perform CPU-intensive operations. It is managed by the CPU_LITE thread pool, which is designed for lightweight CPU operations.

This separation ensures that IO-bound operations do not block CPU-bound operations, maintaining optimal performance and responsiveness in the application.

References:

- * MuleSoft Threading and Thread Pools
- * MuleSoft HTTP Listener Documentation

NEW QUESTION: 26

Refer to the exhibit.

An organization is sizing an Anypoint VPC for the non-production deployments of those Mule applications that connect to the organization's on-premises systems. This applies to approx. 60 Mule applications. Each application is deployed to two CloudHub i workers. The organization currently has three non-production environments (DEV, SIT and UAT) that share this VPC. The AWS region of the VPC has two AZs.

The organization has a very mature DevOps approach which automatically progresses each application through all non-production environments before automatically deploying to production. This process results in several Mule application deployments per hour, using CloudHub's normal zero-downtime deployment feature.

What is a CIDR block for this VPC that results in the smallest usable private IP address range?

- A. 10.0.0.0/26 (64 IPS)
- B. 10.0.0.0/25 (128 IPs)
- C. 10.0.0.0/24 (256 IPs)
- D. 10.0.0.0/22 (1024 IPs)

Answer: D (LEAVE A REPLY)

Mule applications are deployed in CloudHub workers and each worker is assigned with a dedicated IP * For zero downtime deployment, each worker in CloudHub needs additional IP addresses * A few IPs in a VPC are reserved for infrastructure (generally 2 IPs) * The IP addresses are usually in a private range with a subnet block specifier, such as 10.0.0.1/24 * The

smallest CIDR network subnet block you can assign for your VPC is /24 (256 IP addresses) (60*3 env * 2 worker per application) + 50% of (total) for zero downtime = 540 In this case correct answer is 10.0.0.0/22 as this provided 1024 IP's .

Other IP's are insufficient.

NEW QUESTION: 27

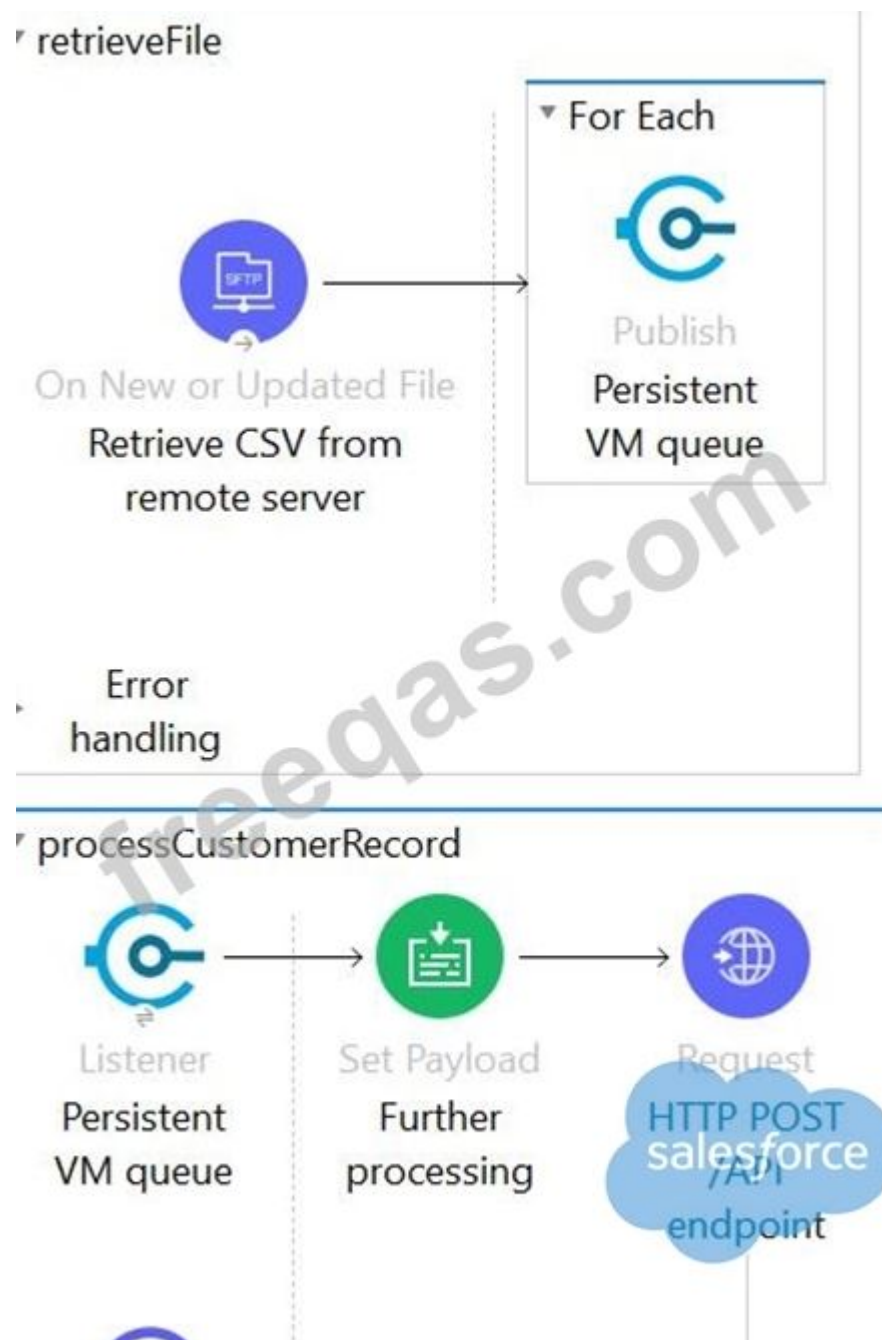
According to MuleSoft, which major benefit does a Center for Enablement (C4E) provide for an enterprise and its lines of business?

- A.** Enabling Edge security between the lines of business and public devices
- B.** Centralizing project management across the lines of business
- C.** Centrally managing return on investment (ROI) reporting from lines of business to leadership
- D.** Accelerating self-service by the lines of business

Answer: B ([LEAVE A REPLY](#))

NEW QUESTION: 28

Refer to the exhibit.



This Mule application is deployed to multiple Cloudhub workers with persistent queue enabled. The retrievefile flow event source reads a CSV file from a remote SFTP server and then publishes each record in the CSV file to a VM queue. The processCustomerRecords flow's VM Listener receives messages from the same VM queue and then processes each message separately.

How are messages routed to the cloudhub workers as messages are received by the VM Listener?

- A. Each message is routed to ONE of the Cloudhub workers in a DETERMINISTIC round robin fashion thereby EXACTLY BALANCING messages among the cloudhub workers
- B. Each message is duplicated to ALL of the Cloudhub workers, thereby SHARING EACH message with ALL the Cloudhub workers.
- C. Each message is routed to the SAME Cloudhub worker that retrieved the file, thereby BINDING ALL messages to ONLY that ONE Cloudhub worker
- D. Each messages routes to ONE of the available Clouhub workers in a NON- DETERMINISTIC non round-robin fashion thereby APPROXIMATELY BALANCING messages among the cloudhub workers

Answer: D ([LEAVE A REPLY](#))

NEW QUESTION: 29

A Mule application is deployed to a cluster of two(2) customer-hosted Mule runtimes. Currently the node name Alice is the primary node and node named Bob is the secondary node. The mule application has a flow that polls a directory on a file system for new files.

The primary node Alice fails for an hour and then restarted.

After the Alice node completely restarts, from what node are the files polled, and what node is now the primary node for the cluster?

A. Files are polled from Alice node

Alice is now the primary node

B. Files are polled from Bob node

Alice is now the primary node

C. Files are polled from Alice node

Bob is now the primary node

D. Files are polled from Bob node

Bob is now the primary node

Answer: (SHOW ANSWER)

* Mule High Availability Clustering provides basic failover capability for Mule. * When the primary Mule Runtime becomes unavailable, for example, because of a fatal JVM or hardware failure or it's taken offline for maintenance, a backup Mule Runtime immediately becomes the primary node and resumes processing where the failed instance left off. * After a system administrator recovers a failed Mule Runtime server and puts it back online, that server automatically becomes the backup node. In this case, Alice, once up, will become backup

----- Reference: <https://docs.mulesoft.com/mule-runtime/4.3/hadr-guide> So correct choice is : Files are polled from Bob node Bob is now the primary node

NEW QUESTION: 30

How does timeout attribute help inform design decisions while using JMS connector listening for incoming messages in an extended architecture (XA) transaction?

A. After the timeout is exceeded, stale JMS consumer threads are destroyed and new threads are created

B. The timeout specifies the time allowed to pass between receiving JMS messages on the same JMS connection and then after the timeout new JMS connection is established

C. The time allowed to pass between committing the transaction and the completion of the mule flow and then after the timeout flow processing triggers an error

D. The timeout defines the time that is allowed to pass without the transaction ending explicitly and after the timeout expires, the transaction rolls back

Answer: D (LEAVE A REPLY)

* Timeout Attribute:

* The timeout attribute in the JMS connector specifies the maximum time allowed for a transaction to be completed. If the transaction does not complete within this specified time, it will be considered a timeout.

* Transaction Management:

* In the context of an Extended Architecture (XA) transaction, this timeout ensures that if the transaction is not explicitly ended within the defined time, the transaction is rolled back to maintain data consistency.

* Implications of Timeout:

* If the timeout is exceeded, the JMS message processing is considered unsuccessful, and all operations within the transaction are undone, which helps in avoiding partial updates and maintaining a reliable state.

References:

* MuleSoft Documentation on JMS Connector

* Understanding XA Transactions

NEW QUESTION: 31

What is an example of data confidentiality?

- A. Signing a file digitally and sending it using a file transfer mechanism
- B. Encrypting a file containing personally identifiable information (PII)
- C. Providing a server's private key to a client for secure decryption of data during a two-way SSL handshake
- D. De-masking a person's Social Security number while inserting it into a database

Answer: B (LEAVE A REPLY)

Data confidentiality involves protecting information from unauthorized access and disclosure. Encrypting a file containing personally identifiable information (PII) is a prime example of ensuring data confidentiality.

Encryption transforms the data into a format that is unreadable without the appropriate decryption key, thereby safeguarding sensitive information such as PII from being accessed by unauthorized parties. This measure is essential for compliance with data protection regulations and maintaining the privacy and security of personal data.

References

- * MuleSoft Security Best Practices
- * Data Protection and Encryption Standards Documentation

Valid MuleSoft-Integration-Architect-I Dumps shared by PrepPdf.com for Helping Passing MuleSoft-Integration-Architect-I Exam! PrepPdf.com now offer the **newest MuleSoft-Integration-Architect-I exam dumps**, the PrepPdf.com MuleSoft-Integration-Architect-I exam **questions have been updated** and **answers have been corrected** get the **newest** PrepPdf.com MuleSoft-Integration-Architect-I dumps with Test Engine here: <https://www.preppdf.com/Salesforce/MuleSoft-Integration-Architect-I-prepaway-exam-dumps.html> (275 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 32

An insurance company is using a CloudHub runtime plane. As a part of requirement, email alert should be sent to internal operations team every time of policy applied to an API instance is deleted As an integration architect suggest on how this requirement be met?

- A. Use audit logs in Anypoint platform to detect a policy deletion and configure the Audit logs alert feature to send an email to the operations team
- B. Use Anypoint monitoring to configure an alert that sends an email to the operations team every time a policy is deleted in API manager
- C. Create a custom connector to be triggered every time of policy is deleted in API manager
- D. Implement a new application that uses the Audit log REST API to detect the policy deletion and send an email to operations team the SMTP connector

Answer: D (LEAVE A REPLY)

* Requirement Analysis: The organization needs to send email alerts to the internal operations team whenever a policy applied to an API instance is deleted in the CloudHub runtime plane.

* Solution: The most effective approach is to implement a new Mule application that uses the Audit Log REST API to detect when a policy is deleted and then sends an email using the SMTP connector.

* Implementation Steps:

* Access Audit Logs:

* Use the Anypoint Platform's Audit Log REST API to monitor and detect policy deletion events.

* Example of accessing the Audit Log REST API:

GET /accounts/api/v2/organizations/{orgId}/audit-logs

* This endpoint will provide logs, including events related to policy deletions.

* Create Email Notification Application:

* Create a Mule application that periodically polls the Audit Log REST API to check for policy deletion events.

* Use the SMTP connector to send email notifications to the operations team.

* Example Mule flow:

```
<flow name="policy-deletion-alert-flow"> <scheduler frequency="60000" /> <http:request config-ref=" AuditLogAPIConfig" path="/audit-logs" method="GET"> <http:request-builder>
<http:uri-params> <http:uri- param key="orgId" value="your-org-id" /> </http:uri-params> <http:query-params> <http:query-param key=" eventType" value="PolicyDeleted" /> </http:query-
params> </http:request-builder> </http:request> <choice>
<when expression="#[payload contains 'PolicyDeleted']"> <smtp:send config-ref="SMTP_Config" to=" operations@company.com" subject="Policy Deletion Alert"> <smtp:body>Policy
deletion detected: #
[payload]</smtp:body> </smtp:send> </when> </choice> </flow>
```

* Scheduler: Configures the application to check the audit logs at regular intervals (e.g., every minute).

* HTTP Request: Makes a GET request to the Audit Log REST API to fetch the logs.

* Choice Router: Checks if the payload contains a policy deletion event.

* SMTP Connector: Sends an email to the operations team with details about the policy deletion.

* Advantages:

* Automation: Automates the detection of policy deletions and the notification process, reducing manual monitoring efforts.

* Timeliness: Ensures the operations team is promptly informed of any policy deletions, enabling them to take immediate action if necessary.

References

* MuleSoft Documentation on Audit Log REST API

* MuleSoft Documentation on SMTP Connector

* MuleSoft Documentation on MuleSoft Scheduler Component

NEW QUESTION: 33

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 invocation of the fallback API provides the best resilience?

A. Redirect client requests through an HTTP 303 temporary redirect status code to the fallback API whenever the Order API is unavailable

B. 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 received from Order API

C. 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

D. Search Anypoint Exchange for a suitable existing fallback API and then implement invocations to their fallback API in addition to the Order API

Answer: (SHOW ANSWER)

* Resilience testing is a type of software testing that observes how applications act under stress. It's meant to ensure the product's ability to perform in chaotic conditions without a loss of core functions or data; it ensures a quick recovery after unforeseen, uncontrollable events.

* In case an API invocation fails - even after a certain number of retries - it might be adequate to invoke a different API as a fallback. A fallback API, by definition, will never be ideal for the purpose of the API client, otherwise it would be the primary API.

* Here are some examples for fallback APIs:

- An old, deprecated version of the same API.

- An alternative endpoint of the same API and version (e.g. API in another CloudHub region).

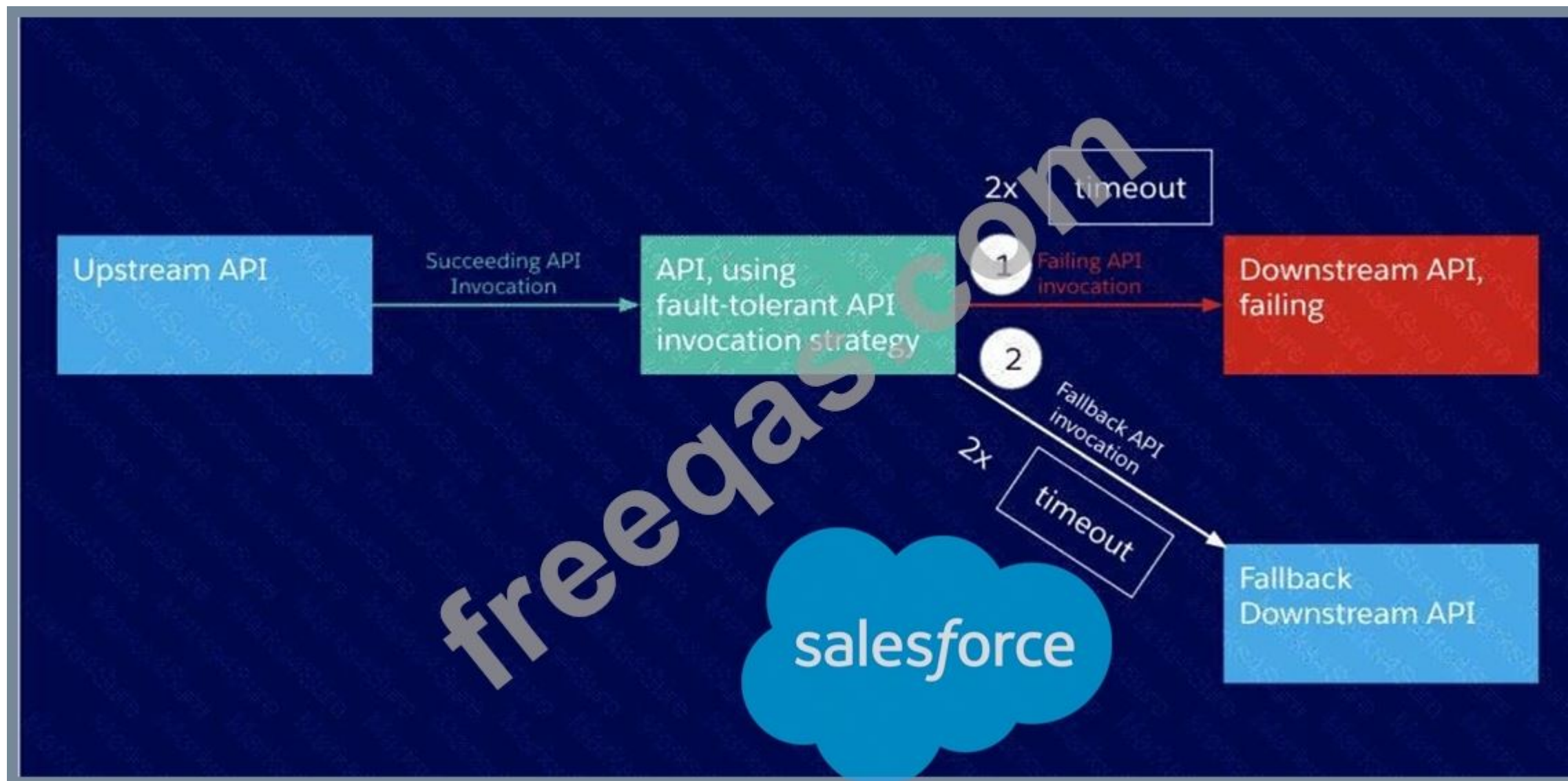
- An API doing more than required, and therefore not as performant as the primary API.

- An API doing less than required and therefore forcing the API Client to offer a degraded service, which is still better than no service at all.

* API clients implemented as Mule applications offer the 'Until Successful Scope and Exception' strategies at their disposal, which together allow configuring fallback actions such as a fallback API invocation.

* All HTTP response status codes within the 3xx category are considered redirection messages. These codes indicate to the user agent (i.e. your web browser) that an additional action is required in order to complete the request and access the desired resource

Diagram Description automatically generated



Hence correct answer is Redirect client requests through an HTTP 303 temporary redirect status code to the fallback API whenever the Order API is unavailable

NEW QUESTION: 34

An organization is in the process of building automated deployments using a CI/CD process. As a part of automated deployments, it wants to apply policies to API Instances. What tool can the organization use to promote and deploy API Manager policies?

- A. MUnit Maven plugin
- B. Runtime Manager agent
- C. Mule Maven plugin
- D. Anypoint CLI

Answer: D ([LEAVE A REPLY](#))

NEW QUESTION: 35

An organization is designing multiple new applications to run on CloudHub in a single Anypoint VPC and that must share data using a common persistent Anypoint object store V2 (OSv2). Which design gives these mule applications access to the same object store instance?

- A. AVM connector configured to directly access the persistence queue of the persistent object store
- B. An Anypoint MQ connector configured to directly access the persistent object store
- C. Object store V2 can be shared across cloudhub applications with the configured osv2 connector

D. The object store V2 rest API configured to access the persistent object store

Answer: C ([LEAVE A REPLY](#))

Object Store v2 (OSv2) in MuleSoft allows multiple applications within the same Anypoint Virtual Private Cloud (VPC) to share data. To achieve this, the OSv2 connector must be configured in each application to access the common persistent object store instance. This configuration enables the applications to read from and write to the same object store, facilitating data sharing. Object Store v2 is designed to be persistent and reliable, making it suitable for scenarios where data needs to be retained and shared across multiple applications.

MuleSoft Documentation on Object Store v2

NEW QUESTION: 36

An auto mobile company want to share inventory updates with dealers D1 and D2 asynchronously and concurrently via queues Q1 and Q2. Dealer D1 must consume the message from the queue Q1 and dealer D2 to must consume a message from the queue Q2.

Dealer D1 has implemented a retry mechanism to reprocess the transaction in case of any errors while processing the inventers updates. Dealer D2 has not implemented any retry mechanism.

How should the dealers acknowledge the message to avoid message loss and minimize impact on the current implementation?

- A.** Dealer D1 must use auto acknowledgement and dealer D2 can use manual acknowledgement and acknowledge the message after successful processing
- B.** Dealer D1 can use auto acknowledgement and dealer D2 can use IMMEDIATE acknowledgement and acknowledge the message of successful processing
- C.** Dealer D1 and dealer D2 must use AUTO acknowledgement and acknowledge the message after successful processing
- D.** Dealer D1 can use AUTO acknowledgement and dealer D2 must use manual acknowledgement and acknowledge the message after successful processing

Answer: ([SHOW ANSWER](#)**)**

* Dealer D1 - AUTO Acknowledgment:

* Dealer D1, which has a retry mechanism, can use AUTO acknowledgment. This means that the message is acknowledged automatically once received. The retry mechanism will handle any processing errors and reprocess the message if needed.

* Dealer D2 - Manual Acknowledgment:

* Dealer D2, which does not have a retry mechanism, must use manual acknowledgment. This approach allows D2 to acknowledge the message only after it has been successfully processed, ensuring no message is lost due to processing errors.

* Ensuring Message Delivery:

* Using manual acknowledgment for Dealer D2 ensures that the message is not lost and can be reprocessed if an error occurs during processing, which is crucial since it lacks a retry mechanism.

References:

* MuleSoft Documentation on JMS Acknowledgment

* Best practices for Reliable Messaging

NEW QUESTION: 37

According to MuleSoft, what is a major distinguishing characteristic of an application network in relation to the integration of systems, data, and devices?

- A.** It uses a well-organized monolithic approach with standards
- B.** It is built for change and self-service
- C.** It leverages well-accepted internet standards like HTTP and JSON
- D.** It uses CI/CD automation for real-time project delivery

Answer: B ([LEAVE A REPLY](#))

A major distinguishing characteristic of an application network, according to MuleSoft, is that it is built for change and self-service. An application network connects applications, data, and devices with APIs, enabling self-service access and reuse of assets. This architecture allows organizations to rapidly adapt to changing business needs, fosters innovation, and reduces time to market by empowering different teams to access and integrate systems independently without waiting for centralized IT.

References:

Application Networks: The Future of Integration

MuleSoft's Approach to Building Application Networks

NEW QUESTION: 38

According to the Internet Engineering Task Force (IETF), which supporting protocol does File Transfer Protocol (FTP) use for reliable communication?

- A. Lightweight Directory Access Protocol (LDAP)
- B. B Transmission Control Protocol (TCP)
- C. A Secure Sockets Layer (SSL)
- D. Hypertext Transfer Protocol (HTTP)

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 39

An organization has chosen Mulesoft for their integration and API platform.

According to the Mulesoft catalyst framework, what would an integration architect do to create achievement goals as part of their business outcomes?

- A. Measure the impact of the centre for enablement
- B. build and publish foundational assets
- C. agree upon KPI's and help develop and overall success plan
- D. evangelize API's

Answer: ([SHOW ANSWER](#))

According to the MuleSoft Catalyst framework, an Integration Architect plays a crucial role in defining and achieving business outcomes. One of their key responsibilities is to agree upon Key Performance Indicators (KPIs) and help develop an overall success plan. This involves working with stakeholders to identify measurable goals and ensure that the integration initiatives align with the organization's strategic objectives.

KPIs are critical for tracking progress, measuring success, and making data-driven decisions. By agreeing on KPIs and developing a success plan, the Integration Architect ensures that the organization can objectively measure the impact of its integration efforts and adjust strategies as needed to achieve desired business outcomes.

MuleSoft Catalyst Knowledge Hub

NEW QUESTION: 40

Which Exchange asset type represents a complete API specification in RAML or OAS format?

- A. REST APIs
- B. API Spec Fragments
- C. SOAP APIs
- D. Connectors

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 41

A company is implementing a new Mule application that supports a set of critical functions driven by a rest API enabled, claims payment rules engine hosted on oracle ERP. As designed the mule application requires many data transformation operations as it performs its batch processing logic.

The company wants to leverage and reuse as many of its existing java-based capabilities (classes, objects, data model etc.) as possible What approach should be considered when implementing required data mappings and transformations between Mule application and Oracle ERP in the new Mule application?

- A. Create a new metadata RAML classes in Mule from the appropriate Java objects and then perform transformations via Dataweave

- B.** From the mule application, transform via theXSLT model
- C.** Transform by calling any suitable Java class from Dataweave
- D.** Invoke any of the appropriate Java methods directly, create metadata RAML classes and then perform required transformations via Dataweave

Answer: C (LEAVE A REPLY)

Leveraging existing Java-based capabilities for data transformations in a Mule application can enhance efficiency and reuse. Here's how to integrate Java classes for transformations:

- * Create Java Classes:

- * Ensure the Java classes that contain the transformation logic are available in the Mule application project.

- * Compile the Java classes if they are not already compiled and place the .class files or JAR file in the Mule project's classpath.

- * Configure DataWeave to Call Java Methods:

- * Use DataWeave's capability to invoke Java methods within the transformation scripts.

- * Import the Java classes and methods in the DataWeave script.

```
%dw 2.0 import * from my.package.ClassName output application/json --- { transformedData: ClassName::
methodName(payload) }
```

- * Perform Transformations:

- * Write DataWeave scripts that call the appropriate Java methods to perform the necessary transformations.

- * Ensure the input and output types match between DataWeave and the Java methods.

- * Test Transformations:

- * Thoroughly test the transformations to ensure the Java methods are correctly invoked and the expected transformations are applied.

This approach allows for seamless integration of existing Java logic into Mule applications, leveraging DataWeave's power for comprehensive data transformations.

References

- * MuleSoft Documentation: DataWeave and Java Integration

- * MuleSoft Documentation: Using Java with Mule

NEW QUESTION: 42

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 1: 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?

- A.** Both Correlation ID values should be stored in a persistent object store

- B.** Both Correlation ID values should be stored In a non-persistent object store

- C.** The Correlation ID value in Step 1 should be stored in a persistent object storeThe Correlation ID value in step 3 should be stored as a Mule event variable/attribute

- D.** Both Correlation ID values should be stored as Mule event variable/attribute

Answer: C (LEAVE A REPLY)

- * If we store Correlation id value in step 1 as Mule event variables/attributes, the values will be cleared after server restart and we want system to be fault tolerant.

- * The Correlation ID value in Step 1 should be stored in a persistent object store.
- * We don't need to store Correlation ID value in Step 3 to persistent object store. We can store it but as we also need to make application performant. We can avoid this step of accessing persistent object store.
- * Accessing persistent object stores slow down the performance as persistent object stores are by default stored in shared file systems.
- * As the SOAP service is idempotent in nature. In case of any failures , using this Correlation ID saved in first step we can make call to SOAP service and validate the Correlation ID.

Top of Form

Additional Information:

- * Competing Consumers are multiple consumers that are all created to receive messages from a single Point- to-Point Channel. When the channel delivers a message, any of the consumers could potentially receive it.

The messaging system's implementation determines which consumer actually receives the message, but in effect the consumers compete with each other to be the receiver. Once a consumer receives a message, it can delegate to the rest of its application to help process the message.

- * In case you are unaware about term idempotent re is more info:

Idempotent operations means their result will always same no matter how many times these operations are invoked.

Bottom of Form

NEW QUESTION: 43

An organization has decided on a cloud migration strategy to minimize the organization's own IT resources.

Currently the organization has all of its new applications running on its own premises and uses an on- premises load balancer that exposes all APIs under the base URL (<https://api.rutujar.com>).

As part of migration strategy, the organization is planning to migrate all of its new applications and load balancer CloudHub.

What is the most straightforward and cost-effective approach to Mule application deployment and load balancing that preserves the public URL's?

A. Deploy the Mule application to Cloudhub

Create a CNAME record for base URL(<https://api.rutujar.com>) in the Cloudhub shared load balancer that points to the A record of the on-premises load balancer Apply mapping rules in SLB to map URL to their corresponding Mule applications

B. Deploy the Mule application to Cloudhub

Update a CNAME record for base URL (<https://api.rutujar.com>) in the organization's DNS server to point to the A record of the Cloudhub dedicated load balancer Apply mapping rules in DLB to map URL to their corresponding Mule applications

C. Deploy the Mule application to Cloudhub

Update a CNAME record for base URL (<https://api.rutujar.com>) in the organization's DNS server to point to the A record of the CloudHub shared load balancer Apply mapping rules in SLB to map URL to their corresponding Mule applications

D. For each migrated Mule application, deploy an API proxy application to Cloudhub with all traffic to the mule applications routed through a Cloud Hub Dedicated load balancer (DLB)

Update a CNAME record for base URL (<https://api.rutujar.com>) in the organization's DNS server to point to the A record of the CloudHub dedicated load balancer Apply mapping rules in DLB to map each API proxy application who is responding new application

Answer: C (LEAVE A REPLY)

For an organization migrating its Mule applications and load balancer to CloudHub while preserving its public URL, the most straightforward and cost-effective approach involves leveraging CloudHub's shared load balancer (SLB). Here's how to do it:

- * Deploying Mule Applications to CloudHub:

* First, deploy your Mule applications to CloudHub. This involves packaging your Mule applications and using the Anypoint Platform to deploy them to CloudHub.

- * DNS Configuration:

* Update your DNS settings to ensure the base URL (<https://api.rutujar.com>) points to the CloudHub shared load balancer. Specifically, create or update a CNAME record for api.rutujar.com to point to the CloudHub SLB's A record.

* Configuring Shared Load Balancer (SLB):

* In CloudHub, configure the SLB to handle incoming traffic and route it to the appropriate Mule applications.

* Apply mapping rules in the SLB to map the base URL and its paths to the corresponding Mule applications. This ensures that requests to <https://api.rutujar.com> are correctly routed to the specific Mule application endpoints.

* Steps to Apply Mapping Rules:

* In Anypoint Platform, navigate to the Runtime Manager and select the application deployed on CloudHub.

* Configure the SLB settings by defining the domain and URL mappings. For instance, set rules that map requests from <https://api.rutujar.com/app1> to the corresponding Mule application app1.

* Benefits of Using SLB:

* Cost-Effective: The shared load balancer is a more economical option compared to a dedicated load balancer (DLB).

* Simplicity: Using SLB simplifies the deployment process, as it requires less configuration and management compared to setting up and maintaining a dedicated load balancer.

* Consistency: Preserves the public-facing URL structure, ensuring a seamless transition for users and clients accessing the APIs.

References:

* MuleSoft Documentation on CloudHub Deployment

* MuleSoft Documentation on Shared Load Balancer

By following these steps, the organization can effectively migrate its applications to CloudHub, ensuring minimal disruption and preserving the existing public URL structure.

NEW QUESTION: 44

A Mule application currently writes to two separate SQL Server database instances across the internet using a single XA transaction. It is proposed to split this one transaction into two separate non-XA transactions with no other changes to the Mule application.

What non-functional requirement can be expected to be negatively affected when implementing this change?

A. Throughput

B. Consistency

C. Response time

D. Availability

Answer: B (LEAVE A REPLY)

Correct answer is Consistency as XA transactions are implemented to achieve this. XA transactions are added in the implementation to achieve goal of ACID properties. In the context of transaction processing, the acronym ACID refers to the four key properties of a transaction: atomicity, consistency, isolation, and durability. Atomicity : All changes to data are performed as if they are a single operation. That is, all the changes are performed, or none of them are. For example, in an application that transfers funds from one account to another, the atomicity property ensures that, if a debit is made successfully from one account, the corresponding credit is made to the other account. Consistency : Data is in a consistent state when a transaction starts and when it ends. For example, in an application that transfers funds from one account to another, the consistency property ensures that the total value of funds in both the accounts is the same at the start and end of each transaction. Isolation : The intermediate state of a transaction is invisible to other transactions. As a result, transactions that run concurrently appear to be serialized. For example, in an application that transfers funds from one account to another, the isolation property ensures that another transaction sees the transferred funds in one account or the other, but not in both, nor in neither. Durability : After a transaction successfully completes, changes to data persist and are not undone, even in the event of a system failure. For example, in an application that transfers funds from one account to another, the durability property ensures that the changes made to each account will not be reversed. MuleSoft reference: <https://docs.mulesoft.com/mule-runtime/4.3/xa-transactions>

NEW QUESTION: 45

According to MuleSoft, what Action should an IT organization take regarding its technology assets in order to close the IT delivery.

A. Create weekly meetings that all members of IT attend to present justification and request approval to use existing assets

B. Hire additional staff to meet the demand for asset creation required for approved projects and timelines

- C. Focus project delivery efforts on custom assets that meet the specific requirements of each individual line of business
- D. Make assets easily discoverable via a central repository

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 46

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?

- A. 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
- B. 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
- C. The time that is allowed to pass without the transaction being ended explicitly After the timeout, the transaction is forcefully rolled-back
- D. 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

Answer: ([SHOW ANSWER](#))

* Setting a transaction timeout for the Bitronix transaction manager

Set the transaction timeout either

- In wrapper.conf

- In CloudHub in the Properties tab of the Mule application deployment

The default is 60 secs. It is defined as

mule.bitronix.transactiontimeout = 120

* This property defines the timeout for each transaction created for this manager.

If the transaction has not terminated before the timeout expires it will be automatically rolled back.

----- Additional Info around Transaction Management:

Bitronix is available as the XA transaction manager for Mule applications

To use Bitronix, declare it as a global configuration element in the Mule application

<bti:transaction-manager />

Each Mule runtime can have only one instance of a Bitronix transaction manager, which is shared by all Mule applications

For customer-hosted deployments, define the XA transaction manager in a Mule domain

- Then share this global element among all Mule applications in the Mule runtime Graphical user interface, table Description automatically generated with medium confidence

Transaction Management		
Characteristics	Local Transactions	Extended Architecture (XA) Transactions
Connector Requisite 1	All operations inside the transaction must belong to same Connector.	Operations inside the transaction may belong to different Connectors
Connector Requisite 2	Connectors may not be XA enabled	Connectors must be XA enabled
Connector Requisite 3	Connectors should use single config reference	Connectors may use multiple config references
Available resources	JMS, VM, JDBC	JMS, VM, JDBC
Uses Two Phase Commit (2PC)	No	Yes
DB Operations	Perform database operation to only one database resource	Perform database operation to one or more transactional resource
Supports Nested Transactions	Does not support nested transactions.	Supports nested transactions.
Bitronix is available	No	Yes
A.C.I.D Properties	No	Yes
Performance	Better than XA	Latency Increases
Thread Pooling	BLOCKING_IO	BLOCKING_IO
Recovery in case of system failure	No	Using Bitronix

NEW QUESTION: 47

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

Answer: D (LEAVE A REPLY)

We need to note a few things about the scenario which will help us in reaching the correct solution.

Point 1 : The APIs are all publicly available and are associated with several mobile applications and web applications. This means Apply an IP blacklist policy is not a viable option, as blacklisting IPs is limited to partial web traffic. It can't be useful for traffic from mobile application Point 2 : The organization does NOT want to use any authentication or compliance policies for these APIs.

This means we can not apply HTTPS mutual authentication scheme.

Header injection or removal will not help the purpose.

By its nature, JSON is vulnerable to JavaScript injection. When you parse the JSON object, the malicious code inflicts its damages. An inordinate increase in the size and depth of the JSON payload can indicate injection. Applying the JSON threat protection policy can limit the size of your JSON payload and thwart recursive additions to the JSON hierarchy.

Hence correct answer is Apply a JSON threat protection policy to all APIs to detect potential threat vectors

NEW QUESTION: 48

What API policy would LEAST likely be applied to a Process API?

- A. Custom circuit breaker
- B. Client ID enforcement
- C. Rate limiting
- D. JSON threat protection

Answer: D (LEAVE A REPLY)

Key to this question lies in the fact that Process API are not meant to be accessed directly by clients. Let's 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'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'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's. This policy is used to safeguard application from malicious attacks by injecting malicious code in JSON object. As ideally Process API's are never called from external world, this policy is never used on Process API's Hence correct answer is JSON threat protection MuleSoft

Documentation Reference : <https://docs.mulesoft.com/api-manager/2.x/policy-mule3-json-threat>

NEW QUESTION: 49

Mule application A receives a request Anypoint MQ message REQU with a payload containing a variable-length list of request objects. Application A uses the For Each scope to split the list into individual objects and sends each object as a message to an Anypoint MQ queue.

Service S listens on that queue, processes each message independently of all other messages, and sends a response message to a response queue.

Application A listens on that response queue and must in turn create and publish a response Anypoint MQ message RESP with a payload containing the list of responses sent by service S in the same order as the request objects originally sent in REQU.

Assume successful response messages are returned by service S for all request messages.

What is required so that application A can ensure that the length and order of the list of objects in RESP and REQU match, while at the same time maximizing message throughput?

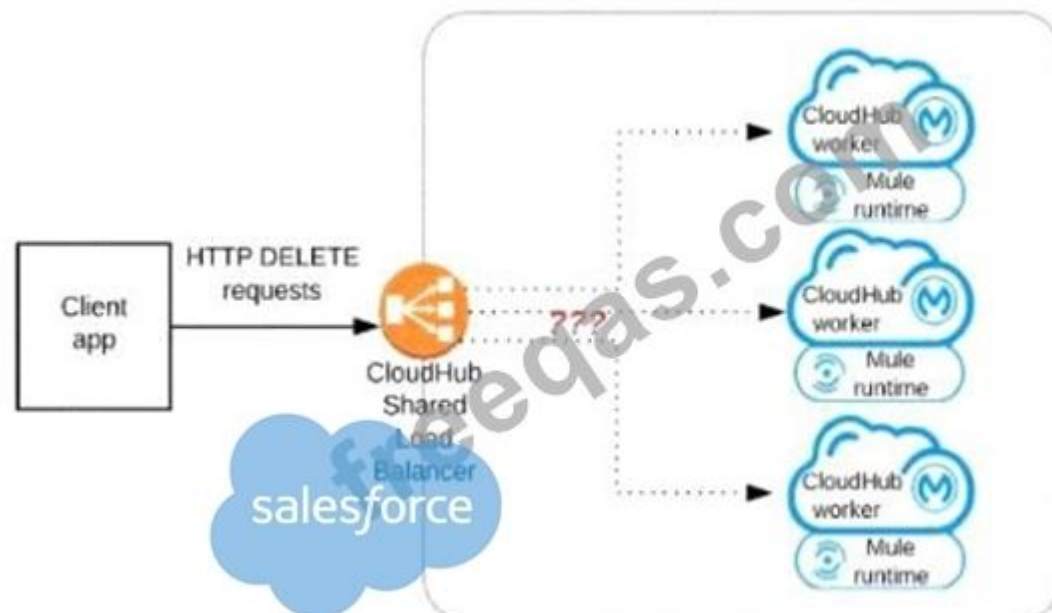
- A. Use a Scatter-Gather within the For Each scope to ensure response message order Configure the Scatter-Gather with a persistent object store
- B. Perform all communication involving service S synchronously from within the For Each scope, so objects in RESP are in the exact same order as request objects in REQU
- C. Use an Async scope within the For Each scope and collect response messages in a second For Each scope in the order In which they arrive, then send RESP using this list of responses
- D. Keep track of the list length and all object indices in REQU, both in the For Each scope and in all communication involving service Use persistent storage when creating RESP

Answer: (SHOW ANSWER)

Correct answer is Perform all communication involving service S synchronously from within the For Each scope, so objects in RESP are in the exact same order as request objects in REQU : Using Anypoint MQ, you can create two types of queues: Standard queue These queues don't guarantee a specific message order. Standard queues are the best fit for applications in which messages must be delivered quickly. FIFO (first in, first out) queue These queues ensure that your messages arrive in order. FIFO queues are the best fit for applications requiring strict message ordering and exactly-once delivery, but in which message delivery speed is of less importance Use of FIFO queue is no where in the option and also it decreased throughput. Similarly persistent object store is not the preferred solution approach when you maximizing message throughput. This rules out one of the options. Scatter Gather does not support ObjectStore. This rules out one of the options. Standard Anypoint MQ queues don't guarantee a specific message order hence using another for each block to collect response wont work as requirement here is to ensure the order. Hence considering all the above factors the feasible approach is Perform all communication involving service S synchronously from within the For Each scope, so objects in RESP are in the exact same order as request objects in REQU

NEW QUESTION: 50

Refer to the exhibit.



A Mule application has an HTTP Listener that accepts HTTP DELETE requests. This Mule application is deployed to three CloudHub workers under the control of the CloudHub Shared Load Balancer.

A web client makes a sequence of requests to the Mule application's public URL.

How is this sequence of web client requests distributed among the HTTP Listeners running in the three CloudHub workers?

- A. Each request is routed to the PRIMARY CloudHub worker in the PRIMARY Availability Zone (AZ)
- B. Each request is routed to ONE ARBITRARY CloudHub worker in the PRIMARY Availability Zone (AZ)
- C. Each request is routed to ONE ARBITRARY CloudHub worker out of ALL three CloudHub workers

D. Each request is routed (scattered) to ALL three CloudHub workers at the same time

Answer: ([SHOW ANSWER](#))

Correct behavior is Each request is routed to ONE ARBITRARY CloudHub worker out of ALL three CloudHub workers

NEW QUESTION: 51

A mule application is being designed to perform product orchestration. The Mule application needs to join together the responses from an inventory API and a Product Sales History API with the least latency.

To minimize the overall latency. What is the most idiomatic (used for its intended purpose) design to call each API request in the Mule application?

A. Call each API request in a separate lookup call from Dataweave reduce operator

B. Call each API request in a separate route of a Scatter-Gather

C. Call each API request in a separate route of a Parallel For Each scope

D. Call each API request in a separate Async scope

Answer: ([SHOW ANSWER](#))

Scatter-Gather sends a request message to multiple targets concurrently. It collects the responses from all routes, and aggregates them into a single message.

Reference: <https://docs.mulesoft.com/mule-runtime/4.3/scatter-gather-concept>

NEW QUESTION: 52

An Order microservice and a Fulfillment microservice are being designed to communicate with their clients through message-based integration (and NOT through API invocations).

The Order microservice publishes an Order message (a kind of command message) containing the details of an order to be fulfilled. The intention is that Order messages are only consumed by one Mule application, the Fulfillment microservice.

The Fulfillment microservice consumes Order messages, fulfills the order described therein, and then publishes an OrderFulfilled message (a kind of event message). Each OrderFulfilled message can be consumed by any interested Mule application, and the Order microservice is one such Mule application.

What is the most appropriate choice of message broker(s) and message destination(s) in this scenario?

A. Order messages are sent to an Anypoint MQ exchange OrderFulfilled messages are sent to an Anypoint MQ queue Both microservices interact with Anypoint MQ as the message broker, which must therefore scale to support the load of both microservices

B. Order messages are sent to a JMS queue. OrderFulfilled messages are sent to a JMS topic Both microservices interact with the same JMS provider (message broker) instance, which must therefore scale to support the load of both microservices

C. Order messages are sent directly to the Fulfillment microservices. OrderFulfilled messages are sent directly to the Order microservice The Order microservice interacts with one AMQP-compatible message broker and the Fulfillment microservice interacts with a different AMQP-compatible message broker, so that both message brokers can be chosen and scaled to best support the load of each microservice

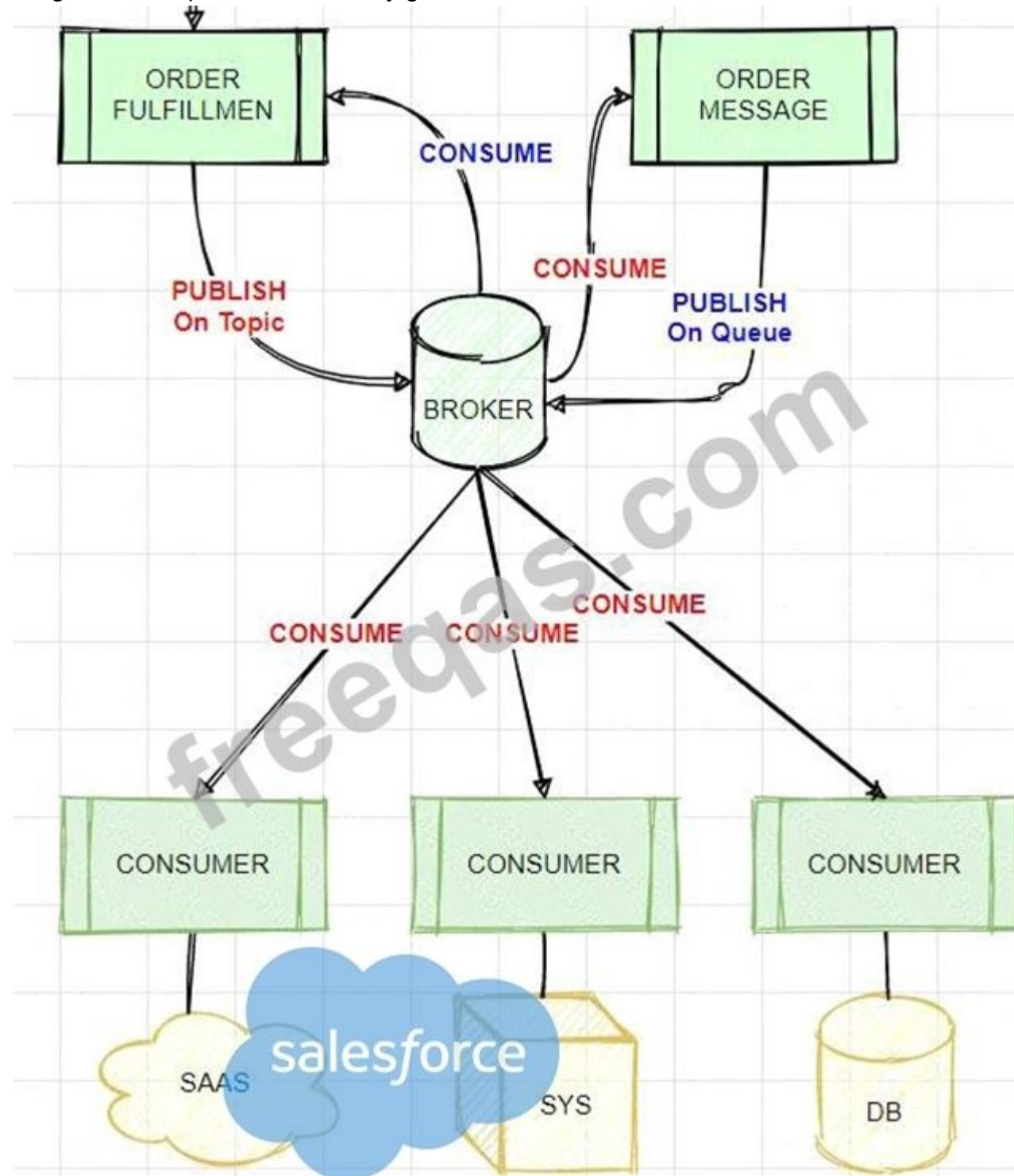
D. Order messages are sent to a JMS queue. OrderFulfilled messages are sent to a JMS topic The Order microservice interacts with one JMS provider (message broker) and the Fulfillment microservice interacts with a different JMS provider, so that both message brokers can be chosen and scaled to best support the load of each microservice

Answer: ([SHOW ANSWER](#))

* If you need to scale a JMS provider/ message broker, - add nodes to scale it horizontally or - add memory to scale it vertically * Cons of adding another JMS provider/ message broker: - adds cost. - adds complexity to use two JMS brokers - adds Operational overhead if we use two brokers, say, ActiveMQ and IBM MQ * So Two options that mention to use two brokers are not best choice. * It's mentioned that "The Fulfillment microservice consumes Order messages, fulfills the order described therein, and then publishes an OrderFulfilled message. Each OrderFulfilled message can be consumed by any interested Mule application." - When you publish a message on a topic, it goes to all the subscribers who are interested - so zero to many subscribers will receive a copy of the message. - When you send a message on a queue, it will be received by exactly one consumer. * As we need multiple consumers to consume the message below option is not valid choice: "Order messages are sent to an Anypoint MQ exchange. OrderFulfilled messages are sent to an Anypoint MQ queue. Both microservices interact with Anypoint MQ as the message broker, which must therefore scale to support the load of both microservices" * Order messages are only consumed by one Mule application, the Fulfillment microservice, so we will publish it on queue and OrderFulfilled message can be consumed by any interested Mule application so it need to be published on Topic using same

broker. * answer: Best choice in this scenario is: "Order messages are sent to a JMS queue. OrderFulfilled messages are sent to a JMS topic. Both microservices interact with the same JMS provider (message broker) instance, which must therefore scale to support the load of both microservices" Tried to depict scenario in diagram:

Diagram Description automatically generated



NEW QUESTION: 53

According to MuleSoft, a synchronous invocation of a RESTful API using HTTP to get an individual customer record from a single system is an example of which system integration interaction pattern?

- A. Request-Reply
- B. Multicast
- C. Batch

D. One-way

Answer: A (LEAVE A REPLY)

A synchronous invocation of a RESTful API using HTTP to get an individual customer record from a single system exemplifies the request-reply interaction pattern. In this pattern, a client sends a request to a service and waits for a response. The interaction is synchronous because the client expects an immediate reply and is blocked until it receives the response.

This pattern is common in scenarios where the client needs to retrieve specific data or perform an action that requires immediate feedback from the server, such as retrieving a customer record from a database.

References

* MuleSoft Integration Patterns Documentation

* Request-Reply Pattern Overview

NEW QUESTION: 54

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?

- A.** 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.
- B.** 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.
- C.** The time that is allowed to pass without the transaction being ended explicitly. After the timeout, the transaction is forcefully rolled-back.
- D.** 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.

Answer: C (LEAVE A REPLY)

* Setting a transaction timeout for the Bitronix transaction manager

Set the transaction timeout either

- In wrapper.conf

- In CloudHub in the Properties tab of the Mule application deployment

The default is 60 secs. It is defined as

mule.bitronix.transactiontimeout = 120

* This property defines the timeout for each transaction created for this manager.

If the transaction has not terminated before the timeout expires, it will be automatically rolled back.

----- Additional Info around Transaction Management:

Bitronix is available as the XA transaction manager for Mule applications

To use Bitronix, declare it as a global configuration element in the Mule application

<bti:transaction-manager />

Each Mule runtime can have only one instance of a Bitronix transaction manager, which is shared by all Mule applications

For customer-hosted deployments, define the XA transaction manager in a Mule domain

- Then share this global element among all Mule applications in the Mule runtime. Graphical user interface, table Description automatically generated with medium confidence

Transaction Management		
Characteristics	Local Transactions	Extended Architecture (XA) Transactions
Connector Requisite 1	All operations inside the transaction must belong to same Connector.	Operations inside the transaction may belong to different Connectors
Connector Requisite 2	Connectors may not be XA enabled	Connectors must be XA enabled
Connector Requisite 3	Connectors should use single config reference	Connectors may use multiple config references
Available resources	JMS, VM, JDBC	JMS, VM, JDBC
Uses Two Phase Commit (2PC)	No	Yes
DB Operations	Perform database operation to only one database resource	Perform database operation to one or more transactional resource
Supports Nested Transactions	Does not support nested transactions.	Supports nested transactions.
Bitronix is available	No	Yes
A.C.I.D Properties	No	Yes
Performance	Better than XA	Latency Increases
Thread Pooling	BLOCKING_IO	BLOCKING_IO
Recovery is case of system failure	No	Using Bitronix

NEW QUESTION: 55

A retail company is implementing a MuleSoft API to get inventory details from two vendors by Invoking each vendor's online applications. Due to network issues, the invocations to the vendor applications are timing out intermittently, but the requests are successful after re-invoking each What is the most performant way of implementing the API to invoke each vendor application and to retry invocations that generate timeout errors?

A. Use a Scatter-Gather scope to invoke each vendor application on a separate route.

Use an Until-Successful scope in each route to retry requests that raise timeout errors.

B. Use a Choice scope to Invoke each vendor application on a separate route.

Place the Choice scope inside an Until-Successful scope to retry requests that raise timeout errors.

C. Use a For-Each scope to invoke the two vendor applications in series, one after the other.

Place the For-Each scope inside an Until-Successful scope to retry requests that raise timeout errors.

D. Use a Round-Robin scope to invoke each vendor application on a separate route.

Use a Try-Catch scope in each route to retry requests that raise timeout errors.

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 56

An architect is designing a Mule application to meet the following two requirements:

1. The application must process files asynchronously and reliably from an FTPS server to a back-end database using VM intermediary queues for load-balancing Mule events.
2. The application must process a medium rate of records from a source to a target system using a Batch Job scope.

To make the Mule application more reliable, the Mule application will be deployed to two CloudHub 1.0 workers.

Following MuleSoft-recommended best practices, how should the Mule application deployment typically be configured in Runtime Manger to best support the performance and reliability goals of both the Batch Job scope and the file processing VM queues?

- A.** Check the Persistent VM queues checkbox in the application deployment configuration
- B.** Check the Non-persistent VM queues checkbox in the application deployment configuration
- C.** In the Runtime Manager Properties tab, disable persistent VM queues for Batch Job scopes
- D.** In the Runtime Manager Properties tab, enable persistent VM queues for the FTPS connector

Answer: A ([LEAVE A REPLY](#))

* Requirements:

* Asynchronous and Reliable File Processing: The application must process files from an FTPS server to a back-end database using VM intermediary queues for load balancing.

* Batch Job Processing: The application must process records from a source to a target system using a Batch Job scope.

* Persistent VM Queues:

* Reliability: Persistent VM queues ensure that messages are not lost even if there is a system failure or restart. This is critical for reliable file processing and load balancing.

* Asynchronous Processing: Persistent queues allow for asynchronous processing, where messages can be stored and processed independently of the producer and consumer lifecycles.

* MuleSoft Best Practices:

* Persistent VM Queues for Load Balancing: Using persistent VM queues aligns with MuleSoft best practices for ensuring reliable message processing and load balancing between Mule events.

* High Availability: With CloudHub 1.0 workers, enabling persistent VM queues helps maintain high availability and reliability of the application.

* Batch Job Scope: Ensuring persistence for VM queues also benefits batch processing by maintaining consistency and ensuring all records are processed even in the event of disruptions.

* Configuration in Runtime Manager:

* Deployment Configuration: When deploying the Mule application in Runtime Manager, check the "Persistent VM queues" checkbox to enable this feature.

* Impact: This configuration ensures that the application meets its performance and reliability goals by safeguarding the integrity and continuity of the processing tasks.

References:

* MuleSoft Documentation on VM Queues: VM Queues

* MuleSoft Best Practices: MuleSoft Best Practices

* CloudHub Deployment Guide: CloudHub Deployment

NEW QUESTION: 57

A team has completed the build and test activities for a Mule application that implements a System API for its application network.

Which Anypoint Platform component should the team now use to both deploy and monitor the System AP\ implementation?

- A.** API Manager
- B.** Design Center
- C.** Anypoint Exchange

D. Runtime Manager

Answer: D ([LEAVE A REPLY](#))

After completing the build and test activities for a Mule application that implements a System API, the team should use Runtime Manager to both deploy and monitor the System API implementation. Runtime Manager is a component of Anypoint Platform that provides capabilities for deploying, managing, and monitoring Mule applications and APIs. It allows teams to manage applications and APIs across various environments, ensuring they operate correctly and efficiently.

References:

- * Anypoint Runtime Manager
- * Deploying and Managing Applications with Runtime Manager

NEW QUESTION: 58

A company is planning to migrate its deployment environment from on-premises cluster to a Runtime Fabric (RTF) cluster. It also has a requirement to enable Mule applications deployed to a Mule runtime instance to store and share data across application replicas and restarts.

How can these requirements be met?

- A.** Anypoint object store V2 to share data between replicas in the RTF cluster
- B.** Install the object store pod on one of the cluster nodes
- C.** Configure Persistence Gateway in any of the servers using Mule Object Store
- D.** Configure Persistent Gateway at the RTF

Answer: A ([LEAVE A REPLY](#))

When migrating from an on-premises cluster to a Runtime Fabric (RTF) cluster, and needing to enable Mule applications to store and share data across application replicas and restarts, the use of Anypoint Object Store V2 is the most suitable option. Here's why and how to implement it:

* Understanding Object Store V2:

* Object Store V2 is designed to store and retrieve key-value pairs in a scalable and highly available manner. It is particularly useful for sharing state and data between different instances of applications running on RTF.

* Setting Up Anypoint Object Store V2:

* First, ensure that your MuleSoft Anypoint Platform account has access to Object Store V2.

* Configure your Mule application to use Object Store V2 by defining an object store in your Mule configuration file. This can be done using the objectstore element in your Mule flow.

* Configuration Steps:

* Add the Object Store connector to your Mule project.

* Define the object store configuration in your Mule flow as follows:

xml

```
<objectstore:config name="ObjectStoreV2" doc:name="ObjectStoreV2"/>
```

* Use the object store in your flows to store and retrieve data:

xml

```
<objectstore:store config-ref="ObjectStoreV2" key="#[key]" value="#[value]" /> <objectstore:retrieve config-ref="ObjectStoreV2" key="#[key]" target="variableName"/>
```

* Deploying on RTF:

* Deploy your Mule application to the RTF cluster. The RTF will handle scaling and ensure that the Object Store V2 is available to all instances of your application.

* Benefits:

* Object Store V2 ensures data is shared and persisted across different application replicas and restarts, which meets the requirement of storing and sharing data across application replicas in the RTF cluster.

References:

- * MuleSoft Documentation on Object Store V2

* MuleSoft Documentation on Runtime Fabric

NEW QUESTION: 59

Insurance organization is planning to deploy Mule application in MuleSoft Hosted runtime plane. As a part of requirement , application should be scalable . highly available. It also has regulatory requirement which demands logs to be retained for at least 2 years. As an Integration Architect what step you will recommend in order to achieve this?

- A. It is not possible to store logs for 2 years in CloudHub deployment. External log management system is required.
- B. When deploying an application to CloudHub , logs retention period should be selected as 2 years
- C. When deploying an application to CloudHub, worker size should be sufficient to store 2 years data
- D. Logging strategy should be configured accordingly in log4j file deployed with the application.

Answer: A ([LEAVE A REPLY](#))

Correct answer is It is not possible to store logs for 2 years in CloudHub deployment. External log management system is required. CloudHub has a specific log retention policy, as described in the documentation: the platform stores logs of up to 100 MB per app & per worker or for up to 30 days, whichever limit is hit first. Once this limit has been reached, the oldest log information is deleted in chunks and is irretrievably lost. The recommended approach is to persist your logs to a external logging system of your choice (such as Splunk, for instance) using a log appender. Please note that this solution results in the logs no longer being stored on our platform, so any support cases you lodge will require for you to provide the appropriate logs for review and case resolution

NEW QUESTION: 60

What aspect of logging is only possible for Mule applications deployed to customer-hosted Mule runtimes, but NOT for Mule applications deployed to CloudHub?

- A. To send Mule application log entries to Splunk
- B. To change log4j2 log levels in Anypoint Runtime Manager without having to restart the Mule application
- C. To log certain messages to a custom log category
- D. To directly reference one shared and customized log4j2.xml file from multiple Mule applications

Answer: D ([LEAVE A REPLY](#))

* Correct answer is To directly reference one shared and customized log4j2.xml file from multiple Mule applications. Key word to note in the answer is directly.

* By default, CloudHub replaces a Mule application's log4j2.xml file with a CloudHub log4j2.xml file. This specifies the CloudHub appender to write logs to the CloudHub logging service.

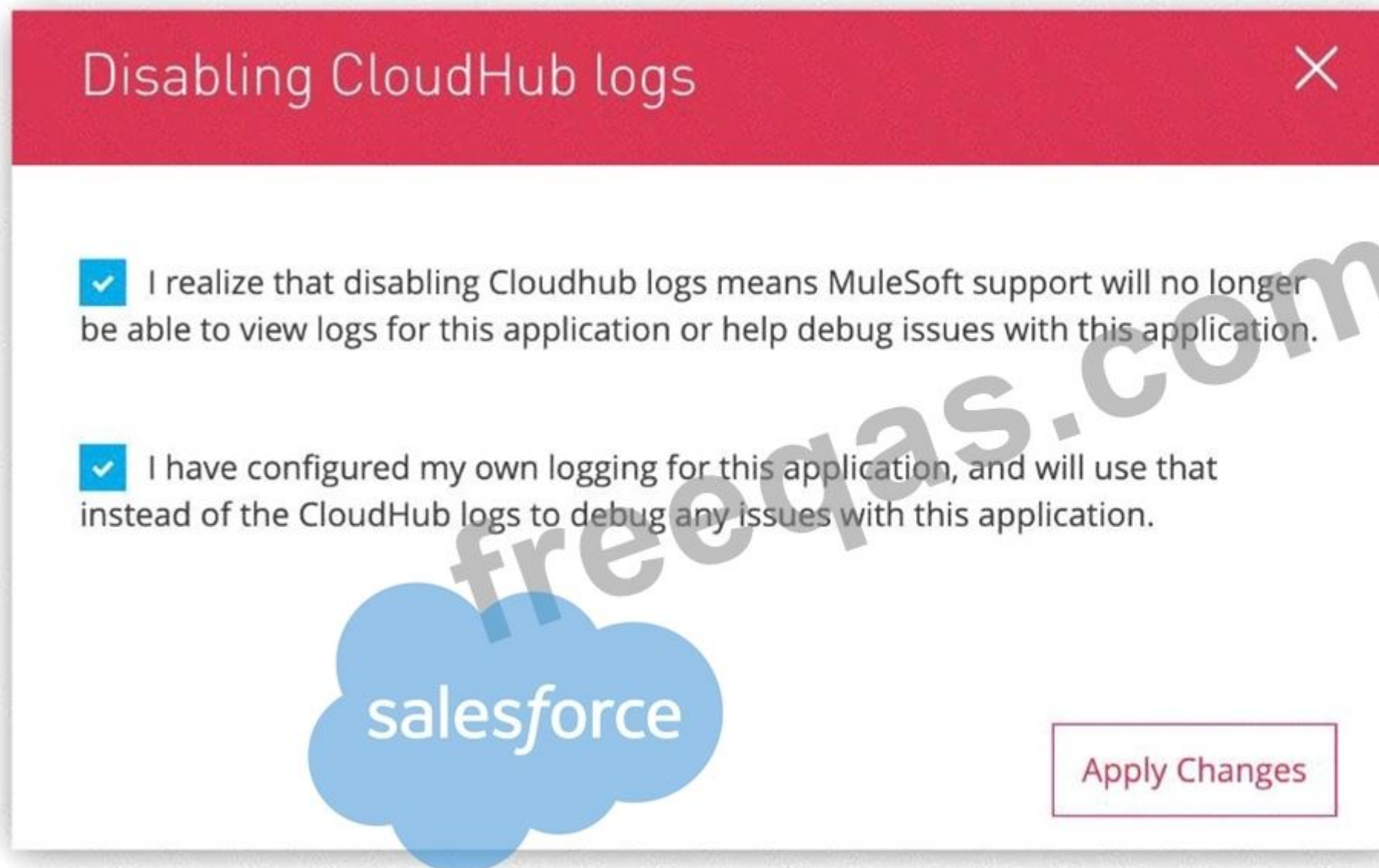
* You cannot modify CloudHub log4j2.xml file to add any custom appender. But there is a process in order to achieve this. You need to raise a request on support portal to disable CloudHub provided Mule application log4j2 file.

Graphical user interface, application, Word Description automatically generated



* Once this is done , Mule application's log4j2.xml file is used which you can use to send/export application logs to other log4j2 appenders, such as a custom logging system MuleSoft does not own any responsibility for lost logging data due to misconfiguration of your own log4j appender if it happens by any chance.

Graphical user interface, text, application, email Description automatically generated



* One more difference between customer-hosted Mule runtimes and CloudHub deployed mule instance is that

- CloudHub system log messages cannot be sent to external log management system without installing custom CH logging configuration through support
- where as Customer-hosted runtime can send system and application log to external log management system MuleSoft Reference:
<https://docs.mulesoft.com/runtime-manager/viewing-log-data>
<https://docs.mulesoft.com/runtime-manager/custom-log-appender>

NEW QUESTION: 61

What is not true about Mule Domain Project?

- A. This allows Mule applications to share resources
- B. Expose multiple services within the Mule domain on the same port
- C. Only available Anypoint Runtime Fabric
- D. Send events (messages) to other Mule applications using VM queues

Answer: C (LEAVE A REPLY)

* 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'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:

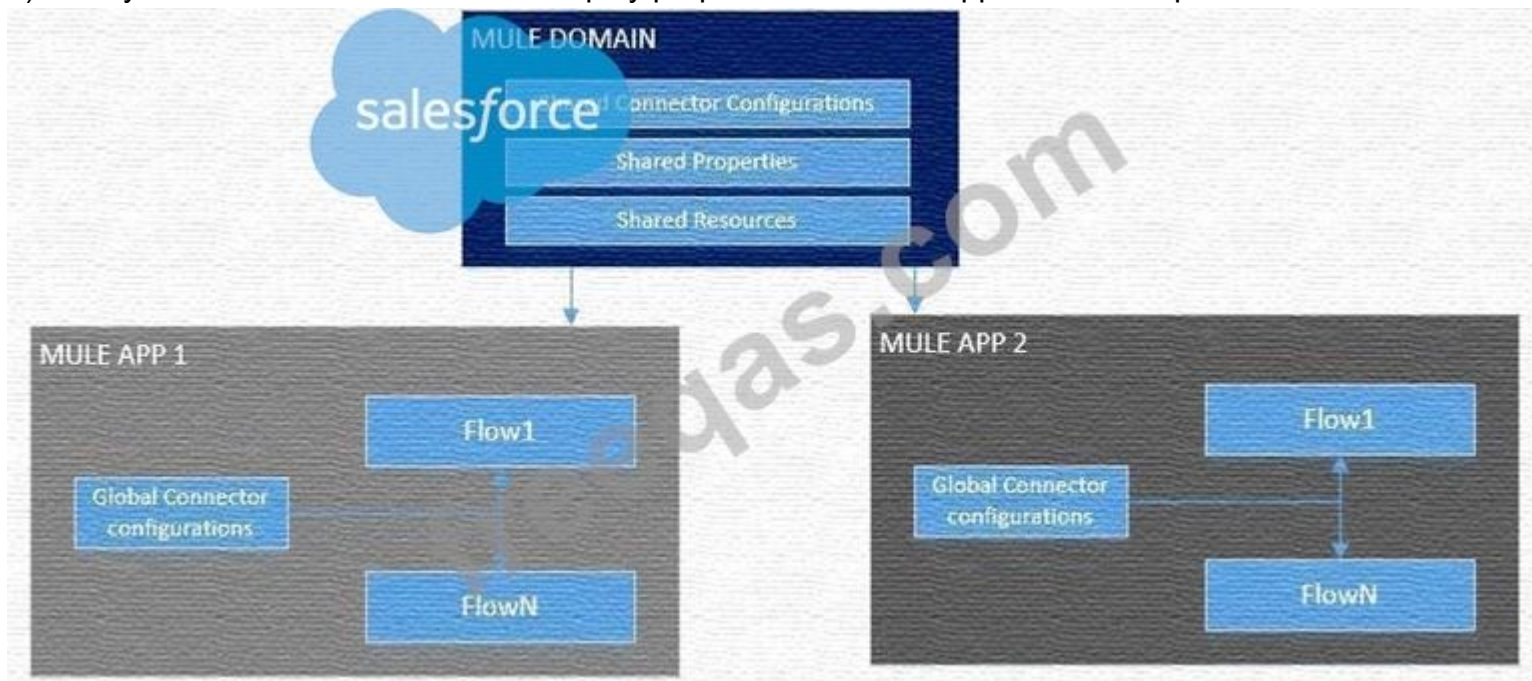
- Expose multiple services within the domain through the same port.
- Share the connection to persistent storage.
- Share services between apps through a well-defined interface.
- Ensure consistency between apps upon any changes because the configuration is only set in one place.

* 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 be 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 Graphical user interface Description automatically generated



Valid MuleSoft-Integration-Architect-I Dumps shared by PrepPdf.com for Helping Passing MuleSoft-Integration-Architect-I Exam! PrepPdf.com now offer the **newest MuleSoft-Integration-Architect-I exam dumps**, the PrepPdf.com MuleSoft-Integration-Architect-I exam **questions have been updated** and **answers have been corrected** get the **newest** PrepPdf.com MuleSoft-Integration-Architect-I dumps with Test Engine here: <https://www.preppdf.com/Salesforce/MuleSoft-Integration-Architect-I-prepaway-exam-dumps.html> (275 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 62

A trading company handles millions of requests a day. Due to nature of its business, it requires excellent performance and reliability within its application.

For this purpose, company uses a number of event-based API's hosted on various mule clusters that communicate across a shared message queue sitting within its network.

Which method should be used to meet the company's requirement for its system?

- A. XA transactions and XA connected components
- B. JMS transactions

C. JMS manual acknowledgements with a reliability pattern

D. VM queues with reliability pattern

Answer: B (LEAVE A REPLY)

For handling millions of requests per day with high performance and reliability, JMS (Java Message Service) transactions are the most suitable method. JMS transactions ensure that messages are processed reliably and can be rolled back in case of failures, providing transactional integrity. This is crucial for a trading company where reliable message processing is critical.

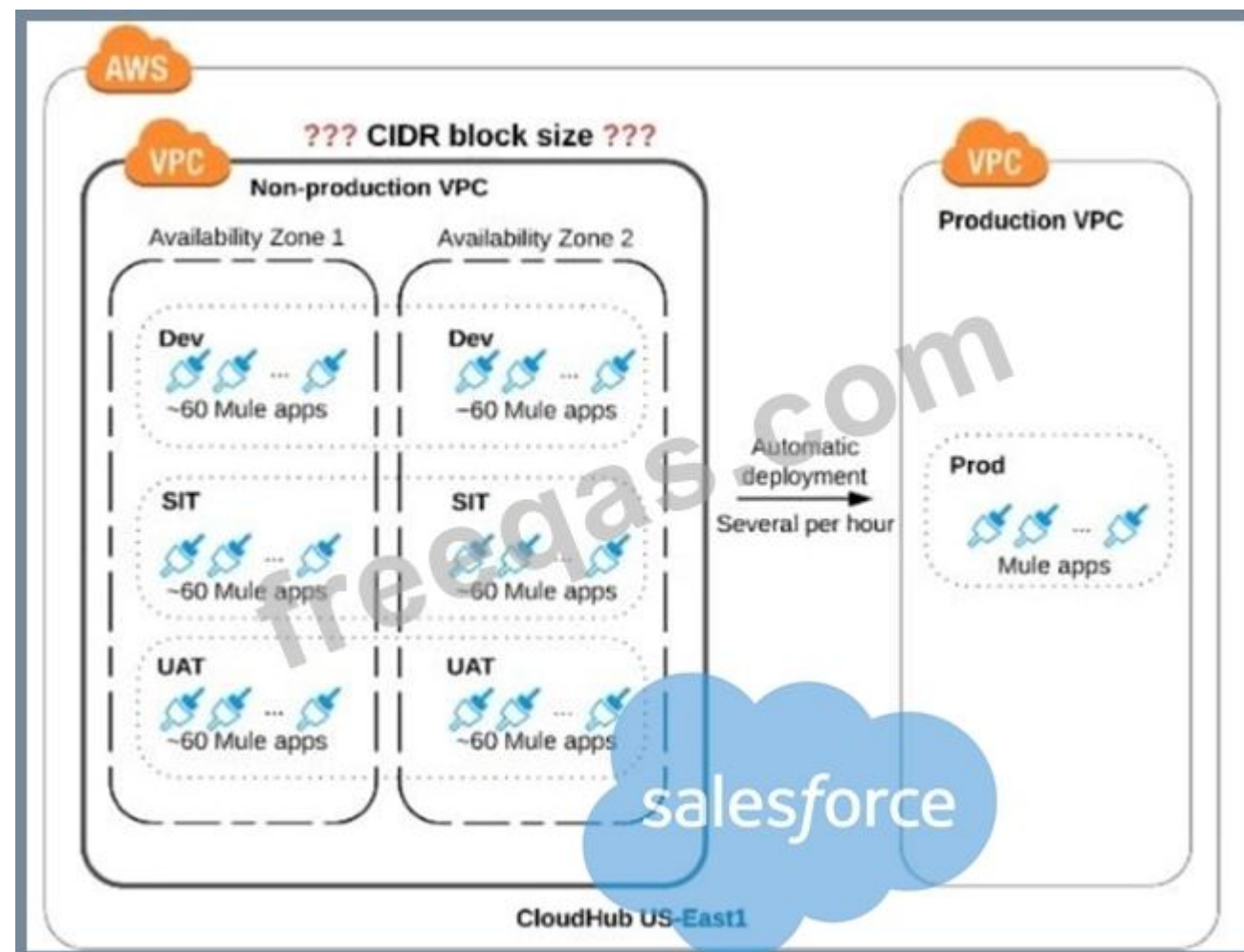
Using JMS transactions within Mule clusters allows for the decoupling of message producers and consumers, improving scalability and fault tolerance. JMS provides built-in mechanisms for handling transactional messaging, ensuring that messages are processed once and only once, even in the event of system failures.

References:

* MuleSoft Documentation on JMS

NEW QUESTION: 63

Refer to the exhibit.



An organization is sizing an Anypoint VPC for the non-production deployments of those Mule applications that connect to the organization's on-premises systems. This applies to approx. 60 Mule applications. Each application is deployed to two CloudHub i workers. The organization currently has three non-production environments (DEV, SIT and UAT) that share this VPC. The AWS region of the VPC has two AZs.

The organization has a very mature DevOps approach which automatically progresses each application through all non-production environments before automatically deploying to production. This process results in several Mule application deployments per hour, using CloudHub's normal zero-downtime deployment feature.

What is a CIDR block for this VPC that results in the smallest usable private IP address range?

A. 10.0.0.0/26 (64 IPS)

- B. 10.0.0.0/25 (128 IPs)
- C. 10.0.0.0/24 (256 IPs)
- D. 10.0.0.0/22 (1024 IPs)

Answer: ([SHOW ANSWER](#))

Mule applications are deployed in CloudHub workers and each worker is assigned with a dedicated IP * For zero downtime deployment, each worker in CloudHub needs additional IP addresses * A few IPs in a VPC are reserved for infrastructure (generally 2 IPs) * The IP addresses are usually in a private range with a subnet block specifier, such as 10.0.0.1/24 * The smallest CIDR network subnet block you can assign for your VPC is /24 (256 IP addresses) (60*3 env * 2 worker per application) + 50% of (total) for zero downtime = 540 In this case correct answer is 10.0.0.0/22 as this provided 1024 IP's .
Other IP's are insufficient.

NEW QUESTION: 64

Which Exchange asset type represents configuration modules that extend the functionality of an API and enforce capabilities such as security?

- A. RESTAPIs
- B. Rulesets
- C. Connectors
- D. Policies

Answer: D ([LEAVE A REPLY](#))

NEW QUESTION: 65

Which Anypoint Platform component should a MuleSoft developer use to create an API specification prior to building the API implementation?

- A. MUnit
- B. API Designer
- C. API Manager
- D. Runtime Manager

Answer: B ([LEAVE A REPLY](#))

A MuleSoft developer should use API Designer to create an API specification prior to building the API implementation. API Designer is a component of Anypoint Platform that provides a web-based interface for designing and documenting APIs. It supports API modeling languages such as RAML and OAS, enabling developers to define the structure, resources, and operations of an API before proceeding with implementation. This approach aligns with the design-first methodology, ensuring a well-defined contract for the API.

References:

- * Anypoint API Designer
- * API Design Best Practices

NEW QUESTION: 66

49 of A popular retailer is designing a public API for its numerous business partners. Each business partner will invoke the API at the URL 58. https://api.acme.com/partnefs/v1. The API implementation is estimated to require deployment to 5 CloudHub workers.

The retailer has obtained a public X.509 certificate for the name api.acme.com, signed by a reputable CA, to be used as the server certificate.

Where and how should the X.509 certificate and Mule applications be used to configure load balancing among the 5 CloudHub workers, and what DNS entries should be configured in order for the retailer to support its numerous business partners?

- A. Add the X.509 certificate to the Mule application's deployable archive, then configure a CloudHub Dedicated Load Balancer (DLB) for each of the Mule application's CloudHub workers
Create a CNAME for api.acme.com pointing to the DLB's A record
- B. Add the X.509 certificate to the CloudHub Shared Load Balancer (SLB), not to the Mule application Create a CNAME for api.acme.com pointing to the SLB's A record

C. Add the X.509 certificate to a CloudHub Dedicated Load Balancer (DLB), not to the Mule application Create a CNAME for api.acme.com pointing to the DLB's A record

D. Add the x.509 certificate to the Mule application's deployable archive, then configure the CloudHub Shared Load Balancer (SLB) for each of the Mule application's CloudHub workers
Create a CNAME for api.acme.com pointing to the SLB's A record

Answer: C (LEAVE A REPLY)

* An X.509 certificate is a vital safeguard against malicious network impersonators. Without x.509 server authentication, man-in-the-middle attacks can be initiated by malicious access points, compromised routers, etc.

* X.509 is most used for SSL/TLS connections to ensure that the client (e.g., a web browser) is not fooled by a malicious impersonator pretending to be a known, trustworthy website.

* Coming to the question , we can not use SLB here as SLB does not allow to define vanity domain names. * Hence we need to use DLB and add certificate in there

----- Hence correct answer is Add the X 509 certificate to the cloudbus Dedicated Load Balancer (DLB),
not the Mule application. Create the CNAME for api.acme.com pointing to the DLB's record

NEW QUESTION: 67

An organization is migrating all its Mule applications to Runtime Fabric (RTF). None of the Mule applications use Mule domain projects.

Currently, all the Mule applications have been manually deployed to a server group among several customer hosted Mule runtimes.

Port conflicts between these Mule application deployments are currently managed by the DevOps team who carefully manage Mule application properties files.

When the Mule applications are migrated from the current customer-hosted server group to Runtime Fabric (RTF), fo the Mule applications need to be rewritten and what DevOps port configuration responsibilities change or stay the same?

A. Yes, the Mule applications Must be rewritten

DevOps No Longer needs to manage port conflicts between the Mule applications

B. Yes, the Mule applications Must be rewritten

DevOps Must Still Manage port conflicts.

C. NO, The Mule applications do NOT need to be rewritten

DevOps MUST STILL manage port conflicts

D. NO, the Mule applications do NO need to be rewritten

DevOps NO LONGER needs to manage port conflicts between the Mule applications.

Answer: C (LEAVE A REPLY)

* Anypoint Runtime Fabric is a container service that automates the deployment and orchestration of your Mule applications and gateways.

* Runtime Fabric runs on customer-managed infrastructure on AWS, Azure, virtual machines (VMs) or bare- metal servers.

* As none of the Mule applications use Mule domain projects. applications are not required to be rewritten.

Also when applications are deployed on RTF, by default ingress is allowed only on 8081.

* Hence port conflicts are not required to be managed by DevOps team

NEW QUESTION: 68

An organization's IT team follows an API-led connectivity approach and must use Anypoint Platform to implement a System AP\ that securely accesses customer dat a. The organization uses Salesforce as the system of record for all customer data, and its most important objective is to reduce the overall development time to release the System API.

The team's integration architect has identified four different approaches to access the customer data from within the implementation of the System API by using different Anypoint Connectors that all meet the technical requirements of the project.

A. Use the Anypoint Connector for Salesforce to connect to the Salesforce APIs to directly access the customer data

B. Use the Anypoint Connector for HTTP to connect to the Salesforce APIs to directly access the customer data

C. Use the Anypoint Connector for Database to connect to a MySQL database to access a copy of the customer data

D. Use the Anypoint Connector tor FTP to download a file containing a recent near-real time extract of the customer data

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 69

A new Mule application under development must implement extensive data transformation logic. Some of the data transformation functionality is already available as external transformation services that are mature and widely used across the organization; the rest is highly specific to the new Mule application.

The organization follows a rigorous testing approach, where every service and application must be extensively acceptance tested before it is allowed to go into production.

What is the best way to implement the data transformation logic for this new Mule application while minimizing the overall testing effort?

- A. Implement and expose all transformation logic as microservices using DataWeave, so it can be reused by any application component that needs it, including the new Mule application
- B. Implement transformation logic in the new Mule application using DataWeave, replicating the transformation logic of existing transformation services
- C. Extend the existing transformation services with new transformation logic and Invoke them from the new Mule application
- D. Implement transformation logic in the new Mule application using DataWeave, invoking existing transformation services when possible

Answer: D ([LEAVE A REPLY](#))

Correct answer is Implement transformation logic in the new Mule application using DataWeave, invoking existing transformation services when possible. * The key here minimal testing effort, "Extend existing transformation logic" is not a feasible option because additional functionality is highly specific to the new Mule application so it should not be a part of commonly used functionality. So this option is ruled out. * "Implement transformation logic in the new Mule application using DataWeave, replicating the transformation logic of existing transformation services" Replicating the transformation logic of existing transformation services will cause duplicity of code. So this option is ruled out. * "Implement and expose all transformation logic as microservices using DataWeave, so it can be reused by any application component that needs it, including the new Mule application" as question specifies that the transformation is app specific and wont be used outside

NEW QUESTION: 70

A company is designing a mule application to consume batch data from a partner's ftps server The data files have been compressed and then digitally signed using PGP.

What inputs are required for the application to securely consumed these files?

- A. ATLS context Key Store requiring the private key and certificate for the company PGP public key of partner PGP private key for the company
- B. TLS context trust or containing a public certificate for the ftps server The FTP username and password The PGP public key of the partner
- C. ATLS context first store containing a public certificate for partner ftps server and the PGP public key of the partner TLS context Key Store containing the FTP credentials
- D. The PGP public key of the partner

The PGP private key for the company

The FTP username and password

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 71

What Is a recommended practice when designing an integration Mule 4 application that reads a large XML payload as a stream?

- A. The payload should be dealt with as a repeatable XML stream, which must only be traversed (iterated- over) once and CANNOT be accessed randomly from DataWeave expressions and scripts
- B. The payload should be dealt with as an XML stream, without converting it to a single Java object (POJO)
- C. The payload size should NOT exceed the maximum available heap memory of the Mule runtime on which the Mule application executes
- D. The payload must be cached using a Cache scope If It Is to be sent to multiple backend systems

Answer: C ([LEAVE A REPLY](#))

If the size of the stream exceeds the maximum, a STREAM_MAXIMUM_SIZE_EXCEEDED error is raised.

NEW QUESTION: 72

An airline is architecting an API connectivity project to integrate its flight data into an online aggregation website. The interface must allow for secure communication high-performance and asynchronous message exchange.

What are suitable interface technologies for this integration assuming that Mulesoft fully supports these technologies and that Anypoint connectors exist for these interfaces?

- A. CSV over FTP YAM L over TLS JSON over HTTPS
- B. SOAP over HTTPS HOP over TLS gRPC over HTTPS
- C. AsyncAPI over HTTPS
AMQP with RabbitMQ JSON/REST over HTTPS
- D. XML over ActiveMQ XML over SFTP XML/REST over HTTPS

Answer: C ([LEAVE A REPLY](#))

NEW QUESTION: 73

An insurance provider is implementing Anypoint platform to manage its application infrastructure and is using the customer hosted runtime for its business due to certain financial requirements it must meet. It has built a number of synchronous API's and is currently hosting these on a mule runtime on one server These applications make use of a number of components including heavy use of object stores and VM queues.

Business has grown rapidly in the last year and the insurance provider is starting to receive reports of reliability issues from its applications.

The DevOps team indicates that the API's are currently handling too many requests and this is over loading the server. The team has also mentioned that there is a significant downtime when the server is down for maintenance.

As an integration architect, which option would you suggest to mitigate these issues?

- A. Add a load balancer and add additional servers in a server group configuration
- B. Add a load balancer and add additional servers in a cluster configuration
- C. Increase physical specifications of server CPU memory and network
- D. Change applications by use an event-driven model

Answer: B ([LEAVE A REPLY](#))

To address the reliability and scalability issues faced by the insurance provider, adding a load balancer and configuring additional servers in a cluster configuration is the optimal solution.

Here's why:

- * Load Balancing: Implementing a load balancer will help distribute incoming API requests evenly across multiple servers. This prevents any single server from becoming a bottleneck, thereby improving the overall performance and reliability of the system.
- * Cluster Configuration: By setting up a cluster configuration, you ensure that multiple servers work together as a single unit. This provides several benefits:
 - * High Availability: If one server goes down, the load balancer can redirect traffic to other available servers in the cluster, minimizing downtime.
 - * Scalability: Additional servers can be added to the cluster to handle increased load as the business grows.
 - * Fault Tolerance: Clustering allows for automatic failover and recovery, ensuring that applications remain available even if one server fails.
 - * Maintenance: With a cluster configuration, servers can be taken offline for maintenance one at a time without affecting the overall availability of the applications, as the load balancer can redirect traffic to the remaining servers.
- * VM Queues and Object Stores: In a clustered environment, the use of VM queues and object stores can be more efficiently managed as these resources are distributed across multiple servers, reducing the risk of contention and improving throughput.

References:

- * MuleSoft Documentation on Clustering: <https://docs.mulesoft.com/mule-runtime/4.3/clustering>
- * Best Practices for Scaling Mule Applications: <https://blogs.mulesoft.com/dev/mule-dev/mule-4-scaling-applications/>

NEW QUESTION: 74

When a Mule application using VM queues is deployed to a customer-hosted cluster or multiple CloudHub v1.0 workers/replicas, how are messages consumed across the nodes?

- A. Sequentially, only from the primary node
- B. Round-robin, within an XA transaction
- C. In a non-deterministic way
- D. Sequentially, from a dedicated Anypoint MQ queue

Answer: D ([LEAVE A REPLY](#))

NEW QUESTION: 75

Which Exchange asset type represents configuration modules that extend the functionality of an API and enforce capabilities such as security?

- A. Rulesets
- B. Policies
- C. REST APIs
- D. Connectors

Answer: B ([LEAVE A REPLY](#))

In Anypoint Exchange, policies are the asset type that represents configuration modules extending the functionality of an API and enforcing capabilities such as security. Policies can be applied to APIs to control access, apply throttling, manage security, and other aspects that modify or extend the behavior of APIs.

Rulesets, REST APIs, and connectors serve different purposes within the Anypoint Platform. Rulesets are used for validation or routing decisions. REST APIs define the endpoints and methods for API interactions, and connectors enable connectivity to various systems and services. Only policies are specifically designed to enforce additional capabilities on APIs.

References

* MuleSoft Anypoint Platform Documentation on API Policies

* Anypoint Exchange Overview

NEW QUESTION: 76

An organization has chosen Mulesoft for their integration and API platform.

According to the Mulesoft catalyst framework, what would an integration architect do to create achievement goals as part of their business outcomes?

- A. Measure the impact of the centre for enablement
- B. build and publish foundational assets
- C. agree upon KPI's and help develop and overall success plan
- D. evangelize API's

Answer: C ([LEAVE A REPLY](#))

According to the MuleSoft Catalyst framework, an Integration Architect plays a crucial role in defining and achieving business outcomes. One of their key responsibilities is to agree upon Key Performance Indicators (KPIs) and help develop an overall success plan. This involves working with stakeholders to identify measurable goals and ensure that the integration initiatives align with the organization's strategic objectives.

KPIs are critical for tracking progress, measuring success, and making data-driven decisions. By agreeing on KPIs and developing a success plan, the Integration Architect ensures that the organization can objectively measure the impact of its integration efforts and adjust strategies as needed to achieve desired business outcomes.

References:

* MuleSoft Catalyst Knowledge Hub

NEW QUESTION: 77

An organization is migrating all its Mule applications to Runtime Fabric (RTF). None of the Mule applications use Mule domain projects.

Currently, all the Mule applications have been manually deployed to a server group among several customer hosted Mule runtimes.

Port conflicts between these Mule application deployments are currently managed by the DevOps team who carefully manage Mule application properties files.

When the Mule applications are migrated from the current customer-hosted server group to Runtime Fabric (RTF), for the Mule applications need to be rewritten and what DevOps port configuration responsibilities change or stay the same?

- A. Yes, the Mule applications Must be rewrittenDevOps No Longer needs to manage port conflicts between the Mule applications
- B. Yes, the Mule applications Must be rewrittenDevOps Must Still Manage port conflicts.
- C. NO, The Mule applications do NOT need to be rewrittenDevOps MUST STILL manage port conflicts
- D. NO, the Mule applications do NO need to be rewrittenDevOps NO LONGER needs to manage port conflicts between the Mule applications.

Answer: C (LEAVE A REPLY)

* Anypoint Runtime Fabric is a container service that automates the deployment and orchestration of your Mule applications and gateways.

* Runtime Fabric runs on customer-managed infrastructure on AWS, Azure, virtual machines (VMs) or bare-metal servers.

* As none of the Mule applications use Mule domain projects. applications are not required to be rewritten.

Also when applications are deployed on RTF, by default ingress is allowed only on 8081.

* Hence port conflicts are not required to be managed by DevOps team

NEW QUESTION: 78

As a part of project requirement, client will send a stream of data to mule application. Payload size can vary between 10mb to 5GB. Mule application is required to transform the data and send across multiple sftp servers. Due to the cost cuttings in the organization, mule application can only be allocated one worker with size of 0.2 vCore.

As an integration architect, which streaming strategy you would suggest to handle this scenario?

- A. In-memory non repeatable stream
- B. File based non-repeatable stream
- C. In-memory repeatable stream
- D. File based repeatable storage

Answer: D (LEAVE A REPLY)

As the question says that data needs to be sent across multiple sftp servers, we cannot use non-repeatable streams. The non-repeatable strategy disables repeatable streams, which enables you to read an input stream only once.

You cannot use in-memory storage because with 0.2 vcore you will get only 1 GB of heap memory. Hence application will error out for file more than 1 GB.

Hence the correct option is file base repeatable stream

NEW QUESTION: 79

Refer to the exhibit.



The HTTP Listener and the Logger are being handled from which thread pools respectively?

- A. Shared Selector Pool and CPU LITE
- B. UBER and NONBLOCKING
- C. BLOCKING _IO and UBER
- D. CPU_INTENSIVE and Dedicated Selector pool

Answer: A ([LEAVE A REPLY](#))

NEW QUESTION: 80

An organization's IT team follows an API-led connectivity approach and must use Anypoint Platform to implement a System API that securely accesses customer data. The organization uses Salesforce as the system of record for all customer data, and its most important objective is to reduce the overall development time to release the System API.

The team's integration architect has identified four different approaches to access the customer data from within the implementation of the System API by using different Anypoint Connectors that all meet the technical requirements of the project.

- A. Use the Anypoint Connector for Database to connect to a MySQL database to access a copy of the customer data
- B. Use the Anypoint Connector for HTTP to connect to the Salesforce APIs to directly access the customer data
- C. Use the Anypoint Connector for Salesforce to connect to the Salesforce APIs to directly access the customer data
- D. Use the Anypoint Connector for FTP to download a file containing a recent near-real time extract of the customer data

Answer: ([SHOW ANSWER](#))

To reduce the overall development time to release the System API, the most efficient approach is to use the Anypoint Connector for Salesforce. This connector is specifically designed to interact with Salesforce APIs, providing a seamless and optimized way to access customer data directly from Salesforce. Using this connector simplifies development, reduces complexity, and leverages built-in capabilities for authentication and data retrieval, ensuring a faster and more secure implementation.

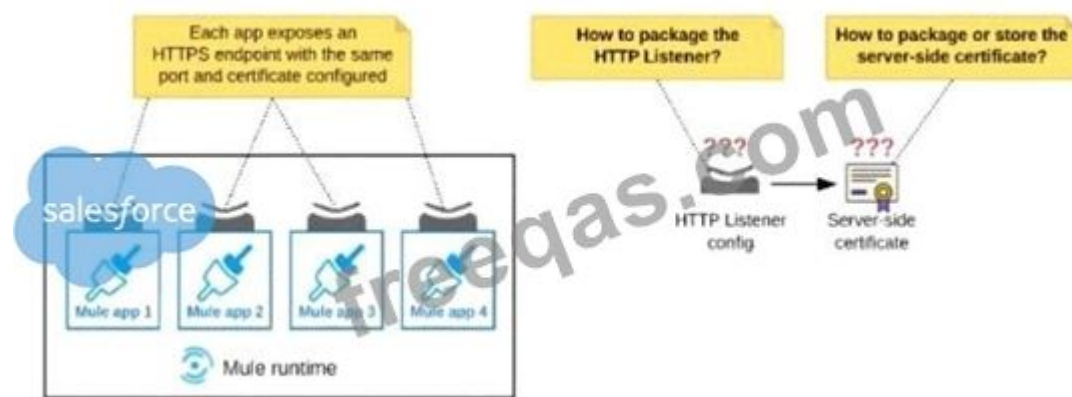
References:

Anypoint Connector for Salesforce

Connecting to Salesforce with Anypoint Platform

NEW QUESTION: 81

Refer to the exhibit.



An organization deploys multiple Mule applications to the same customer -hosted Mule runtime. Many of these Mule applications must expose an HTTPS endpoint on the same port using a server-side certificate that rotates often.

What is the most effective way to package the HTTP Listener and package or store the server-side certificate when deploying these Mule applications, so the disruption caused by certificate rotation is minimized?

- A.** Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint Package the server-side certificate in ALL Mule APPLICATIONS that need to expose an HTTPS endpoint
- B.** Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint. Store the server-side certificate in a shared filesystem location in the Mule runtime's classpath, OUTSIDE the Mule DOMAIN or any Mule APPLICATION
- C.** Package an HTTPS Listener configuration In all Mule APPLICATIONS that need to expose an HTTPS endpoint Package the server-side certificate in a NEW Mule DOMAIN project
- D.** Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing It from all Mule applications that need to expose an HTTPS endpoint. Package the server-side certificate in the SAME Mule DOMAIN project Go to Set

Answer: B (LEAVE A REPLY)

In this scenario, both A & C will work, but A is better as it does not require repackaging to the domain project at all.

Correct answer is Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint. Store the server-side certificate in a shared filesystem location in the Mule runtime's classpath, OUTSIDE the Mule DOMAIN or any Mule APPLICATION.

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:

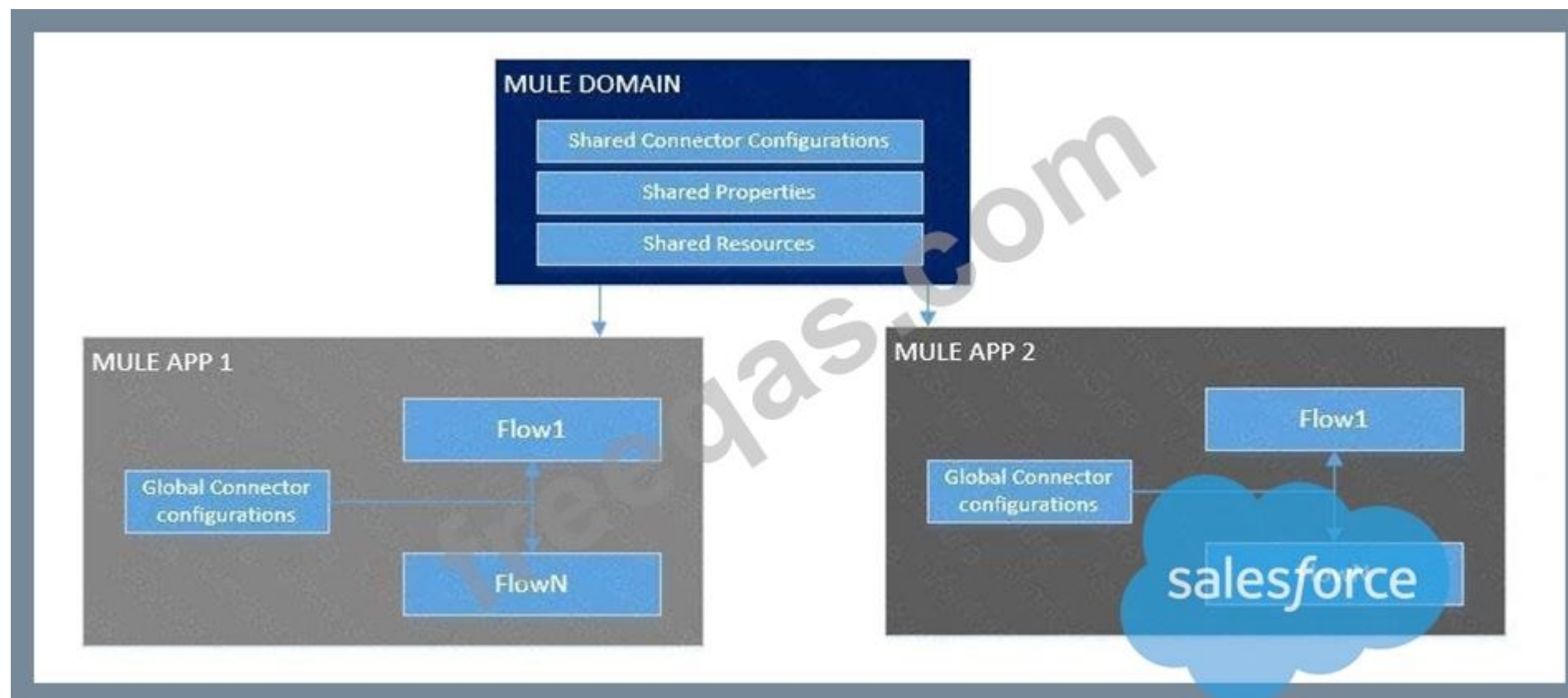
- Expose multiple services within the domain through the same port.
- Share the connection to persistent storage.
- Share services between apps through a well-defined interface.
- Ensure consistency between apps upon any changes because the configuration is only set in one place.

* 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 be redeployed instead of all the applications.

You can start using domains in only three steps:

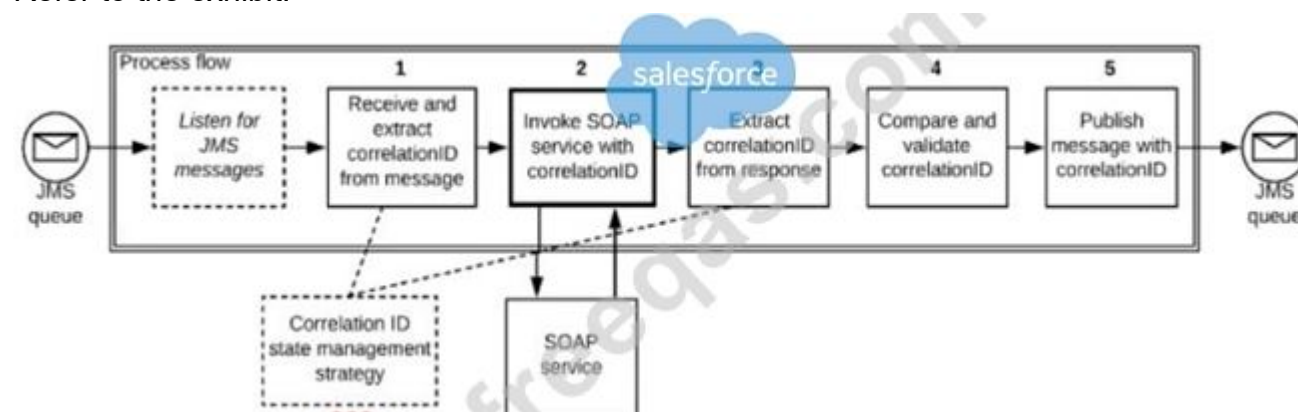
- 1) Create a Mule Domain project
- 2) Create the global connector configurations which needs to be shared across the applications inside the Mule Domain project
- 3) Modify the value of domain in mule-deploy.properties file of the applications Graphical user interface Description automatically generated



Use a certificate defined in already deployed Mule domain. Configure the certificate in the domain so that the API proxy HTTPS Listener references it, and then deploy the secure API proxy to the target Runtime Fabric, or on-premises target. (CloudHub is not supported with this approach because it does not support Mule domains.)

NEW QUESTION: 82

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 1: 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?

- A. Both Correlation ID values should be stored in a persistent object store
- B. Both Correlation ID values should be stored In a non-persistent object store
- C. 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 variable/attribute
- D. Both Correlation ID values should be stored as Mule event variable/attribute

Answer: ([SHOW ANSWER](#))

- * If we store Correlation id value in step 1 as Mule event variables/attributes, the values will be cleared after server restart and we want system to be fault tolerant.
- * The Correlation ID value in Step 1 should be stored in a persistent object store.
- * We don't need to store Correlation ID value in Step 3 to persistent object store. We can store it but as we also need to make application performant. We can avoid this step of accessing persistent object store.
- * Accessing persistent object stores slow down the performance as persistent object stores are by default stored in shared file systems.
- * As the SOAP service is idempotent in nature. In case of any failures , using this Correlation ID saved in first step we can make call to SOAP service and validate the Correlation ID.

Top of Form

Additional Information:

- * Competing Consumers are multiple consumers that are all created to receive messages from a single Point-to-Point Channel. When the channel delivers a message, any of the consumers could potentially receive it. The messaging system's implementation determines which consumer actually receives the message, but in effect the consumers compete with each other to be the receiver. Once a consumer receives a message, it can delegate to the rest of its application to help process the message.



- * In case you are unaware about term idempotent re is more info:

Idempotent operations means their result will always same no matter how many times these operations are invoked.

IDEMPOTENCE

WHEN PERFORMING AN OPERATION AGAIN GIVES THE SAME RESULT

HTTP METHOD	IDEMPOTENCE	SAFETY
GET	YES	YES
HEAD	YES	YES
PUT	YES	NO
DELETE	YES	NO
POST	NO	NO
PATCH	NO	NO

Bottom of Form

NEW QUESTION: 83

A rate limiting policy has been applied to a soap V1.2 API published in Cloudfoundry. The API implementation catches errors in a global error handler and propagates them in the main flow for HTTP:

RETRY_EXHAUSTED with HTTP status set to 429 and any with the HTTP status set to 500.

What is the expected HTTP status when the client exceeds the quota of the API calls?

- A. HTTP status 429 as defined in the HTTP:RETRY_EXHAUSTED error handler in the API
- B. HTTP status 500 as defined in the ANY error handler in the API since an API:RETRY_EXHAUSTED will be generated
- C. HTTP status 401 unauthorized for policy violation
- D. HTTP status 400 from the rate-limiting policy violation since the call does not reach the back-end

Answer: ([SHOW ANSWER](#))

In the given scenario, a rate limiting policy has been applied to the SOAP API and a global error handler is configured to handle HTTP:RETRY_EXHAUSTED errors with a 429 status code. The rate limiting policy will trigger when the client exceeds the allowed quota of API calls. Since the HTTP:RETRY_EXHAUSTED error specifically catches quota exhaustion errors and the error handler is configured to return a 429 status code, the expected HTTP status returned to the client when the quota is exceeded will be 429. This error code indicates that the user has sent too many requests in a given amount of time ("rate limiting").

References:

- * MuleSoft Documentation on Error Handling
- * HTTP Status Codes

NEW QUESTION: 84

A team has completed the build and test activities for a Mule application that implements a System API for its application network.

Which Anypoint Platform component should the team now use to both deploy and monitor the System API implementation?

- A. Design Center
- B. Runtime Manager
- C. API Manager
- D. Anypoint Exchange

Answer: ([SHOW ANSWER](#))

NEW QUESTION: 85

An organization is building a test suite for their applications using m-unit. The integration architect has recommended using test recorder in studio to record the processing flows and then configure unit tests based on the capture events. What are the two considerations that must be kept in mind while using test recorder (Choose two answers)?

- A. Tests for flows cannot be created with Mule errors raised inside the flow or already existing in the incoming event
- B. Recorder supports mocking a message before or inside a ForEach processor
- C. The recorder support loops where the structure of the data being tested changes inside the iteration
- D. A recorded flow execution ends successfully but the result does not reach its destination because the application is killed
- E. Mocking values resulting from parallel processes are possible and will not affect the execution of the processes that follow in the test

Answer: A,D ([LEAVE A REPLY](#))

* When using MUnit's test recorder in Anypoint Studio to create unit tests, consider the following points:

A: Tests for flows cannot be created with Mule errors raised inside the flow or already existing in the incoming event:

* Explanation: The test recorder cannot record flows if Mule errors are raised during the flow execution or if the incoming event already contains errors. This limitation requires users to handle or clear errors before recording the flow to ensure accurate test creation.

D: A recorded flow execution ends successfully but the result does not reach its destination because the application is killed:

* Explanation: If the application is killed before the recorded flow execution completes, the recorder captures the flow up to the point of termination. However, the final result may not be reached or recorded. This scenario should be avoided to ensure complete and reliable test recordings.

These considerations help ensure the accuracy and reliability of tests created using the test recorder.

References:

- * MUnit Documentation: <https://docs.mulesoft.com/munit/2.2/>
- * MUnit Test Recorder: <https://blogs.mulesoft.com/dev/mule-dev/using-the-munit-test-recorder/>

NEW QUESTION: 86

What requires configuration of both a key store and a trust store for an HTTP Listener?

- A. Support for TLS mutual (two-way) authentication with HTTP clients
- B. Encryption of requests to both subdomains and API resource endpoints fhttpPs://aDi.customer.com/ and https://customer.com/api)
- C. Encryption of both HTTP request and HTTP response bodies for all HTTP clients
- D. Encryption of both HTTP request header and HTTP request body for all HTTP clients

Answer: A (LEAVE A REPLY)

1 way SSL : The server presents its certificate to the client and the client adds it to its list of trusted certificate. And so, the client can talk to the server.

2-way SSL: The same principle but both ways. i.e. both the client and the server has to establish trust between themselves using a trusted certificate. In this way of a digital handshake, the server needs to present a certificate to authenticate itself to client and client has to present its certificate to server.

* TLS is a cryptographic protocol that provides communications security for your Mule app.

* TLS offers many different ways of exchanging keys for authentication, encrypting data, and guaranteeing message integrity Keystores and Truststores Truststore and keystore contents differ depending on whether they are used for clients or servers:

For servers: the truststore contains certificates of the trusted clients, the keystore contains the private and public key of the server. For clients: the truststore contains certificates of the trusted servers, the keystore contains the private and public key of the client.

Adding both a keystore and a truststore to the configuration implements two-way TLS authentication also known as mutual authentication.

* in this case, correct answer is Support for TLS mutual (two-way) authentication with HTTP clients.



NEW QUESTION: 87

An organization's security requirements mandate centralized control at all times over authentication and authorization of external applications when invoking web APIs managed on Anypoint Platform.

What Anypoint Platform feature is most idiomatic (used for its intended purpose), straightforward, and maintainable to use to meet this requirement?

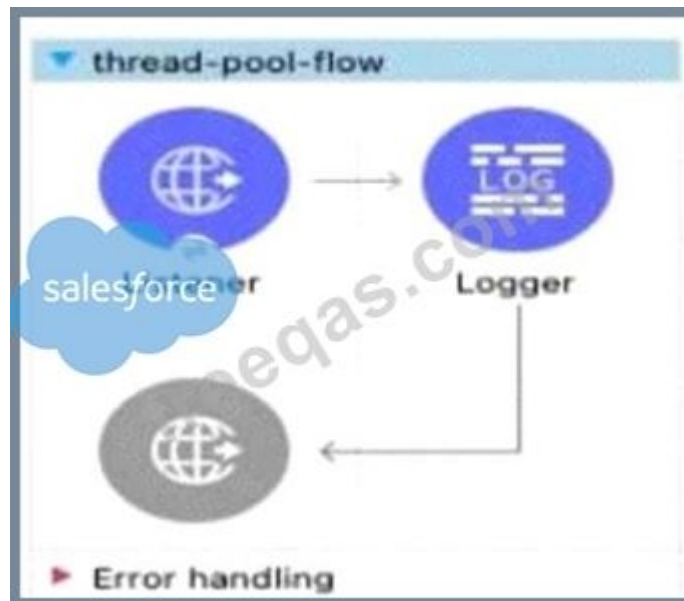
- A. Client management configured in access management
- B. Identity management configured in access management
- C. Enterprise Security module coded in Mule applications
- D. External access configured in API Manager

Answer: (SHOW ANSWER)

Reference: <https://blogs.mulesoft.com/dev-guides/api-security-ways-to-authenticate-and-authorize/>

NEW QUESTION: 88

Refer to the exhibit.



A customer is running Mule applications on Runtime Fabric for Self-Managed Kubernetes (RTF-BYOKS) in a multi-cloud environment.

Based on this configuration, how do Agents and Runtime Manager communicate, and what is exchanged between them?

- A. BLOCKING_IO, UBER
- B. UBER, Dedicated NIO Selector Pool
- C. CPU_LITE, CPU_INTENSIVE
- D. Shared NIO Selector Pool, CPU_LITE

Answer: D (LEAVE A REPLY)

In the context of Mule applications running on Runtime Fabric for Self-Managed Kubernetes (RTF-BYOKS) in a multi-cloud environment, understanding the thread pools used for communication between Agents and Runtime Manager is crucial:

* Shared NIO Selector Pool: This pool is responsible for handling non-blocking IO operations, such as network communication. It ensures efficient handling of IO operations by using a small number of threads to manage multiple IO tasks simultaneously.

* CPU_LITE: This thread pool is used for lightweight CPU operations. It is designed to handle tasks that do not require significant computational resources, ensuring that lightweight operations are processed efficiently without overwhelming the system.

The combination of the Shared NIO Selector Pool and CPU_LITE thread pool ensures efficient and reliable communication between Agents and Runtime Manager in the RTF environment.

References:

- * MuleSoft Threading and Thread Pools
- * Runtime Fabric Architecture

NEW QUESTION: 89

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?

- A. 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
- B. 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
- C. They must be assigned as an API version owner of the API in the Staging environment
- D. 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

Answer: (SHOW ANSWER)

* It's mentioned that the API is governed by a Client ID Enforcement policy in all environments.

- * Client ID Enforcement policy allows only authorized applications to access the deployed API implementation.
- * Each authorized application is configured with credentials: client_id and client_secret.
- * At runtime, authorized applications provide the credentials with each request to the API implementation.

MuleSoft Reference: <https://docs.mulesoft.com/api-manager/2.x/policy-mule3-client-id-based-policies>

NEW QUESTION: 90

According to MuleSoft, which system integration term describes the method, format, and protocol used for communication between two system?

- A. Component
- B. interaction
- C. Message
- D. Interface

Answer: D ([LEAVE A REPLY](#))

According to MuleSoft, the term "interface" describes the method, format, and protocol used for communication between two systems. An interface defines how systems interact, specifying the data formats (e.g., JSON, XML), protocols (e.g., HTTP, FTP), and methods (e.g., GET, POST) that are used to exchange information. Properly designed interfaces ensure compatibility and seamless communication between integrated systems.

References:

- * MuleSoft Glossary of Integration Terms
- * System Interfaces and APIs

NEW QUESTION: 91

A MuleSoft developer must implement an API as a Mule application, run the application locally, and execute unit tests against the Running application.

Which Anypoint Platform component can the developer use to full all of these requirements?

- A. API Manager
- B. API Designer
- C. Anypoint CLI
- D. Anypoint Studio

Answer: ([SHOW ANSWER](#)**)**

A MuleSoft developer can use Anypoint Studio to implement an API as a Mule application, run the application locally, and execute unit tests against the running application. Anypoint Studio is an integrated development environment (IDE) for designing, building, testing, and deploying Mule applications. It provides tools for local development and testing, including MUnit for creating and running unit tests, making it a comprehensive solution for developing MuleSoft applications.

References:

- * Anypoint Studio Overview
- * Using MUnit for Unit Testing

Valid MuleSoft-Integration-Architect-I Dumps shared by PrepPdf.com for Helping Passing MuleSoft-Integration-Architect-I Exam! PrepPdf.com now offer the **newest MuleSoft-Integration-Architect-I exam dumps**, the PrepPdf.com MuleSoft-Integration-Architect-I exam **questions have been updated** and **answers have been corrected** get the **newest** PrepPdf.com MuleSoft-Integration-Architect-I dumps with Test Engine here: <https://www.preppdf.com/Salesforce/MuleSoft-Integration-Architect-I-prepaway-exam-dumps.html> (275 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 92

A manufacturing company is planning to deploy Mule applications to its own Azure Kubernetes Service infrastructure.

The organization wants to make the Mule applications more available and robust by deploying each Mule application to an isolated Mule runtime in a Docker container while managing all the Mule applications from the MuleSoft-hosted control plane.

What is the most idiomatic (used for its intended purpose) choice of runtime plane to meet these organizational requirements?

- A. CloudHub
- B. Anypoint Platform Private Cloud Edition
- C. Anypoint Runtime Fabric
- D. Anypoint Service Mesh

Answer: C ([LEAVE A REPLY](#))

NEW QUESTION: 93

An organization needs to procure an enterprise software system to increase cross-selling opportunities and better track prospect data.

Which category of enterprise software has these core capabilities, when used for its typical and intended purpose?

- A. Supply Chain Management (SCM)
- B. IT Service Management (ITSM)
- C. Business-to-Business (B2B)
- D. Customer Relationship Management (CRM)

Answer: D ([LEAVE A REPLY](#))

Customer Relationship Management (CRM) systems are designed to manage a company's interactions with current and potential customers. They provide capabilities for tracking prospect data, managing customer interactions, and analyzing customer information to improve cross-selling opportunities. By using a CRM system, organizations can consolidate customer information, enhance customer relationships, and leverage data analytics to drive sales and marketing efforts.

References:

- * What is CRM?
- * Benefits of CRM Systems

NEW QUESTION: 94

Refer to the exhibit.

```
traits:  
  error-responses: traits/error-responses.raml  
  jwt-required:  
    headers:  
      x-jwt:  
        type: string  
        description: JWT token string
```

What is the type data format shown in the exhibit?

- A. XML
- B. YAML
- C. CSV
- D. JSON

Answer: D ([LEAVE A REPLY](#))

NEW QUESTION: 95

A global organization operates datacenters in many countries. There are private network links between these datacenters because all business data (but NOT metadata) must be exchanged over these private network connections.

The organization does not currently use AWS in any way.

The strategic decision has Just been made to rigorously minimize IT operations effort and investment going forward.

What combination of deployment options of the Anypoint Platform control plane and runtime plane(s) best serves this organization at the start of this strategic journey?

- A. MuleSoft-hosted Anypoint Platform control plane CloudHub Shared Worker Cloud in multiple AWS regions
- B. Anypoint Platform - Private Cloud Edition Customer-hosted runtime plane in each datacenter
- C. MuleSoft-hosted Anypoint Platform control plane Customer-hosted runtime plane in multiple AWS regions
- D. MuleSoft-hosted Anypoint Platform control plane Customer-hosted runtime plane in each datacenter

Answer: ([SHOW ANSWER](#))

Correct answer is MuleSoft-hosted Anypoint Platform control plane Customer-hosted runtime plane in each datacenter There are two things to note about the question which can help us figure out correct answer.. * Business data must be exchanged over these private network connections which means we can not use MuleSoft provided Cloudhub option. So we are left with either customer hosted runtime in external cloud provider or customer hosted runtime in their own premises. As customer does not use AWS at the moment.

Hence that don't have the immediate option of using Customer-hosted runtime plane in multiple AWS regions. hence the most suitable option for runtime plane is Customer-hosted runtime plane in each datacenter

* Metadata has no limitation to reside in organization premises. Hence for control plane MuleSoft hosted Anypoint platform can be used as a strategic solution.

Hybrid is the best choice to start. Mule hosted Control plane and Customer hosted Runtime to start with.

Once they mature in cloud migration, everything can be in Mule hosted.

NEW QUESTION: 96

An insurance company is using a CloudHub runtime plane. As a part of requirement, email alert should be sent to internal operations team every time of policy applied to an API instance is deleted As an integration architect suggest on how this requirement be met?

- A. Use audit logs in Anypoint platform to detect a policy deletion and configure the Audit logs alert feature to send an email to the operations team
- B. Use Anypoint monitoring to configure an alert that sends an email to the operations team every time a policy is deleted in API manager
- C. Create a custom connector to be triggered every time of policy is deleted in API manager
- D. Implement a new application that uses the Audit log REST API to detect the policy deletion and send an email to operations team the SMTP connector

Answer: D (LEAVE A REPLY)

* Requirement Analysis: The organization needs to send email alerts to the internal operations team whenever a policy applied to an API instance is deleted in the CloudHub runtime plane.

* Solution: The most effective approach is to implement a new Mule application that uses the Audit Log REST API to detect when a policy is deleted and then sends an email using the SMTP connector.

* Implementation Steps:

* Access Audit Logs:

* Use the Anypoint Platform's Audit Log REST API to monitor and detect policy deletion events.

* Example of accessing the Audit Log REST API:

GET /accounts/api/v2/organizations/{orgId}/audit-logs

* This endpoint will provide logs, including events related to policy deletions.

* Create Email Notification Application:

* Create a Mule application that periodically polls the Audit Log REST API to check for policy deletion events.

* Use the SMTP connector to send email notifications to the operations team.

* Example Mule flow:

```
<flow name="policy-deletion-alert-flow"> <scheduler frequency="60000" /> <http:request config-ref=" AuditLogAPIConfig" path="/audit-logs" method="GET"> <http:request-builder>
<http:uri-params> <http:uri- param key="orgId" value="your-org-id" /> </http:uri-params> <http:query-params> <http:query-param key=" eventType" value="PolicyDeleted" /> </http:query-
params> </http:request-builder> </http:request> <choice>
<when expression="#[payload contains 'PolicyDeleted']"> <smtp:send config-ref="SMTP_Config" to=" operations@company.com" subject="Policy Deletion Alert"> <smtp:body>Policy
deletion detected: #
[payload]</smtp:body> </smtp:send> </when> </choice> </flow>
```

* Scheduler: Configures the application to check the audit logs at regular intervals (e.g., every minute).

* HTTP Request: Makes a GET request to the Audit Log REST API to fetch the logs.

* Choice Router: Checks if the payload contains a policy deletion event.

* SMTP Connector: Sends an email to the operations team with details about the policy deletion.

* Advantages:

* Automation: Automates the detection of policy deletions and the notification process, reducing manual monitoring efforts.

* Timeliness: Ensures the operations team is promptly informed of any policy deletions, enabling them to take immediate action if necessary.

References

* MuleSoft Documentation on Audit Log REST API

* MuleSoft Documentation on SMTP Connector

* MuleSoft Documentation on MuleSoft Scheduler Component

NEW QUESTION: 97

What aspect of logging is only possible for Mule applications deployed to customer-hosted Mule runtimes, but NOT for Mule applications deployed to CloudHub?

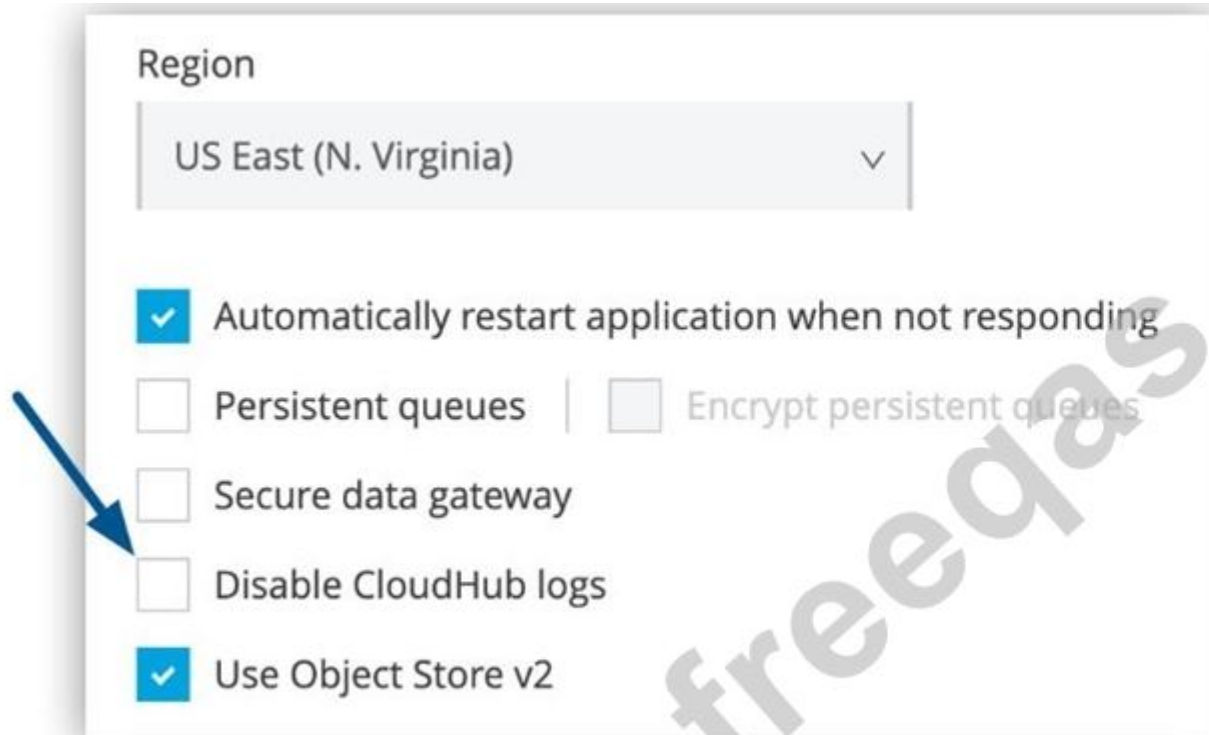
- A. To send Mule application log entries to Splunk
- B. To change log4j2 log levels in Anypoint Runtime Manager without having to restart the Mule application
- C. To log certain messages to a custom log category
- D. To directly reference one shared and customized log4j2.xml file from multiple Mule applications

Answer: D (LEAVE A REPLY)

* Correct answer is To directly reference one shared and customized log4j2.xml file from multiple Mule applications. Key word to note in the answer is directly.

* By default, CloudHub replaces a Mule application's log4j2.xml file with a CloudHub log4j2.xml file. This specifies the CloudHub appender to write logs to the CloudHub logging service.

* You cannot modify CloudHub log4j2.xml file to add any custom appender. But there is a process in order to achieve this. You need to raise a request on support portal to disable CloudHub provided Mule application log4j2 file.



* Once this is done, Mule application's log4j2.xml file is used which you can use to send/export application logs to other log4j2 appenders, such as a custom logging system MuleSoft does not own any responsibility for lost logging data due to misconfiguration of your own log4j appender if it happens by any chance.

Disabling CloudHub logs

×

salesforce

☒

I realize that disabling Cloudhub logs means MuleSoft support will no longer be able to view logs for this application or help debug issues with this application.

☒

I have configured my own logging for this application, and will use that instead of the CloudHub logs to debug any issues with this application.

Apply Changes

* One more difference between customer-hosted Mule runtimes and CloudHub deployed mule instance is that

- CloudHub system log messages cannot be sent to external log management system without installing custom CH logging configuration through support
- where as Customer-hosted runtime can send system and application log to external log management system MuleSoft Reference:

<https://docs.mulesoft.com/runtime-manager/viewing-log-data>

<https://docs.mulesoft.com/runtime-manager/custom-log-appender>

NEW QUESTION: 98

An integration architect is designing an API that must accept requests from API clients for both XML and JSON content over HTTP/1.1 by default.

Which API architectural style, when used for its intended and typical purposes, should the architect choose to meet these requirements?

- A. REST
- B. GraphQL
- C. SOAP
- D. grRPC

Answer: A ([LEAVE A REPLY](#))

NEW QUESTION: 99

A leading bank implementing new mule API.

The purpose of API to fetch the customer account balances from the backend application and display them on the online platform the online banking platform. The online banking platform will send an array of accounts to Mule API get the account balances.

As a part of the processing the Mule API needs to insert the data into the database for auditing purposes and this process should not have any performance related implications on the account balance retrieval flow How should this requirement be implemented to achieve better throughput?

- A. Implement parallel for each scope to fetch the data from the backend application and use Async scope to insert the records into the Audit database
- B. Implement the Async scope fetch the data from the backend application and to insert records in the Audit database
- C. Implement a for each scope to fetch the data from the back-end application and to insert records into the Audit database
- D. Implement a try-catch scope to fetch the data from the back-end application and use the Async scope to insert records into the Audit database

Answer: A ([LEAVE A REPLY](#))

NEW QUESTION: 100

An automation engineer needs to write scripts to automate the steps of the API lifecycle, including steps to create, publish, deploy and manage APIs and their implementations in Anypoint Platform.

What Anypoint Platform feature can be used to automate the execution of all these actions in scripts in the easiest way without needing to directly invoke the Anypoint Platform REST APIs?

- A. Automated Policies in API Manager
- B. Runtime Manager agent
- C. The Mule Maven Plugin
- D. Anypoint CLI

Answer: D ([LEAVE A REPLY](#))

Anypoint Platform provides a scripting and command-line tool for both Anypoint Platform and Anypoint Platform Private Cloud Edition (Anypoint Platform PCE). The command-line interface (CLI) supports both the interactive shell and standard CLI modes and works with: Anypoint Exchange Access management Anypoint Runtime Manager

Valid MuleSoft-Integration-Architect-I Dumps shared by PrepPdf.com for Helping Passing MuleSoft-Integration-Architect-I Exam! PrepPdf.com now offer the **newest MuleSoft-Integration-Architect-I exam dumps**, the PrepPdf.com MuleSoft-Integration-Architect-I exam **questions have been updated** and **answers have been corrected** get the **newest** PrepPdf.com MuleSoft-Integration-Architect-I dumps with Test Engine here: <https://www.preppdf.com/Salesforce/MuleSoft-Integration-Architect-I-prepaway-exam-dumps.html> (275 Q&As Dumps, **40%OFF** Special Discount: **Exam-Tests**)